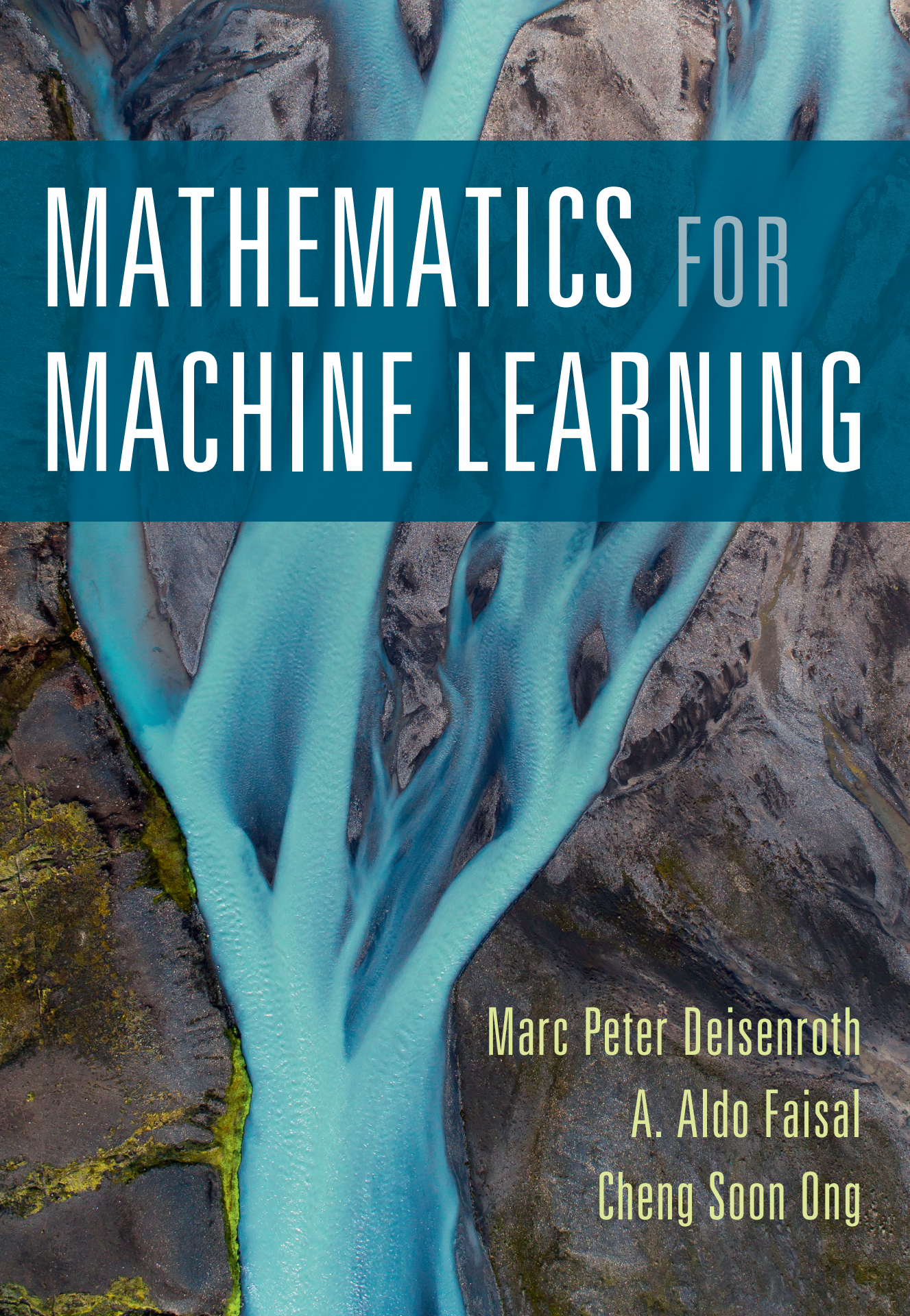

MATHEMATICS FOR
MACHINE LEARNING
Marc Peter Deisenroth
A. Aldo Faisal
Cheng Soon Ong

Contents

Foreword

Machine learning is the latest in a long line of attempts to distill human knowledge and reasoning into a form that is suitable for constructing machines and engineering automated systems. As machine learning becomes more ubiquitous and its software packages become easier to use, it is natural and desirable that the low-level technical details are abstracted away and hidden from the practitioner. However, this brings with it the danger that a practitioner becomes unaware of the design decisions and, hence, the limits of machine learning algorithms.

The enthusiastic practitioner who is interested to learn more about the magic behind successful machine learning algorithms currently faces a daunting set of pre-requisite knowledge:

- Programming languages and data analysis tools
- Large-scale computation and the associated frameworks
- Mathematics and statistics and how machine learning builds on it

At universities, introductory courses on machine learning tend to spend early parts of the course covering some of these pre-requisites. For historical reasons, courses in machine learning tend to be taught in the computer science department, where students are often trained in the first two areas of knowledge, but not so much in mathematics and statistics.

Current machine learning textbooks primarily focus on machine learning algorithms and methodologies and assume that the reader is competent in mathematics and statistics. Therefore, these books only spend one or two chapters on background mathematics, either at the beginning of the book or as appendices. We have found many people who want to delve into the foundations of basic machine learning methods who struggle with the mathematical knowledge required to read a machine learning textbook. Having taught undergraduate and graduate courses at universities, we find that the gap between high school mathematics and the mathematics level required to read a standard machine learning textbook is too big for many people.

This book brings the mathematical foundations of basic machine learning concepts to the fore and collects the information in a single place so that this skills gap is narrowed or even closed.

Why Another Book on Machine Learning?

Machine learning builds upon the language of mathematics to express concepts that seem intuitively obvious but that are surprisingly difficult to formalize. Once formalized properly, we can gain insights into the task we want to solve. One common complaint of students of mathematics around the globe is that the topics covered seem to have little relevance to practical problems. We believe that machine learning is an obvious and direct motivation for people to learn mathematics.

"Math is linked in the popular mind with phobia and anxiety. You'd think we're discussing spiders." (Strogatz, 2014, page 281)

This book is intended to be a guidebook to the vast mathematical literature that forms the foundations of modern machine learning. We motivate the need for mathematical concepts by directly pointing out their usefulness in the context of fundamental machine learning problems. In the interest of keeping the book short, many details and more advanced concepts have been left out. Equipped with the basic concepts presented here, and how they fit into the larger context of machine learning, the reader can find numerous resources for further study, which we provide at the end of the respective chapters. For readers with a mathematical background, this book provides a brief but precisely stated glimpse of machine learning. In contrast to other books that focus on methods and models of machine learning (MacKay, 2003; Bishop, 2006; Alpaydin, 2010; Barber, 2012; Murphy, 2012; Shalev-Shwartz and Ben-David, 2014; Rogers and Girolami, 2016) or programmatic aspects of machine learning (Müller and Guido, 2016; Raschka and Mirjalili, 2017; Chollet and Allaire, 2018), we provide only four representative examples of machine learning algorithms. Instead, we focus on the mathematical concepts behind the models themselves. We hope that readers will be able to gain a deeper understanding of the basic questions in machine learning and connect practical questions arising from the use of machine learning with fundamental choices in the mathematical model.

We do not aim to write a classical machine learning book. Instead, our intention is to provide the mathematical background, applied to four central machine learning problems, to make it easier to read other machine learning textbooks.

Who Is the Target Audience?

As applications of machine learning become widespread in society, we believe that everybody should have some understanding of its underlying principles. This book is written in an academic mathematical style, which enables us to be precise about the concepts behind machine learning. We encourage readers unfamiliar with this seemingly terse style to persevere and to keep the goals of each topic in mind. We sprinkle comments and remarks throughout the text, in the hope that it provides useful guidance with respect to the big picture.

The book assumes the reader to have mathematical knowledge commonly

covered in high school mathematics and physics. For example, the reader should have seen derivatives and integrals before, and geometric vectors in two or three dimensions. Starting from there, we generalize these concepts. Therefore, the target audience of the book includes undergraduate university students, evening learners and learners participating in online machine learning courses.

In analogy to music, there are three types of interaction that people have with machine learning:

Astute Listener The democratization of machine learning by the provision of open-source software, online tutorials and cloud-based tools allows users to not worry about the specifics of pipelines. Users can focus on extracting insights from data using off-the-shelf tools. This enables non-tech-savvy domain experts to benefit from machine learning. This is similar to listening to music; the user is able to choose and discern between different types of machine learning, and benefits from it. More experienced users are like music critics, asking important questions about the application of machine learning in society such as ethics, fairness, and privacy of the individual. We hope that this book provides a foundation for thinking about the certification and risk management of machine learning systems, and allows them to use their domain expertise to build better machine learning systems.

Experienced Artist Skilled practitioners of machine learning can plug and play different tools and libraries into an analysis pipeline. The stereotypical practitioner would be a data scientist or engineer who understands machine learning interfaces and their use cases, and is able to perform wonderful feats of prediction from data. This is similar to a virtuoso playing music, where highly skilled practitioners can bring existing instruments to life and bring enjoyment to their audience. Using the mathematics presented here as a primer, practitioners would be able to understand the benefits and limits of their favorite method, and to extend and generalize existing machine learning algorithms. We hope that this book provides the impetus for more rigorous and principled development of machine learning methods.

Fledgling Composer As machine learning is applied to new domains, developers of machine learning need to develop new methods and extend existing algorithms. They are often researchers who need to understand the mathematical basis of machine learning and uncover relationships between different tasks. This is similar to composers of music who, within the rules and structure of musical theory, create new and amazing pieces. We hope this book provides a high-level overview of other technical books for people who want to become composers of machine learning. There is a great need in society for new researchers who are able to propose and explore novel approaches for attacking the many challenges of learning from data.

Acknowledgments

We are grateful to many people who looked at early drafts of the book and suffered through painful expositions of concepts. We tried to implement their ideas that we did not vehemently disagree with. We would like to especially acknowledge Christfried Webers for his careful reading of many parts of the book, and his detailed suggestions on structure and presentation. Many friends and colleagues have also been kind enough to provide their time and energy on different versions of each chapter. We have been lucky to benefit from the generosity of the online community, who have suggested improvements via `https://github.com`, which greatly improved the book.

The following people have found bugs, proposed clarifications and suggested relevant literature, either via `https://github.com` or personal communication. Their names are sorted alphabetically.

Abdul-Ganiy Usman
Adam Gaier
Adele Jackson
Aditya Menon
Alasdair Tran
Aleksandar Krnjaic
Alexander Makrigiorgos
Alfredo Canziani
Ali Shafti
Amr Khalifa
Andrew Tanggara
Angus Gruen
Antal A. Buss
Antoine Toisoul Le Cann
Areg Sarvazyan
Artem Artemev
Artyom Stepanov
Bill Kromydas
Bob Williamson
Boon Ping Lim
Chao Qu
Cheng Li
Chris Sherlock
Christopher Gray
Daniel McNamara
Daniel Wood
Darren Siegel
David Johnston
Dawei Chen
Ellen Broad
Fengkuangtian Zhu
Fiona Condon
Georgios Theodorou
He Xin
Irene Raissa Kameni
Jakub Nabaglo
James Hensman
Jamie Liu
Jean Kaddour
Jean-Paul Ebejer
Jerry Qiang
Jitesh Sindhare
John Lloyd
Jonas Ngnawe
Jon Martin
Justin Hsi
Kai Arulkumaran
Kamil Dreczkowski
Lily Wang
Lionel Tondji Ngoupeyou
Lydia Knüfing
Mahmoud Aslan
Mark Hartenstein
Mark van der Wilk
Markus Hegland
Martin Hewing
Matthew Alger
Matthew Lee

Maximus McCann
Mengyan Zhang
Michael Bennett
Michael Pedersen
Minjeong Shin
Mohammad Malekzadeh
Naveen Kumar
Nico Montali
Oscar Armas
Patrick Henriksen
Patrick Wieschollek
Pattarawat Chormai
Paul Kelly
Petros Christodoulou
Piotr Januszewski
Pranav Subramani
Quyu Kong
Ragib Zaman
Rui Zhang
Ryan-Rhys Griffiths
Salomon Kabongo
Samuel Ogunmola
Sandeep Mavadia
Sarvesh Nikumbh
Sebastian Raschka
Senanayak Sesh Kumar Karri
Seung-Heon Baek
Shahbaz Chaudhary
Shakir Mohamed
Shawn Berry
Sheikh Abdul Raheem Ali
Sheng Xue
Sridhar Thiagarajan
Syed Nouman Hasany
Szymon Brych
Thomas Bühler
Timur Sharapov
Tom Melamed
Vincent Adam
Vincent Dutordoir
Vu Minh
Wasim Aftab
Wen Zhi
Wojciech Stokowiec
Xiaonan Chong
Xiaowei Zhang
Yazhou Hao
Yicheng Luo
Young Lee
Yu Lu
Yun Cheng
Yuxiao Huang
Zac Cranko
Zijian Cao
Zoe Nolan

Contributors through GitHub, whose real names were not listed on their GitHub profile, are:

SamDataMad	insad	empet
bumptiousmonkey	HorizonP	victorBigand
idoamihai	cs-maillist	17SKYE
deepakiim	kudo23	jessjing1995

We are also very grateful to Parameswaran Raman and the many anonymous reviewers, organized by Cambridge University Press, who read one or more chapters of earlier versions of the manuscript, and provided constructive criticism that led to considerable improvements. A special mention goes to Dinesh Singh Negi, our LaTeX support, for detailed and prompt advice about LaTeX-related issues. Last but not least, we are very grateful to our editor Lauren Cowles, who has been patiently guiding us through the gestation process of this book.

Table of Symbols

Symbol	Typical meaning
$a, b, c, \alpha, \beta, \gamma$	Scalars are lowercase
$\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$	Vectors are bold lowercase
$\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$	Matrices are bold uppercase
$\boldsymbol{x}^\top, \boldsymbol{A}^\top$	Transpose of a vector or matrix
$\boldsymbol{A}^{-1}$	Inverse of a matrix
$\langle \boldsymbol{x}, \boldsymbol{y} \rangle$	Inner product of $\boldsymbol{x}$ and $\boldsymbol{y}$
$\boldsymbol{x}^\top \boldsymbol{y}$	Dot product of $\boldsymbol{x}$ and $\boldsymbol{y}$
$B = (\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3)$	(Ordered) tuple
$\boldsymbol{B} = [\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3]$	Matrix of column vectors stacked horizontally
$\mathcal{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3\}$	Set of vectors (unordered)
$\mathbb{Z}, \mathbb{N}$	Integers and natural numbers, respectively
$\mathbb{R}, \mathbb{C}$	Real and complex numbers, respectively
$\mathbb{R}^n$	n-dimensional vector space of real numbers
$\forall x$	Universal quantifier: for all x
$\exists x$	Existential quantifier: there exists x
$a := b$	a is defined as b
$a =: b$	b is defined as a
$a \propto b$	a is proportional to b, i.e., $a =$ constant $\cdot\, b$
$g \circ f$	Function composition: "g after f"
$\iff$	If and only if
$\implies$	Implies
$\mathcal{A}, \mathcal{C}$	Sets
$a \in \mathcal{A}$	a is an element of set $\mathcal{A}$
$\emptyset$	Empty set
$\mathcal{A} \backslash \mathcal{B}$	$\mathcal{A}$ without $\mathcal{B}$: the set of elements in $\mathcal{A}$ but not in $\mathcal{B}$
D	Number of dimensions; indexed by $d = 1, \ldots, D$
N	Number of data points; indexed by $n = 1, \ldots, N$
$\boldsymbol{I}_m$	Identity matrix of size $m \times m$
$\boldsymbol{0}_{m,n}$	Matrix of zeros of size $m \times n$
$\boldsymbol{1}_{m,n}$	Matrix of ones of size $m \times n$
$\boldsymbol{e}_i$	Standard/canonical vector (where i is the component that is 1)
$\dim$	Dimensionality of vector space
$\mathrm{rk}(\boldsymbol{A})$	Rank of matrix $\boldsymbol{A}$
$\mathrm{Im}(\Phi)$	Image of linear mapping Φ
$\ker(\Phi)$	Kernel (null space) of a linear mapping Φ
$\mathrm{span}[\boldsymbol{b}_1]$	Span (generating set) of $\boldsymbol{b}_1$
$\mathrm{tr}(\boldsymbol{A})$	Trace of $\boldsymbol{A}$
$\det(\boldsymbol{A})$	Determinant of $\boldsymbol{A}$
$\lvert \cdot \rvert$	Absolute value or determinant (depending on context)
$\lVert \cdot \rVert$	Norm; Euclidean, unless specified
λ	Eigenvalue or Lagrange multiplier
E_λ	Eigenspace corresponding to eigenvalue λ

Symbol	Typical meaning
$\boldsymbol{x} \perp \boldsymbol{y}$	Vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ are orthogonal
V	Vector space
$V^\perp$	Orthogonal complement of vector space V
$\sum_{n=1}^{N} x_n$	Sum of the x_n: $x_1 + \ldots + x_N$
$\prod_{n=1}^{N} x_n$	Product of the x_n: $x_1 \cdot \ldots \cdot x_N$
$\boldsymbol{\theta}$	Parameter vector
$\frac{\partial f}{\partial x}$	Partial derivative of f with respect to x
$\frac{\mathrm{d}f}{\mathrm{d}x}$	Total derivative of f with respect to x
∇	Gradient
$f_* = \min_x f(x)$	The smallest function value of f
$x_* \in \arg\min_x f(x)$	The value x_* that minimizes f (note: $\arg\min$ returns a set of values)
$\mathfrak{L}$	Lagrangian
$\mathcal{L}$	Negative log-likelihood
$\binom{n}{k}$	Binomial coefficient, n choose k
$\mathbb{V}_X[\boldsymbol{x}]$	Variance of $\boldsymbol{x}$ with respect to the random variable X
$\mathbb{E}_X[\boldsymbol{x}]$	Expectation of $\boldsymbol{x}$ with respect to the random variable X
$\mathrm{Cov}_{X,Y}[\boldsymbol{x}, \boldsymbol{y}]$	Covariance between $\boldsymbol{x}$ and $\boldsymbol{y}$.
$X \perp\!\!\!\perp Y \mid Z$	X is conditionally independent of Y given Z
$X \sim p$	Random variable X is distributed according to p
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
$\mathrm{Ber}(\mu)$	Bernoulli distribution with parameter μ
$\mathrm{Bin}(N, \mu)$	Binomial distribution with parameters N, μ
$\mathrm{Beta}(\alpha, \beta)$	Beta distribution with parameters α, β

Table of Abbreviations and Acronyms

Acronym	Meaning
e.g.	Exempli gratia (Latin: for example)
GMM	Gaussian mixture model
i.e.	Id est (Latin: this means)
i.i.d.	Independent, identically distributed
MAP	Maximum a posteriori
MLE	Maximum likelihood estimation/estimator
ONB	Orthonormal basis
PCA	Principal component analysis
PPCA	Probabilistic principal component analysis
REF	Row-echelon form
SPD	Symmetric, positive definite
SVM	Support vector machine

Part I

Mathematical Foundations

1

Introduction and Motivation

Machine learning is about designing algorithms that automatically extract valuable information from data. The emphasis here is on "automatic", i.e., machine learning is concerned about general-purpose methodologies that can be applied to many datasets, while producing something that is meaningful. There are three concepts that are at the core of machine learning: data, a model, and learning.

Since machine learning is inherently data driven, *data* is at the core of machine learning. The goal of machine learning is to design general-purpose methodologies to extract valuable patterns from data, ideally without much domain-specific expertise. For example, given a large corpus of documents (e.g., books in many libraries), machine learning methods can be used to automatically find relevant topics that are shared across documents (Hoffman et al., 2010). To achieve this goal, we design *models* that are typically related to the process that generates data, similar to the dataset we are given. For example, in a regression setting, the model would describe a function that maps inputs to real-valued outputs. To paraphrase Mitchell (1997): A model is said to learn from data if its performance on a given task improves after the data is taken into account. The goal is to find good models that generalize well to yet unseen data, which we may care about in the future. *Learning* can be understood as a way to automatically find patterns and structure in data by optimizing the parameters of the model.

data

model

learning

While machine learning has seen many success stories, and software is readily available to design and train rich and flexible machine learning systems, we believe that the mathematical foundations of machine learning are important in order to understand fundamental principles upon which more complicated machine learning systems are built. Understanding these principles can facilitate creating new machine learning solutions, understanding and debugging existing approaches, and learning about the inherent assumptions and limitations of the methodologies we are working with.

1.1 Finding Words for Intuitions

A challenge we face regularly in machine learning is that concepts and words are slippery, and a particular component of the machine learning system can be abstracted to different mathematical concepts. For example, the word "algorithm" is used in at least two different senses in the context of machine learning. In the first sense, we use the phrase "machine learning algorithm" to mean a system that makes predictions based on input data. We refer to these algorithms as *predictors*. In the second sense, we use the exact same phrase "machine learning algorithm" to mean a system that adapts some internal parameters of the predictor so that it performs well on future unseen input data. Here we refer to this adaptation as *training* a system.

predictor

training

This book will not resolve the issue of ambiguity, but we want to highlight upfront that, depending on the context, the same expressions can mean different things. However, we attempt to make the context sufficiently clear to reduce the level of ambiguity.

The first part of this book introduces the mathematical concepts and foundations needed to talk about the three main components of a machine learning system: data, models, and learning. We will briefly outline these components here, and we will revisit them again in Chapter 8 once we have discussed the necessary mathematical concepts.

While not all data is numerical, it is often useful to consider data in a number format. In this book, we assume that *data* has already been appropriately converted into a numerical representation suitable for reading into a computer program. Therefore, we think of data as vectors. As another illustration of how subtle words are, there are (at least) three different ways to think about vectors: a vector as an array of numbers (a computer science view), a vector as an arrow with a direction and magnitude (a physics view), and a vector as an object that obeys addition and scaling (a mathematical view).

data as vectors

model

A *model* is typically used to describe a process for generating data, similar to the dataset at hand. Therefore, good models can also be thought of as simplified versions of the real (unknown) data-generating process, capturing aspects that are relevant for modeling the data and extracting hidden patterns from it. A good model can then be used to predict what would happen in the real world without performing real-world experiments.

model
prediction
hidden patterns
unseen data
data generation

learning

We now come to the crux of the matter, the *learning* component of machine learning. Assume we are given a dataset and a suitable model. *Training* the model means to use the data available to optimize some parameters of the model with respect to a utility function that evaluates how well the model predicts the training data. Most training methods can be thought of as an approach analogous to climbing a hill to reach its peak. In this analogy, the peak of the hill corresponds to a maximum of some

desired performance measure. However, in practice, we are interested in the model to perform well on unseen data. Performing well on data that we have already seen (training data) may only mean that we found a good way to memorize the data. However, this may not generalize well to unseen data, and, in practical applications, we often need to expose our machine learning system to situations that it has not encountered before.

Let us summarize the main concepts of machine learning that we cover in this book:

- We represent data as vectors.
- We choose an appropriate model, either using the probabilistic or optimization view.
- We learn from available data by using numerical optimization methods with the aim that the model performs well on data not used for training.

1.2 Two Ways to Read This Book

We can consider two strategies for understanding the mathematics for machine learning:

- **Bottom-up:** Building up the concepts from foundational to more advanced. This is often the preferred approach in more technical fields, such as mathematics. This strategy has the advantage that the reader at all times is able to rely on their previously learned concepts. Unfortunately, for a practitioner many of the foundational concepts are not particularly interesting by themselves, and the lack of motivation means that most foundational definitions are quickly forgotten.
- **Top-down:** Drilling down from practical needs to more basic requirements. This goal-driven approach has the advantage that the readers know at all times why they need to work on a particular concept, and there is a clear path of required knowledge. The downside of this strategy is that the knowledge is built on potentially shaky foundations, and the readers have to remember a set of words that they do not have any way of understanding.

We decided to write this book in a modular way to separate foundational (mathematical) concepts from applications so that this book can be read in both ways. The book is split into two parts, where Part I lays the mathematical foundations and Part II applies the concepts from Part I to a set of fundamental machine learning problems, which form four pillars of machine learning as illustrated in Figure 1.1: regression, dimensionality reduction, density estimation, and classification. Chapters in Part I mostly build upon the previous ones, but it is possible to skip a chapter and work backward if necessary. Chapters in Part II are only loosely coupled and can be read in any order. There are many pointers forward and backward

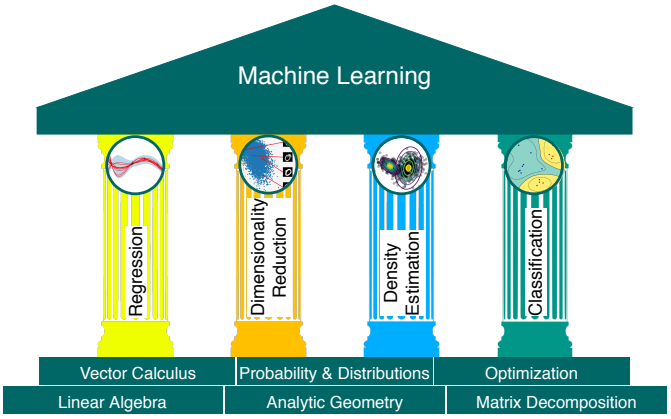


Figure 1.1 The foundations and four pillars of machine learning.

between the two parts of the book to link mathematical concepts with machine learning algorithms.

Of course there are more than two ways to read this book. Most readers learn using a combination of top-down and bottom-up approaches, sometimes building up basic mathematical skills before attempting more complex concepts, but also choosing topics based on applications of machine learning.

Part I Is about Mathematics

The four pillars of machine learning we cover in this book (see Figure 1.1) require a solid mathematical foundation, which is laid out in Part I.

We represent numerical data as vectors and represent a table of such data as a matrix. The study of vectors and matrices is called *linear algebra*, which we introduce in Chapter 2. The collection of vectors as a matrix is also described there.

linear algebra

Given two vectors representing two objects in the real world, we want to make statements about their similarity. The idea is that vectors that are similar should be predicted to have similar outputs by our machine learning algorithm (our predictor). To formalize the idea of similarity between vectors, we need to introduce operations that take two vectors as input and return a numerical value representing their similarity. The construction of similarity and distances is central to *analytic geometry* and is discussed in Chapter 3.

measure similarity (as a numerical value)

analytic geometry

In Chapter 4, we introduce some fundamental concepts about matrices and *matrix decomposition*. Some operations on matrices are extremely useful in machine learning, and they allow for an intuitive interpretation of the data and more efficient learning.

matrix decomposition

We often consider data to be noisy observations of some true underlying signal. We hope that by applying machine learning we can identify the signal from the noise. This requires us to have a language for quantifying what "noise" means. We often would also like to have predictors that

allow us to express some sort of uncertainty, e.g., to quantify the confidence we have about the value of the prediction at a particular test data point. Quantification of uncertainty is the realm of *probability theory* and is covered in Chapter 6.

probability theory

To train machine learning models, we typically find parameters that maximize some performance measure. Many optimization techniques require the concept of a gradient, which tells us the direction in which to search for a solution. Chapter 5 is about *vector calculus* and details the concept of gradients, which we subsequently use in Chapter 7, where we talk about *optimization* to find maxima/minima of functions.

vector calculus

optimization

Part II Is about Machine Learning

The second part of the book introduces *four pillars of machine learning* as shown in Figure 1.1. We illustrate how the mathematical concepts introduced in the first part of the book are the foundation for each pillar. Broadly speaking, chapters are ordered by difficulty (in ascending order).

In Chapter 8, we restate the three components of machine learning (data, models, and parameter estimation) in a mathematical fashion. In addition, we provide some guidelines for building experimental set-ups that guard against overly optimistic evaluations of machine learning systems. Recall that the goal is to build a predictor that performs well on unseen data.

In Chapter 9, we will have a close look at *linear regression*, where our objective is to find functions that map inputs $\boldsymbol{x} \in \mathbb{R}^D$ to corresponding observed function values $y \in \mathbb{R}$, which we can interpret as the labels of their respective inputs. We will discuss classical model fitting (parameter estimation) via maximum likelihood and maximum a posteriori estimation, as well as Bayesian linear regression, where we integrate the parameters out instead of optimizing them.

linear regression

Chapter 10 focuses on *dimensionality reduction*, the second pillar in Figure 1.1, using principal component analysis. The key objective of dimensionality reduction is to find a compact, lower-dimensional representation of high-dimensional data $\boldsymbol{x} \in \mathbb{R}^D$, which is often easier to analyze than the original data. Unlike regression, dimensionality reduction is only concerned about modeling the data – there are no labels associated with a data point $\boldsymbol{x}$.

dimensionality reduction

In Chapter 11, we will move to our third pillar: *density estimation*. The objective of density estimation is to find a probability distribution that describes a given dataset. We will focus on Gaussian mixture models for this purpose, and we will discuss an iterative scheme to find the parameters of this model. As in dimensionality reduction, there are no labels associated with the data points $\boldsymbol{x} \in \mathbb{R}^D$. However, we do not seek a low-dimensional representation of the data. Instead, we are interested in a density model that describes the data.

density estimation

Chapter 12 concludes the book with an in-depth discussion of the fourth

classification

pillar: *classification*. We will discuss classification in the context of support vector machines. Similar to regression (Chapter 9), we have inputs $\boldsymbol{x}$ and corresponding labels y. However, unlike regression, where the labels were real-valued, the labels in classification are integers, which requires special care.

1.3 Exercises and Feedback

We provide some exercises in Part I, which can be done mostly by pen and paper. For Part II, we provide programming tutorials (jupyter notebooks) to explore some properties of the machine learning algorithms we discuss in this book.

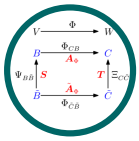
We appreciate that Cambridge University Press strongly supports our aim to democratize education and learning by making this book freely available for download at

`https://mml-book.com`

where tutorials, errata, and additional materials can be found. Mistakes can be reported and feedback provided using the preceding URL.

2

Linear Algebra

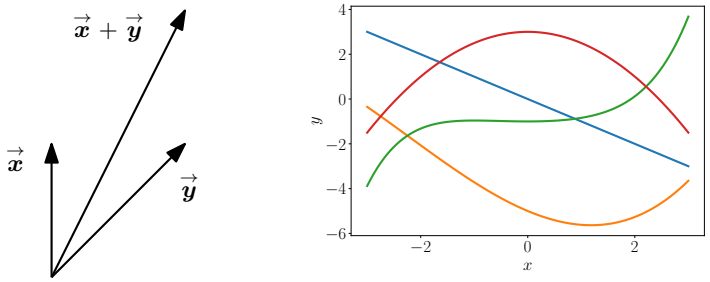


When formalizing intuitive concepts, a common approach is to construct a set of objects (symbols) and a set of rules to manipulate these objects. This is known as an *algebra*. Linear algebra is the study of vectors and certain rules to manipulate vectors. The vectors many of us know from school are called "geometric vectors", which are usually denoted by a small arrow above the letter, e.g., $\vec{x}$ and $\vec{y}$. In this book, we discuss more general concepts of vectors and use a bold letter to represent them, e.g., $\boldsymbol{x}$ and $\boldsymbol{y}$.

algebra

In general, vectors are special objects that can be added together and multiplied by scalars to produce another object of the same kind. From an abstract mathematical viewpoint, any object that satisfies these two properties can be considered a vector. Here are some examples of such vector objects:

1. Geometric vectors. This example of a vector may be familiar from high school mathematics and physics. Geometric vectors – see Figure 2.1(a) – are directed segments, which can be drawn (at least in two dimensions). Two geometric vectors $\vec{\boldsymbol{x}}, \vec{\boldsymbol{y}}$ can be added, such that $\vec{\boldsymbol{x}} + \vec{\boldsymbol{y}} = \vec{\boldsymbol{z}}$ is another geometric vector. Furthermore, multiplication by a scalar $\lambda \vec{\boldsymbol{x}}$, $\lambda \in \mathbb{R}$, is also a geometric vector. In fact, it is the original vector scaled by λ. Therefore, geometric vectors are instances of the vector concepts introduced previously. Interpreting vectors as geometric vectors enables us to use our intuitions about direction and magnitude to reason about mathematical operations.
2. Polynomials are also vectors; see Figure 2.1(b): Two polynomials can



(a) Geometric vectors.

(b) Polynomials.

Figure 2.1 Different types of vectors. Vectors can be surprising objects, including (a) geometric vectors and (b) polynomials.

be added together, which results in another polynomial; and they can be multiplied by a scalar $\lambda \in \mathbb{R}$, and the result is a polynomial as well. Therefore, polynomials are (rather unusual) instances of vectors. Note that polynomials are very different from geometric vectors. While geometric vectors are concrete "drawings", polynomials are abstract concepts. However, they are both vectors in the sense previously described.

3. Audio signals are vectors. Audio signals are represented as a series of numbers. We can add audio signals together, and their sum is a new audio signal. If we scale an audio signal, we also obtain an audio signal. Therefore, audio signals are a type of vector, too.
4. Elements of $\mathbb{R}^n$ (tuples of n real numbers) are vectors. $\mathbb{R}^n$ is more abstract than polynomials, and it is the concept we focus on in this book. For instance,

$$\boldsymbol{a} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \in \mathbb{R}^3 \tag{2.1}$$

is an example of a triplet of numbers. Adding two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^n$ component-wise results in another vector: $\boldsymbol{a} + \boldsymbol{b} = \boldsymbol{c} \in \mathbb{R}^n$. Moreover, multiplying $\boldsymbol{a} \in \mathbb{R}^n$ by $\lambda \in \mathbb{R}$ results in a scaled vector $\lambda\boldsymbol{a} \in \mathbb{R}^n$. Considering vectors as elements of $\mathbb{R}^n$ has an additional benefit that it loosely corresponds to arrays of real numbers on a computer. Many programming languages support array operations, which allow for convenient implementation of algorithms that involve vector operations.

Be careful to check whether array operations actually perform vector operations when implementing on a computer.

Linear algebra focuses on the similarities between these vector concepts. We can add them together and multiply them by scalars. We will largely focus on vectors in $\mathbb{R}^n$ since most algorithms in linear algebra are formulated in $\mathbb{R}^n$. We will see in Chapter 8 that we often consider data to be represented as vectors in $\mathbb{R}^n$. In this book, we will focus on finite-dimensional vector spaces, in which case there is a 1:1 correspondence between any kind of vector and $\mathbb{R}^n$. When it is convenient, we will use intuitions about geometric vectors and consider array-based algorithms.

Pavel Grinfeld's series on linear algebra: `http://tinyurl.com/nahclwm`
Gilbert Strang's course on linear algebra: `http://tinyurl.com/bdfbu8s5`
3Blue1Brown series on linear algebra: `https://tinyurl.com/h5g4kps`

One major idea in mathematics is the idea of "closure". This is the question: What is the set of all things that can result from my proposed operations? In the case of vectors: What is the set of vectors that can result by starting with a small set of vectors, and adding them to each other and scaling them? This results in a vector space (Section 2.4). The concept of a vector space and its properties underlie much of machine learning. The concepts introduced in this chapter are summarized in Figure 2.2.

This chapter is mostly based on the lecture notes and books by Drumm and Weil (2001), Strang (2003), Hogben (2013), Liesen and Mehrmann (2015), as well as Pavel Grinfeld's Linear Algebra series. Other excellent

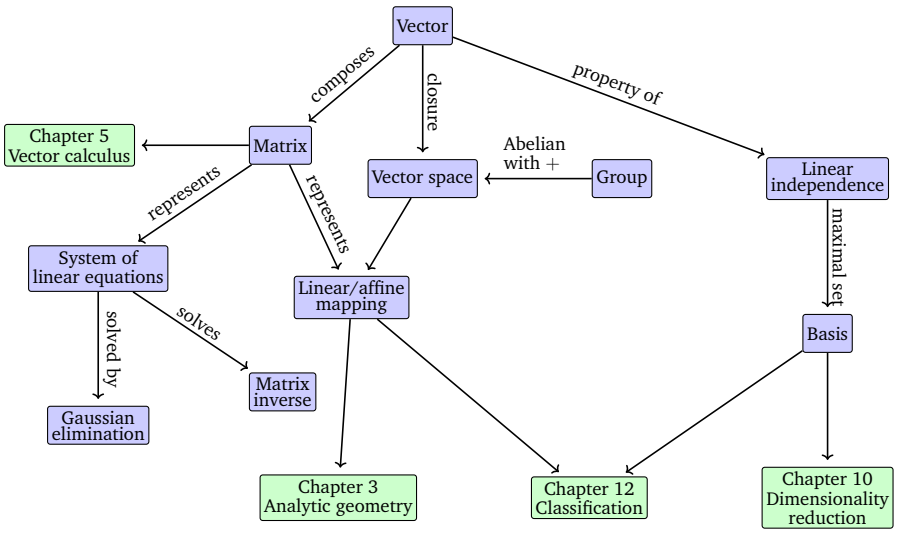


Figure 2.2 A mind map of the concepts introduced in this chapter, along with where they are used in other parts of the book.

resources are Gilbert Strang's Linear Algebra course at MIT and the Linear Algebra Series by 3Blue1Brown.

Linear algebra plays an important role in machine learning and general mathematics. The concepts introduced in this chapter are further expanded to include the idea of geometry in Chapter 3. In Chapter 5, we will discuss vector calculus, where a principled knowledge of matrix operations is essential. In Chapter 10, we will use projections (to be introduced in Section 3.8) for dimensionality reduction with principal component analysis (PCA). In Chapter 9, we will discuss linear regression, where linear algebra plays a central role for solving least-squares problems.

2.1 Systems of Linear Equations

Systems of linear equations play a central part of linear algebra. Many problems can be formulated as systems of linear equations, and linear algebra gives us the tools for solving them.

Example 2.1

A company produces products $N_1, \ldots, N_n$ for which resources $R_1, \ldots, R_m$ are required. To produce a unit of product N_j, a_{ij} units of resource R_i are needed, where $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

The objective is to find an optimal production plan, i.e., a plan of how many units x_j of product N_j should be produced if a total of b_i units of resource R_i are available and (ideally) no resources are left over.

If we produce $x_1, \ldots, x_n$ units of the corresponding products, we need

a total of

$$a_{i1}x_1 + \cdots + a_{in}x_n \tag{2.2}$$

many units of resource R_i. An optimal production plan $(x_1, \ldots, x_n) \in \mathbb{R}^n$, therefore, has to satisfy the following system of equations:

$$\begin{aligned} a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\ &\vdots \\ a_{m1}x_1 + \cdots + a_{mn}x_n &= b_m \end{aligned}, \tag{2.3}$$

where $a_{ij} \in \mathbb{R}$ and $b_i \in \mathbb{R}$.

system of linear equations

solution

Equation (2.3) is the general form of a *system of linear equations*, and $x_1, \ldots, x_n$ are the *unknowns* of this system. Every n-tuple $(x_1, \ldots, x_n) \in \mathbb{R}^n$ that satisfies (2.3) is a *solution* of the linear equation system.

Example 2.2
The system of linear equations

$$\begin{array}{rcrcrcll} x_1 & + & x_2 & + & x_3 & = & 3 & (1) \\ x_1 & - & x_2 & + & 2x_3 & = & 2 & (2) \\ 2x_1 & & & + & 3x_3 & = & 1 & (3) \end{array} \tag{2.4}$$

has *no solution:* Adding the first two equations yields $2x_1 + 3x_3 = 5$, which contradicts the third equation (3).

Let us have a look at the system of linear equations

$$\begin{array}{rcrcrcll} x_1 & + & x_2 & + & x_3 & = & 3 & (1) \\ x_1 & - & x_2 & + & 2x_3 & = & 2 & (2) \\ & & x_2 & + & x_3 & = & 2 & (3) \end{array} . \tag{2.5}$$

From the first and third equation, it follows that $x_1 = 1$. From (1)+(2), we get $2x_1 + 3x_3 = 5$, i.e., $x_3 = 1$. From (3), we then get that $x_2 = 1$. Therefore, $(1, 1, 1)$ is the only possible and *unique solution* (verify that $(1, 1, 1)$ is a solution by plugging in).

As a third example, we consider

$$\begin{array}{rcrcrcll} x_1 & + & x_2 & + & x_3 & = & 3 & (1) \\ x_1 & - & x_2 & + & 2x_3 & = & 2 & (2) \\ 2x_1 & & & + & 3x_3 & = & 5 & (3) \end{array} . \tag{2.6}$$

Since (1)+(2)=(3), we can omit the third equation (redundancy). From (1) and (2), we get $2x_1 = 5 - 3x_3$ and $2x_2 = 1 + x_3$. We define $x_3 = a \in \mathbb{R}$ as a free variable, such that any triplet

$$\left(\frac{5}{2} - \frac{3}{2}a, \frac{1}{2} + \frac{1}{2}a, a\right), \quad a \in \mathbb{R} \tag{2.7}$$

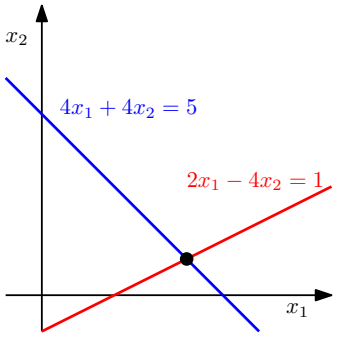


Figure 2.3 The solution space of a system of two linear equations with two variables can be geometrically interpreted as the intersection of two lines. Every linear equation represents a line.

is a solution of the system of linear equations, i.e., we obtain a solution set that contains *infinitely many* solutions.

In general, for a real-valued system of linear equations we obtain either no, exactly one, or infinitely many solutions. Linear regression (Chapter 9) solves a version of Example 2.1 when we cannot solve the system of linear equations.

Remark (Geometric Interpretation of Systems of Linear Equations). In a system of linear equations with two variables x_1, x_2, each linear equation defines a line on the x_1x_2-plane. Since a solution to a system of linear equations must satisfy all equations simultaneously, the solution set is the intersection of these lines. This intersection set can be a line (if the linear equations describe the same line), a point, or empty (when the lines are parallel). An illustration is given in Figure 2.3 for the system

$$\begin{aligned} 4x_1 + 4x_2 &= 5 \\ 2x_1 - 4x_2 &= 1 \end{aligned} \tag{2.8}$$

where the solution space is the point $(x_1, x_2) = (1, \frac{1}{4})$. Similarly, for three variables, each linear equation determines a plane in three-dimensional space. When we intersect these planes, i.e., satisfy all linear equations at the same time, we can obtain a solution set that is a plane, a line, a point or empty (when the planes have no common intersection). ♢

For a systematic approach to solving systems of linear equations, we will introduce a useful compact notation. We collect the coefficients a_{ij} into vectors and collect the vectors into matrices. In other words, we write the system from (2.3) in the following form:

$$\begin{bmatrix} a_{11} \\ \vdots \\ a_{m1} \end{bmatrix} x_1 + \begin{bmatrix} a_{12} \\ \vdots \\ a_{m2} \end{bmatrix} x_2 + \cdots + \begin{bmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{bmatrix} x_n = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix} \tag{2.9}$$

$$\iff \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}. \tag{2.10}$$

In the following, we will have a close look at these *matrices* and define computation rules. We will return to solving linear equations in Section 2.3.

2.2 Matrices

Matrices play a central role in linear algebra. They can be used to compactly represent systems of linear equations, but they also represent linear functions (linear mappings) as we will see later in Section 2.7. Before we discuss some of these interesting topics, let us first define what a matrix is and what kind of operations we can do with matrices. We will see more properties of matrices in Chapter 4.

matrix

Definition 2.1 (Matrix)**.** With $m, n \in \mathbb{N}$ a real-valued (m, n) *matrix* $\boldsymbol{A}$ is an $m \cdot n$-tuple of elements a_{ij}, $i = 1, \ldots, m$, $j = 1, \ldots, n$, which is ordered according to a rectangular scheme consisting of m rows and n columns:

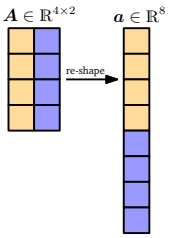
$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad a_{ij} \in \mathbb{R}. \tag{2.11}$$

row
column
row vector
column vector

By convention $(1, n)$-matrices are called *rows* and $(m, 1)$-matrices are called *columns*. These special matrices are also called *row/column vectors*.

$\mathbb{R}^{m \times n}$ is the set of all real-valued (m, n)-matrices. $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ can be equivalently represented as $\boldsymbol{a} \in \mathbb{R}^{mn}$ by stacking all n columns of the matrix into a long vector; see Figure 2.4.

Figure 2.4 By stacking its columns, a matrix $\boldsymbol{A}$ can be represented as a long vector $\boldsymbol{a}$.



Note the size of the matrices.

```
C = np.einsum('il, lj', A, B)
```

2.2.1 Matrix Addition and Multiplication

The sum of two matrices $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{B} \in \mathbb{R}^{m \times n}$ is defined as the element-wise sum, i.e.,

$$\boldsymbol{A} + \boldsymbol{B} := \begin{bmatrix} a_{11} + b_{11} & \cdots & a_{1n} + b_{1n} \\ \vdots & & \vdots \\ a_{m1} + b_{m1} & \cdots & a_{mn} + b_{mn} \end{bmatrix} \in \mathbb{R}^{m \times n}. \tag{2.12}$$

For matrices $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{B} \in \mathbb{R}^{n \times k}$, the elements c_{ij} of the product $\boldsymbol{C} = \boldsymbol{A}\boldsymbol{B} \in \mathbb{R}^{m \times k}$ are computed as

$$c_{ij} = \sum_{l=1}^{n} a_{il} b_{lj}, \qquad i = 1, \ldots, m, \quad j = 1, \ldots, k. \tag{2.13}$$

This means, to compute element c_{ij} we multiply the elements of the ith row of $\boldsymbol{A}$ with the jth column of $\boldsymbol{B}$ and sum them up. Later in Section 3.2, we will call this the *dot product* of the corresponding row and column. In cases, where we need to be explicit that we are performing multiplication, we use the notation $\boldsymbol{A} \cdot \boldsymbol{B}$ to denote multiplication (explicitly showing "·").

There are n columns in $\boldsymbol{A}$ and n rows in $\boldsymbol{B}$ so that we can compute $a_{il}b_{lj}$ for $l = 1, \ldots, n$.

Commonly, the dot product between two vectors $\boldsymbol{a}, \boldsymbol{b}$ is denoted by $\boldsymbol{a}^\top \boldsymbol{b}$ or $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$.

Remark. Matrices can only be multiplied if their "neighboring" dimensions match. For instance, an $n \times k$-matrix $\boldsymbol{A}$ can be multiplied with a $k \times m$-matrix $\boldsymbol{B}$, but only from the left side:

$$\underbrace{\boldsymbol{A}}_{n\times k}\underbrace{\boldsymbol{B}}_{k\times m} = \underbrace{\boldsymbol{C}}_{n\times m} \tag{2.14}$$

The product $\boldsymbol{BA}$ is not defined if $m \neq n$ since the neighboring dimensions do not match. ♢

Remark. Matrix multiplication is *not* defined as an element-wise operation on matrix elements, i.e., $c_{ij} \neq a_{ij}b_{ij}$ (even if the size of $\boldsymbol{A}, \boldsymbol{B}$ was chosen appropriately). This kind of element-wise multiplication often appears in programming languages when we multiply (multi-dimensional) arrays with each other, and is called a *Hadamard product*. ♢

Hadamard product

Example 2.3

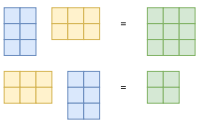
For $\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix} \in \mathbb{R}^{2\times 3}$, $\boldsymbol{B} = \begin{bmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 2}$, we obtain

$$\boldsymbol{AB} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 3 \\ 2 & 5 \end{bmatrix} \in \mathbb{R}^{2\times 2}, \tag{2.15}$$

$$\boldsymbol{BA} = \begin{bmatrix} 0 & 2 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 6 & 4 & 2 \\ -2 & 0 & 2 \\ 3 & 2 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3}. \tag{2.16}$$

From this example, we can already see that matrix multiplication is not commutative, i.e., $\boldsymbol{AB} \neq \boldsymbol{BA}$; see also Figure 2.5 for an illustration.

Figure 2.5 Even if both matrix multiplications $\boldsymbol{AB}$ and $\boldsymbol{BA}$ are defined, the dimensions of the results can be different.



Definition 2.2 (Identity Matrix). In $\mathbb{R}^{n\times n}$, we define the *identity matrix*

$$\boldsymbol{I}_n := \begin{bmatrix} 1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{n\times n} \tag{2.17}$$

identity matrix

as the $n \times n$-matrix containing 1 on the diagonal and 0 everywhere else.

Now that we defined matrix multiplication, matrix addition and the identity matrix, let us have a look at some properties of matrices:

associativity

- *Associativity:*

$$\forall \boldsymbol{A} \in \mathbb{R}^{m \times n}, \boldsymbol{B} \in \mathbb{R}^{n \times p}, \boldsymbol{C} \in \mathbb{R}^{p \times q} : (\boldsymbol{AB})\boldsymbol{C} = \boldsymbol{A}(\boldsymbol{BC}) \tag{2.18}$$

distributivity

- *Distributivity:*

$$\forall \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m \times n}, \boldsymbol{C}, \boldsymbol{D} \in \mathbb{R}^{n \times p} : (\boldsymbol{A} + \boldsymbol{B})\boldsymbol{C} = \boldsymbol{AC} + \boldsymbol{BC} \tag{2.19a}$$

$$\boldsymbol{A}(\boldsymbol{C} + \boldsymbol{D}) = \boldsymbol{AC} + \boldsymbol{AD} \tag{2.19b}$$

- Multiplication with the identity matrix:

$$\forall \boldsymbol{A} \in \mathbb{R}^{m \times n} : \boldsymbol{I}_m \boldsymbol{A} = \boldsymbol{A}\boldsymbol{I}_n = \boldsymbol{A} \tag{2.20}$$

Note that $\boldsymbol{I}_m \neq \boldsymbol{I}_n$ for $m \neq n$.

2.2.2 Inverse and Transpose

A square matrix possesses the same number of columns and rows.

inverse

Definition 2.3 (Inverse)**.** Consider a square matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$. Let matrix $\boldsymbol{B} \in \mathbb{R}^{n \times n}$ have the property that $\boldsymbol{AB} = \boldsymbol{I}_n = \boldsymbol{BA}$. $\boldsymbol{B}$ is called the *inverse* of $\boldsymbol{A}$ and denoted by $\boldsymbol{A}^{-1}$.

regular
invertible
nonsingular
singular
noninvertible

Unfortunately, not every matrix $\boldsymbol{A}$ possesses an inverse $\boldsymbol{A}^{-1}$. If this inverse does exist, $\boldsymbol{A}$ is called *regular/invertible/nonsingular*, otherwise *singular/noninvertible*. When the matrix inverse exists, it is unique. In Section 2.3, we will discuss a general way to compute the inverse of a matrix by solving a system of linear equations.

Remark (Existence of the Inverse of a 2×2-matrix). Consider a matrix

$$\boldsymbol{A} := \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \in \mathbb{R}^{2 \times 2}. \tag{2.21}$$

If we multiply $\boldsymbol{A}$ with

$$\boldsymbol{A}' := \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \tag{2.22}$$

we obtain

$$\boldsymbol{A}\boldsymbol{A}' = \begin{bmatrix} a_{11}a_{22} - a_{12}a_{21} & 0 \\ 0 & a_{11}a_{22} - a_{12}a_{21} \end{bmatrix} = (a_{11}a_{22} - a_{12}a_{21})\boldsymbol{I}. \tag{2.23}$$

Therefore,

$$\boldsymbol{A}^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix} \tag{2.24}$$

if and only if $a_{11}a_{22} - a_{12}a_{21} \neq 0$. In Section 4.1, we will see that $a_{11}a_{22} -$

$a_{12}a_{21}$ is the determinant of a 2×2-matrix. Furthermore, we can generally use the determinant to check whether a matrix is invertible. ♢

Example 2.4 (Inverse Matrix)
The matrices

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 1 \\ 4 & 4 & 5 \\ 6 & 7 & 7 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} -7 & -7 & 6 \\ 2 & 1 & -1 \\ 4 & 5 & -4 \end{bmatrix} \tag{2.25}$$

are inverse to each other since $\boldsymbol{AB} = \boldsymbol{I} = \boldsymbol{BA}$.

Definition 2.4 (Transpose). For $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ the matrix $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ with $b_{ij} = a_{ji}$ is called the *transpose* of $\boldsymbol{A}$. We write $\boldsymbol{B} = \boldsymbol{A}^\top$.

transpose

The main diagonal (sometimes called "principal diagonal", "primary diagonal", "leading diagonal", or "major diagonal") of a matrix $\boldsymbol{A}$ is the collection of entries A_{ij} where $i = j$.

In general, $\boldsymbol{A}^\top$ can be obtained by writing the columns of $\boldsymbol{A}$ as the rows of $\boldsymbol{A}^\top$. The following are important properties of inverses and transposes:

$$\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I} = \boldsymbol{A}^{-1}\boldsymbol{A} \tag{2.26}$$
$$(\boldsymbol{AB})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1} \tag{2.27}$$
$$(\boldsymbol{A}+\boldsymbol{B})^{-1} \neq \boldsymbol{A}^{-1} + \boldsymbol{B}^{-1} \tag{2.28}$$
$$(\boldsymbol{A}^\top)^\top = \boldsymbol{A} \tag{2.29}$$
$$(\boldsymbol{AB})^\top = \boldsymbol{B}^\top\boldsymbol{A}^\top \tag{2.30}$$
$$(\boldsymbol{A}+\boldsymbol{B})^\top = \boldsymbol{A}^\top + \boldsymbol{B}^\top \tag{2.31}$$

The scalar case of (2.28) is $\frac{1}{2+4} = \frac{1}{6} \neq \frac{1}{2} + \frac{1}{4}$.

Definition 2.5 (Symmetric Matrix). A matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is *symmetric* if $\boldsymbol{A} = \boldsymbol{A}^\top$.

symmetric matrix

Note that only (n, n)-matrices can be symmetric. Generally, we call (n, n)-matrices also *square matrices* because they possess the same number of rows and columns. Moreover, if $\boldsymbol{A}$ is invertible, then so is $\boldsymbol{A}^\top$, and $(\boldsymbol{A}^{-1})^\top = (\boldsymbol{A}^\top)^{-1} =: \boldsymbol{A}^{-\top}$.

square matrix

Remark (Sum and Product of Symmetric Matrices). The sum of symmetric matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$ is always symmetric. However, although their product is always defined, it is generally not symmetric:

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}. \tag{2.32}$$

♢

2.2.3 Multiplication by a Scalar

Let us look at what happens to matrices when they are multiplied by a scalar $\lambda \in \mathbb{R}$. Let $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $\lambda \in \mathbb{R}$. Then $\lambda\boldsymbol{A} = \boldsymbol{K}$, $K_{ij} = \lambda a_{ij}$. Practically, λ scales each element of $\boldsymbol{A}$. For $\lambda, \psi \in \mathbb{R}$, the following holds:

associativity

- *Associativity:*
  $(\lambda\psi)\boldsymbol{C} = \lambda(\psi\boldsymbol{C}), \quad \boldsymbol{C} \in \mathbb{R}^{m\times n}$
- $\lambda(\boldsymbol{B}\boldsymbol{C}) = (\lambda\boldsymbol{B})\boldsymbol{C} = \boldsymbol{B}(\lambda\boldsymbol{C}) = (\boldsymbol{B}\boldsymbol{C})\lambda, \quad \boldsymbol{B} \in \mathbb{R}^{m\times n}, \boldsymbol{C} \in \mathbb{R}^{n\times k}$.
  Note that this allows us to move scalar values around.

distributivity

- $(\lambda\boldsymbol{C})^\top = \boldsymbol{C}^\top\lambda^\top = \boldsymbol{C}^\top\lambda = \lambda\boldsymbol{C}^\top$ since $\lambda = \lambda^\top$ for all $\lambda \in \mathbb{R}$.
- *Distributivity:*
  $(\lambda + \psi)\boldsymbol{C} = \lambda\boldsymbol{C} + \psi\boldsymbol{C}, \quad \boldsymbol{C} \in \mathbb{R}^{m\times n}$
  $\lambda(\boldsymbol{B} + \boldsymbol{C}) = \lambda\boldsymbol{B} + \lambda\boldsymbol{C}, \quad \boldsymbol{B}, \boldsymbol{C} \in \mathbb{R}^{m\times n}$

Example 2.5 (Distributivity)
If we define

$$\boldsymbol{C} := \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \tag{2.33}$$

then for any $\lambda, \psi \in \mathbb{R}$ we obtain

$$(\lambda + \psi)\boldsymbol{C} = \begin{bmatrix} (\lambda+\psi)1 & (\lambda+\psi)2 \\ (\lambda+\psi)3 & (\lambda+\psi)4 \end{bmatrix} = \begin{bmatrix} \lambda+\psi & 2\lambda+2\psi \\ 3\lambda+3\psi & 4\lambda+4\psi \end{bmatrix} \tag{2.34a}$$

$$= \begin{bmatrix} \lambda & 2\lambda \\ 3\lambda & 4\lambda \end{bmatrix} + \begin{bmatrix} \psi & 2\psi \\ 3\psi & 4\psi \end{bmatrix} = \lambda\boldsymbol{C} + \psi\boldsymbol{C}\,. \tag{2.34b}$$

2.2.4 Compact Representations of Systems of Linear Equations

If we consider the system of linear equations

$$\begin{aligned} 2x_1 + 3x_2 + 5x_3 &= 1 \\ 4x_1 - 2x_2 - 7x_3 &= 8 \\ 9x_1 + 5x_2 - 3x_3 &= 2 \end{aligned} \tag{2.35}$$

and use the rules for matrix multiplication, we can write this equation system in a more compact form as

$$\begin{bmatrix} 2 & 3 & 5 \\ 4 & -2 & -7 \\ 9 & 5 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 8 \\ 2 \end{bmatrix}. \tag{2.36}$$

Note that x_1 scales the first column, x_2 the second one, and x_3 the third one.

Generally, a system of linear equations can be compactly represented in their matrix form as $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$; see (2.3), and the product $\boldsymbol{A}\boldsymbol{x}$ is a (linear) combination of the columns of $\boldsymbol{A}$. We will discuss linear combinations in more detail in Section 2.5.

2.3 Solving Systems of Linear Equations

In (2.3), we introduced the general form of an equation system, i.e.,

$$\begin{aligned} a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\ &\vdots \\ a_{m1}x_1 + \cdots + a_{mn}x_n &= b_m \,, \end{aligned} \tag{2.37}$$

where $a_{ij} \in \mathbb{R}$ and $b_i \in \mathbb{R}$ are known constants and x_j are unknowns, $i = 1, \ldots, m$, $j = 1, \ldots, n$. Thus far, we saw that matrices can be used as a compact way of formulating systems of linear equations so that we can write $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, see (2.10). Moreover, we defined basic matrix operations, such as addition and multiplication of matrices. In the following, we will focus on solving systems of linear equations and provide an algorithm for finding the inverse of a matrix.

2.3.1 Particular and General Solution

Before discussing how to generally solve systems of linear equations, let us have a look at an example. Consider the system of equations

$$\begin{bmatrix} 1 & 0 & 8 & -4 \\ 0 & 1 & 2 & 12 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 42 \\ 8 \end{bmatrix}. \tag{2.38}$$

The system has two equations and four unknowns. Therefore, in general we would expect infinitely many solutions. This system of equations is in a particularly easy form, where the first two columns consist of a 1 and a 0. Remember that we want to find scalars $x_1, \ldots, x_4$, such that $\sum_{i=1}^{4} x_i \boldsymbol{c}_i = \boldsymbol{b}$, where we define $\boldsymbol{c}_i$ to be the ith column of the matrix and $\boldsymbol{b}$ the right-hand-side of (2.38). A solution to the problem in (2.38) can be found immediately by taking 42 times the first column and 8 times the second column so that

$$\boldsymbol{b} = \begin{bmatrix} 42 \\ 8 \end{bmatrix} = 42 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 8 \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{2.39}$$

Therefore, a solution is $[42, 8, 0, 0]^\top$. This solution is called a *particular solution* or *special solution*. However, this is not the only solution of this system of linear equations. To capture all the other solutions, we need to be creative in generating $\mathbf{0}$ in a non-trivial way using the columns of the matrix: Adding $\mathbf{0}$ to our special solution does not change the special solution. To do so, we express the third column using the first two columns (which are of this very simple form)

particular solution
special solution

$$\begin{bmatrix} 8 \\ 2 \end{bmatrix} = 8 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{2.40}$$

so that $\mathbf{0} = 8\boldsymbol{c}_1 + 2\boldsymbol{c}_2 - 1\boldsymbol{c}_3 + 0\boldsymbol{c}_4$ and $(x_1, x_2, x_3, x_4) = (8, 2, -1, 0)$. In fact, any scaling of this solution by $\lambda_1 \in \mathbb{R}$ produces the $\mathbf{0}$ vector, i.e.,

$$\begin{bmatrix} 1 & 0 & 8 & -4 \\ 0 & 1 & 2 & 12 \end{bmatrix} \left(\lambda_1 \begin{bmatrix} 8 \\ 2 \\ -1 \\ 0 \end{bmatrix} \right) = \lambda_1(8\boldsymbol{c}_1 + 2\boldsymbol{c}_2 - \boldsymbol{c}_3) = \mathbf{0}\,. \tag{2.41}$$

Following the same line of reasoning, we express the fourth column of the matrix in (2.38) using the first two columns and generate another set of non-trivial versions of $\mathbf{0}$ as

$$\begin{bmatrix} 1 & 0 & 8 & -4 \\ 0 & 1 & 2 & 12 \end{bmatrix} \left(\lambda_2 \begin{bmatrix} -4 \\ 12 \\ 0 \\ -1 \end{bmatrix} \right) = \lambda_2(-4\boldsymbol{c}_1 + 12\boldsymbol{c}_2 - \boldsymbol{c}_4) = \mathbf{0} \tag{2.42}$$

for any $\lambda_2 \in \mathbb{R}$. Putting everything together, we obtain all solutions of the equation system in (2.38), which is called the *general solution*, as the set

general solution

$$\left\{ \boldsymbol{x} \in \mathbb{R}^4 : \boldsymbol{x} = \begin{bmatrix} 42 \\ 8 \\ 0 \\ 0 \end{bmatrix} + \lambda_1 \begin{bmatrix} 8 \\ 2 \\ -1 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} -4 \\ 12 \\ 0 \\ -1 \end{bmatrix}, \lambda_1, \lambda_2 \in \mathbb{R} \right\}. \tag{2.43}$$

Remark. The general approach we followed consisted of the following three steps:

1. Find a particular solution to $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.
2. Find all solutions to $\boldsymbol{A}\boldsymbol{x} = \mathbf{0}$.
3. Combine the solutions from steps 1. and 2. to the general solution.

Neither the general nor the particular solution is unique. ♢

The system of linear equations in the preceding example was easy to solve because the matrix in (2.38) has this particularly convenient form, which allowed us to find the particular and the general solution by inspection. However, general equation systems are not of this simple form. Fortunately, there exists a constructive algorithmic way of transforming any system of linear equations into this particularly simple form: Gaussian elimination. Key to Gaussian elimination are elementary transformations of systems of linear equations, which transform the equation system into a simple form. Then, we can apply the three steps to the simple form that we just discussed in the context of the example in (2.38).

2.3.2 Elementary Transformations

elementary transformations

Key to solving a system of linear equations are *elementary transformations* that keep the solution set the same, but that transform the equation system into a simpler form:

- Exchange of two equations (rows in the matrix representing the system of equations)
- Multiplication of an equation (row) with a constant $\lambda \in \mathbb{R}\backslash\{0\}$
- Addition of two equations (rows)

Example 2.6

For $a \in \mathbb{R}$, we seek all solutions of the following system of equations:

$$\begin{array}{rcrcrcrcrcl} -2x_1 & + & 4x_2 & - & 2x_3 & - & x_4 & + & 4x_5 & = & -3 \\ 4x_1 & - & 8x_2 & + & 3x_3 & - & 3x_4 & + & x_5 & = & 2 \\ x_1 & - & 2x_2 & + & x_3 & - & x_4 & + & x_5 & = & 0 \\ x_1 & - & 2x_2 & & & - & 3x_4 & + & 4x_5 & = & a \end{array} . \tag{2.44}$$

We start by converting this system of equations into the compact matrix notation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. We no longer mention the variables $\boldsymbol{x}$ explicitly and build the *augmented matrix* (in the form $[\boldsymbol{A} \,|\, \boldsymbol{b}]$)

augmented matrix

$$\left[\begin{array}{rrrrr|r} -2 & 4 & -2 & -1 & 4 & -3 \\ 4 & -8 & 3 & -3 & 1 & 2 \\ 1 & -2 & 1 & -1 & 1 & 0 \\ 1 & -2 & 0 & -3 & 4 & a \end{array}\right] \begin{array}{l} \text{Swap with } R_3 \\ \\ \text{Swap with } R_1 \\ \\ \end{array}$$

where we used the vertical line to separate the left-hand side from the right-hand side in (2.44). We use $\rightsquigarrow$ to indicate a transformation of the augmented matrix using elementary transformations.

The augmented matrix $[\boldsymbol{A} \,|\, \boldsymbol{b}]$ compactly represents the system of linear equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.

Swapping Rows 1 and 3 leads to

$$\left[\begin{array}{rrrrr|r} 1 & -2 & 1 & -1 & 1 & 0 \\ 4 & -8 & 3 & -3 & 1 & 2 \\ -2 & 4 & -2 & -1 & 4 & -3 \\ 1 & -2 & 0 & -3 & 4 & a \end{array}\right] \begin{array}{l} \\ -4R_1 \\ +2R_1 \\ -R_1 \end{array}$$

When we now apply the indicated transformations (e.g., subtract Row 1 four times from Row 2), we obtain

$$\left[\begin{array}{rrrrr|r} 1 & -2 & 1 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 & -3 & 2 \\ 0 & 0 & 0 & -3 & 6 & -3 \\ 0 & 0 & -1 & -2 & 3 & a \end{array}\right] -R_2 - R_3$$

$$\rightsquigarrow \left[\begin{array}{rrrrr|r} 1 & -2 & 1 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 & -3 & 2 \\ 0 & 0 & 0 & -3 & 6 & -3 \\ 0 & 0 & 0 & 0 & 0 & a+1 \end{array}\right] \begin{array}{l} \\ \cdot(-1) \\ \cdot(-\frac{1}{3}) \\ \\ \end{array}$$

$$\rightsquigarrow \left[\begin{array}{rrrrr|r} 1 & -2 & 1 & -1 & 1 & 0 \\ 0 & 0 & 1 & -1 & 3 & -2 \\ 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & a+1 \end{array}\right]$$

row-echelon form

This (augmented) matrix is in a convenient form, the *row-echelon form* (REF). Reverting this compact notation back into the explicit notation with the variables we seek, we obtain

$$\begin{array}{cccccccccc} x_1 & - & 2x_2 & + & x_3 & - & x_4 & + & x_5 & = & 0 \\ & & & & x_3 & - & x_4 & + & 3x_5 & = & -2 \\ & & & & & & x_4 & - & 2x_5 & = & 1 \\ & & & & & & & & 0 & = & a+1 \end{array} \quad . \tag{2.45}$$

particular solution

Only for $a = -1$ this system can be solved. A *particular solution* is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ -1 \\ 1 \\ 0 \end{bmatrix} . \tag{2.46}$$

general solution

The *general solution*, which captures the set of all possible solutions, is

$$\left\{ \boldsymbol{x} \in \mathbb{R}^5 : \boldsymbol{x} = \begin{bmatrix} 2 \\ 0 \\ -1 \\ 1 \\ 0 \end{bmatrix} + \lambda_1 \begin{bmatrix} 2 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 2 \\ 0 \\ -1 \\ 2 \\ 1 \end{bmatrix} , \quad \lambda_1, \lambda_2 \in \mathbb{R} \right\} . \tag{2.47}$$

In the following, we will detail a constructive way to obtain a particular and general solution of a system of linear equations.

pivot

Remark (Pivots and Staircase Structure). The leading coefficient of a row (first nonzero number from the left) is called the *pivot* and is always strictly to the right of the pivot of the row above it. Therefore, any equation system in row-echelon form always has a "staircase" structure. ♢

row-echelon form

Definition 2.6 (Row-Echelon Form). A matrix is in *row-echelon form* if

- All rows that contain only zeros are at the bottom of the matrix; correspondingly, all rows that contain at least one nonzero element are on top of rows that contain only zeros.
- Looking at nonzero rows only, the first nonzero number from the left (also called the *pivot* or the *leading coefficient*) is always strictly to the right of the pivot of the row above it.

pivot
leading coefficient
In other texts, it is sometimes required that the pivot is 1.

basic variable
free variable

Remark (Basic and Free Variables). The variables corresponding to the pivots in the row-echelon form are called *basic variables* and the other variables are *free variables*. For example, in (2.45), x_1, x_3, x_4 are basic variables, whereas x_2, x_5 are free variables. ♢

Remark (Obtaining a Particular Solution). The row-echelon form makes

our lives easier when we need to determine a particular solution. To do this, we express the right-hand side of the equation system using the pivot columns, such that $\boldsymbol{b} = \sum_{i=1}^{P} \lambda_i \boldsymbol{p}_i$, where $\boldsymbol{p}_i,\ i = 1, \ldots, P$, are the pivot columns. The λ_i are determined easiest if we start with the rightmost pivot column and work our way to the left.

In the previous example, we would try to find $\lambda_1, \lambda_2, \lambda_3$ so that

$$\lambda_1 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \lambda_3 \begin{bmatrix} -1 \\ -1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \\ 1 \\ 0 \end{bmatrix} . \tag{2.48}$$

From here, we find relatively directly that $\lambda_3 = 1, \lambda_2 = -1, \lambda_1 = 2$. When we put everything together, we must not forget the non-pivot columns for which we set the coefficients implicitly to 0. Therefore, we get the particular solution $\boldsymbol{x} = [2, 0, -1, 1, 0]^\top$. ♢

reduced row-echelon form

Remark (Reduced Row Echelon Form). An equation system is in *reduced row-echelon form* (also: *row-reduced echelon form* or *row canonical form*) if

- It is in row-echelon form.
- Every pivot is 1.
- The pivot is the only nonzero entry in its column.

♢

The reduced row-echelon form will play an important role later in Section 2.3.3 because it allows us to determine the general solution of a system of linear equations in a straightforward way.

Gaussian elimination

Remark (Gaussian Elimination). *Gaussian elimination* is an algorithm that performs elementary transformations to bring a system of linear equations into reduced row-echelon form. ♢

Example 2.7 (Reduced Row Echelon Form)
Verify that the following matrix is in reduced row-echelon form (the pivots are in **bold**):

$$\boldsymbol{A} = \begin{bmatrix} \mathbf{1} & 3 & 0 & 0 & 3 \\ 0 & 0 & \mathbf{1} & 0 & 9 \\ 0 & 0 & 0 & \mathbf{1} & -4 \end{bmatrix} . \tag{2.49}$$

The key idea for finding the solutions of $\boldsymbol{A}\boldsymbol{x} = \mathbf{0}$ is to look at the *non-pivot columns*, which we will need to express as a (linear) combination of the pivot columns. The reduced row echelon form makes this relatively straightforward, and we express the non-pivot columns in terms of sums and multiples of the pivot columns that are on their left: The second column is 3 times the first column (we can ignore the pivot columns on the right of the second column). Therefore, to obtain $\mathbf{0}$, we need to subtract

the second column from three times the first column. Now, we look at the fifth column, which is our second non-pivot column. The fifth column can be expressed as 3 times the first pivot column, 9 times the second pivot column, and -4 times the third pivot column. We need to keep track of the indices of the pivot columns and translate this into 3 times the first column, 0 times the second column (which is a non-pivot column), 9 times the third column (which is our second pivot column), and -4 times the fourth column (which is the third pivot column). Then we need to subtract the fifth column to obtain $\mathbf{0}$. In the end, we are still solving a homogeneous equation system.

To summarize, all solutions of $\boldsymbol{A}\boldsymbol{x} = \mathbf{0}, \boldsymbol{x} \in \mathbb{R}^5$ are given by

$$\left\{ \boldsymbol{x} \in \mathbb{R}^5 : \boldsymbol{x} = \lambda_1 \begin{bmatrix} 3 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 3 \\ 0 \\ 9 \\ -4 \\ -1 \end{bmatrix}, \quad \lambda_1, \lambda_2 \in \mathbb{R} \right\}. \tag{2.50}$$

2.3.3 The Minus-1 Trick

In the following, we introduce a practical trick for reading out the solutions $\boldsymbol{x}$ of a homogeneous system of linear equations $\boldsymbol{A}\boldsymbol{x} = \mathbf{0}$, where $\boldsymbol{A} \in \mathbb{R}^{k \times n}, \boldsymbol{x} \in \mathbb{R}^n$.

To start, we assume that $\boldsymbol{A}$ is in reduced row-echelon form without any rows that just contain zeros, i.e.,

$$\boldsymbol{A} = \begin{bmatrix} 0 & \cdots & 0 & \mathbf{1} & * & \cdots & * & 0 & * & \cdots & * & 0 & * & \cdots & * \\ \vdots & & \vdots & 0 & 0 & \cdots & 0 & \mathbf{1} & * & \cdots & * & \vdots & \vdots & & \vdots \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & 0 & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & 0 & \vdots & & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \mathbf{1} & * & \cdots & * \end{bmatrix}, \tag{2.51}$$

where $*$ can be an arbitrary real number, with the constraints that the first nonzero entry per row must be 1 and all other entries in the corresponding column must be 0. The columns $j_1, \ldots, j_k$ with the pivots (marked in **bold**) are the standard unit vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k \in \mathbb{R}^k$. We extend this matrix to an $n \times n$-matrix $\tilde{\boldsymbol{A}}$ by adding $n - k$ rows of the form

$$\begin{bmatrix} 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \end{bmatrix} \tag{2.52}$$

so that the diagonal of the augmented matrix $\tilde{\boldsymbol{A}}$ contains either 1 or -1. Then, the columns of $\tilde{\boldsymbol{A}}$ that contain the -1 as pivots are solutions of

the homogeneous equation system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$. To be more precise, these columns form a basis (Section 2.6.1) of the solution space of $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$, which we will later call the *kernel* or *null space* (see Section 2.7.3).

kernel
null space

Example 2.8 (Minus-1 Trick)
Let us revisit the matrix in (2.49), which is already in reduced REF:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 3 & 0 & 0 & 3 \\ 0 & 0 & 1 & 0 & 9 \\ 0 & 0 & 0 & 1 & -4 \end{bmatrix} . \tag{2.53}$$

We now augment this matrix to a 5×5 matrix by adding rows of the form (2.52) at the places where the pivots on the diagonal are missing and obtain

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} 1 & 3 & 0 & 0 & 3 \\ 0 & \mathbf{-1} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 9 \\ 0 & 0 & 0 & 1 & -4 \\ 0 & 0 & 0 & 0 & \mathbf{-1} \end{bmatrix} . \tag{2.54}$$

From this form, we can immediately read out the solutions of $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ by taking the columns of $\tilde{\boldsymbol{A}}$, which contain -1 on the diagonal:

$$\left\{ \boldsymbol{x} \in \mathbb{R}^5 : \boldsymbol{x} = \lambda_1 \begin{bmatrix} 3 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 3 \\ 0 \\ 9 \\ -4 \\ -1 \end{bmatrix} , \quad \lambda_1, \lambda_2 \in \mathbb{R} \right\} , \tag{2.55}$$

which is identical to the solution in (2.50) that we obtained by "insight".

Calculating the Inverse

To compute the inverse $\boldsymbol{A}^{-1}$ of $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, we need to find a matrix $\boldsymbol{X}$ that satisfies $\boldsymbol{A}\boldsymbol{X} = \boldsymbol{I}_n$. Then, $\boldsymbol{X} = \boldsymbol{A}^{-1}$. We can write this down as a set of simultaneous linear equations $\boldsymbol{A}\boldsymbol{X} = \boldsymbol{I}_n$, where we solve for $\boldsymbol{X} = [\boldsymbol{x}_1 | \cdots | \boldsymbol{x}_n]$. We use the augmented matrix notation for a compact representation of this set of systems of linear equations and obtain

$$\left[\boldsymbol{A} | \boldsymbol{I}_n\right] \quad \rightsquigarrow \cdots \rightsquigarrow \quad \left[\boldsymbol{I}_n | \boldsymbol{A}^{-1}\right] . \tag{2.56}$$

RREF

This means that if we bring the augmented equation system into reduced row-echelon form, we can read out the inverse on the right-hand side of the equation system. Hence, determining the inverse of a matrix is equivalent to solving systems of linear equations.

Example 2.9 (Calculating an Inverse Matrix by Gaussian Elimination)
To determine the inverse of

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 2 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{2.57}$$

we write down the augmented matrix

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

and use Gaussian elimination to bring it into reduced row-echelon form

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

such that the desired inverse is given as its right-hand side:

$$\boldsymbol{A}^{-1} = \begin{bmatrix} -1 & 2 & -2 & 2 \\ 1 & -1 & 2 & -2 \\ 1 & -1 & 1 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix}. \tag{2.58}$$

We can verify that (2.58) is indeed the inverse by performing the multiplication $\boldsymbol{A}\boldsymbol{A}^{-1}$ and observing that we recover $\boldsymbol{I}_4$.

2.3.4 Algorithms for Solving a System of Linear Equations

In the following, we briefly discuss approaches to solving a system of linear equations of the form $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. We make the assumption that a solution exists. Should there be no solution, we need to resort to approximate solutions, which we do not cover in this chapter. One way to solve the approximate problem is using the approach of linear regression, which we discuss in detail in Chapter 9.

In special cases, we may be able to determine the inverse $\boldsymbol{A}^{-1}$, such that the solution of $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ is given as $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$. However, this is only possible if $\boldsymbol{A}$ is a square matrix and invertible, which is often not the case. Otherwise, under mild assumptions (i.e., $\boldsymbol{A}$ needs to have linearly independent columns) we can use the transformation

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \iff \boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}^\top\boldsymbol{b} \iff \boldsymbol{x} = (\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top\boldsymbol{b} \tag{2.59}$$

and use the *Moore-Penrose pseudo-inverse* $(\boldsymbol{A}^\top \boldsymbol{A})^{-1}\boldsymbol{A}^\top$ to determine the solution (2.59) that solves $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, which also corresponds to the minimum norm least-squares solution. A disadvantage of this approach is that it requires many computations for the matrix-matrix product and computing the inverse of $\boldsymbol{A}^\top \boldsymbol{A}$. Moreover, for reasons of numerical precision it is generally not recommended to compute the inverse or pseudo-inverse. In the following, we therefore briefly discuss alternative approaches to solving systems of linear equations.

Moore-Penrose pseudo-inverse

Gaussian elimination plays an important role when computing determinants (Section 4.1), checking whether a set of vectors is linearly independent (Section 2.5), computing the inverse of a matrix (Section 2.2.2), computing the rank of a matrix (Section 2.6.2), and determining a basis of a vector space (Section 2.6.1). Gaussian elimination is an intuitive and constructive way to solve a system of linear equations with thousands of variables. However, for systems with millions of variables, it is impractical as the required number of arithmetic operations scales cubically in the number of simultaneous equations.

In practice, systems of many linear equations are solved indirectly, by either stationary iterative methods, such as the Richardson method, the Jacobi method, the Gauß-Seidel method, and the successive over-relaxation method, or Krylov subspace methods, such as conjugate gradients, generalized minimal residual, or biconjugate gradients. We refer to the books by Stoer and Burlirsch (2002), Strang (2003), and Liesen and Mehrmann (2015) for further details.

Let $\boldsymbol{x}_*$ be a solution of $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. The key idea of these iterative methods is to set up an iteration of the form

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{C}\boldsymbol{x}^{(k)} + \boldsymbol{d} \tag{2.60}$$

for suitable $\boldsymbol{C}$ and $\boldsymbol{d}$ that reduces the residual error $\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}_*\|$ in every iteration and converges to $\boldsymbol{x}_*$. We will introduce norms $\|\cdot\|$, which allow us to compute similarities between vectors, in Section 3.1.

2.4 Vector Spaces

Thus far, we have looked at systems of linear equations and how to solve them (Section 2.3). We saw that systems of linear equations can be compactly represented using matrix-vector notation (2.10). In the following, we will have a closer look at vector spaces, i.e., a structured space in which vectors live.

In the beginning of this chapter, we informally characterized vectors as objects that can be added together and multiplied by a scalar, and they remain objects of the same type. Now, we are ready to formalize this, and we will start by introducing the concept of a group, which is a set of elements and an operation defined on these elements that keeps some structure of the set intact.

closure

2.4.1 Groups

Groups play an important role in computer science. Besides providing a fundamental framework for operations on sets, they are heavily used in cryptography, coding theory, and graphics.

group
closure
associativity
neutral element
inverse element

Definition 2.7 (Group)**.** Consider a set $\mathcal{G}$ and an operation $\otimes : \mathcal{G} \times \mathcal{G} \to \mathcal{G}$ defined on $\mathcal{G}$. Then $G := (\mathcal{G}, \otimes)$ is called a *group* if the following hold:

1. *Closure* of $\mathcal{G}$ under $\otimes$: $\forall x, y \in \mathcal{G} : x \otimes y \in \mathcal{G}$
2. *Associativity:* $\forall x, y, z \in \mathcal{G} : (x \otimes y) \otimes z = x \otimes (y \otimes z)$
3. *Neutral element:* $\exists e \in \mathcal{G} \, \forall x \in \mathcal{G} : x \otimes e = x$ and $e \otimes x = x$
4. *Inverse element:* $\forall x \in \mathcal{G} \, \exists y \in \mathcal{G} : x \otimes y = e$ and $y \otimes x = e$, where e is the neutral element. We often write x^{-1} to denote the inverse element of x.

Remark. The inverse element is defined with respect to the operation $\otimes$ and does not necessarily mean $\frac{1}{x}$. ♢

Abelian group

If additionally $\forall x, y \in \mathcal{G} : x \otimes y = y \otimes x$, then $G = (\mathcal{G}, \otimes)$ is an *Abelian group* (commutative).

Example 2.10 (Groups)
Let us have a look at some examples of sets with associated operations and see whether they are groups:

$\mathbb{N}_0 := \mathbb{N} \cup \{0\}$

- $(\mathbb{Z}, +)$ is an Abelian group.
- $(\mathbb{N}_0, +)$ is not a group: Although $(\mathbb{N}_0, +)$ possesses a neutral element (0), the inverse elements are missing.
- $(\mathbb{Z}, \cdot)$ is not a group: Although $(\mathbb{Z}, \cdot)$ contains a neutral element (1), the inverse elements for any $z \in \mathbb{Z}, z \neq \pm 1$, are missing.
- $(\mathbb{R}, \cdot)$ is not a group since 0 does not possess an inverse element.
- $(\mathbb{R} \backslash \{0\}, \cdot)$ is Abelian.
- $(\mathbb{R}^n, +), (\mathbb{Z}^n, +), n \in \mathbb{N}$ are Abelian if $+$ is defined componentwise, i.e.,

$$(x_1, \cdots, x_n) + (y_1, \cdots, y_n) = (x_1 + y_1, \cdots, x_n + y_n). \tag{2.61}$$

  Then, $(x_1, \cdots, x_n)^{-1} := (-x_1, \cdots, -x_n)$ is the inverse element and $e = (0, \cdots, 0)$ is the neutral element.
- $(\mathbb{R}^{m \times n}, +)$, the set of $m \times n$-matrices is Abelian (with componentwise addition as defined in (2.61)).
- Let us have a closer look at $(\mathbb{R}^{n \times n}, \cdot)$, i.e., the set of $n \times n$-matrices with matrix multiplication as defined in (2.13).
  - Closure and associativity follow directly from the definition of matrix multiplication.
  - Neutral element: The identity matrix $\boldsymbol{I}_n$ is the neutral element with respect to matrix multiplication "$\cdot$" in $(\mathbb{R}^{n \times n}, \cdot)$.

– Inverse element: If the inverse exists ($\boldsymbol{A}$ is regular), then $\boldsymbol{A}^{-1}$ is the inverse element of $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, and in exactly this case $(\mathbb{R}^{n\times n}, \cdot)$ is a group, called the *general linear group*.

Definition 2.8 (General Linear Group)**.** The set of regular (invertible) matrices $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is a group with respect to matrix multiplication as defined in (2.13) and is called *general linear group* $GL(n, \mathbb{R})$. However, since matrix multiplication is not commutative, the group is not Abelian. general linear group

2.4.2 Vector Spaces

When we discussed groups, we looked at sets $\mathcal{G}$ and inner operations on $\mathcal{G}$, i.e., mappings $\mathcal{G} \times \mathcal{G} \to \mathcal{G}$ that only operate on elements in $\mathcal{G}$. In the following, we will consider sets that in addition to an inner operation $+$ also contain an outer operation $\cdot$, the multiplication of a vector $\boldsymbol{x} \in \mathcal{G}$ by a scalar $\lambda \in \mathbb{R}$. We can think of the inner operation as a form of addition, and the outer operation as a form of scaling. Note that the inner/outer operations have nothing to do with inner/outer products.

Definition 2.9 (Vector Space)**.** A real-valued *vector space* $V = (\mathcal{V}, +, \cdot)$ is a set $\mathcal{V}$ with two operations vector space

$$+ : \mathcal{V} \times \mathcal{V} \to \mathcal{V} \tag{2.62}$$
$$\cdot : \mathbb{R} \times \mathcal{V} \to \mathcal{V} \tag{2.63}$$

where

1. $(\mathcal{V}, +)$ is an Abelian group
2. Distributivity:
   1. $\forall \lambda \in \mathbb{R}, \boldsymbol{x}, \boldsymbol{y} \in \mathcal{V} : \lambda \cdot (\boldsymbol{x} + \boldsymbol{y}) = \lambda \cdot \boldsymbol{x} + \lambda \cdot \boldsymbol{y}$
   2. $\forall \lambda, \psi \in \mathbb{R}, \boldsymbol{x} \in \mathcal{V} : (\lambda + \psi) \cdot \boldsymbol{x} = \lambda \cdot \boldsymbol{x} + \psi \cdot \boldsymbol{x}$
3. Associativity (outer operation): $\forall \lambda, \psi \in \mathbb{R}, \boldsymbol{x} \in \mathcal{V} : \lambda \cdot (\psi \cdot \boldsymbol{x}) = (\lambda\psi) \cdot \boldsymbol{x}$
4. Neutral element with respect to the outer operation: $\forall \boldsymbol{x} \in \mathcal{V} : 1 \cdot \boldsymbol{x} = \boldsymbol{x}$

The elements $\boldsymbol{x} \in V$ are called *vectors*. The neutral element of $(\mathcal{V}, +)$ is the zero vector $\boldsymbol{0} = [0, \ldots, 0]^\top$, and the inner operation $+$ is called *vector addition*. The elements $\lambda \in \mathbb{R}$ are called *scalars* and the outer operation $\cdot$ is a *multiplication by scalars*. Note that a scalar product is something different, and we will get to this in Section 3.2. vector; vector addition; scalar; multiplication by scalars

Remark. A "vector multiplication" $\boldsymbol{ab}$, $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^n$, is not defined. Theoretically, we could define an element-wise multiplication, such that $\boldsymbol{c} = \boldsymbol{ab}$ with $c_j = a_j b_j$. This "array multiplication" is common to many programming languages but makes mathematically limited sense using the standard rules for matrix multiplication: By treating vectors as $n \times 1$ matrices

(which we usually do), we can use the matrix multiplication as defined in (2.13). However, then the dimensions of the vectors do not match. Only the following multiplications for vectors are defined: $\boldsymbol{a}\boldsymbol{b}^\top \in \mathbb{R}^{n\times n}$ (*outer product*), $\boldsymbol{a}^\top\boldsymbol{b} \in \mathbb{R}$ (inner/scalar/dot product). ♢

outer product

Example 2.11 (Vector Spaces)
Let us have a look at some important examples:

- $\mathcal{V} = \mathbb{R}^n, n \in \mathbb{N}$ is a vector space with operations defined as follows:
  - Addition: $\boldsymbol{x}+\boldsymbol{y} = (x_1,\dots,x_n)+(y_1,\dots,y_n) = (x_1+y_1,\dots,x_n+y_n)$ for all $\boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^n$
  - Multiplication by scalars: $\lambda\boldsymbol{x} = \lambda(x_1,\dots,x_n) = (\lambda x_1,\dots,\lambda x_n)$ for all $\lambda \in \mathbb{R}, \boldsymbol{x} \in \mathbb{R}^n$
- $\mathcal{V} = \mathbb{R}^{m\times n}, m,n \in \mathbb{N}$ is a vector space with
  - Addition: $\boldsymbol{A}+\boldsymbol{B} = \begin{bmatrix} a_{11}+b_{11} & \cdots & a_{1n}+b_{1n} \\ \vdots & & \vdots \\ a_{m1}+b_{m1} & \cdots & a_{mn}+b_{mn} \end{bmatrix}$ is defined elementwise for all $\boldsymbol{A},\boldsymbol{B} \in \mathcal{V}$
  - Multiplication by scalars: $\lambda\boldsymbol{A} = \begin{bmatrix} \lambda a_{11} & \cdots & \lambda a_{1n} \\ \vdots & & \vdots \\ \lambda a_{m1} & \cdots & \lambda a_{mn} \end{bmatrix}$ as defined in Section 2.2. Remember that $\mathbb{R}^{m\times n}$ is equivalent to $\mathbb{R}^{mn}$.
- $\mathcal{V} = \mathbb{C}$, with the standard definition of addition of complex numbers.

Remark. In the following, we will denote a vector space $(\mathcal{V},+,\cdot)$ by V when $+$ and $\cdot$ are the standard vector addition and scalar multiplication. Moreover, we will use the notation $\boldsymbol{x} \in V$ for vectors in $\mathcal{V}$ to simplify notation. ♢

Remark. The vector spaces $\mathbb{R}^n, \mathbb{R}^{n\times 1}, \mathbb{R}^{1\times n}$ are only different in the way we write vectors. In the following, we will not make a distinction between $\mathbb{R}^n$ and $\mathbb{R}^{n\times 1}$, which allows us to write n-tuples as *column vectors*

column vector

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}. \tag{2.64}$$

This simplifies the notation regarding vector space operations. However, we do distinguish between $\mathbb{R}^{n\times 1}$ and $\mathbb{R}^{1\times n}$ (the *row vectors*) to avoid confusion with matrix multiplication. By default, we write $\boldsymbol{x}$ to denote a column vector, and a row vector is denoted by $\boldsymbol{x}^\top$, the *transpose* of $\boldsymbol{x}$. ♢

row vector

transpose

2.4.3 Vector Subspaces

In the following, we will introduce vector subspaces. Intuitively, they are sets contained in the original vector space with the property that when we perform vector space operations on elements within this subspace, we will never leave it. In this sense, they are "closed". Vector subspaces are a key idea in machine learning. For example, Chapter 10 demonstrates how to use vector subspaces for dimensionality reduction.

Definition 2.10 (Vector Subspace)**.** Let $V = (\mathcal{V}, +, \cdot)$ be a vector space and $\mathcal{U} \subseteq \mathcal{V}$, $\mathcal{U} \neq \emptyset$. Then $U = (\mathcal{U}, +, \cdot)$ is called *vector subspace* of V (or *linear subspace*) if U is a vector space with the vector space operations $+$ and $\cdot$ restricted to $\mathcal{U} \times \mathcal{U}$ and $\mathbb{R} \times \mathcal{U}$. We write $U \subseteq V$ to denote a subspace U of V.

vector subspace
linear subspace

If $\mathcal{U} \subseteq \mathcal{V}$ and V is a vector space, then U naturally inherits many properties directly from V because they hold for all $\boldsymbol{x} \in \mathcal{V}$, and in particular for all $\boldsymbol{x} \in \mathcal{U} \subseteq \mathcal{V}$. This includes the Abelian group properties, the distributivity, the associativity and the neutral element. To determine whether $(\mathcal{U}, +, \cdot)$ is a subspace of V we still do need to show

1. $\mathcal{U} \neq \emptyset$, in particular: $\mathbf{0} \in \mathcal{U}$
2. Closure of U:
   a. With respect to the outer operation: $\forall \lambda \in \mathbb{R} \, \forall \boldsymbol{x} \in \mathcal{U} : \lambda \boldsymbol{x} \in \mathcal{U}$.
   b. With respect to the inner operation: $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{U} : \boldsymbol{x} + \boldsymbol{y} \in \mathcal{U}$.

Example 2.12 (Vector Subspaces)
Let us have a look at some examples:

- For every vector space V, the trivial subspaces are V itself and $\{\mathbf{0}\}$.
- Only example D in Figure 2.6 is a subspace of $\mathbb{R}^2$ (with the usual inner/outer operations). In A and C, the closure property is violated; B does not contain $\mathbf{0}$.
- The solution set of a homogeneous system of linear equations $\boldsymbol{A}\boldsymbol{x} = \mathbf{0}$ with n unknowns $\boldsymbol{x} = [x_1, \ldots, x_n]^\top$ is a subspace of $\mathbb{R}^n$.
- The solution of an inhomogeneous system of linear equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{b} \neq \mathbf{0}$ is not a subspace of $\mathbb{R}^n$.
- The intersection of arbitrarily many subspaces is a subspace itself.

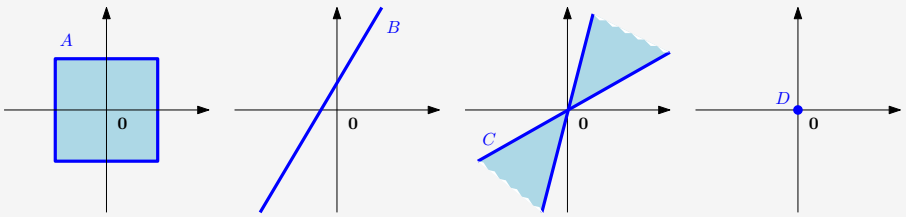


Figure 2.6 Not all subsets of $\mathbb{R}^2$ are subspaces. In A and C, the closure property is violated; B does not contain $\mathbf{0}$. Only D is a subspace.

Remark. Every subspace $U \subseteq (\mathbb{R}^n, +, \cdot)$ is the solution space of a homogeneous system of linear equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ for $\boldsymbol{x} \in \mathbb{R}^n$. ♢

2.5 Linear Independence

In the following, we will have a close look at what we can do with vectors (elements of the vector space). In particular, we can add vectors together and multiply them with scalars. The closure property guarantees that we end up with another vector in the same vector space. It is possible to find a set of vectors with which we can represent every vector in the vector space by adding them together and scaling them. This set of vectors is a *basis*, and we will discuss them in Section 2.6.1. Before we get there, we will need to introduce the concepts of linear combinations and linear independence.

Definition 2.11 (Linear Combination)**.** Consider a vector space V and a finite number of vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in V$. Then, every $\boldsymbol{v} \in V$ of the form

$$\boldsymbol{v} = \lambda_1 \boldsymbol{x}_1 + \cdots + \lambda_k \boldsymbol{x}_k = \sum_{i=1}^{k} \lambda_i \boldsymbol{x}_i \in V \tag{2.65}$$

linear combination

with $\lambda_1, \ldots, \lambda_k \in \mathbb{R}$ is a *linear combination* of the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$.

The $\boldsymbol{0}$-vector can always be written as the linear combination of k vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ because $\boldsymbol{0} = \sum_{i=1}^{k} 0\boldsymbol{x}_i$ is always true. In the following, we are interested in non-trivial linear combinations of a set of vectors to represent $\boldsymbol{0}$, i.e., linear combinations of vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$, where not all coefficients λ_i in (2.65) are 0.

Definition 2.12 (Linear (In)dependence)**.** Let us consider a vector space V with $k \in \mathbb{N}$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in V$. If there is a non-trivial linear combination, such that $\boldsymbol{0} = \sum_{i=1}^{k} \lambda_i \boldsymbol{x}_i$ with at least one $\lambda_i \neq 0$, the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ are *linearly dependent*. If only the trivial solution exists, i.e., $\lambda_1 = \ldots = \lambda_k = 0$ the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ are *linearly independent*.

linearly dependent

linearly independent

Linear independence is one of the most important concepts in linear algebra. Intuitively, a set of linearly independent vectors consists of vectors that have no redundancy, i.e., if we remove any of those vectors from the set, we will lose something. Throughout the next sections, we will formalize this intuition more.

Example 2.13 (Linearly Dependent Vectors)
A geographic example may help to clarify the concept of linear independence. A person in Nairobi (Kenya) describing where Kigali (Rwanda) is might say ,"You can get to Kigali by first going $506\,\mathrm{km}$ Northwest to Kampala (Uganda) and then $374\,\mathrm{km}$ Southwest.". This is sufficient information

to describe the location of Kigali because the geographic coordinate system may be considered a two-dimensional vector space (ignoring altitude and the Earth's curved surface). The person may add, "It is about $751\,\mathrm{km}$ West of here." Although this last statement is true, it is not necessary to find Kigali given the previous information (see Figure 2.7 for an illustration). In this example, the "$506\,\mathrm{km}$ Northwest" vector (blue) and the "$374\,\mathrm{km}$ Southwest" vector (purple) are linearly independent. This means the Southwest vector cannot be described in terms of the Northwest vector, and vice versa. However, the third "$751\,\mathrm{km}$ West" vector (black) is a linear combination of the other two vectors, and it makes the set of vectors linearly dependent. Equivalently, given "$751\,\mathrm{km}$ West" and "$374\,\mathrm{km}$ Southwest" can be linearly combined to obtain "$506\,\mathrm{km}$ Northwest".

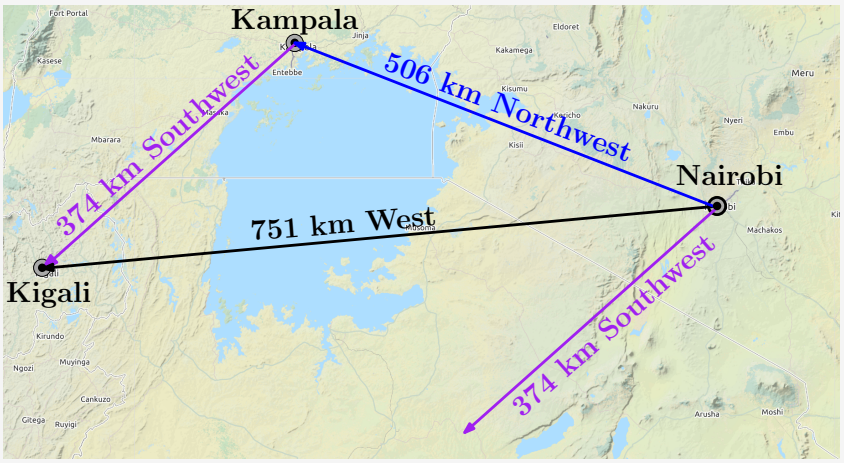


Figure 2.7 Geographic example (with crude approximations to cardinal directions) of linearly dependent vectors in a two-dimensional space (plane).

Remark. The following properties are useful to find out whether vectors are linearly independent:

- k vectors are either linearly dependent or linearly independent. There is no third option.
- If at least one of the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ is $\mathbf{0}$ then they are linearly dependent. The same holds if two vectors are identical.
- The vectors $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k : \boldsymbol{x}_i \neq \mathbf{0}, i = 1, \ldots, k\}$, $k \geqslant 2$, are linearly dependent if and only if (at least) one of them is a linear combination of the others. In particular, if one vector is a multiple of another vector, i.e., $\boldsymbol{x}_i = \lambda \boldsymbol{x}_j$, $\lambda \in \mathbb{R}$ then the set $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k : \boldsymbol{x}_i \neq \mathbf{0}, i = 1, \ldots, k\}$ is linearly dependent.
- A practical way of checking whether vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in V$ are linearly independent is to use Gaussian elimination: Write all vectors as columns of a matrix $\boldsymbol{A}$ and perform Gaussian elimination until the matrix is in row echelon form (the reduced row-echelon form is unnecessary here):

- The pivot columns indicate the vectors, which are linearly independent of the vectors on the left. Note that there is an ordering of vectors when the matrix is built.
- The non-pivot columns can be expressed as linear combinations of the pivot columns on their left. For instance, the row-echelon form

$$\begin{bmatrix} 1 & 3 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{2.66}$$

tells us that the first and third columns are pivot columns. The second column is a non-pivot column because it is three times the first column.

All column vectors are linearly independent if and only if all columns are pivot columns. If there is at least one non-pivot column, the columns (and, therefore, the corresponding vectors) are linearly dependent.

♢

Example 2.14
Consider $\mathbb{R}^4$ with

$$\boldsymbol{x}_1 = \begin{bmatrix} 1 \\ 2 \\ -3 \\ 4 \end{bmatrix}, \quad \boldsymbol{x}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 2 \end{bmatrix}, \quad \boldsymbol{x}_3 = \begin{bmatrix} -1 \\ -2 \\ 1 \\ 1 \end{bmatrix}. \tag{2.67}$$

To check whether they are linearly dependent, we follow the general approach and solve

$$\lambda_1 \boldsymbol{x}_1 + \lambda_2 \boldsymbol{x}_2 + \lambda_3 \boldsymbol{x}_3 = \lambda_1 \begin{bmatrix} 1 \\ 2 \\ -3 \\ 4 \end{bmatrix} + \lambda_2 \begin{bmatrix} 1 \\ 1 \\ 0 \\ 2 \end{bmatrix} + \lambda_3 \begin{bmatrix} -1 \\ -2 \\ 1 \\ 1 \end{bmatrix} = \mathbf{0} \tag{2.68}$$

for $\lambda_1, \ldots, \lambda_3$. We write the vectors $\boldsymbol{x}_i$, $i = 1, 2, 3$, as the columns of a matrix and apply elementary row operations until we identify the pivot columns:

$$\begin{bmatrix} 1 & 1 & -1 \\ 2 & 1 & -2 \\ -3 & 0 & 1 \\ 4 & 2 & 1 \end{bmatrix} \rightsquigarrow \cdots \rightsquigarrow \begin{bmatrix} 1 & 1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}. \tag{2.69}$$

Here, every column of the matrix is a pivot column. Therefore, there is no non-trivial solution, and we require $\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$ to solve the equation system. Hence, the vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3$ are linearly independent.

Remark. Consider a vector space V with k linearly independent vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k$ and m linear combinations

$$\begin{aligned} \boldsymbol{x}_1 &= \sum_{i=1}^{k} \lambda_{i1} \boldsymbol{b}_i \,, \\ &\vdots \\ \boldsymbol{x}_m &= \sum_{i=1}^{k} \lambda_{im} \boldsymbol{b}_i \,. \end{aligned} \tag{2.70}$$

Defining $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k]$ as the matrix whose columns are the linearly independent vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k$, we can write

$$\boldsymbol{x}_j = \boldsymbol{B}\boldsymbol{\lambda}_j \,, \quad \boldsymbol{\lambda}_j = \begin{bmatrix} \lambda_{1j} \\ \vdots \\ \lambda_{kj} \end{bmatrix} \,, \quad j = 1, \ldots, m \,, \tag{2.71}$$

in a more compact form.

We want to test whether $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ are linearly independent. For this purpose, we follow the general approach of testing when $\sum_{j=1}^{m} \psi_j \boldsymbol{x}_j = \boldsymbol{0}$. With (2.71), we obtain

$$\sum_{j=1}^{m} \psi_j \boldsymbol{x}_j = \sum_{j=1}^{m} \psi_j \boldsymbol{B}\boldsymbol{\lambda}_j = \boldsymbol{B} \sum_{j=1}^{m} \psi_j \boldsymbol{\lambda}_j \,. \tag{2.72}$$

This means that $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\}$ are linearly independent if and only if the column vectors $\{\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_m\}$ are linearly independent.

♢

Remark. In a vector space V, m linear combinations of k vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ are linearly dependent if $m > k$. ♢

Example 2.15

Consider a set of linearly independent vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3, \boldsymbol{b}_4 \in \mathbb{R}^n$ and

$$\begin{array}{rcrcrcrcl} \boldsymbol{x}_1 &=& \boldsymbol{b}_1 &-& 2\boldsymbol{b}_2 &+& \boldsymbol{b}_3 &-& \boldsymbol{b}_4 \\ \boldsymbol{x}_2 &=& -4\boldsymbol{b}_1 &-& 2\boldsymbol{b}_2 & & &+& 4\boldsymbol{b}_4 \\ \boldsymbol{x}_3 &=& 2\boldsymbol{b}_1 &+& 3\boldsymbol{b}_2 &-& \boldsymbol{b}_3 &-& 3\boldsymbol{b}_4 \\ \boldsymbol{x}_4 &=& 17\boldsymbol{b}_1 &-& 10\boldsymbol{b}_2 &+& 11\boldsymbol{b}_3 &+& \boldsymbol{b}_4 \end{array} \,. \tag{2.73}$$

Are the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_4 \in \mathbb{R}^n$ linearly independent? To answer this question, we investigate whether the column vectors

$$\left\{ \begin{bmatrix} 1 \\ -2 \\ 1 \\ -1 \end{bmatrix}, \begin{bmatrix} -4 \\ -2 \\ 0 \\ 4 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \\ -1 \\ -3 \end{bmatrix}, \begin{bmatrix} 17 \\ -10 \\ 11 \\ 1 \end{bmatrix} \right\} \tag{2.74}$$

are linearly independent. The reduced row-echelon form of the corresponding linear equation system with coefficient matrix

$$\boldsymbol{A} = \begin{bmatrix} 1 & -4 & 2 & 17 \\ -2 & -2 & 3 & -10 \\ 1 & 0 & -1 & 11 \\ -1 & 4 & -3 & 1 \end{bmatrix} \tag{2.75}$$

is given as

$$\begin{bmatrix} 1 & 0 & 0 & -7 \\ 0 & 1 & 0 & -15 \\ 0 & 0 & 1 & -18 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.76}$$

We see that the corresponding linear equation system is non-trivially solvable: The last column is not a pivot column, and $\boldsymbol{x}_4 = -7\boldsymbol{x}_1 - 15\boldsymbol{x}_2 - 18\boldsymbol{x}_3$. Therefore, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_4$ are linearly dependent as $\boldsymbol{x}_4$ can be expressed as a linear combination of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_3$.

2.6 Basis and Rank

In a vector space V, we are particularly interested in sets of vectors $\mathcal{A}$ that possess the property that any vector $\boldsymbol{v} \in V$ can be obtained by a linear combination of vectors in $\mathcal{A}$. These vectors are special vectors, and in the following, we will characterize them.

2.6.1 Generating Set and Basis

Definition 2.13 (Generating Set and Span)**.** Consider a vector space $V = (\mathcal{V}, +, \cdot)$ and set of vectors $\mathcal{A} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\} \subseteq \mathcal{V}$. If every vector $\boldsymbol{v} \in \mathcal{V}$ can be expressed as a linear combination of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$, $\mathcal{A}$ is called a *generating set* of V. The set of all linear combinations of vectors in $\mathcal{A}$ is called the *span* of $\mathcal{A}$. If $\mathcal{A}$ spans the vector space V, we write $V = \operatorname{span}[\mathcal{A}]$ or $V = \operatorname{span}[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k]$.

generating set

span

Generating sets are sets of vectors that span vector (sub)spaces, i.e., every vector can be represented as a linear combination of the vectors in the generating set. Now, we will be more specific and characterize the smallest generating set that spans a vector (sub)space.

Definition 2.14 (Basis)**.** Consider a vector space $V = (\mathcal{V}, +, \cdot)$ and $\mathcal{A} \subseteq \mathcal{V}$. A generating set $\mathcal{A}$ of V is called *minimal* if there exists no smaller set $\tilde{\mathcal{A}} \subsetneq \mathcal{A} \subseteq \mathcal{V}$ that spans V. Every linearly independent generating set of V is minimal and is called a *basis* of V.

minimal

basis

Let $V = (\mathcal{V}, +, \cdot)$ be a vector space and $\mathcal{B} \subseteq \mathcal{V}, \mathcal{B} \neq \emptyset$. Then, the following statements are equivalent:

A basis is a minimal generating set and a maximal linearly independent set of vectors.

- $\mathcal{B}$ is a basis of V.
- $\mathcal{B}$ is a minimal generating set.
- $\mathcal{B}$ is a maximal linearly independent set of vectors in V, i.e., adding any other vector to this set will make it linearly dependent.
- Every vector $\boldsymbol{x} \in V$ is a linear combination of vectors from $\mathcal{B}$, and every linear combination is unique, i.e., with

$$\boldsymbol{x} = \sum_{i=1}^{k} \lambda_i \boldsymbol{b}_i = \sum_{i=1}^{k} \psi_i \boldsymbol{b}_i \tag{2.77}$$

and $\lambda_i, \psi_i \in \mathbb{R}$, $\boldsymbol{b}_i \in \mathcal{B}$ it follows that $\lambda_i = \psi_i,\ i = 1, \ldots, k$.

Example 2.16

- In $\mathbb{R}^3$, the *canonical/standard basis* is

canonical basis

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}. \tag{2.78}$$

- Different bases in $\mathbb{R}^3$ are

$$\mathcal{B}_1 = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}, \mathcal{B}_2 = \left\{ \begin{bmatrix} 0.5 \\ 0.8 \\ 0.4 \end{bmatrix}, \begin{bmatrix} 1.8 \\ 0.3 \\ 0.3 \end{bmatrix}, \begin{bmatrix} -2.2 \\ -1.3 \\ 3.5 \end{bmatrix} \right\}. \tag{2.79}$$

- The set

$$\mathcal{A} = \left\{ \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \begin{bmatrix} 2 \\ -1 \\ 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \\ -4 \end{bmatrix} \right\} \tag{2.80}$$

is linearly independent, but not a generating set (and no basis) of $\mathbb{R}^4$: For instance, the vector $[1, 0, 0, 0]^\top$ cannot be obtained by a linear combination of elements in $\mathcal{A}$.

Remark. Every vector space V possesses a basis $\mathcal{B}$. The preceding examples show that there can be many bases of a vector space V, i.e., there is no unique basis. However, all bases possess the same number of elements, the *basis vectors*. ♢

basis vector

We only consider finite-dimensional vector spaces V. In this case, the *dimension* of V is the number of basis vectors of V, and we write $\dim(V)$. If $U \subseteq V$ is a subspace of V, then $\dim(U) \leqslant \dim(V)$ and $\dim(U) =$

dimension

$\dim(V)$ if and only if $U = V$. Intuitively, the dimension of a vector space can be thought of as the number of independent directions in this vector space.

The dimension of a vector space corresponds to the number of its basis vectors.

Remark. The dimension of a vector space is not necessarily the number of elements in a vector. For instance, the vector space $V = \operatorname{span}[\begin{bmatrix}0\\1\end{bmatrix}]$ is one-dimensional, although the basis vector possesses two elements. ♢

Remark. A basis of a subspace $U = \operatorname{span}[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m] \subseteq \mathbb{R}^n$ can be found by executing the following steps:

1. Write the spanning vectors as columns of a matrix $\boldsymbol{A}$
2. Determine the row-echelon form of $\boldsymbol{A}$.
3. The spanning vectors associated with the pivot columns are a basis of U.

♢

Example 2.17 (Determining a Basis)
For a vector subspace $U \subseteq \mathbb{R}^5$, spanned by the vectors

$$\boldsymbol{x}_1 = \begin{bmatrix}1\\2\\-1\\-1\\-1\end{bmatrix}, \quad \boldsymbol{x}_2 = \begin{bmatrix}2\\-1\\1\\2\\-2\end{bmatrix}, \quad \boldsymbol{x}_3 = \begin{bmatrix}3\\-4\\3\\5\\-3\end{bmatrix}, \quad \boldsymbol{x}_4 = \begin{bmatrix}-1\\8\\-5\\-6\\1\end{bmatrix} \in \mathbb{R}^5, \tag{2.81}$$

we are interested in finding out which vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_4$ are a basis for U. For this, we need to check whether $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_4$ are linearly independent. Therefore, we need to solve

$$\sum_{i=1}^{4} \lambda_i \boldsymbol{x}_i = \boldsymbol{0}\,, \tag{2.82}$$

which leads to a homogeneous system of equations with matrix

$$\left[\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4\right] = \begin{bmatrix}1 & 2 & 3 & -1\\2 & -1 & -4 & 8\\-1 & 1 & 3 & -5\\-1 & 2 & 5 & -6\\-1 & -2 & -3 & 1\end{bmatrix}. \tag{2.83}$$

With the basic transformation rules for systems of linear equations, we obtain the row-echelon form

$$\begin{bmatrix}1 & 2 & 3 & -1\\2 & -1 & -4 & 8\\-1 & 1 & 3 & -5\\-1 & 2 & 5 & -6\\-1 & -2 & -3 & 1\end{bmatrix} \rightsquigarrow \cdots \rightsquigarrow \begin{bmatrix}1 & 2 & 3 & -1\\0 & 1 & 2 & -2\\0 & 0 & 0 & 1\\0 & 0 & 0 & 0\\0 & 0 & 0 & 0\end{bmatrix}.$$

Since the pivot columns indicate which set of vectors is linearly independent, we see from the row-echelon form that $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_4$ are linearly independent (because the system of linear equations $\lambda_1\boldsymbol{x}_1 + \lambda_2\boldsymbol{x}_2 + \lambda_4\boldsymbol{x}_4 = \boldsymbol{0}$ can only be solved with $\lambda_1 = \lambda_2 = \lambda_4 = 0$). Therefore, $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_4\}$ is a basis of U.

2.6.2 Rank

The number of linearly independent columns of a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ equals the number of linearly independent rows and is called the *rank* of $\boldsymbol{A}$ and is denoted by $\mathrm{rk}(\boldsymbol{A})$. rank

Remark. The rank of a matrix has some important properties:

- $\mathrm{rk}(\boldsymbol{A}) = \mathrm{rk}(\boldsymbol{A}^\top)$, i.e., the column rank equals the row rank.
- The columns of $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ span a subspace $U \subseteq \mathbb{R}^m$ with $\dim(U) = \mathrm{rk}(\boldsymbol{A})$. Later we will call this subspace the *image* or *range*. A basis of U can be found by applying Gaussian elimination to $\boldsymbol{A}$ to identify the pivot columns.
- The rows of $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ span a subspace $W \subseteq \mathbb{R}^n$ with $\dim(W) = \mathrm{rk}(\boldsymbol{A})$. A basis of W can be found by applying Gaussian elimination to $\boldsymbol{A}^\top$.
- For all $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ it holds that $\boldsymbol{A}$ is regular (invertible) if and only if $\mathrm{rk}(\boldsymbol{A}) = n$.
- For all $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and all $\boldsymbol{b} \in \mathbb{R}^m$ it holds that the linear equation system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ can be solved if and only if $\mathrm{rk}(\boldsymbol{A}) = \mathrm{rk}(\boldsymbol{A}|\boldsymbol{b})$, where $\boldsymbol{A}|\boldsymbol{b}$ denotes the augmented system.
- For $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ the subspace of solutions for $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ possesses dimension $n - \mathrm{rk}(\boldsymbol{A})$. Later, we will call this subspace the *kernel* or the *null space*. kernel, null space
- A matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ has *full rank* if its rank equals the largest possible rank for a matrix of the same dimensions. This means that the rank of a full-rank matrix is the lesser of the number of rows and columns, i.e., $\mathrm{rk}(\boldsymbol{A}) = \min(m, n)$. A matrix is said to be *rank deficient* if it does not have full rank. full rank, rank deficient

♢

Example 2.18 (Rank)

- $\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$.
  $\boldsymbol{A}$ has two linearly independent rows/columns so that $\mathrm{rk}(\boldsymbol{A}) = 2$.

- $\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix}$.

  We use Gaussian elimination to determine the rank:

$$\begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix} \rightsquigarrow \cdots \rightsquigarrow \begin{bmatrix} 1 & 2 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 0 \end{bmatrix}. \tag{2.84}$$

  Here, we see that the number of linearly independent rows and columns is 2, such that $\mathrm{rk}(\boldsymbol{A}) = 2$.

2.7 Linear Mappings

In the following, we will study mappings on vector spaces that preserve their structure, which will allow us to define the concept of a coordinate. In the beginning of the chapter, we said that vectors are objects that can be added together and multiplied by a scalar, and the resulting object is still a vector. We wish to preserve this property when applying the mapping: Consider two real vector spaces V, W. A mapping $\Phi : V \to W$ preserves the structure of the vector space if

$$\Phi(\boldsymbol{x} + \boldsymbol{y}) = \Phi(\boldsymbol{x}) + \Phi(\boldsymbol{y}) \tag{2.85}$$
$$\Phi(\lambda \boldsymbol{x}) = \lambda \Phi(\boldsymbol{x}) \tag{2.86}$$

for all $\boldsymbol{x}, \boldsymbol{y} \in V$ and $\lambda \in \mathbb{R}$. We can summarize this in the following definition:

linear mapping
vector space homomorphism
linear transformation

Definition 2.15 (Linear Mapping)**.** For vector spaces V, W, a mapping $\Phi : V \to W$ is called a *linear mapping* (or *vector space homomorphism/ linear transformation*) if

$$\forall \boldsymbol{x}, \boldsymbol{y} \in V \, \forall \lambda, \psi \in \mathbb{R} : \Phi(\lambda \boldsymbol{x} + \psi \boldsymbol{y}) = \lambda \Phi(\boldsymbol{x}) + \psi \Phi(\boldsymbol{y}) \,. \tag{2.87}$$

It turns out that we can represent linear mappings as matrices (Section 2.7.1). Recall that we can also collect a set of vectors as columns of a matrix. When working with matrices, we have to keep in mind what the matrix represents: a linear mapping or a collection of vectors. We will see more about linear mappings in Chapter 4. Before we continue, we will briefly introduce special mappings.

Definition 2.16 (Injective, Surjective, Bijective)**.** Consider a mapping $\Phi : \mathcal{V} \to \mathcal{W}$, where $\mathcal{V}, \mathcal{W}$ can be arbitrary sets. Then Φ is called

injective
surjective
bijective

- *Injective* if $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{V} : \Phi(\boldsymbol{x}) = \Phi(\boldsymbol{y}) \implies \boldsymbol{x} = \boldsymbol{y}$.
- *Surjective* if $\Phi(\mathcal{V}) = \mathcal{W}$.
- *Bijective* if it is injective and surjective.

If Φ is surjective, then every element in $\mathcal{W}$ can be "reached" from $\mathcal{V}$ using Φ. A bijective Φ can be "undone", i.e., there exists a mapping $\Psi : \mathcal{W} \to \mathcal{V}$ so that $\Psi \circ \Phi(\boldsymbol{x}) = \boldsymbol{x}$. This mapping Ψ is then called the inverse of Φ and normally denoted by Φ^{-1}.

With these definitions, we introduce the following special cases of linear mappings between vector spaces V and W:

isomorphism
endomorphism
automorphism
identity mapping
identity automorphism

- *Isomorphism:* $\Phi : V \to W$ linear and bijective
- *Endomorphism:* $\Phi : V \to V$ linear
- *Automorphism:* $\Phi : V \to V$ linear and bijective
- We define $\mathrm{id}_V : V \to V$, $\boldsymbol{x} \mapsto \boldsymbol{x}$ as the *identity mapping* or *identity automorphism* in V.

Example 2.19 (Homomorphism)
The mapping $\Phi : \mathbb{R}^2 \to \mathbb{C},\ \Phi(\boldsymbol{x}) = x_1 + ix_2$, is a homomorphism:

$$\begin{aligned}\Phi\left(\begin{bmatrix}x_1\\x_2\end{bmatrix} + \begin{bmatrix}y_1\\y_2\end{bmatrix}\right) &= (x_1 + y_1) + i(x_2 + y_2) = x_1 + ix_2 + y_1 + iy_2 \\ &= \Phi\left(\begin{bmatrix}x_1\\x_2\end{bmatrix}\right) + \Phi\left(\begin{bmatrix}y_1\\y_2\end{bmatrix}\right) \\ \Phi\left(\lambda\begin{bmatrix}x_1\\x_2\end{bmatrix}\right) &= \lambda x_1 + \lambda i x_2 = \lambda(x_1 + ix_2) = \lambda\Phi\left(\begin{bmatrix}x_1\\x_2\end{bmatrix}\right).\end{aligned} \tag{2.88}$$

This also justifies why complex numbers can be represented as tuples in $\mathbb{R}^2$: There is a bijective linear mapping that converts the elementwise addition of tuples in $\mathbb{R}^2$ into the set of complex numbers with the corresponding addition. Note that we only showed linearity, but not the bijection.

Theorem 2.17 (Theorem 3.59 in Axler (2015))**.** *Finite-dimensional vector spaces V and W are isomorphic if and only if* $\dim(V) = \dim(W)$.

Theorem 2.17 states that there exists a linear, bijective mapping between two vector spaces of the same dimension. Intuitively, this means that vector spaces of the same dimension are kind of the same thing, as they can be transformed into each other without incurring any loss.

Theorem 2.17 also gives us the justification to treat $\mathbb{R}^{m\times n}$ (the vector space of $m \times n$-matrices) and $\mathbb{R}^{mn}$ (the vector space of vectors of length mn) the same, as their dimensions are mn, and there exists a linear, bijective mapping that transforms one into the other.

Remark. Consider vector spaces V, W, X. Then:

- For linear mappings $\Phi : V \to W$ and $\Psi : W \to X$, the mapping $\Psi \circ \Phi : V \to X$ is also linear.
- If $\Phi : V \to W$ is an isomorphism, then $\Phi^{-1} : W \to V$ is an isomorphism, too.

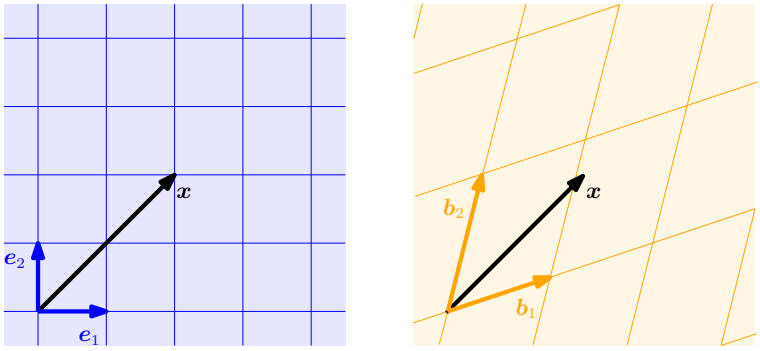


Figure 2.8 Two different coordinate systems defined by two sets of basis vectors. A vector $\boldsymbol{x}$ has different coordinate representations depending on which coordinate system is chosen.

- If $\Phi : V \to W,\ \Psi : V \to W$ are linear, then $\Phi + \Psi$ and $\lambda\Phi,\ \lambda \in \mathbb{R}$, are linear, too.

♢

2.7.1 Matrix Representation of Linear Mappings

Any n-dimensional vector space is isomorphic to $\mathbb{R}^n$ (Theorem 2.17). We consider a basis $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ of an n-dimensional vector space V. In the following, the order of the basis vectors will be important. Therefore, we write

$$B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \tag{2.89}$$

ordered basis

and call this n-tuple an *ordered basis* of V.

Remark (Notation). We are at the point where notation gets a bit tricky. Therefore, we summarize some parts here. $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ is an ordered basis, $\mathcal{B} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ is an (unordered) basis, and $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n]$ is a matrix whose columns are the vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$. ♢

Definition 2.18 (Coordinates)**.** Consider a vector space V and an ordered basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ of V. For any $\boldsymbol{x} \in V$ we obtain a unique representation (linear combination)

$$\boldsymbol{x} = \alpha_1\boldsymbol{b}_1 + \ldots + \alpha_n\boldsymbol{b}_n \tag{2.90}$$

coordinate

of $\boldsymbol{x}$ with respect to B. Then $\alpha_1, \ldots, \alpha_n$ are the *coordinates* of $\boldsymbol{x}$ with respect to B, and the vector

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} \in \mathbb{R}^n \tag{2.91}$$

coordinate vector
coordinate representation

is the *coordinate vector/coordinate representation* of $\boldsymbol{x}$ with respect to the ordered basis B.

A basis effectively defines a coordinate system. We are familiar with the Cartesian coordinate system in two dimensions, which is spanned by the canonical basis vectors $\boldsymbol{e}_1, \boldsymbol{e}_2$. In this coordinate system, a vector $\boldsymbol{x} \in \mathbb{R}^2$ has a representation that tells us how to linearly combine $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ to obtain $\boldsymbol{x}$. However, any basis of $\mathbb{R}^2$ defines a valid coordinate system, and the same vector $\boldsymbol{x}$ from before may have a different coordinate representation in the $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ basis. In Figure 2.8, the coordinates of $\boldsymbol{x}$ with respect to the standard basis $(\boldsymbol{e}_1, \boldsymbol{e}_2)$ is $[2, 2]^\top$. However, with respect to the basis $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ the same vector $\boldsymbol{x}$ is represented as $[1.09, 0.72]^\top$, i.e., $\boldsymbol{x} = 1.09\boldsymbol{b}_1 + 0.72\boldsymbol{b}_2$. In the following sections, we will discover how to obtain this representation.

Example 2.20

Let us have a look at a geometric vector $\boldsymbol{x} \in \mathbb{R}^2$ with coordinates $[2, 3]^\top$ with respect to the standard basis $(\boldsymbol{e}_1, \boldsymbol{e}_2)$ of $\mathbb{R}^2$. This means, we can write $\boldsymbol{x} = 2\boldsymbol{e}_1 + 3\boldsymbol{e}_2$. However, we do not have to choose the standard basis to represent this vector. If we use the basis vectors $\boldsymbol{b}_1 = [1, -1]^\top, \boldsymbol{b}_2 = [1, 1]^\top$ we will obtain the coordinates $\frac{1}{2}[-1, 5]^\top$ to represent the same vector with respect to $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ (see Figure 2.9).

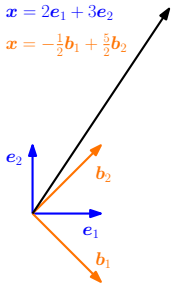


Figure 2.9 Different coordinate representations of a vector $\boldsymbol{x}$, depending on the choice of basis.

Remark. For an n-dimensional vector space V and an ordered basis B of V, the mapping $\Phi : \mathbb{R}^n \to V$, $\Phi(\boldsymbol{e}_i) = \boldsymbol{b}_i$, $i = 1, \ldots, n$, is linear (and because of Theorem 2.17 an isomorphism), where $(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n)$ is the standard basis of $\mathbb{R}^n$. ♢

Now we are ready to make an explicit connection between matrices and linear mappings between finite-dimensional vector spaces.

Definition 2.19 (Transformation Matrix)**.** Consider vector spaces V, W with corresponding (ordered) bases $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ and $C = (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_m)$. Moreover, we consider a linear mapping $\Phi : V \to W$. For $j \in \{1, \ldots, n\}$,

$$\Phi(\boldsymbol{b}_j) = \alpha_{1j}\boldsymbol{c}_1 + \cdots + \alpha_{mj}\boldsymbol{c}_m = \sum_{i=1}^{m} \alpha_{ij}\boldsymbol{c}_i \tag{2.92}$$

is the unique representation of $\Phi(\boldsymbol{b}_j)$ with respect to C. Then, we call the $m \times n$-matrix $\boldsymbol{A}_\Phi$, whose elements are given by

$$A_\Phi(i, j) = \alpha_{ij}\,, \tag{2.93}$$

the *transformation matrix* of Φ (with respect to the ordered bases B of V and C of W). transformation matrix

The coordinates of $\Phi(\boldsymbol{b}_j)$ with respect to the ordered basis C of W are the j-th column of $\boldsymbol{A}_\Phi$. Consider (finite-dimensional) vector spaces V, W with ordered bases B, C and a linear mapping $\Phi : V \to W$ with

transformation matrix $\boldsymbol{A}_\Phi$. If $\hat{\boldsymbol{x}}$ is the coordinate vector of $\boldsymbol{x} \in V$ with respect to B and $\hat{\boldsymbol{y}}$ the coordinate vector of $\boldsymbol{y} = \Phi(\boldsymbol{x}) \in W$ with respect to C, then

$$\hat{\boldsymbol{y}} = \boldsymbol{A}_\Phi \hat{\boldsymbol{x}} \,. \tag{2.94}$$

This means that the transformation matrix can be used to map coordinates with respect to an ordered basis in V to coordinates with respect to an ordered basis in W.

Example 2.21 (Transformation Matrix)
Consider a homomorphism $\Phi : V \to W$ and ordered bases $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_3)$ of V and $C = (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_4)$ of W. With

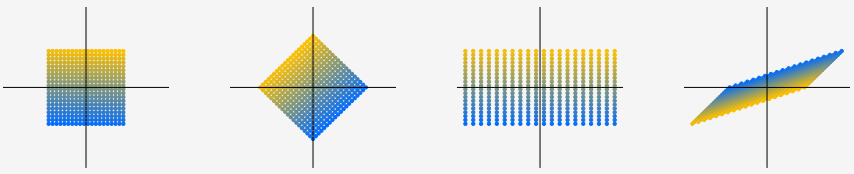
$$\begin{aligned} \Phi(\boldsymbol{b}_1) &= \boldsymbol{c}_1 - \boldsymbol{c}_2 + 3\boldsymbol{c}_3 - \boldsymbol{c}_4 \\ \Phi(\boldsymbol{b}_2) &= 2\boldsymbol{c}_1 + \boldsymbol{c}_2 + 7\boldsymbol{c}_3 + 2\boldsymbol{c}_4 \\ \Phi(\boldsymbol{b}_3) &= 3\boldsymbol{c}_2 + \boldsymbol{c}_3 + 4\boldsymbol{c}_4 \end{aligned} \tag{2.95}$$

the transformation matrix $\boldsymbol{A}_\Phi$ with respect to B and C satisfies $\Phi(\boldsymbol{b}_k) = \sum_{i=1}^{4} \alpha_{ik} \boldsymbol{c}_i$ for $k = 1, \ldots, 3$ and is given as

$$\boldsymbol{A}_\Phi = [\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha}_3] = \begin{bmatrix} 1 & 2 & 0 \\ -1 & 1 & 3 \\ 3 & 7 & 1 \\ -1 & 2 & 4 \end{bmatrix} , \tag{2.96}$$

where the $\boldsymbol{\alpha}_j,\ j = 1, 2, 3$, are the coordinate vectors of $\Phi(\boldsymbol{b}_j)$ with respect to C.

Example 2.22 (Linear Transformations of Vectors)



(a) Original data. (b) Rotation by $45°$. (c) Stretch along the horizontal axis. (d) General linear mapping.

Figure 2.10 Three examples of linear transformations of the vectors shown as dots in (a); (b) Rotation by $45°$; (c) Stretching of the horizontal coordinates by 2; (d) Combination of reflection, rotation and stretching.

We consider three linear transformations of a set of vectors in $\mathbb{R}^2$ with the transformation matrices

$$\boldsymbol{A}_1 = \begin{bmatrix} \cos(\frac{\pi}{4}) & -\sin(\frac{\pi}{4}) \\ \sin(\frac{\pi}{4}) & \cos(\frac{\pi}{4}) \end{bmatrix} , \quad \boldsymbol{A}_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} , \quad \boldsymbol{A}_3 = \frac{1}{2} \begin{bmatrix} 3 & -1 \\ 1 & -1 \end{bmatrix} . \tag{2.97}$$

Figure 2.10 gives three examples of linear transformations of a set of vectors. Figure 2.10(a) shows 400 vectors in $\mathbb{R}^2$, each of which is represented by a dot at the corresponding (x_1, x_2)-coordinates. The vectors are arranged in a square. When we use matrix $\boldsymbol{A}_1$ in (2.97) to linearly transform each of these vectors, we obtain the rotated square in Figure 2.10(b). If we apply the linear mapping represented by $\boldsymbol{A}_2$, we obtain the rectangle in Figure 2.10(c) where each x_1-coordinate is stretched by 2. Figure 2.10(d) shows the original square from Figure 2.10(a) when linearly transformed using $\boldsymbol{A}_3$, which is a combination of a reflection, a rotation, and a stretch.

2.7.2 Basis Change

In the following, we will have a closer look at how transformation matrices of a linear mapping $\Phi : V \to W$ change if we change the bases in V and W. Consider two ordered bases

$$B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n), \quad \tilde{B} = (\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n) \tag{2.98}$$

of V and two ordered bases

$$C = (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_m), \quad \tilde{C} = (\tilde{\boldsymbol{c}}_1, \ldots, \tilde{\boldsymbol{c}}_m) \tag{2.99}$$

of W. Moreover, $\boldsymbol{A}_\Phi \in \mathbb{R}^{m \times n}$ is the transformation matrix of the linear mapping $\Phi : V \to W$ with respect to the bases B and C, and $\tilde{\boldsymbol{A}}_\Phi \in \mathbb{R}^{m \times n}$ is the corresponding transformation mapping with respect to $\tilde{B}$ and $\tilde{C}$. In the following, we will investigate how $\boldsymbol{A}$ and $\tilde{\boldsymbol{A}}$ are related, i.e., how/ whether we can transform $\boldsymbol{A}_\Phi$ into $\tilde{\boldsymbol{A}}_\Phi$ if we choose to perform a basis change from B, C to $\tilde{B}, \tilde{C}$.

Remark. We effectively get different coordinate representations of the identity mapping id_V. In the context of Figure 2.9, this would mean to map coordinates with respect to $(\boldsymbol{e}_1, \boldsymbol{e}_2)$ onto coordinates with respect to $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ without changing the vector $\boldsymbol{x}$. By changing the basis and correspondingly the representation of vectors, the transformation matrix with respect to this new basis can have a particularly simple form that allows for straightforward computation. ♢

Example 2.23 (Basis Change)
Consider a transformation matrix

$$\boldsymbol{A} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{2.100}$$

with respect to the canonical basis in $\mathbb{R}^2$. If we define a new basis

$$B = (\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \end{bmatrix}) \tag{2.101}$$

we obtain a diagonal transformation matrix

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \tag{2.102}$$

with respect to B, which is easier to work with than $\boldsymbol{A}$.

In the following, we will look at mappings that transform coordinate vectors with respect to one basis into coordinate vectors with respect to a different basis. We will state our main result first and then provide an explanation.

Theorem 2.20 (Basis Change)**.** *For a linear mapping $\Phi : V \to W$, ordered bases*

$$B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n), \quad \tilde{B} = (\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n) \tag{2.103}$$

of V and

$$C = (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_m), \quad \tilde{C} = (\tilde{\boldsymbol{c}}_1, \ldots, \tilde{\boldsymbol{c}}_m) \tag{2.104}$$

of W, and a transformation matrix $\boldsymbol{A}_\Phi$ of Φ with respect to B and C, the corresponding transformation matrix $\tilde{\boldsymbol{A}}_\Phi$ with respect to the bases $\tilde{B}$ and $\tilde{C}$ is given as

$$\tilde{\boldsymbol{A}}_\Phi = \boldsymbol{T}^{-1} \boldsymbol{A}_\Phi \boldsymbol{S} \,. \tag{2.105}$$

Here, $\boldsymbol{S} \in \mathbb{R}^{n \times n}$ is the transformation matrix of id_V that maps coordinates with respect to $\tilde{B}$ onto coordinates with respect to B, and $\boldsymbol{T} \in \mathbb{R}^{m \times m}$ is the transformation matrix of id_W that maps coordinates with respect to $\tilde{C}$ onto coordinates with respect to C.

Proof Following Drumm and Weil (2001), we can write the vectors of the new basis $\tilde{B}$ of V as a linear combination of the basis vectors of B, such that

$$\tilde{\boldsymbol{b}}_j = s_{1j}\boldsymbol{b}_1 + \cdots + s_{nj}\boldsymbol{b}_n = \sum_{i=1}^{n} s_{ij}\boldsymbol{b}_i \,, \quad j = 1, \ldots, n \,. \tag{2.106}$$

Similarly, we write the new basis vectors $\tilde{C}$ of W as a linear combination of the basis vectors of C, which yields

$$\tilde{\boldsymbol{c}}_k = t_{1k}\boldsymbol{c}_1 + \cdots + t_{mk}\boldsymbol{c}_m = \sum_{l=1}^{m} t_{lk}\boldsymbol{c}_l \,, \quad k = 1, \ldots, m \,. \tag{2.107}$$

We define $\boldsymbol{S} = ((s_{ij})) \in \mathbb{R}^{n \times n}$ as the transformation matrix that maps coordinates with respect to $\tilde{B}$ onto coordinates with respect to B and $\boldsymbol{T} = ((t_{lk})) \in \mathbb{R}^{m \times m}$ as the transformation matrix that maps coordinates with respect to $\tilde{C}$ onto coordinates with respect to C. In particular, the jth column of $\boldsymbol{S}$ is the coordinate representation of $\tilde{\boldsymbol{b}}_j$ with respect to B and

the kth column of $\boldsymbol{T}$ is the coordinate representation of $\tilde{\boldsymbol{c}}_k$ with respect to C. Note that both $\boldsymbol{S}$ and $\boldsymbol{T}$ are regular.

We are going to look at $\Phi(\tilde{\boldsymbol{b}}_j)$ from two perspectives. First, applying the mapping Φ, we get that for all $j = 1, \dots, n$

$$\Phi(\tilde{\boldsymbol{b}}_j) = \sum_{k=1}^{m} \underbrace{\tilde{a}_{kj} \tilde{\boldsymbol{c}}_k}_{\in W} \overset{(2.107)}{=} \sum_{k=1}^{m} \tilde{a}_{kj} \sum_{l=1}^{m} t_{lk} \boldsymbol{c}_l = \sum_{l=1}^{m} \left(\sum_{k=1}^{m} t_{lk} \tilde{a}_{kj} \right) \boldsymbol{c}_l \,, \quad (2.108)$$

where we first expressed the new basis vectors $\tilde{\boldsymbol{c}}_k \in W$ as linear combinations of the basis vectors $\boldsymbol{c}_l \in W$ and then swapped the order of summation.

Alternatively, when we express the $\tilde{\boldsymbol{b}}_j \in V$ as linear combinations of $\boldsymbol{b}_j \in V$, we arrive at

$$\Phi(\tilde{\boldsymbol{b}}_j) \overset{(2.106)}{=} \Phi \left(\sum_{i=1}^{n} s_{ij} \boldsymbol{b}_i \right) = \sum_{i=1}^{n} s_{ij} \Phi(\boldsymbol{b}_i) = \sum_{i=1}^{n} s_{ij} \sum_{l=1}^{m} a_{li} \boldsymbol{c}_l \quad (2.109a)$$

$$= \sum_{l=1}^{m} \left(\sum_{i=1}^{n} a_{li} s_{ij} \right) \boldsymbol{c}_l \,, \quad j = 1, \dots, n \,, \quad (2.109b)$$

where we exploited the linearity of Φ. Comparing (2.108) and (2.109b), it follows for all $j = 1, \dots, n$ and $l = 1, \dots, m$ that

$$\sum_{k=1}^{m} t_{lk} \tilde{a}_{kj} = \sum_{i=1}^{n} a_{li} s_{ij} \quad (2.110)$$

and, therefore,

$$\boldsymbol{T} \tilde{\boldsymbol{A}}_\Phi = \boldsymbol{A}_\Phi \boldsymbol{S} \in \mathbb{R}^{m \times n} \,, \quad (2.111)$$

such that

$$\tilde{\boldsymbol{A}}_\Phi = \boldsymbol{T}^{-1} \boldsymbol{A}_\Phi \boldsymbol{S} \,, \quad (2.112)$$

which proves Theorem 2.20. □

Theorem 2.20 tells us that with a basis change in V (B is replaced with $\tilde{B}$) and W (C is replaced with $\tilde{C}$), the transformation matrix $\boldsymbol{A}_\Phi$ of a linear mapping $\Phi : V \to W$ is replaced by an equivalent matrix $\tilde{\boldsymbol{A}}_\Phi$ with

$$\tilde{\boldsymbol{A}}_\Phi = \boldsymbol{T}^{-1} \boldsymbol{A}_\Phi \boldsymbol{S}. \quad (2.113)$$

Figure 2.11 illustrates this relation: Consider a homomorphism $\Phi : V \to W$ and ordered bases $B, \tilde{B}$ of V and $C, \tilde{C}$ of W. The mapping Φ_{CB} is an instantiation of Φ and maps basis vectors of B onto linear combinations of basis vectors of C. Assume that we know the transformation matrix $\boldsymbol{A}_\Phi$ of Φ_{CB} with respect to the ordered bases B, C. When we perform a basis change from B to $\tilde{B}$ in V and from C to $\tilde{C}$ in W, we can determine the

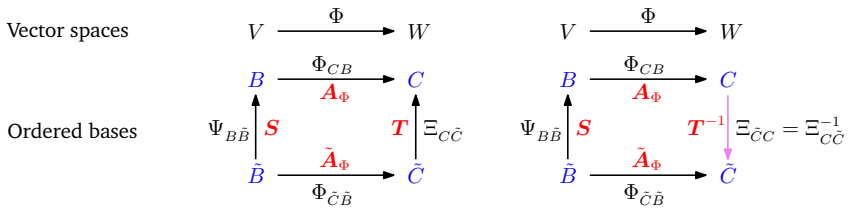


Figure 2.11 For a homomorphism $\Phi : V \to W$ and ordered bases $B, \tilde{B}$ of V and $C, \tilde{C}$ of W (marked in blue), we can express the mapping $\Phi_{\tilde{C}\tilde{B}}$ with respect to the bases $\tilde{B}, \tilde{C}$ equivalently as a composition of the homomorphisms $\Phi_{\tilde{C}\tilde{B}} = \Xi_{\tilde{C}C} \circ \Phi_{CB} \circ \Psi_{B\tilde{B}}$ with respect to the bases in the subscripts. The corresponding transformation matrices are in red.

corresponding transformation matrix $\tilde{\boldsymbol{A}}_\Phi$ as follows: First, we find the matrix representation of the linear mapping $\Psi_{B\tilde{B}} : V \to V$ that maps coordinates with respect to the new basis $\tilde{B}$ onto the (unique) coordinates with respect to the "old" basis B (in V). Then, we use the transformation matrix $\boldsymbol{A}_\Phi$ of $\Phi_{CB} : V \to W$ to map these coordinates onto the coordinates with respect to C in W. Finally, we use a linear mapping $\Xi_{\tilde{C}C} : W \to W$ to map the coordinates with respect to C onto coordinates with respect to $\tilde{C}$. Therefore, we can express the linear mapping $\Phi_{\tilde{C}\tilde{B}}$ as a composition of linear mappings that involve the "old" basis:

$$\Phi_{\tilde{C}\tilde{B}} = \Xi_{\tilde{C}C} \circ \Phi_{CB} \circ \Psi_{B\tilde{B}} = \Xi^{-1}_{C\tilde{C}} \circ \Phi_{CB} \circ \Psi_{B\tilde{B}} \,. \tag{2.114}$$

Concretely, we use $\Psi_{B\tilde{B}} = \mathrm{id}_V$ and $\Xi_{C\tilde{C}} = \mathrm{id}_W$, i.e., the identity mappings that map vectors onto themselves, but with respect to a different basis.

equivalent

Definition 2.21 (Equivalence)**.** Two matrices $\boldsymbol{A}, \tilde{\boldsymbol{A}} \in \mathbb{R}^{m \times n}$ are *equivalent* if there exist regular matrices $\boldsymbol{S} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{T} \in \mathbb{R}^{m \times m}$, such that $\tilde{\boldsymbol{A}} = \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{S}$.

similar

Definition 2.22 (Similarity)**.** Two matrices $\boldsymbol{A}, \tilde{\boldsymbol{A}} \in \mathbb{R}^{n \times n}$ are *similar* if there exists a regular matrix $\boldsymbol{S} \in \mathbb{R}^{n \times n}$ with $\tilde{\boldsymbol{A}} = \boldsymbol{S}^{-1}\boldsymbol{A}\boldsymbol{S}$

Remark. Similar matrices are always equivalent. However, equivalent matrices are not necessarily similar. ♢

Remark. Consider vector spaces V, W, X. From the remark that follows Theorem 2.17, we already know that for linear mappings $\Phi : V \to W$ and $\Psi : W \to X$ the mapping $\Psi \circ \Phi : V \to X$ is also linear. With transformation matrices $\boldsymbol{A}_\Phi$ and $\boldsymbol{A}_\Psi$ of the corresponding mappings, the overall transformation matrix is $\boldsymbol{A}_{\Psi \circ \Phi} = \boldsymbol{A}_\Psi \boldsymbol{A}_\Phi$. ♢

In light of this remark, we can look at basis changes from the perspective of composing linear mappings:

- $\boldsymbol{A}_\Phi$ is the transformation matrix of a linear mapping $\Phi_{CB} : V \to W$ with respect to the bases B, C.
- $\tilde{\boldsymbol{A}}_\Phi$ is the transformation matrix of the linear mapping $\Phi_{\tilde{C}\tilde{B}} : V \to W$ with respect to the bases $\tilde{B}, \tilde{C}$.
- $\boldsymbol{S}$ is the transformation matrix of a linear mapping $\Psi_{B\tilde{B}} : V \to V$ (automorphism) that represents $\tilde{B}$ in terms of B. Normally, $\Psi = \mathrm{id}_V$ is the identity mapping in V.

- $\boldsymbol{T}$ is the transformation matrix of a linear mapping $\Xi_{C\tilde{C}} : W \to W$ (automorphism) that represents $\tilde{C}$ in terms of C. Normally, $\Xi = \mathrm{id}_W$ is the identity mapping in W.

If we (informally) write down the transformations just in terms of bases, then $\boldsymbol{A}_\Phi : B \to C$, $\tilde{\boldsymbol{A}}_\Phi : \tilde{B} \to \tilde{C}$, $\boldsymbol{S} : \tilde{B} \to B$, $\boldsymbol{T} : \tilde{C} \to C$ and $\boldsymbol{T}^{-1} : C \to \tilde{C}$, and

$$\tilde{B} \to \tilde{C} = \tilde{B} \to B \to C \to \tilde{C} \tag{2.115}$$

$$\tilde{\boldsymbol{A}}_\Phi = \boldsymbol{T}^{-1} \boldsymbol{A}_\Phi \boldsymbol{S} . \tag{2.116}$$

Note that the execution order in (2.116) is from right to left because vectors are multiplied at the right-hand side so that $\boldsymbol{x} \mapsto \boldsymbol{S}\boldsymbol{x} \mapsto \boldsymbol{A}_\Phi(\boldsymbol{S}\boldsymbol{x}) \mapsto \boldsymbol{T}^{-1}\big(\boldsymbol{A}_\Phi(\boldsymbol{S}\boldsymbol{x})\big) = \tilde{\boldsymbol{A}}_\Phi \boldsymbol{x}$.

Example 2.24 (Basis Change)
Consider a linear mapping $\Phi : \mathbb{R}^3 \to \mathbb{R}^4$ whose transformation matrix is

$$\boldsymbol{A}_\Phi = \begin{bmatrix} 1 & 2 & 0 \\ -1 & 1 & 3 \\ 3 & 7 & 1 \\ -1 & 2 & 4 \end{bmatrix} \tag{2.117}$$

with respect to the standard bases

$$B = \left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right), \quad C = \left(\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right). \tag{2.118}$$

We seek the transformation matrix $\tilde{\boldsymbol{A}}_\Phi$ of Φ with respect to the new bases

$$\tilde{B} = \left(\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \right) \in \mathbb{R}^3, \quad \tilde{C} = \left(\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right). \tag{2.119}$$

Then,

$$\boldsymbol{S} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \qquad \boldsymbol{T} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.120}$$

where the ith column of $\boldsymbol{S}$ is the coordinate representation of $\tilde{\boldsymbol{b}}_i$ in terms of the basis vectors of B. Since B is the standard basis, the coordinate representation is straightforward to find. For a general basis B, we would need to solve a linear equation system to find the λ_i such that

$\sum_{i=1}^{3} \lambda_i \boldsymbol{b}_i = \tilde{\boldsymbol{b}}_j$, $j = 1, \ldots, 3$. Similarly, the jth column of $\boldsymbol{T}$ is the coordinate representation of $\tilde{\boldsymbol{c}}_j$ in terms of the basis vectors of C.

Therefore, we obtain

$$\tilde{\boldsymbol{A}}_\Phi = \boldsymbol{T}^{-1} \boldsymbol{A}_\Phi \boldsymbol{S} = \frac{1}{2} \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} 3 & 2 & 1 \\ 0 & 4 & 2 \\ 10 & 8 & 4 \\ 1 & 6 & 3 \end{bmatrix} \tag{2.121a}$$

$$= \begin{bmatrix} -4 & -4 & -2 \\ 6 & 0 & 0 \\ 4 & 8 & 4 \\ 1 & 6 & 3 \end{bmatrix}. \tag{2.121b}$$

In Chapter 4, we will be able to exploit the concept of a basis change to find a basis with respect to which the transformation matrix of an endomorphism has a particularly simple (diagonal) form. In Chapter 10, we will look at a data compression problem and find a convenient basis onto which we can project the data while minimizing the compression loss.

2.7.3 Image and Kernel

The image and kernel of a linear mapping are vector subspaces with certain important properties. In the following, we will characterize them more carefully.

Definition 2.23 (Image and Kernel)**.**

kernel
null space

For $\Phi : V \to W$, we define the *kernel/null space*

$$\ker(\Phi) := \Phi^{-1}(\boldsymbol{0}_W) = \{\boldsymbol{v} \in V : \Phi(\boldsymbol{v}) = \boldsymbol{0}_W\} \tag{2.122}$$

image
range

and the *image/range*

$$\operatorname{Im}(\Phi) := \Phi(V) = \{\boldsymbol{w} \in W | \exists \boldsymbol{v} \in V : \Phi(\boldsymbol{v}) = \boldsymbol{w}\}. \tag{2.123}$$

domain
codomain

We also call V and W also the *domain* and *codomain* of Φ, respectively.

Intuitively, the kernel is the set of vectors $\boldsymbol{v} \in V$ that Φ maps onto the neutral element $\boldsymbol{0}_W \in W$. The image is the set of vectors $\boldsymbol{w} \in W$ that can be "reached" by Φ from any vector in V. An illustration is given in Figure 2.12.

Remark. Consider a linear mapping $\Phi : V \to W$, where V, W are vector spaces.

- It always holds that $\Phi(\boldsymbol{0}_V) = \boldsymbol{0}_W$ and, therefore, $\boldsymbol{0}_V \in \ker(\Phi)$. In particular, the null space is never empty.
- $\operatorname{Im}(\Phi) \subseteq W$ is a subspace of W, and $\ker(\Phi) \subseteq V$ is a subspace of V.

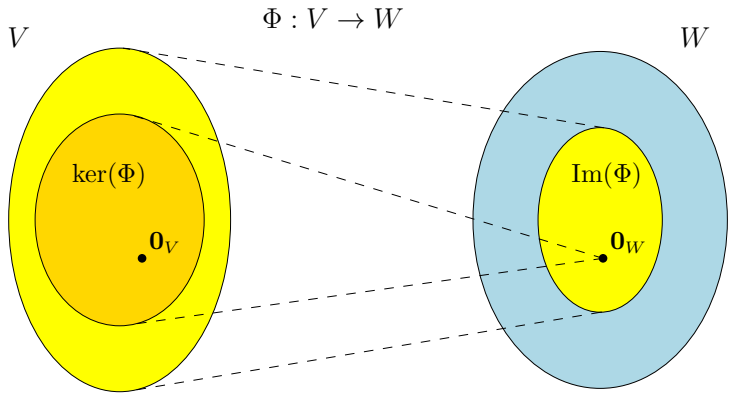


Figure 2.12 Kernel and image of a linear mapping $\Phi : V \to W$.

- Φ is injective (one-to-one) if and only if $\ker(\Phi) = \{\mathbf{0}\}$.

♢

Remark (Null Space and Column Space). Let us consider $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and a linear mapping $\Phi : \mathbb{R}^n \to \mathbb{R}^m$, $\boldsymbol{x} \mapsto \boldsymbol{A}\boldsymbol{x}$.

- For $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n]$, where $\boldsymbol{a}_i$ are the columns of $\boldsymbol{A}$, we obtain

$$\mathrm{Im}(\Phi) = \{\boldsymbol{A}\boldsymbol{x} : \boldsymbol{x} \in \mathbb{R}^n\} = \left\{\sum_{i=1}^{n} x_i \boldsymbol{a}_i : x_1, \ldots, x_n \in \mathbb{R}\right\} \tag{2.124a}$$

$$= \mathrm{span}[\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n] \subseteq \mathbb{R}^m\,, \tag{2.124b}$$

  i.e., the image is the span of the columns of $\boldsymbol{A}$, also called the *column space*. Therefore, the column space (image) is a subspace of $\mathbb{R}^m$, where m is the "height" of the matrix. column space
- $\mathrm{rk}(\boldsymbol{A}) = \dim(\mathrm{Im}(\Phi))$.
- The kernel/null space $\ker(\Phi)$ is the general solution to the homogeneous system of linear equations $\boldsymbol{A}\boldsymbol{x} = \mathbf{0}$ and captures all possible linear combinations of the elements in $\mathbb{R}^n$ that produce $\mathbf{0} \in \mathbb{R}^m$.
- The kernel is a subspace of $\mathbb{R}^n$, where n is the "width" of the matrix.
- The kernel focuses on the relationship among the columns, and we can use it to determine whether/how we can express a column as a linear combination of other columns.

♢

Example 2.25 (Image and Kernel of a Linear Mapping)
The mapping

$$\Phi : \mathbb{R}^4 \to \mathbb{R}^2, \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \mapsto \begin{bmatrix} 1 & 2 & -1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 + 2x_2 - x_3 \\ x_1 + x_4 \end{bmatrix} \tag{2.125a}$$

$$= x_1 \begin{bmatrix}1\\1\end{bmatrix} + x_2 \begin{bmatrix}2\\0\end{bmatrix} + x_3 \begin{bmatrix}-1\\0\end{bmatrix} + x_4 \begin{bmatrix}0\\1\end{bmatrix} \quad (2.125b)$$

is linear. To determine $\mathrm{Im}(\Phi)$, we can take the span of the columns of the transformation matrix and obtain

$$\mathrm{Im}(\Phi) = \mathrm{span}[\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}2\\0\end{bmatrix}, \begin{bmatrix}-1\\0\end{bmatrix}, \begin{bmatrix}0\\1\end{bmatrix}]. \quad (2.126)$$

To compute the kernel (null space) of Φ, we need to solve $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$, i.e., we need to solve a homogeneous equation system. To do this, we use Gaussian elimination to transform $\boldsymbol{A}$ into reduced row-echelon form:

$$\begin{bmatrix}1 & 2 & -1 & 0\\ 1 & 0 & 0 & 1\end{bmatrix} \rightsquigarrow \cdots \rightsquigarrow \begin{bmatrix}1 & 0 & 0 & 1\\ 0 & 1 & -\frac{1}{2} & -\frac{1}{2}\end{bmatrix}. \quad (2.127)$$

This matrix is in reduced row-echelon form, and we can use the Minus-1 Trick to compute a basis of the kernel (see Section 2.3.3). Alternatively, we can express the non-pivot columns (columns 3 and 4) as linear combinations of the pivot columns (columns 1 and 2). The third column $\boldsymbol{a}_3$ is equivalent to $-\frac{1}{2}$ times the second column $\boldsymbol{a}_2$. Therefore, $\boldsymbol{0} = \boldsymbol{a}_3 + \frac{1}{2}\boldsymbol{a}_2$. In the same way, we see that $\boldsymbol{a}_4 = \boldsymbol{a}_1 - \frac{1}{2}\boldsymbol{a}_2$ and, therefore, $\boldsymbol{0} = \boldsymbol{a}_1 - \frac{1}{2}\boldsymbol{a}_2 - \boldsymbol{a}_4$. Overall, this gives us the kernel (null space) as

$$\ker(\Phi) = \mathrm{span}[\begin{bmatrix}0\\ \frac{1}{2}\\ 1\\ 0\end{bmatrix}, \begin{bmatrix}-1\\ \frac{1}{2}\\ 0\\ 1\end{bmatrix}]. \quad (2.128)$$

rank-nullity theorem

Theorem 2.24 (Rank-Nullity Theorem)**.** *For vector spaces V, W and a linear mapping $\Phi : V \to W$ it holds that*

$$\dim(\ker(\Phi)) + \dim(\mathrm{Im}(\Phi)) = \dim(V). \quad (2.129)$$

fundamental theorem of linear mappings

The rank-nullity theorem is also referred to as the *fundamental theorem of linear mappings* (Axler, 2015, theorem 3.22). The following are direct consequences of Theorem 2.24:

- If $\dim(\mathrm{Im}(\Phi)) < \dim(V)$, then $\ker(\Phi)$ is non-trivial, i.e., the kernel contains more than $\boldsymbol{0}_V$ and $\dim(\ker(\Phi)) \geqslant 1$.
- If $\boldsymbol{A}_\Phi$ is the transformation matrix of Φ with respect to an ordered basis and $\dim(\mathrm{Im}(\Phi)) < \dim(V)$, then the system of linear equations $\boldsymbol{A}_\Phi \boldsymbol{x} = \boldsymbol{0}$ has infinitely many solutions.
- If $\dim(V) = \dim(W)$, then the three-way equivalence

  $$\Phi \text{ is injective} \iff \Phi \text{ is surjective} \iff \Phi \text{ is bijective}$$

  holds since $\mathrm{Im}(\Phi) \subseteq W$.

2.8 Affine Spaces

In the following, we will take a closer look at spaces that are offset from the origin, i.e., spaces that are no longer vector subspaces. Moreover, we will briefly discuss properties of mappings between these affine spaces, which resemble linear mappings.

Remark. In the machine learning literature, the distinction between linear and affine is sometimes not clear so that we can find references to affine spaces/mappings as linear spaces/mappings. ♢

2.8.1 Affine Subspaces

Definition 2.25 (Affine Subspace)**.** Let V be a vector space, $\boldsymbol{x}_0 \in V$ and $U \subseteq V$ a subspace. Then the subset

$$L = \boldsymbol{x}_0 + U := \{\boldsymbol{x}_0 + \boldsymbol{u} : \boldsymbol{u} \in U\} \tag{2.130a}$$
$$= \{\boldsymbol{v} \in V | \exists \boldsymbol{u} \in U : \boldsymbol{v} = \boldsymbol{x}_0 + \boldsymbol{u}\} \subseteq V \tag{2.130b}$$

is called *affine subspace* or *linear manifold* of V. U is called *direction* or *direction space*, and $\boldsymbol{x}_0$ is called *support point*. In Chapter 12, we refer to such a subspace as a *hyperplane*.

affine subspace
linear manifold
direction
direction space
support point
hyperplane

Note that the definition of an affine subspace excludes $\mathbf{0}$ if $\boldsymbol{x}_0 \notin U$. Therefore, an affine subspace is not a (linear) subspace (vector subspace) of V for $\boldsymbol{x}_0 \notin U$.

Examples of affine subspaces are points, lines, and planes in $\mathbb{R}^3$, which do not (necessarily) go through the origin.

Remark. Consider two affine subspaces $L = \boldsymbol{x}_0 + U$ and $\tilde{L} = \tilde{\boldsymbol{x}}_0 + \tilde{U}$ of a vector space V. Then, $L \subseteq \tilde{L}$ if and only if $U \subseteq \tilde{U}$ and $\boldsymbol{x}_0 - \tilde{\boldsymbol{x}}_0 \in \tilde{U}$.

Affine subspaces are often described by *parameters*: Consider a k-dimensional affine space $L = \boldsymbol{x}_0 + U$ of V. If $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$ is an ordered basis of U, then every element $\boldsymbol{x} \in L$ can be uniquely described as

$$\boldsymbol{x} = \boldsymbol{x}_0 + \lambda_1 \boldsymbol{b}_1 + \ldots + \lambda_k \boldsymbol{b}_k \,, \tag{2.131}$$

where $\lambda_1, \ldots, \lambda_k \in \mathbb{R}$. This representation is called *parametric equation* of L with directional vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k$ and *parameters* $\lambda_1, \ldots, \lambda_k$. ♢

parametric equation
parameters

Example 2.26 (Affine Subspaces)

- One-dimensional affine subspaces are called *lines* and can be written as $\boldsymbol{y} = \boldsymbol{x}_0 + \lambda \boldsymbol{b}_1$, where $\lambda \in \mathbb{R}$ and $U = \operatorname{span}[\boldsymbol{b}_1] \subseteq \mathbb{R}^n$ is a one-dimensional subspace of $\mathbb{R}^n$. This means that a line is defined by a support point $\boldsymbol{x}_0$ and a vector $\boldsymbol{b}_1$ that defines the direction. See Figure 2.13 for an illustration.

line

plane

- Two-dimensional affine subspaces of $\mathbb{R}^n$ are called *planes*. The parametric equation for planes is $\boldsymbol{y} = \boldsymbol{x}_0 + \lambda_1 \boldsymbol{b}_1 + \lambda_2 \boldsymbol{b}_2$, where $\lambda_1, \lambda_2 \in \mathbb{R}$ and $U = \operatorname{span}[\boldsymbol{b}_1, \boldsymbol{b}_2] \subseteq \mathbb{R}^n$. This means that a plane is defined by a support point $\boldsymbol{x}_0$ and two linearly independent vectors $\boldsymbol{b}_1, \boldsymbol{b}_2$ that span the direction space.

hyperplane

- In $\mathbb{R}^n$, the $(n-1)$-dimensional affine subspaces are called *hyperplanes*, and the corresponding parametric equation is $\boldsymbol{y} = \boldsymbol{x}_0 + \sum_{i=1}^{n-1} \lambda_i \boldsymbol{b}_i$, where $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n-1}$ form a basis of an $(n-1)$-dimensional subspace U of $\mathbb{R}^n$. This means that a hyperplane is defined by a support point $\boldsymbol{x}_0$ and $(n-1)$ linearly independent vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n-1}$ that span the direction space. In $\mathbb{R}^2$, a line is also a hyperplane. In $\mathbb{R}^3$, a plane is also a hyperplane.

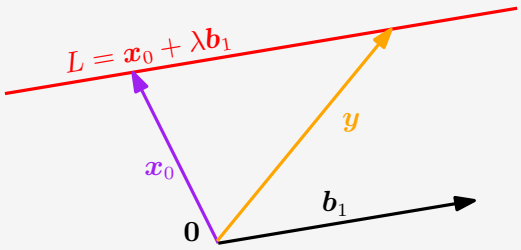


Figure 2.13 Lines are affine subspaces. Vectors $\boldsymbol{y}$ on a line $\boldsymbol{x}_0 + \lambda \boldsymbol{b}_1$ lie in an affine subspace L with support point $\boldsymbol{x}_0$ and direction $\boldsymbol{b}_1$.

Remark (Inhomogeneous systems of linear equations and affine subspaces). For $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{x} \in \mathbb{R}^m$, the solution of the system of linear equations $\boldsymbol{A}\boldsymbol{\lambda} = \boldsymbol{x}$ is either the empty set or an affine subspace of $\mathbb{R}^n$ of dimension $n - \operatorname{rk}(\boldsymbol{A})$. In particular, the solution of the linear equation $\lambda_1 \boldsymbol{b}_1 + \ldots + \lambda_n \boldsymbol{b}_n = \boldsymbol{x}$, where $(\lambda_1, \ldots, \lambda_n) \neq (0, \ldots, 0)$, is a hyperplane in $\mathbb{R}^n$.

In $\mathbb{R}^n$, every k-dimensional affine subspace is the solution of an inhomogeneous system of linear equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where $\boldsymbol{A} \in \mathbb{R}^{m \times n}, \boldsymbol{b} \in \mathbb{R}^m$ and $\operatorname{rk}(\boldsymbol{A}) = n - k$. Recall that for homogeneous equation systems $\boldsymbol{A}\boldsymbol{x} = \mathbf{0}$ the solution was a vector subspace, which we can also think of as a special affine space with support point $\boldsymbol{x}_0 = \mathbf{0}$. ♢

2.8.2 Affine Mappings

Similar to linear mappings between vector spaces, which we discussed in Section 2.7, we can define affine mappings between two affine spaces. Linear and affine mappings are closely related. Therefore, many properties that we already know from linear mappings, e.g., that the composition of linear mappings is a linear mapping, also hold for affine mappings.

Definition 2.26 (Affine Mapping)**.** For two vector spaces V, W, a linear

mapping $\Phi : V \to W$, and $\boldsymbol{a} \in W$, the mapping

$$\phi : V \to W \tag{2.132}$$
$$\boldsymbol{x} \mapsto \boldsymbol{a} + \Phi(\boldsymbol{x}) \tag{2.133}$$

is an *affine mapping* from V to W. The vector $\boldsymbol{a}$ is called the *translation vector* of ϕ.

affine mapping
translation vector

- Every affine mapping $\phi : V \to W$ is also the composition of a linear mapping $\Phi : V \to W$ and a translation $\tau : W \to W$ in W, such that $\phi = \tau \circ \Phi$. The mappings Φ and τ are uniquely determined.
- The composition $\phi' \circ \phi$ of affine mappings $\phi : V \to W$, $\phi' : W \to X$ is affine.
- If ϕ is bijective, affine mappings keep the geometric structure invariant. They then also preserve the dimension and parallelism.

2.9 Further Reading

There are many resources for learning linear algebra, including the textbooks by Strang (2003), Golan (2007), Axler (2015), and Liesen and Mehrmann (2015). There are also several online resources that we mentioned in the introduction to this chapter. We only covered Gaussian elimination here, but there are many other approaches for solving systems of linear equations, and we refer to numerical linear algebra textbooks by Stoer and Burlirsch (2002), Golub and Van Loan (2012), and Horn and Johnson (2013) for an in-depth discussion.

In this book, we distinguish between the topics of linear algebra (e.g., vectors, matrices, linear independence, basis) and topics related to the geometry of a vector space. In Chapter 3, we will introduce the inner product, which induces a norm. These concepts allow us to define angles, lengths and distances, which we will use for orthogonal projections. Projections turn out to be key in many machine learning algorithms, such as linear regression and principal component analysis, both of which we will cover in Chapters 9 and 10, respectively.

Exercises

2.1 We consider $(\mathbb{R}\backslash\{-1\}, \star)$, where

$$a \star b := ab + a + b, \qquad a, b \in \mathbb{R}\backslash\{-1\} \tag{2.134}$$

a. Show that $(\mathbb{R}\backslash\{-1\}, \star)$ is an Abelian group.
b. Solve

$$3 \star x \star x = 15$$

in the Abelian group $(\mathbb{R}\backslash\{-1\}, \star)$, where $\star$ is defined in (2.134).

2.2 Let n be in $\mathbb{N}\backslash\{0\}$. Let k, x be in $\mathbb{Z}$. We define the congruence class $\bar{k}$ of the integer k as the set

$$\begin{aligned}\bar{k} &= \{x \in \mathbb{Z} \mid x - k = 0 \pmod{n}\} \\ &= \{x \in \mathbb{Z} \mid \exists a \in \mathbb{Z}\colon (x - k = n \cdot a)\}\,.\end{aligned}$$

We now define $\mathbb{Z}/n\mathbb{Z}$ (sometimes written $\mathbb{Z}_n$) as the set of all congruence classes modulo n. Euclidean division implies that this set is a finite set containing n elements:

$$\mathbb{Z}_n = \{\bar{0}, \bar{1}, \ldots, \overline{n-1}\}$$

For all $\bar{a}, \bar{b} \in \mathbb{Z}_n$, we define

$$\bar{a} \oplus \bar{b} := \overline{a + b}$$

a. Show that $(\mathbb{Z}_n, \oplus)$ is a group. Is it Abelian?
b. We now define another operation $\otimes$ for all $\bar{a}$ and $\bar{b}$ in $\mathbb{Z}_n$ as

$$\bar{a} \otimes \bar{b} = \overline{a \times b}\,, \tag{2.135}$$

where $a \times b$ represents the usual multiplication in $\mathbb{Z}$.
Let $n = 5$. Draw the times table of the elements of $\mathbb{Z}_5\backslash\{\bar{0}\}$ under $\otimes$, i.e., calculate the products $\bar{a} \otimes \bar{b}$ for all $\bar{a}$ and $\bar{b}$ in $\mathbb{Z}_5\backslash\{\bar{0}\}$.
Hence, show that $\mathbb{Z}_5\backslash\{\bar{0}\}$ is closed under $\otimes$ and possesses a neutral element for $\otimes$. Display the inverse of all elements in $\mathbb{Z}_5\backslash\{\bar{0}\}$ under $\otimes$. Conclude that $(\mathbb{Z}_5\backslash\{\bar{0}\}, \otimes)$ is an Abelian group.
c. Show that $(\mathbb{Z}_8\backslash\{\bar{0}\}, \otimes)$ is not a group.
d. We recall that the Bézout theorem states that two integers a and b are relatively prime (i.e., $gcd(a, b) = 1$) if and only if there exist two integers u and v such that $au + bv = 1$. Show that $(\mathbb{Z}_n\backslash\{\bar{0}\}, \otimes)$ is a group if and only if $n \in \mathbb{N}\backslash\{0\}$ is prime.

2.3 Consider the set $\mathcal{G}$ of 3×3 matrices defined as follows:

$$\mathcal{G} = \left\{ \begin{bmatrix} 1 & x & z \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3} \,\middle|\, x, y, z \in \mathbb{R} \right\}$$

We define $\cdot$ as the standard matrix multiplication.
Is $(\mathcal{G}, \cdot)$ a group? If yes, is it Abelian? Justify your answer.

2.4 Compute the following matrix products, if possible:

a.

$$\begin{bmatrix}1 & 2\\ 4 & 5\\ 7 & 8\end{bmatrix}\begin{bmatrix}1 & 1 & 0\\ 0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix}$$

b.

$$\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{bmatrix}\begin{bmatrix}1 & 1 & 0\\ 0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix}$$

c.

$$\begin{bmatrix}1 & 1 & 0\\ 0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{bmatrix}$$

d.

$$\begin{bmatrix}1 & 2 & 1 & 2\\ 4 & 1 & -1 & -4\end{bmatrix}\begin{bmatrix}0 & 3\\ 1 & -1\\ 2 & 1\\ 5 & 2\end{bmatrix}$$

e.

$$\begin{bmatrix}0 & 3\\ 1 & -1\\ 2 & 1\\ 5 & 2\end{bmatrix}\begin{bmatrix}1 & 2 & 1 & 2\\ 4 & 1 & -1 & -4\end{bmatrix}$$

2.5 Find the set $\mathcal{S}$ of all solutions in $\boldsymbol{x}$ of the following inhomogeneous linear systems $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, where $\boldsymbol{A}$ and $\boldsymbol{b}$ are defined as follows:

a.

$$\boldsymbol{A} = \begin{bmatrix}1 & 1 & -1 & -1\\ 2 & 5 & -7 & -5\\ 2 & -1 & 1 & 3\\ 5 & 2 & -4 & 2\end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix}1\\ -2\\ 4\\ 6\end{bmatrix}$$

b.

$$\boldsymbol{A} = \begin{bmatrix}1 & -1 & 0 & 0 & 1\\ 1 & 1 & 0 & -3 & 0\\ 2 & -1 & 0 & 1 & -1\\ -1 & 2 & 0 & -2 & -1\end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix}3\\ 6\\ 5\\ -1\end{bmatrix}$$

2.6 Using Gaussian elimination, find all solutions of the inhomogeneous equation system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ with

$$\boldsymbol{A} = \begin{bmatrix}0 & 1 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1 & 1 & 0\\ 0 & 1 & 0 & 0 & 0 & 1\end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix}2\\ -1\\ 1\end{bmatrix}.$$

2.7 Find all solutions in $\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in \mathbb{R}^3$ of the equation system $\boldsymbol{A}\boldsymbol{x} = 12\boldsymbol{x}$, where

$$\boldsymbol{A} = \begin{bmatrix} 6 & 4 & 3 \\ 6 & 0 & 9 \\ 0 & 8 & 0 \end{bmatrix}$$

and $\sum_{i=1}^{3} x_i = 1$.

2.8 Determine the inverses of the following matrices if possible:

a.

$$\boldsymbol{A} = \begin{bmatrix} 2 & 3 & 4 \\ 3 & 4 & 5 \\ 4 & 5 & 6 \end{bmatrix}$$

b.

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

2.9 Which of the following sets are subspaces of $\mathbb{R}^3$?

a. $A = \{(\lambda, \lambda + \mu^3, \lambda - \mu^3) \mid \lambda, \mu \in \mathbb{R}\}$
b. $B = \{(\lambda^2, -\lambda^2, 0) \mid \lambda \in \mathbb{R}\}$
c. Let γ be in $\mathbb{R}$.
   $C = \{(\xi_1, \xi_2, \xi_3) \in \mathbb{R}^3 \mid \xi_1 - 2\xi_2 + 3\xi_3 = \gamma\}$
d. $D = \{(\xi_1, \xi_2, \xi_3) \in \mathbb{R}^3 \mid \xi_2 \in \mathbb{Z}\}$

2.10 Are the following sets of vectors linearly independent?

a.

$$\boldsymbol{x}_1 = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}, \quad \boldsymbol{x}_2 = \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix}, \quad \boldsymbol{x}_3 = \begin{bmatrix} 3 \\ -3 \\ 8 \end{bmatrix}$$

b.

$$\boldsymbol{x}_1 = \begin{bmatrix} 1 \\ 2 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad \boldsymbol{x}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \quad \boldsymbol{x}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

2.11 Write

$$\boldsymbol{y} = \begin{bmatrix} 1 \\ -2 \\ 5 \end{bmatrix}$$

as linear combination of

$$\boldsymbol{x}_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \boldsymbol{x}_2 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \quad \boldsymbol{x}_3 = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$$

2.12 Consider two subspaces of $\mathbb{R}^4$:

$$U_1 = \operatorname{span}[\begin{bmatrix}1\\1\\-3\\1\end{bmatrix}, \begin{bmatrix}2\\-1\\0\\-1\end{bmatrix}, \begin{bmatrix}-1\\1\\-1\\1\end{bmatrix}], \quad U_2 = \operatorname{span}[\begin{bmatrix}-1\\-2\\2\\1\end{bmatrix}, \begin{bmatrix}2\\-2\\0\\0\end{bmatrix}, \begin{bmatrix}-3\\6\\-2\\-1\end{bmatrix}].$$

Determine a basis of $U_1 \cap U_2$.

2.13 Consider two subspaces U_1 and U_2, where U_1 is the solution space of the homogeneous equation system $\boldsymbol{A}_1\boldsymbol{x} = \boldsymbol{0}$ and U_2 is the solution space of the homogeneous equation system $\boldsymbol{A}_2\boldsymbol{x} = \boldsymbol{0}$ with

$$\boldsymbol{A}_1 = \begin{bmatrix}1 & 0 & 1\\1 & -2 & -1\\2 & 1 & 3\\1 & 0 & 1\end{bmatrix}, \quad \boldsymbol{A}_2 = \begin{bmatrix}3 & -3 & 0\\1 & 2 & 3\\7 & -5 & 2\\3 & -1 & 2\end{bmatrix}.$$

a. Determine the dimension of U_1, U_2.
b. Determine bases of U_1 and U_2.
c. Determine a basis of $U_1 \cap U_2$.

2.14 Consider two subspaces U_1 and U_2, where U_1 is spanned by the columns of $\boldsymbol{A}_1$ and U_2 is spanned by the columns of $\boldsymbol{A}_2$ with

$$\boldsymbol{A}_1 = \begin{bmatrix}1 & 0 & 1\\1 & -2 & -1\\2 & 1 & 3\\1 & 0 & 1\end{bmatrix}, \quad \boldsymbol{A}_2 = \begin{bmatrix}3 & -3 & 0\\1 & 2 & 3\\7 & -5 & 2\\3 & -1 & 2\end{bmatrix}.$$

a. Determine the dimension of U_1, U_2
b. Determine bases of U_1 and U_2
c. Determine a basis of $U_1 \cap U_2$

2.15 Let $F = \{(x, y, z) \in \mathbb{R}^3 \mid x+y-z = 0\}$ and $G = \{(a-b, a+b, a-3b) \mid a, b \in \mathbb{R}\}$.

a. Show that F and G are subspaces of $\mathbb{R}^3$.
b. Calculate $F \cap G$ without resorting to any basis vector.
c. Find one basis for F and one for G, calculate $F \cap G$ using the basis vectors previously found and check your result with the previous question.

2.16 Are the following mappings linear?

a. Let $a, b \in \mathbb{R}$.

$$\Phi : L^1([a,b]) \to \mathbb{R}$$
$$f \mapsto \Phi(f) = \int_a^b f(x)dx\,,$$

where $L^1([a,b])$ denotes the set of integrable functions on $[a,b]$.

b.

$$\Phi : C^1 \to C^0$$
$$f \mapsto \Phi(f) = f'\,,$$

where for $k \geqslant 1$, C^k denotes the set of k times continuously differentiable functions, and C^0 denotes the set of continuous functions.

c.

$$\begin{aligned}\Phi : \mathbb{R} &\to \mathbb{R}\\ x &\mapsto \Phi(x) = \cos(x)\end{aligned}$$

d.

$$\begin{aligned}\Phi : \mathbb{R}^3 &\to \mathbb{R}^2\\ \boldsymbol{x} &\mapsto \begin{bmatrix}1 & 2 & 3\\ 1 & 4 & 3\end{bmatrix}\boldsymbol{x}\end{aligned}$$

e. Let θ be in $[0, 2\pi[$ and

$$\begin{aligned}\Phi : \mathbb{R}^2 &\to \mathbb{R}^2\\ \boldsymbol{x} &\mapsto \begin{bmatrix}\cos(\theta) & \sin(\theta)\\ -\sin(\theta) & \cos(\theta)\end{bmatrix}\boldsymbol{x}\end{aligned}$$

2.17 Consider the linear mapping

$$\begin{aligned}\Phi : \mathbb{R}^3 &\to \mathbb{R}^4\\ \Phi\left(\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}\right) &= \begin{bmatrix}3x_1 + 2x_2 + x_3\\ x_1 + x_2 + x_3\\ x_1 - 3x_2\\ 2x_1 + 3x_2 + x_3\end{bmatrix}\end{aligned}$$

- Find the transformation matrix $\boldsymbol{A}_\Phi$.
- Determine $\mathrm{rk}(\boldsymbol{A}_\Phi)$.
- Compute the kernel and image of Φ. What are $\dim(\ker(\Phi))$ and $\dim(\mathrm{Im}(\Phi))$?

2.18 Let E be a vector space. Let f and g be two automorphisms on E such that $f \circ g = \mathrm{id}_E$ (i.e., $f \circ g$ is the identity mapping id_E). Show that $\ker(f) = \ker(g \circ f)$, $\mathrm{Im}(g) = \mathrm{Im}(g \circ f)$ and that $\ker(f) \cap \mathrm{Im}(g) = \{\mathbf{0}_E\}$.

2.19 Consider an endomorphism $\Phi : \mathbb{R}^3 \to \mathbb{R}^3$ whose transformation matrix (with respect to the standard basis in $\mathbb{R}^3$) is

$$\boldsymbol{A}_\Phi = \begin{bmatrix}1 & 1 & 0\\ 1 & -1 & 0\\ 1 & 1 & 1\end{bmatrix}.$$

a. Determine $\ker(\Phi)$ and $\mathrm{Im}(\Phi)$.
b. Determine the transformation matrix $\tilde{\boldsymbol{A}}_\Phi$ with respect to the basis

$$B = (\begin{bmatrix}1\\ 1\\ 1\end{bmatrix}, \begin{bmatrix}1\\ 2\\ 1\end{bmatrix}, \begin{bmatrix}1\\ 0\\ 0\end{bmatrix}),$$

i.e., perform a basis change toward the new basis B.

2.20 Let us consider $\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}'_1, \boldsymbol{b}'_2$, 4 vectors of $\mathbb{R}^2$ expressed in the standard basis of $\mathbb{R}^2$ as

$$\boldsymbol{b}_1 = \begin{bmatrix}2\\ 1\end{bmatrix}, \quad \boldsymbol{b}_2 = \begin{bmatrix}-1\\ -1\end{bmatrix}, \quad \boldsymbol{b}'_1 = \begin{bmatrix}2\\ -2\end{bmatrix}, \quad \boldsymbol{b}'_2 = \begin{bmatrix}1\\ 1\end{bmatrix}$$

and let us define two ordered bases $B = (\boldsymbol{b}_1, \boldsymbol{b}_2)$ and $B' = (\boldsymbol{b}'_1, \boldsymbol{b}'_2)$ of $\mathbb{R}^2$.

a. Show that B and B' are two bases of $\mathbb{R}^2$ and draw those basis vectors.
b. Compute the matrix $\boldsymbol{P}_1$ that performs a basis change from B' to B.
c. We consider $\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3$, three vectors of $\mathbb{R}^3$ defined in the standard basis of $\mathbb{R}^3$ as

$$\boldsymbol{c}_1 = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}, \quad \boldsymbol{c}_2 = \begin{bmatrix} 0 \\ -1 \\ 2 \end{bmatrix}, \quad \boldsymbol{c}_3 = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

and we define $C = (\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3)$.

   (i) Show that C is a basis of $\mathbb{R}^3$, e.g., by using determinants (see Section 4.1).
   (ii) Let us call $C' = (\boldsymbol{c}'_1, \boldsymbol{c}'_2, \boldsymbol{c}'_3)$ the standard basis of $\mathbb{R}^3$. Determine the matrix $\boldsymbol{P}_2$ that performs the basis change from C to C'.

d. We consider a homomorphism $\Phi : \mathbb{R}^2 \longrightarrow \mathbb{R}^3$, such that

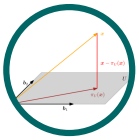
$$\begin{aligned} \Phi(\boldsymbol{b}_1 + \boldsymbol{b}_2) &= \boldsymbol{c}_2 + \boldsymbol{c}_3 \\ \Phi(\boldsymbol{b}_1 - \boldsymbol{b}_2) &= 2\boldsymbol{c}_1 - \boldsymbol{c}_2 + 3\boldsymbol{c}_3 \end{aligned}$$

where $B = (\boldsymbol{b}_1, \boldsymbol{b}_2)$ and $C = (\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3)$ are ordered bases of $\mathbb{R}^2$ and $\mathbb{R}^3$, respectively.
Determine the transformation matrix $\boldsymbol{A}_\Phi$ of Φ with respect to the ordered bases B and C.
e. Determine $\boldsymbol{A}'$, the transformation matrix of Φ with respect to the bases $\boldsymbol{B}'$ and $\boldsymbol{C}'$.
f. Let us consider the vector $\boldsymbol{x} \in \mathbb{R}^2$ whose coordinates in B' are $[2, 3]^\top$. In other words, $\boldsymbol{x} = 2\boldsymbol{b}'_1 + 3\boldsymbol{b}'_2$.

   (i) Calculate the coordinates of $\boldsymbol{x}$ in B.
   (ii) Based on that, compute the coordinates of $\Phi(\boldsymbol{x})$ expressed in C.
   (iii) Then, write $\Phi(\boldsymbol{x})$ in terms of $\boldsymbol{c}'_1, \boldsymbol{c}'_2, \boldsymbol{c}'_3$.
   (iv) Use the representation of $\boldsymbol{x}$ in B' and the matrix $\boldsymbol{A}'$ to find this result directly.

3

Analytic Geometry



In Chapter 2, we studied vectors, vector spaces, and linear mappings at a general but abstract level. In this chapter, we will add some geometric interpretation and intuition to all of these concepts. In particular, we will look at geometric vectors and compute their lengths and distances or angles between two vectors. To be able to do this, we equip the vector space with an inner product that induces the geometry of the vector space. Inner products and their corresponding norms and metrics capture the intuitive notions of similarity and distances, which we use to develop the support vector machine in Chapter 12. We will then use the concepts of lengths and angles between vectors to discuss orthogonal projections, which will play a central role when we discuss principal component analysis in Chapter 10 and regression via maximum likelihood estimation in Chapter 9. Figure 3.1 gives an overview of how concepts in this chapter are related and how they are connected to other chapters of the book.

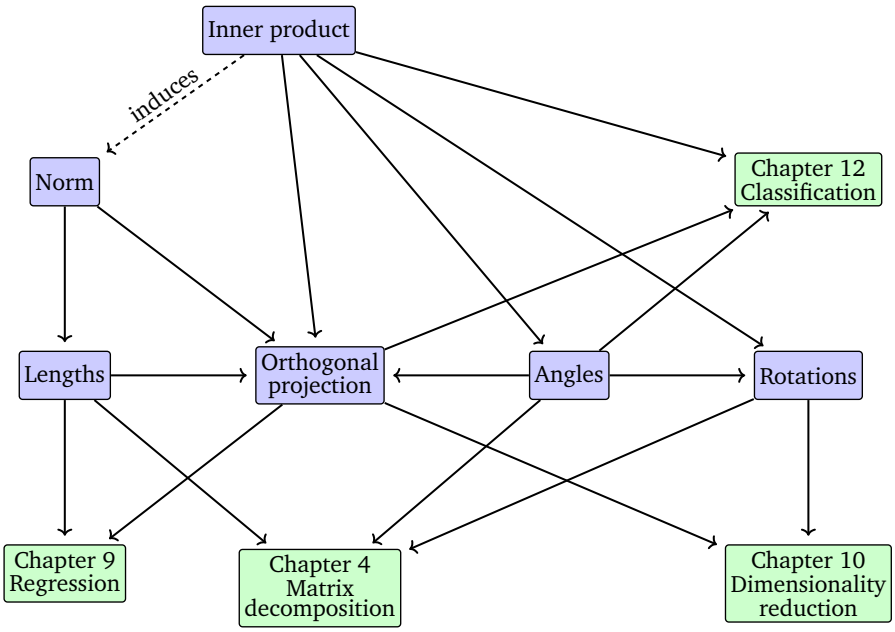


Figure 3.1 A mind map of the concepts introduced in this chapter, along with when they are used in other parts of the book.

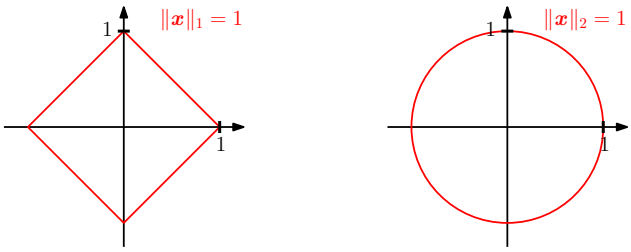


Figure 3.3 For different norms, the red lines indicate the set of vectors with norm 1. Left: Manhattan norm; Right: Euclidean distance.

3.1 Norms

When we think of geometric vectors, i.e., directed line segments that start at the origin, then intuitively the length of a vector is the distance of the "end" of this directed line segment from the origin. In the following, we will discuss the notion of the length of vectors using the concept of a norm.

Definition 3.1 (Norm)**.** A *norm* on a vector space V is a function

norm

$$\|\cdot\| : V \to \mathbb{R}\,, \tag{3.1}$$
$$\boldsymbol{x} \mapsto \|\boldsymbol{x}\|\,, \tag{3.2}$$

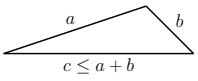
which assigns each vector $\boldsymbol{x}$ its *length* $\|\boldsymbol{x}\| \in \mathbb{R}$, such that for all $\lambda \in \mathbb{R}$ and $\boldsymbol{x}, \boldsymbol{y} \in V$ the following hold:

length

- *Absolutely homogeneous:* $\|\lambda\boldsymbol{x}\| = |\lambda|\|\boldsymbol{x}\|$ — absolutely homogeneous
- *Triangle inequality:* $\|\boldsymbol{x} + \boldsymbol{y}\| \leqslant \|\boldsymbol{x}\| + \|\boldsymbol{y}\|$ — triangle inequality
- *Positive definite:* $\|\boldsymbol{x}\| \geqslant 0$ and $\|\boldsymbol{x}\| = 0 \iff \boldsymbol{x} = \boldsymbol{0}$ — positive definite

In geometric terms, the triangle inequality states that for any triangle, the sum of the lengths of any two sides must be greater than or equal to the length of the remaining side; see Figure 3.2 for an illustration. Definition 3.1 is in terms of a general vector space V (Section 2.4), but in this book we will only consider a finite-dimensional vector space $\mathbb{R}^n$. Recall that for a vector $\boldsymbol{x} \in \mathbb{R}^n$ we denote the elements of the vector using a subscript, that is, x_i is the i^{th} element of the vector $\boldsymbol{x}$.

Figure 3.2 Triangle inequality.



Example 3.1 (Manhattan Norm)

The *Manhattan norm* on $\mathbb{R}^n$ is defined for $\boldsymbol{x} \in \mathbb{R}^n$ as

Manhattan norm

$$\|\boldsymbol{x}\|_1 := \sum_{i=1}^{n} |x_i|\,, \tag{3.3}$$

where $|\cdot|$ is the absolute value. The left panel of Figure 3.3 shows all vectors $\boldsymbol{x} \in \mathbb{R}^2$ with $\|\boldsymbol{x}\|_1 = 1$. The Manhattan norm is also called ℓ_1 *norm*.

ℓ_1 norm

Example 3.2 (Euclidean Norm)

Euclidean norm

The *Euclidean norm* of $\boldsymbol{x} \in \mathbb{R}^n$ is defined as

$$\|\boldsymbol{x}\|_2 := \sqrt{\sum_{i=1}^{n} x_i^2} = \sqrt{\boldsymbol{x}^\top \boldsymbol{x}} \tag{3.4}$$

Euclidean distance

ℓ_2 norm

and computes the *Euclidean distance* of $\boldsymbol{x}$ from the origin. The right panel of Figure 3.3 shows all vectors $\boldsymbol{x} \in \mathbb{R}^2$ with $\|\boldsymbol{x}\|_2 = 1$. The Euclidean norm is also called *ℓ_2 norm*.

Remark. Throughout this book, we will use the Euclidean norm (3.4) by default if not stated otherwise. ♢

3.2 Inner Products

Inner products allow for the introduction of intuitive geometrical concepts, such as the length of a vector and the angle or distance between two vectors. A major purpose of inner products is to determine whether vectors are orthogonal to each other.

distance & angle blw two vectors

3.2.1 Dot Product

scalar product

dot product

We may already be familiar with a particular type of inner product, the *scalar product/dot product* in $\mathbb{R}^n$, which is given by

$$\boldsymbol{x}^\top \boldsymbol{y} = \sum_{i=1}^{n} x_i y_i \,. \tag{3.5}$$

We will refer to this particular inner product as the dot product in this book. However, inner products are more general concepts with specific properties, which we will now introduce.

3.2.2 General Inner Products

bilinear mapping

Recall the linear mapping from Section 2.7, where we can rearrange the mapping with respect to addition and multiplication with a scalar. A *bilinear mapping* Ω is a mapping with two arguments, and it is linear in each argument, i.e., when we look at a vector space V then it holds that for all $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in V,\ \lambda, \psi \in \mathbb{R}$ that

$$\Omega(\lambda \boldsymbol{x} + \psi \boldsymbol{y}, \boldsymbol{z}) = \lambda \Omega(\boldsymbol{x}, \boldsymbol{z}) + \psi \Omega(\boldsymbol{y}, \boldsymbol{z}) \tag{3.6}$$

$$\Omega(\boldsymbol{x}, \lambda \boldsymbol{y} + \psi \boldsymbol{z}) = \lambda \Omega(\boldsymbol{x}, \boldsymbol{y}) + \psi \Omega(\boldsymbol{x}, \boldsymbol{z}) \,. \tag{3.7}$$

Here, (3.6) asserts that Ω is linear in the first argument, and (3.7) asserts that Ω is linear in the second argument (see also (2.87)).

Definition 3.2. Let V be a vector space and $\Omega : V \times V \to \mathbb{R}$ be a bilinear mapping that takes two vectors and maps them onto a real number. Then

- Ω is called *symmetric* if $\Omega(\boldsymbol{x}, \boldsymbol{y}) = \Omega(\boldsymbol{y}, \boldsymbol{x})$ for all $\boldsymbol{x}, \boldsymbol{y} \in V$, i.e., the order of the arguments does not matter. — symmetric
- Ω is called *positive definite* if — positive definite

$$\forall \boldsymbol{x} \in V \backslash \{\boldsymbol{0}\} : \Omega(\boldsymbol{x}, \boldsymbol{x}) > 0\,, \quad \Omega(\boldsymbol{0}, \boldsymbol{0}) = 0\,. \tag{3.8}$$

Definition 3.3. Let V be a vector space and $\Omega : V \times V \to \mathbb{R}$ be a bilinear mapping that takes two vectors and maps them onto a real number. Then

- A positive definite, symmetric bilinear mapping $\Omega : V \times V \to \mathbb{R}$ is called an *inner product* on V. We typically write $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ instead of $\Omega(\boldsymbol{x}, \boldsymbol{y})$. — inner product
- The pair $(V, \langle \cdot, \cdot \rangle)$ is called an *inner product space* or (real) *vector space with inner product*. If we use the dot product defined in (3.5), we call $(V, \langle \cdot, \cdot \rangle)$ a *Euclidean vector space*. — inner product space; vector space with inner product; Euclidean vector space

We will refer to these spaces as inner product spaces in this book.

Example 3.3 (Inner Product That Is Not the Dot Product)
Consider $V = \mathbb{R}^2$. If we define

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle := x_1 y_1 - (x_1 y_2 + x_2 y_1) + 2 x_2 y_2 \tag{3.9}$$

then $\langle \cdot, \cdot \rangle$ is an inner product but different from the dot product. The proof will be an exercise.

3.2.3 Symmetric, Positive Definite Matrices

Symmetric, positive definite matrices play an important role in machine learning, and they are defined via the inner product. In Section 4.3, we will return to symmetric, positive definite matrices in the context of matrix decompositions. The idea of symmetric positive semidefinite matrices is key in the definition of kernels (Section 12.4).

Consider an n-dimensional vector space V with an inner product $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$ (see Definition 3.3) and an ordered basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ of V. Recall from Section 2.6.1 that any vectors $\boldsymbol{x}, \boldsymbol{y} \in V$ can be written as linear combinations of the basis vectors so that $\boldsymbol{x} = \sum_{i=1}^{n} \psi_i \boldsymbol{b}_i \in V$ and $\boldsymbol{y} = \sum_{j=1}^{n} \lambda_j \boldsymbol{b}_j \in V$ for suitable $\psi_i, \lambda_j \in \mathbb{R}$. Due to the bilinearity of the inner product, it holds for all $\boldsymbol{x}, \boldsymbol{y} \in V$ that

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \left\langle \sum_{i=1}^{n} \psi_i \boldsymbol{b}_i, \sum_{j=1}^{n} \lambda_j \boldsymbol{b}_j \right\rangle = \sum_{i=1}^{n} \sum_{j=1}^{n} \psi_i \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \lambda_j = \hat{\boldsymbol{x}}^\top \boldsymbol{A} \hat{\boldsymbol{y}}\,, \tag{3.10}$$

where $A_{ij} := \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle$ and $\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}$ are the coordinates of $\boldsymbol{x}$ and $\boldsymbol{y}$ with respect to the basis B. This implies that the inner product $\langle \cdot, \cdot \rangle$ is uniquely determined through $\boldsymbol{A}$. The symmetry of the inner product also means that $\boldsymbol{A}$

is symmetric. Furthermore, the positive definiteness of the inner product implies that

$$\forall \boldsymbol{x} \in V \backslash \{\boldsymbol{0}\} : \boldsymbol{x}^\top \boldsymbol{A} \boldsymbol{x} > 0\,. \tag{3.11}$$

symmetric, positive definite
positive definite
symmetric, positive semidefinite

Definition 3.4 (Symmetric, Positive Definite Matrix)**.** A symmetric matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ that satisfies (3.11) is called *symmetric, positive definite*, or just *positive definite*. If only $\geqslant$ holds in (3.11), then $\boldsymbol{A}$ is called *symmetric, positive semidefinite*.

Example 3.4 (Symmetric, Positive Definite Matrices)
Consider the matrices

$$\boldsymbol{A}_1 = \begin{bmatrix} 9 & 6 \\ 6 & 5 \end{bmatrix}, \quad \boldsymbol{A}_2 = \begin{bmatrix} 9 & 6 \\ 6 & 3 \end{bmatrix}. \tag{3.12}$$

$\boldsymbol{A}_1$ is positive definite because it is symmetric and

$$\boldsymbol{x}^\top \boldsymbol{A}_1 \boldsymbol{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 9 & 6 \\ 6 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{3.13a}$$

$$= 9x_1^2 + 12x_1x_2 + 5x_2^2 = (3x_1 + 2x_2)^2 + x_2^2 > 0 \tag{3.13b}$$

for all $\boldsymbol{x} \in V \backslash \{\boldsymbol{0}\}$. In contrast, $\boldsymbol{A}_2$ is symmetric but not positive definite because $\boldsymbol{x}^\top \boldsymbol{A}_2 \boldsymbol{x} = 9x_1^2 + 12x_1x_2 + 3x_2^2 = (3x_1 + 2x_2)^2 - x_2^2$ can be less than 0, e.g., for $\boldsymbol{x} = [2, -3]^\top$.

Sum of squares is always non negative $x_i \neq \{0\}$ then the sum is always +ve.

If $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ is symmetric, positive definite, then

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \hat{\boldsymbol{x}}^\top \boldsymbol{A} \hat{\boldsymbol{y}} \tag{3.14}$$

defines an inner product with respect to an ordered basis B, where $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ are the coordinate representations of $\boldsymbol{x}, \boldsymbol{y} \in V$ with respect to B.

Theorem 3.5. *For a real-valued, finite-dimensional vector space V and an ordered basis B of V, it holds that $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$ is an inner product if and only if there exists a symmetric, positive definite matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ with*

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \hat{\boldsymbol{x}}^\top \boldsymbol{A} \hat{\boldsymbol{y}}\,. \tag{3.15}$$

The following properties hold if $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ is symmetric and positive definite:

- The null space (kernel) of $\boldsymbol{A}$ consists only of $\boldsymbol{0}$ because $\boldsymbol{x}^\top \boldsymbol{A} \boldsymbol{x} > 0$ for all $\boldsymbol{x} \neq \boldsymbol{0}$. This implies that $\boldsymbol{A}\boldsymbol{x} \neq \boldsymbol{0}$ if $\boldsymbol{x} \neq \boldsymbol{0}$.
- The diagonal elements a_{ii} of $\boldsymbol{A}$ are positive because $a_{ii} = \boldsymbol{e}_i^\top \boldsymbol{A} \boldsymbol{e}_i > 0$, where $\boldsymbol{e}_i$ is the ith vector of the standard basis in $\mathbb{R}^n$.

3.3 Lengths and Distances

In Section 3.1, we already discussed norms that we can use to compute the length of a vector. Inner products and norms are closely related in the sense that any inner product induces a norm

Inner products induce norms.

$$\|\boldsymbol{x}\| := \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle} \tag{3.16}$$

in a natural way, such that we can compute lengths of vectors using the inner product. However, not every norm is induced by an inner product. The Manhattan norm (3.3) is an example of a norm without a corresponding inner product. In the following, we will focus on norms that are induced by inner products and introduce geometric concepts, such as lengths, distances, and angles.

Remark (Cauchy-Schwarz Inequality). For an inner product vector space $(V, \langle \cdot, \cdot \rangle)$ the induced norm $\| \cdot \|$ satisfies the *Cauchy-Schwarz inequality*

Cauchy-Schwarz inequality

$$| \langle \boldsymbol{x}, \boldsymbol{y} \rangle | \leqslant \|\boldsymbol{x}\| \|\boldsymbol{y}\| \,. \tag{3.17}$$

♢

Example 3.5 (Lengths of Vectors Using Inner Products)
In geometry, we are often interested in lengths of vectors. We can now use an inner product to compute them using (3.16). Let us take $\boldsymbol{x} = [1, 1]^\top \in \mathbb{R}^2$. If we use the dot product as the inner product, with (3.16) we obtain

$$\|\boldsymbol{x}\| = \sqrt{\boldsymbol{x}^\top \boldsymbol{x}} = \sqrt{1^2 + 1^2} = \sqrt{2} \tag{3.18}$$

as the length of $\boldsymbol{x}$. Let us now choose a different inner product:

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\top \begin{bmatrix} 1 & -\frac{1}{2} \\ -\frac{1}{2} & 1 \end{bmatrix} \boldsymbol{y} = x_1 y_1 - \frac{1}{2}(x_1 y_2 + x_2 y_1) + x_2 y_2 \,. \tag{3.19}$$

If we compute the norm of a vector, then this inner product returns smaller values than the dot product if x_1 and x_2 have the same sign (and $x_1 x_2 > 0$); otherwise, it returns greater values than the dot product. With this inner product, we obtain

$$\langle \boldsymbol{x}, \boldsymbol{x} \rangle = x_1^2 - x_1 x_2 + x_2^2 = 1 - 1 + 1 = 1 \implies \|\boldsymbol{x}\| = \sqrt{1} = 1 \,, \tag{3.20}$$

such that $\boldsymbol{x}$ is "shorter" with this inner product than with the dot product.

Definition 3.6 (Distance and Metric)**.** Consider an inner product space $(V, \langle \cdot, \cdot \rangle)$. Then

$$d(\boldsymbol{x}, \boldsymbol{y}) := \|\boldsymbol{x} - \boldsymbol{y}\| = \sqrt{\langle \boldsymbol{x} - \boldsymbol{y}, \boldsymbol{x} - \boldsymbol{y} \rangle} \tag{3.21}$$

is called the *distance* between $\boldsymbol{x}$ and $\boldsymbol{y}$ for $\boldsymbol{x}, \boldsymbol{y} \in V$. If we use the dot product as the inner product, then the distance is called *Euclidean distance*.

distance
Euclidean distance

The mapping

$$d : V \times V \to \mathbb{R} \tag{3.22}$$
$$(\boldsymbol{x}, \boldsymbol{y}) \mapsto d(\boldsymbol{x}, \boldsymbol{y}) \tag{3.23}$$

metric

is called a *metric*.

Remark. Similar to the length of a vector, the distance between vectors does not require an inner product: a norm is sufficient. If we have a norm induced by an inner product, the distance may vary depending on the choice of the inner product. ♢

A metric d satisfies the following:

positive definite

1. d is *positive definite*, i.e., $d(\boldsymbol{x}, \boldsymbol{y}) \geqslant 0$ for all $\boldsymbol{x}, \boldsymbol{y} \in V$ and $d(\boldsymbol{x}, \boldsymbol{y}) = 0 \iff \boldsymbol{x} = \boldsymbol{y}$.

symmetric

2. d is *symmetric*, i.e., $d(\boldsymbol{x}, \boldsymbol{y}) = d(\boldsymbol{y}, \boldsymbol{x})$ for all $\boldsymbol{x}, \boldsymbol{y} \in V$.

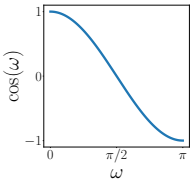
triangle inequality

3. *Triangle inequality:* $d(\boldsymbol{x}, \boldsymbol{z}) \leqslant d(\boldsymbol{x}, \boldsymbol{y}) + d(\boldsymbol{y}, \boldsymbol{z})$ for all $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in V$.

Remark. At first glance, the lists of properties of inner products and metrics look very similar. However, by comparing Definition 3.3 with Definition 3.6 we observe that $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ and $d(\boldsymbol{x}, \boldsymbol{y})$ behave in opposite directions. Very similar $\boldsymbol{x}$ and $\boldsymbol{y}$ will result in a large value for the inner product and a small value for the metric. ♢

3.4 Angles and Orthogonality

Figure 3.4 When restricted to $[0, \pi]$ then $f(\omega) = \cos(\omega)$ returns a unique number in the interval $[-1, 1]$.



In addition to enabling the definition of lengths of vectors, as well as the distance between two vectors, inner products also capture the geometry of a vector space by defining the angle ω between two vectors. We use the Cauchy-Schwarz inequality (3.17) to define angles ω in inner product spaces between two vectors $\boldsymbol{x}, \boldsymbol{y}$, and this notion coincides with our intuition in $\mathbb{R}^2$ and $\mathbb{R}^3$. Assume that $\boldsymbol{x} \neq \boldsymbol{0}, \boldsymbol{y} \neq \boldsymbol{0}$. Then

$$-1 \leqslant \frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{\|\boldsymbol{x}\| \, \|\boldsymbol{y}\|} \leqslant 1 \,. \tag{3.24}$$

Therefore, there exists a unique $\omega \in [0, \pi]$, illustrated in Figure 3.4, with

$$\cos \omega = \frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{\|\boldsymbol{x}\| \, \|\boldsymbol{y}\|} \,. \tag{3.25}$$

angle

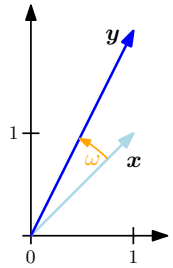
The number ω is the *angle* between the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. Intuitively, the angle between two vectors tells us how similar their orientations are. For example, using the dot product, the angle between $\boldsymbol{x}$ and $\boldsymbol{y} = 4\boldsymbol{x}$, i.e., $\boldsymbol{y}$ is a scaled version of $\boldsymbol{x}$, is 0: Their orientation is the same.

Example 3.6 (Angle between Vectors)
Let us compute the angle between $\boldsymbol{x} = [1, 1]^\top \in \mathbb{R}^2$ and $\boldsymbol{y} = [1, 2]^\top \in \mathbb{R}^2$; see Figure 3.5, where we use the dot product as the inner product. Then we get

$$\cos\omega = \frac{\langle \boldsymbol{x}, \boldsymbol{y}\rangle}{\sqrt{\langle \boldsymbol{x}, \boldsymbol{x}\rangle \langle \boldsymbol{y}, \boldsymbol{y}\rangle}} = \frac{\boldsymbol{x}^\top \boldsymbol{y}}{\sqrt{\boldsymbol{x}^\top \boldsymbol{x}\boldsymbol{y}^\top \boldsymbol{y}}} = \frac{3}{\sqrt{10}}, \tag{3.26}$$

and the angle between the two vectors is $\arccos(\frac{3}{\sqrt{10}}) \approx 0.32\,\text{rad}$, which corresponds to about $18°$.

Figure 3.5 The angle ω between two vectors $\boldsymbol{x}, \boldsymbol{y}$ is computed using the inner product.



A key feature of the inner product is that it also allows us to characterize vectors that are orthogonal.

Definition 3.7 (Orthogonality)**.** Two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ are *orthogonal* if and only if $\langle \boldsymbol{x}, \boldsymbol{y}\rangle = 0$, and we write $\boldsymbol{x} \perp \boldsymbol{y}$. If additionally $\|\boldsymbol{x}\| = 1 = \|\boldsymbol{y}\|$, i.e., the vectors are unit vectors, then $\boldsymbol{x}$ and $\boldsymbol{y}$ are *orthonormal*.

orthogonal

orthonormal

An implication of this definition is that the $\boldsymbol{0}$-vector is orthogonal to every vector in the vector space.

Remark. Orthogonality is the generalization of the concept of perpendicularity to bilinear forms that do not have to be the dot product. In our context, geometrically, we can think of orthogonal vectors as having a right angle with respect to a specific inner product. ♢

Example 3.7 (Orthogonal Vectors)

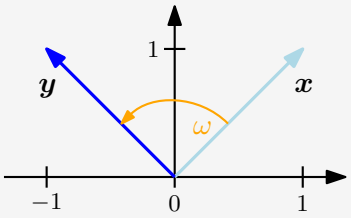


Figure 3.6 The angle ω between two vectors $\boldsymbol{x}, \boldsymbol{y}$ can change depending on the inner product.

Consider two vectors $\boldsymbol{x} = [1, 1]^\top, \boldsymbol{y} = [-1, 1]^\top \in \mathbb{R}^2$; see Figure 3.6. We are interested in determining the angle ω between them using two different inner products. Using the dot product as the inner product yields an angle ω between $\boldsymbol{x}$ and $\boldsymbol{y}$ of $90°$, such that $\boldsymbol{x} \perp \boldsymbol{y}$. However, if we choose the inner product

$$\langle \boldsymbol{x}, \boldsymbol{y}\rangle = \boldsymbol{x}^\top \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \boldsymbol{y}, \tag{3.27}$$

we get that the angle ω between $\boldsymbol{x}$ and $\boldsymbol{y}$ is given by

$$\cos\omega = \frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|} = -\frac{1}{3} \implies \omega \approx 1.91\,\text{rad} \approx 109.5^\circ\,, \tag{3.28}$$

and $\boldsymbol{x}$ and $\boldsymbol{y}$ are not orthogonal. Therefore, vectors that are orthogonal with respect to one inner product do not have to be orthogonal with respect to a different inner product.

orthogonal matrix

Definition 3.8 (Orthogonal Matrix)**.** A square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is an *orthogonal matrix* if and only if its columns are orthonormal so that

$$\boldsymbol{A}\boldsymbol{A}^\top = \boldsymbol{I} = \boldsymbol{A}^\top\boldsymbol{A}\,, \tag{3.29}$$

which implies that

$$\boldsymbol{A}^{-1} = \boldsymbol{A}^\top\,, \tag{3.30}$$

i.e., the inverse is obtained by simply transposing the matrix.

It is convention to call these matrices "orthogonal" but a more precise description would be "orthonormal". Transformations with orthogonal matrices preserve distances and angles.

Transformations by orthogonal matrices are special because the length of a vector $\boldsymbol{x}$ is not changed when transforming it using an orthogonal matrix $\boldsymbol{A}$. For the dot product, we obtain

$$\|\boldsymbol{A}\boldsymbol{x}\|^2 = (\boldsymbol{A}\boldsymbol{x})^\top(\boldsymbol{A}\boldsymbol{x}) = \boldsymbol{x}^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x} = \boldsymbol{x}^\top\boldsymbol{I}\boldsymbol{x} = \boldsymbol{x}^\top\boldsymbol{x} = \|\boldsymbol{x}\|^2\,. \tag{3.31}$$

Moreover, the angle between any two vectors $\boldsymbol{x}, \boldsymbol{y}$, as measured by their inner product, is also unchanged when transforming both of them using an orthogonal matrix $\boldsymbol{A}$. Assuming the dot product as the inner product, the angle of the images $\boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{A}\boldsymbol{y}$ is given as

$$\cos\omega = \frac{(\boldsymbol{A}\boldsymbol{x})^\top(\boldsymbol{A}\boldsymbol{y})}{\|\boldsymbol{A}\boldsymbol{x}\|\,\|\boldsymbol{A}\boldsymbol{y}\|} = \frac{\boldsymbol{x}^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{y}}{\sqrt{\boldsymbol{x}^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x}\boldsymbol{y}^\top\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{y}}} = \frac{\boldsymbol{x}^\top\boldsymbol{y}}{\|\boldsymbol{x}\|\,\|\boldsymbol{y}\|}\,, \tag{3.32}$$

which gives exactly the angle between $\boldsymbol{x}$ and $\boldsymbol{y}$. This means that orthogonal matrices $\boldsymbol{A}$ with $\boldsymbol{A}^\top = \boldsymbol{A}^{-1}$ preserve both angles and distances. It turns out that orthogonal matrices define transformations that are rotations (with the possibility of flips). In Section 3.9, we will discuss more details about rotations.

3.5 Orthonormal Basis

In Section 2.6.1, we characterized properties of basis vectors and found that in an n-dimensional vector space, we need n basis vectors, i.e., n vectors that are linearly independent. In Sections 3.3 and 3.4, we used inner products to compute the length of vectors and the angle between vectors. In the following, we will discuss the special case where the basis vectors are orthogonal to each other and where the length of each basis vector is 1. We will call this basis then an orthonormal basis.

Let us introduce this more formally.

Definition 3.9 (Orthonormal Basis)**.** Consider an n-dimensional vector space V and a basis $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ of V. If

$$\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle = 0 \quad \text{for } i \neq j \tag{3.33}$$
$$\langle \boldsymbol{b}_i, \boldsymbol{b}_i \rangle = 1 \tag{3.34}$$

for all $i, j = 1, \ldots, n$ then the basis is called an *orthonormal basis* (*ONB*). If only (3.33) is satisfied, then the basis is called an *orthogonal basis*. Note that (3.34) implies that every basis vector has length/norm 1.

orthonormal basis
ONB
orthogonal basis

Recall from Section 2.6.1 that we can use Gaussian elimination to find a basis for a vector space spanned by a set of vectors. Assume we are given a set $\{\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n\}$ of non-orthogonal and unnormalized basis vectors. We concatenate them into a matrix $\tilde{\boldsymbol{B}} = [\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n]$ and apply Gaussian elimination to the augmented matrix (Section 2.3.2) $[\tilde{\boldsymbol{B}}\tilde{\boldsymbol{B}}^\top | \tilde{\boldsymbol{B}}]$ to obtain an orthonormal basis. This constructive way to iteratively build an orthonormal basis $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ is called the *Gram-Schmidt process* (Strang, 2003).

Example 3.8 (Orthonormal Basis)
The canonical/standard basis for a Euclidean vector space $\mathbb{R}^n$ is an orthonormal basis, where the inner product is the dot product of vectors.

In $\mathbb{R}^2$, the vectors

$$\boldsymbol{b}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}, \quad \boldsymbol{b}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix} \tag{3.35}$$

form an orthonormal basis since $\boldsymbol{b}_1^\top \boldsymbol{b}_2 = 0$ and $\|\boldsymbol{b}_1\| = 1 = \|\boldsymbol{b}_2\|$.

We will exploit the concept of an orthonormal basis in Chapter 12 and Chapter 10 when we discuss support vector machines and principal component analysis.

3.6 Orthogonal Complement

Having defined orthogonality, we will now look at vector spaces that are orthogonal to each other. This will play an important role in Chapter 10, when we discuss linear dimensionality reduction from a geometric perspective.

Consider a D-dimensional vector space V and an M-dimensional subspace $U \subseteq V$. Then its *orthogonal complement* $U^\perp$ is a $(D-M)$-dimensional subspace of V and contains all vectors in V that are orthogonal to every vector in U. Furthermore, $U \cap U^\perp = \{\boldsymbol{0}\}$ so that any vector $\boldsymbol{x} \in V$ can be

orthogonal complement

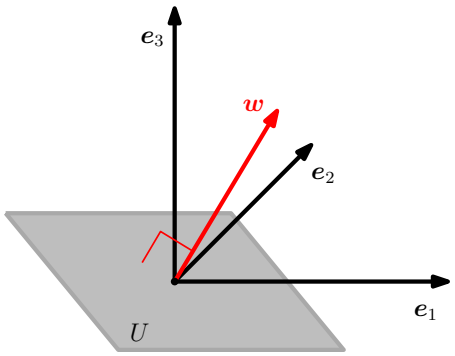


Figure 3.7 A plane U in a three-dimensional vector space can be described by its normal vector, which spans its orthogonal complement $U^\perp$.

uniquely decomposed into

$$\boldsymbol{x} = \sum_{m=1}^{M} \lambda_m \boldsymbol{b}_m + \sum_{j=1}^{D-M} \psi_j \boldsymbol{b}_j^\perp, \quad \lambda_m,\ \psi_j \in \mathbb{R}\,, \tag{3.36}$$

where $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M)$ is a basis of U and $(\boldsymbol{b}_1^\perp, \ldots, \boldsymbol{b}_{D-M}^\perp)$ is a basis of $U^\perp$.

Therefore, the orthogonal complement can also be used to describe a plane U (two-dimensional subspace) in a three-dimensional vector space. More specifically, the vector $\boldsymbol{w}$ with $\|\boldsymbol{w}\| = 1$, which is orthogonal to the plane U, is the basis vector of $U^\perp$. Figure 3.7 illustrates this setting. All vectors that are orthogonal to $\boldsymbol{w}$ must (by construction) lie in the plane U. The vector $\boldsymbol{w}$ is called the *normal vector* of U.

normal vector

Generally, orthogonal complements can be used to describe hyperplanes in n-dimensional vector and affine spaces.

3.7 Inner Product of Functions

Thus far, we looked at properties of inner products to compute lengths, angles and distances. We focused on inner products of finite-dimensional vectors. In the following, we will look at an example of inner products of a different type of vectors: inner products of functions.

The inner products we discussed so far were defined for vectors with a finite number of entries. We can think of a vector $\boldsymbol{x} \in \mathbb{R}^n$ as a function with n function values. The concept of an inner product can be generalized to vectors with an infinite number of entries (countably infinite) and also continuous-valued functions (uncountably infinite). Then the sum over individual components of vectors (see Equation (3.5) for example) turns into an integral.

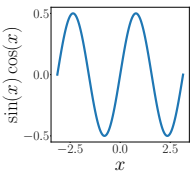
An inner product of two functions $u : \mathbb{R} \to \mathbb{R}$ and $v : \mathbb{R} \to \mathbb{R}$ can be defined as the definite integral

$$\langle u, v \rangle := \int_a^b u(x)v(x)dx \tag{3.37}$$

for lower and upper limits $a, b < \infty$, respectively. As with our usual inner product, we can define norms and orthogonality by looking at the inner product. If (3.37) evaluates to 0, the functions u and v are orthogonal. To make the preceding inner product mathematically precise, we need to take care of measures and the definition of integrals, leading to the definition of a Hilbert space. Furthermore, unlike inner products on finite-dimensional vectors, inner products on functions may diverge (have infinite value). All this requires diving into some more intricate details of real and functional analysis, which we do not cover in this book.

Example 3.9 (Inner Product of Functions)
If we choose $u = \sin(x)$ and $v = \cos(x)$, the integrand $f(x) = u(x)v(x)$ of (3.37), is shown in Figure 3.8. We see that this function is odd, i.e., $f(-x) = -f(x)$. Therefore, the integral with limits $a = -\pi, b = \pi$ of this product evaluates to 0. Therefore, $\sin$ and $\cos$ are orthogonal functions.

Figure 3.8 $f(x) = \sin(x)\cos(x)$.



Remark. It also holds that the collection of functions

$$\{1, \cos(x), \cos(2x), \cos(3x), \dots\} \tag{3.38}$$

is orthogonal if we integrate from $-\pi$ to π, i.e., any pair of functions are orthogonal to each other. The collection of functions in (3.38) spans a large subspace of the functions that are even and periodic on $[-\pi, \pi)$, and projecting functions onto this subspace is the fundamental idea behind Fourier series. ♢

In Section 6.4.6, we will have a look at a second type of unconventional inner products: the inner product of random variables.

3.8 Orthogonal Projections

Projections are an important class of linear transformations (besides rotations and reflections) and play an important role in graphics, coding theory, statistics and machine learning. In machine learning, we often deal with data that is high-dimensional. High-dimensional data is often hard to analyze or visualize. However, high-dimensional data quite often possesses the property that only a few dimensions contain most information, and most other dimensions are not essential to describe key properties of the data. When we compress or visualize high-dimensional data, we will lose information. To minimize this compression loss, we ideally find the most informative dimensions in the data. As discussed in Chapter 1, data can be represented as vectors, and in this chapter, we will discuss some of the fundamental tools for data compression. More specifically, we can project the original high-dimensional data onto a lower-dimensional feature space and work in this lower-dimensional space to learn more about the dataset and extract relevant patterns. For example, machine

"Feature" is a common expression for data representation.

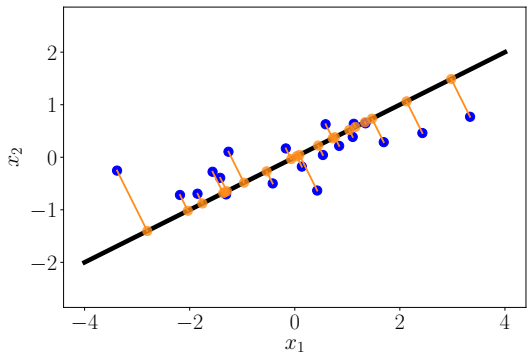


Figure 3.9 Orthogonal projection (orange dots) of a two-dimensional dataset (blue dots) onto a one-dimensional subspace (straight line).

learning algorithms, such as principal component analysis (PCA) by Pearson (1901) and Hotelling (1933) and deep neural networks (e.g., deep auto-encoders (Deng et al., 2010)), heavily exploit the idea of dimensionality reduction. In the following, we will focus on orthogonal projections, which we will use in Chapter 10 for linear dimensionality reduction and in Chapter 12 for classification. Even linear regression, which we discuss in Chapter 9, can be interpreted using orthogonal projections. For a given lower-dimensional subspace, orthogonal projections of high-dimensional data retain as much information as possible and minimize the difference/ error between the original data and the corresponding projection. An illustration of such an orthogonal projection is given in Figure 3.9. Before we detail how to obtain these projections, let us define what a projection actually is.

projection

Definition 3.10 (Projection)**.** Let V be a vector space and $U \subseteq V$ a subspace of V. A linear mapping $\pi : V \to U$ is called a *projection* if $\pi^2 = \pi \circ \pi = \pi$.

projection matrix

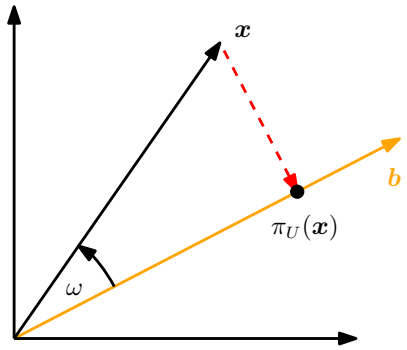
Since linear mappings can be expressed by transformation matrices (see Section 2.7), the preceding definition applies equally to a special kind of transformation matrices, the *projection matrices* $\boldsymbol{P}_\pi$, which exhibit the property that $\boldsymbol{P}_\pi^2 = \boldsymbol{P}_\pi$.

line

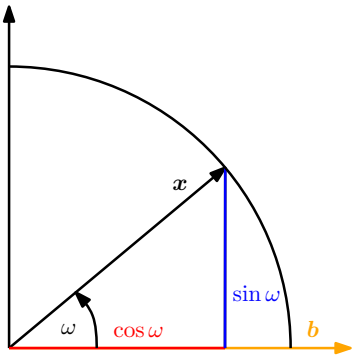
In the following, we will derive orthogonal projections of vectors in the inner product space $(\mathbb{R}^n, \langle \cdot, \cdot \rangle)$ onto subspaces. We will start with one-dimensional subspaces, which are also called *lines*. If not mentioned otherwise, we assume the dot product $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^\top \boldsymbol{y}$ as the inner product.

3.8.1 Projection onto One-Dimensional Subspaces (Lines)

Assume we are given a line (one-dimensional subspace) through the origin with basis vector $\boldsymbol{b} \in \mathbb{R}^n$. The line is a one-dimensional subspace $U \subseteq \mathbb{R}^n$ spanned by $\boldsymbol{b}$. When we project $\boldsymbol{x} \in \mathbb{R}^n$ onto U, we seek the vector $\pi_U(\boldsymbol{x}) \in U$ that is closest to $\boldsymbol{x}$. Using geometric arguments, let



(a) Projection of $\boldsymbol{x} \in \mathbb{R}^2$ onto a subspace U with basis vector $\boldsymbol{b}$.



(b) Projection of a two-dimensional vector $\boldsymbol{x}$ with $\|\boldsymbol{x}\| = 1$ onto a one-dimensional subspace spanned by $\boldsymbol{b}$.

Figure 3.10 Examples of projections onto one-dimensional subspaces.

us characterize some properties of the projection $\pi_U(\boldsymbol{x})$ (Figure 3.10(a) serves as an illustration):

- The projection $\pi_U(\boldsymbol{x})$ is closest to $\boldsymbol{x}$, where "closest" implies that the distance $\|\boldsymbol{x} - \pi_U(\boldsymbol{x})\|$ is minimal. It follows that the segment $\pi_U(\boldsymbol{x}) - \boldsymbol{x}$ from $\pi_U(\boldsymbol{x})$ to $\boldsymbol{x}$ is orthogonal to U, and therefore the basis vector $\boldsymbol{b}$ of U. The orthogonality condition yields $\langle \pi_U(\boldsymbol{x}) - \boldsymbol{x}, \boldsymbol{b} \rangle = 0$ since angles between vectors are defined via the inner product.
- The projection $\pi_U(\boldsymbol{x})$ of $\boldsymbol{x}$ onto U must be an element of U and, therefore, a multiple of the basis vector $\boldsymbol{b}$ that spans U. Hence, $\pi_U(\boldsymbol{x}) = \lambda \boldsymbol{b}$, for some $\lambda \in \mathbb{R}$.

λ is then the coordinate of $\pi_U(\boldsymbol{x})$ with respect to $\boldsymbol{b}$.

In the following three steps, we determine the coordinate λ, the projection $\pi_U(\boldsymbol{x}) \in U$, and the projection matrix $\boldsymbol{P}_\pi$ that maps any $\boldsymbol{x} \in \mathbb{R}^n$ onto U:

1. Finding the coordinate λ. The orthogonality condition yields

$$\langle \boldsymbol{x} - \pi_U(\boldsymbol{x}), \boldsymbol{b} \rangle = 0 \overset{\pi_U(\boldsymbol{x}) = \lambda \boldsymbol{b}}{\iff} \langle \boldsymbol{x} - \lambda \boldsymbol{b}, \boldsymbol{b} \rangle = 0 \,. \tag{3.39}$$

   We can now exploit the bilinearity of the inner product and arrive at

$$\langle \boldsymbol{x}, \boldsymbol{b} \rangle - \lambda \langle \boldsymbol{b}, \boldsymbol{b} \rangle = 0 \iff \lambda = \frac{\langle \boldsymbol{x}, \boldsymbol{b} \rangle}{\langle \boldsymbol{b}, \boldsymbol{b} \rangle} = \frac{\langle \boldsymbol{b}, \boldsymbol{x} \rangle}{\|\boldsymbol{b}\|^2} \,. \tag{3.40}$$

   With a general inner product, we get $\lambda = \langle \boldsymbol{x}, \boldsymbol{b} \rangle$ if $\|\boldsymbol{b}\| = 1$.

   In the last step, we exploited the fact that inner products are symmetric. If we choose $\langle \cdot, \cdot \rangle$ to be the dot product, we obtain

$$\lambda = \frac{\boldsymbol{b}^\top \boldsymbol{x}}{\boldsymbol{b}^\top \boldsymbol{b}} = \frac{\boldsymbol{b}^\top \boldsymbol{x}}{\|\boldsymbol{b}\|^2} \,. \tag{3.41}$$

   If $\|\boldsymbol{b}\| = 1$, then the coordinate λ of the projection is given by $\boldsymbol{b}^\top \boldsymbol{x}$.

2. Finding the projection point $\pi_U(\boldsymbol{x}) \in U$. Since $\pi_U(\boldsymbol{x}) = \lambda\boldsymbol{b}$, we immediately obtain with (3.40) that

$$\pi_U(\boldsymbol{x}) = \lambda\boldsymbol{b} = \frac{\langle \boldsymbol{x}, \boldsymbol{b}\rangle}{\|\boldsymbol{b}\|^2}\boldsymbol{b} = \frac{\boldsymbol{b}^\top \boldsymbol{x}}{\|\boldsymbol{b}\|^2}\boldsymbol{b}\,, \tag{3.42}$$

where the last equality holds for the dot product only. We can also compute the length of $\pi_U(\boldsymbol{x})$ by means of Definition 3.1 as

$$\|\pi_U(\boldsymbol{x})\| = \|\lambda\boldsymbol{b}\| = |\lambda|\,\|\boldsymbol{b}\|\,. \tag{3.43}$$

Hence, our projection is of length $|\lambda|$ times the length of $\boldsymbol{b}$. This also adds the intuition that λ is the coordinate of $\pi_U(\boldsymbol{x})$ with respect to the basis vector $\boldsymbol{b}$ that spans our one-dimensional subspace U.

If we use the dot product as an inner product, we get

$$\|\pi_U(\boldsymbol{x})\| \overset{(3.42)}{=} \frac{|\boldsymbol{b}^\top \boldsymbol{x}|}{\|\boldsymbol{b}\|^2}\|\boldsymbol{b}\| \overset{(3.25)}{=} |\cos\omega|\,\|\boldsymbol{x}\|\,\|\boldsymbol{b}\|\frac{\|\boldsymbol{b}\|}{\|\boldsymbol{b}\|^2} = |\cos\omega|\,\|\boldsymbol{x}\|\,. \tag{3.44}$$

Here, ω is the angle between $\boldsymbol{x}$ and $\boldsymbol{b}$. This equation should be familiar from trigonometry: If $\|\boldsymbol{x}\| = 1$, then $\boldsymbol{x}$ lies on the unit circle. It follows that the projection onto the horizontal axis spanned by $\boldsymbol{b}$ is exactly $\cos\omega$, and the length of the corresponding vector $\pi_U(\boldsymbol{x}) = |\cos\omega|$. An illustration is given in Figure 3.10(b).

The horizontal axis is a one-dimensional subspace.

3. Finding the projection matrix $\boldsymbol{P}_\pi$. We know that a projection is a linear mapping (see Definition 3.10). Therefore, there exists a projection matrix $\boldsymbol{P}_\pi$, such that $\pi_U(\boldsymbol{x}) = \boldsymbol{P}_\pi\boldsymbol{x}$. With the dot product as inner product and

$$\pi_U(\boldsymbol{x}) = \lambda\boldsymbol{b} = \boldsymbol{b}\lambda = \boldsymbol{b}\frac{\boldsymbol{b}^\top\boldsymbol{x}}{\|\boldsymbol{b}\|^2} = \frac{\boldsymbol{b}\boldsymbol{b}^\top}{\|\boldsymbol{b}\|^2}\boldsymbol{x}\,, \tag{3.45}$$

we immediately see that

$$\boldsymbol{P}_\pi = \frac{\boldsymbol{b}\boldsymbol{b}^\top}{\|\boldsymbol{b}\|^2}\,. \tag{3.46}$$

Note that $\boldsymbol{b}\boldsymbol{b}^\top$ (and, consequently, $\boldsymbol{P}_\pi$) is a symmetric matrix (of rank 1), and $\|\boldsymbol{b}\|^2 = \langle \boldsymbol{b}, \boldsymbol{b}\rangle$ is a scalar.

Projection matrices are always symmetric.

The projection matrix $\boldsymbol{P}_\pi$ projects any vector $\boldsymbol{x} \in \mathbb{R}^n$ onto the line through the origin with direction $\boldsymbol{b}$ (equivalently, the subspace U spanned by $\boldsymbol{b}$).

Remark. The projection $\pi_U(\boldsymbol{x}) \in \mathbb{R}^n$ is still an n-dimensional vector and not a scalar. However, we no longer require n coordinates to represent the projection, but only a single one if we want to express it with respect to the basis vector $\boldsymbol{b}$ that spans the subspace U: λ. ♢

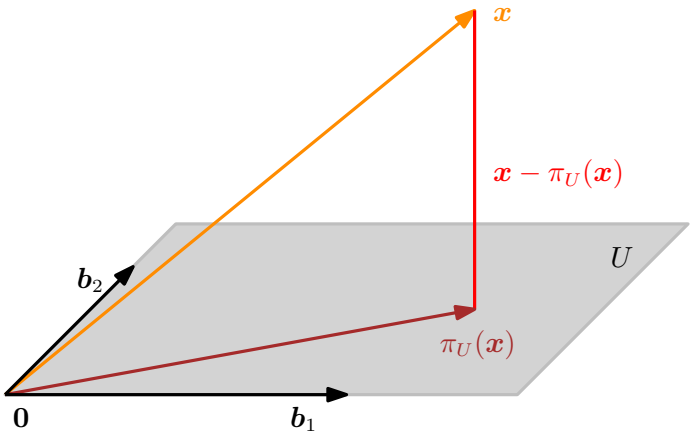


Figure 3.11 Projection onto a two-dimensional subspace U with basis b_1, b_2. The projection $\pi_U(x)$ of $x \in \mathbb{R}^3$ onto U can be expressed as a linear combination of b_1, b_2 and the displacement vector $x - \pi_U(x)$ is orthogonal to both b_1 and b_2.

Example 3.10 (Projection onto a Line)
Find the projection matrix P_π onto the line through the origin spanned by $b = \begin{bmatrix}1 & 2 & 2\end{bmatrix}^\top$. b is a direction and a basis of the one-dimensional subspace (line through origin).

With (3.46), we obtain

$$P_\pi = \frac{bb^\top}{b^\top b} = \frac{1}{9}\begin{bmatrix}1\\2\\2\end{bmatrix}\begin{bmatrix}1 & 2 & 2\end{bmatrix} = \frac{1}{9}\begin{bmatrix}1 & 2 & 2\\2 & 4 & 4\\2 & 4 & 4\end{bmatrix}. \tag{3.47}$$

Let us now choose a particular x and see whether it lies in the subspace spanned by b. For $x = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^\top$, the projection is

$$\pi_U(x) = P_\pi x = \frac{1}{9}\begin{bmatrix}1 & 2 & 2\\2 & 4 & 4\\2 & 4 & 4\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = \frac{1}{9}\begin{bmatrix}5\\10\\10\end{bmatrix} \in \operatorname{span}\left[\begin{bmatrix}1\\2\\2\end{bmatrix}\right]. \tag{3.48}$$

Note that the application of P_π to $\pi_U(x)$ does not change anything, i.e., $P_\pi \pi_U(x) = \pi_U(x)$. This is expected because according to Definition 3.10, we know that a projection matrix P_π satisfies $P_\pi^2 x = P_\pi x$ for all x.

Remark. With the results from Chapter 4, we can show that $\pi_U(x)$ is an eigenvector of P_π, and the corresponding eigenvalue is 1. ♢

3.8.2 Projection onto General Subspaces

In the following, we look at orthogonal projections of vectors $x \in \mathbb{R}^n$ onto lower-dimensional subspaces $U \subseteq \mathbb{R}^n$ with $\dim(U) = m \geqslant 1$. An illustration is given in Figure 3.11.

Assume that $(b_1, \ldots, b_m)$ is an ordered basis of U. Any projection $\pi_U(x)$ onto U is necessarily an element of U. Therefore, they can be represented

If U is given by a set of spanning vectors, which are not a basis, make sure you determine a basis $b_1, \ldots, b_m$ before proceeding.

as linear combinations of the basis vectors $\boldsymbol{b}_1, \dots, \boldsymbol{b}_m$ of U, such that $\pi_U(\boldsymbol{x}) = \sum_{i=1}^{m} \lambda_i \boldsymbol{b}_i$.

The basis vectors form the columns of $\boldsymbol{B} \in \mathbb{R}^{n \times m}$, where $\boldsymbol{B} = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_m]$.

As in the 1D case, we follow a three-step procedure to find the projection $\pi_U(\boldsymbol{x})$ and the projection matrix $\boldsymbol{P}_\pi$:

1. Find the coordinates $\lambda_1, \dots, \lambda_m$ of the projection (with respect to the basis of U), such that the linear combination

$$\pi_U(\boldsymbol{x}) = \sum_{i=1}^{m} \lambda_i \boldsymbol{b}_i = \boldsymbol{B}\boldsymbol{\lambda}\,, \tag{3.49}$$

$$\boldsymbol{B} = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_m] \in \mathbb{R}^{n \times m}, \quad \boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_m]^\top \in \mathbb{R}^m\,, \tag{3.50}$$

is closest to $\boldsymbol{x} \in \mathbb{R}^n$. As in the 1D case, "closest" means "minimum distance", which implies that the vector connecting $\pi_U(\boldsymbol{x}) \in U$ and $\boldsymbol{x} \in \mathbb{R}^n$ must be orthogonal to all basis vectors of U. Therefore, we obtain m simultaneous conditions (assuming the dot product as the inner product)

$$\langle \boldsymbol{b}_1, \boldsymbol{x} - \pi_U(\boldsymbol{x}) \rangle = \boldsymbol{b}_1^\top (\boldsymbol{x} - \pi_U(\boldsymbol{x})) = 0 \tag{3.51}$$

$$\vdots$$

$$\langle \boldsymbol{b}_m, \boldsymbol{x} - \pi_U(\boldsymbol{x}) \rangle = \boldsymbol{b}_m^\top (\boldsymbol{x} - \pi_U(\boldsymbol{x})) = 0 \tag{3.52}$$

which, with $\pi_U(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{\lambda}$, can be written as

$$\boldsymbol{b}_1^\top (\boldsymbol{x} - \boldsymbol{B}\boldsymbol{\lambda}) = 0 \tag{3.53}$$

$$\vdots$$

$$\boldsymbol{b}_m^\top (\boldsymbol{x} - \boldsymbol{B}\boldsymbol{\lambda}) = 0 \tag{3.54}$$

such that we obtain a homogeneous linear equation system

$$\begin{bmatrix} \boldsymbol{b}_1^\top \\ \vdots \\ \boldsymbol{b}_m^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{x} - \boldsymbol{B}\boldsymbol{\lambda} \end{bmatrix} = \boldsymbol{0} \iff \boldsymbol{B}^\top (\boldsymbol{x} - \boldsymbol{B}\boldsymbol{\lambda}) = \boldsymbol{0} \tag{3.55}$$

$$\iff \boldsymbol{B}^\top \boldsymbol{B}\boldsymbol{\lambda} = \boldsymbol{B}^\top \boldsymbol{x}\,. \tag{3.56}$$

normal equation

The last expression is called *normal equation*. Since $\boldsymbol{b}_1, \dots, \boldsymbol{b}_m$ are a basis of U and, therefore, linearly independent, $\boldsymbol{B}^\top \boldsymbol{B} \in \mathbb{R}^{m \times m}$ is regular and can be inverted. This allows us to solve for the coefficients/ coordinates

$$\boldsymbol{\lambda} = (\boldsymbol{B}^\top \boldsymbol{B})^{-1} \boldsymbol{B}^\top \boldsymbol{x}\,. \tag{3.57}$$

pseudo-inverse

The matrix $(\boldsymbol{B}^\top \boldsymbol{B})^{-1} \boldsymbol{B}^\top$ is also called the *pseudo-inverse* of $\boldsymbol{B}$, which can be computed for non-square matrices $\boldsymbol{B}$. It only requires that $\boldsymbol{B}^\top \boldsymbol{B}$ is positive definite, which is the case if $\boldsymbol{B}$ is full rank. In practical applications (e.g., linear regression), we often add a "jitter term" $\epsilon \boldsymbol{I}$ to

$\boldsymbol{B}^\top\boldsymbol{B}$ to guarantee increased numerical stability and positive definiteness. This "ridge" can be rigorously derived using Bayesian inference. See Chapter 9 for details.

2. Find the projection $\pi_U(\boldsymbol{x}) \in U$. We already established that $\pi_U(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{\lambda}$. Therefore, with (3.57)

$$\pi_U(\boldsymbol{x}) = \boldsymbol{B}(\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top\boldsymbol{x}\,. \tag{3.58}$$

3. Find the projection matrix $\boldsymbol{P}_\pi$. From (3.58), we can immediately see that the projection matrix that solves $\boldsymbol{P}_\pi\boldsymbol{x} = \pi_U(\boldsymbol{x})$ must be

$$\boldsymbol{P}_\pi = \boldsymbol{B}(\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top\,. \tag{3.59}$$

Remark. The solution for projecting onto general subspaces includes the 1D case as a special case: If $\dim(U) = 1$, then $\boldsymbol{B}^\top\boldsymbol{B} \in \mathbb{R}$ is a scalar and we can rewrite the projection matrix in (3.59) $\boldsymbol{P}_\pi = \boldsymbol{B}(\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top$ as $\boldsymbol{P}_\pi = \frac{\boldsymbol{B}\boldsymbol{B}^\top}{\boldsymbol{B}^\top\boldsymbol{B}}$, which is exactly the projection matrix in (3.46). ♢

Example 3.11 (Projection onto a Two-dimensional Subspace)

For a subspace $U = \mathrm{span}[\begin{bmatrix}1\\1\\1\end{bmatrix}, \begin{bmatrix}0\\1\\2\end{bmatrix}] \subseteq \mathbb{R}^3$ and $\boldsymbol{x} = \begin{bmatrix}6\\0\\0\end{bmatrix} \in \mathbb{R}^3$ find the coordinates $\boldsymbol{\lambda}$ of $\boldsymbol{x}$ in terms of the subspace U, the projection point $\pi_U(\boldsymbol{x})$ and the projection matrix $\boldsymbol{P}_\pi$.

First, we see that the generating set of U is a basis (linear independence) and write the basis vectors of U into a matrix $\boldsymbol{B} = \begin{bmatrix}1 & 0\\1 & 1\\1 & 2\end{bmatrix}$.

Second, we compute the matrix $\boldsymbol{B}^\top\boldsymbol{B}$ and the vector $\boldsymbol{B}^\top\boldsymbol{x}$ as

$$\boldsymbol{B}^\top\boldsymbol{B} = \begin{bmatrix}1 & 1 & 1\\0 & 1 & 2\end{bmatrix}\begin{bmatrix}1 & 0\\1 & 1\\1 & 2\end{bmatrix} = \begin{bmatrix}3 & 3\\3 & 5\end{bmatrix}, \quad \boldsymbol{B}^\top\boldsymbol{x} = \begin{bmatrix}1 & 1 & 1\\0 & 1 & 2\end{bmatrix}\begin{bmatrix}6\\0\\0\end{bmatrix} = \begin{bmatrix}6\\0\end{bmatrix}. \tag{3.60}$$

Third, we solve the normal equation $\boldsymbol{B}^\top\boldsymbol{B}\boldsymbol{\lambda} = \boldsymbol{B}^\top\boldsymbol{x}$ to find $\boldsymbol{\lambda}$:

$$\begin{bmatrix}3 & 3\\3 & 5\end{bmatrix}\begin{bmatrix}\lambda_1\\\lambda_2\end{bmatrix} = \begin{bmatrix}6\\0\end{bmatrix} \iff \boldsymbol{\lambda} = \begin{bmatrix}5\\-3\end{bmatrix}. \tag{3.61}$$

Fourth, the projection $\pi_U(\boldsymbol{x})$ of $\boldsymbol{x}$ onto U, i.e., into the column space of $\boldsymbol{B}$, can be directly computed via

$$\pi_U(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{\lambda} = \begin{bmatrix}5\\2\\-1\end{bmatrix}. \tag{3.62}$$

projection error

The projection error is also called the *reconstruction error*.

The corresponding *projection error* is the norm of the difference vector between the original vector and its projection onto U, i.e.,

$$\|\boldsymbol{x} - \pi_U(\boldsymbol{x})\| = \left\| \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}^\top \right\| = \sqrt{6}\,. \tag{3.63}$$

Fifth, the projection matrix (for any $\boldsymbol{x} \in \mathbb{R}^3$) is given by

$$\boldsymbol{P}_\pi = \boldsymbol{B}(\boldsymbol{B}^\top\boldsymbol{B})^{-1}\boldsymbol{B}^\top = \frac{1}{6}\begin{bmatrix} 5 & 2 & -1 \\ 2 & 2 & 2 \\ -1 & 2 & 5 \end{bmatrix}. \tag{3.64}$$

To verify the results, we can (a) check whether the displacement vector $\pi_U(\boldsymbol{x}) - \boldsymbol{x}$ is orthogonal to all basis vectors of U, and (b) verify that $\boldsymbol{P}_\pi = \boldsymbol{P}_\pi^2$ (see Definition 3.10).

Remark. The projections $\pi_U(\boldsymbol{x})$ are still vectors in $\mathbb{R}^n$ although they lie in an m-dimensional subspace $U \subseteq \mathbb{R}^n$. However, to represent a projected vector we only need the m coordinates $\lambda_1, \ldots, \lambda_m$ with respect to the basis vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m$ of U. ♢

Remark. In vector spaces with general inner products, we have to pay attention when computing angles and distances, which are defined by means of the inner product. ♢

We can find approximate solutions to unsolvable linear equation systems using projections.

Projections allow us to look at situations where we have a linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ without a solution. Recall that this means that $\boldsymbol{b}$ does not lie in the span of $\boldsymbol{A}$, i.e., the vector $\boldsymbol{b}$ does not lie in the subspace spanned by the columns of $\boldsymbol{A}$. Given that the linear equation cannot be solved exactly, we can find an *approximate solution*. The idea is to find the vector in the subspace spanned by the columns of $\boldsymbol{A}$ that is closest to $\boldsymbol{b}$, i.e., we compute the orthogonal projection of $\boldsymbol{b}$ onto the subspace spanned by the columns of $\boldsymbol{A}$. This problem arises often in practice, and the solution is called the *least-squares solution* (assuming the dot product as the inner product) of an overdetermined system. This is discussed further in Section 9.4. Using reconstruction errors (3.63) is one possible approach to derive principal component analysis (Section 10.3).

least-squares solution

Remark. We just looked at projections of vectors $\boldsymbol{x}$ onto a subspace U with basis vectors $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}$. If this basis is an ONB, i.e., (3.33) and (3.34) are satisfied, the projection equation (3.58) simplifies greatly to

$$\pi_U(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{B}^\top\boldsymbol{x} \tag{3.65}$$

since $\boldsymbol{B}^\top\boldsymbol{B} = \boldsymbol{I}$ with coordinates

$$\boldsymbol{\lambda} = \boldsymbol{B}^\top\boldsymbol{x}\,. \tag{3.66}$$

This means that we no longer have to compute the inverse from (3.58), which saves computation time. ♢

3.8.3 Gram-Schmidt Orthogonalization

Projections are at the core of the Gram-Schmidt method that allows us to constructively transform any basis $(\boldsymbol{b}_1, \dots, \boldsymbol{b}_n)$ of an n-dimensional vector space V into an orthogonal/orthonormal basis $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_n)$ of V. This basis always exists (Liesen and Mehrmann, 2015) and $\operatorname{span}[\boldsymbol{b}_1, \dots, \boldsymbol{b}_n] = \operatorname{span}[\boldsymbol{u}_1, \dots, \boldsymbol{u}_n]$. The *Gram-Schmidt orthogonalization* method iteratively constructs an orthogonal basis $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_n)$ from any basis $(\boldsymbol{b}_1, \dots, \boldsymbol{b}_n)$ of V as follows:

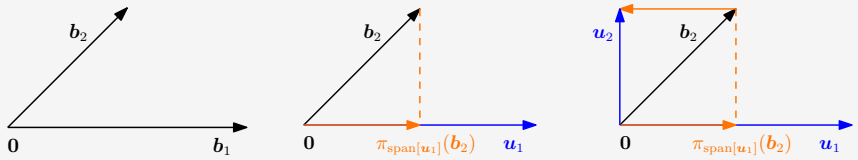
Gram-Schmidt orthogonalization

$$\boldsymbol{u}_1 := \boldsymbol{b}_1 \tag{3.67}$$

$$\boldsymbol{u}_k := \boldsymbol{b}_k - \pi_{\operatorname{span}[\boldsymbol{u}_1, \dots, \boldsymbol{u}_{k-1}]}(\boldsymbol{b}_k)\,, \quad k = 2, \dots, n\,. \tag{3.68}$$

In (3.68), the kth basis vector $\boldsymbol{b}_k$ is projected onto the subspace spanned by the first $k-1$ constructed orthogonal vectors $\boldsymbol{u}_1, \dots, \boldsymbol{u}_{k-1}$; see Section 3.8.2. This projection is then subtracted from $\boldsymbol{b}_k$ and yields a vector $\boldsymbol{u}_k$ that is orthogonal to the $(k-1)$-dimensional subspace spanned by $\boldsymbol{u}_1, \dots, \boldsymbol{u}_{k-1}$. Repeating this procedure for all n basis vectors $\boldsymbol{b}_1, \dots, \boldsymbol{b}_n$ yields an orthogonal basis $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_n)$ of V. If we normalize the $\boldsymbol{u}_k$, we obtain an ONB where $\|\boldsymbol{u}_k\| = 1$ for $k = 1, \dots, n$.

Example 3.12 (Gram-Schmidt Orthogonalization)



(a) Original non-orthogonal basis vectors $\boldsymbol{b}_1, \boldsymbol{b}_2$.

(b) First new basis vector $\boldsymbol{u}_1 = \boldsymbol{b}_1$ and projection of $\boldsymbol{b}_2$ onto the subspace spanned by $\boldsymbol{u}_1$.

(c) Orthogonal basis vectors $\boldsymbol{u}_1$ and $\boldsymbol{u}_2 = \boldsymbol{b}_2 - \pi_{\operatorname{span}[\boldsymbol{u}_1]}(\boldsymbol{b}_2)$.

Figure 3.12 Gram-Schmidt orthogonalization. (a) non-orthogonal basis $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ of $\mathbb{R}^2$; (b) first constructed basis vector $\boldsymbol{u}_1$ and orthogonal projection of $\boldsymbol{b}_2$ onto $\operatorname{span}[\boldsymbol{u}_1]$; (c) orthogonal basis $(\boldsymbol{u}_1, \boldsymbol{u}_2)$ of $\mathbb{R}^2$.

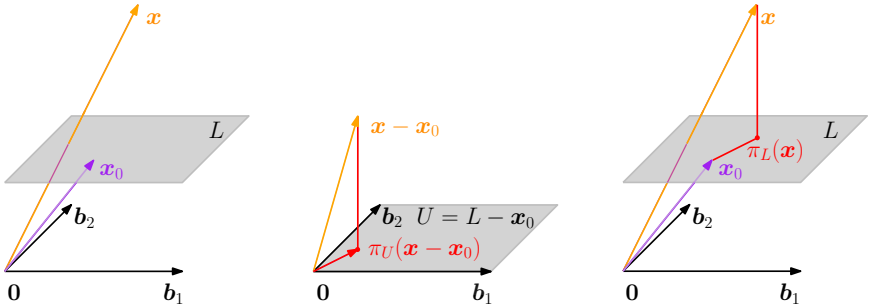
Consider a basis $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ of $\mathbb{R}^2$, where

$$\boldsymbol{b}_1 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}\,, \quad \boldsymbol{b}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\,; \tag{3.69}$$

see also Figure 3.12(a). Using the Gram-Schmidt method, we construct an orthogonal basis $(\boldsymbol{u}_1, \boldsymbol{u}_2)$ of $\mathbb{R}^2$ as follows (assuming the dot product as the inner product):

$$\boldsymbol{u}_1 := \boldsymbol{b}_1 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}\,, \tag{3.70}$$

$$\boldsymbol{u}_2 := \boldsymbol{b}_2 - \pi_{\operatorname{span}[\boldsymbol{u}_1]}(\boldsymbol{b}_2) \overset{(3.45)}{=} \boldsymbol{b}_2 - \frac{\boldsymbol{u}_1 \boldsymbol{u}_1^\top}{\|\boldsymbol{u}_1\|^2}\boldsymbol{b}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\,. \tag{3.71}$$



(a) Setting.

(b) Reduce problem to projection π_U onto vector subspace.

(c) Add support point back in to get affine projection π_L.

Figure 3.13 Projection onto an affine space. (a) original setting; (b) setting shifted by $-\boldsymbol{x}_0$ so that $\boldsymbol{x} - \boldsymbol{x}_0$ can be projected onto the direction space U; (c) projection is translated back to $\boldsymbol{x}_0 + \pi_U(\boldsymbol{x} - \boldsymbol{x}_0)$, which gives the final orthogonal projection $\pi_L(\boldsymbol{x})$.

These steps are illustrated in Figures 3.12(b) and (c). We immediately see that $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are orthogonal, i.e., $\boldsymbol{u}_1^\top \boldsymbol{u}_2 = 0$.

3.8.4 Projection onto Affine Subspaces

Thus far, we discussed how to project a vector onto a lower-dimensional subspace U. In the following, we provide a solution to projecting a vector onto an affine subspace.

Consider the setting in Figure 3.13(a). We are given an affine space $L = \boldsymbol{x}_0 + U$, where $\boldsymbol{b}_1, \boldsymbol{b}_2$ are basis vectors of U. To determine the orthogonal projection $\pi_L(\boldsymbol{x})$ of $\boldsymbol{x}$ onto L, we transform the problem into a problem that we know how to solve: the projection onto a vector subspace. In order to get there, we subtract the support point $\boldsymbol{x}_0$ from $\boldsymbol{x}$ and from L, so that $L - \boldsymbol{x}_0 = U$ is exactly the vector subspace U. We can now use the orthogonal projections onto a subspace we discussed in Section 3.8.2 and obtain the projection $\pi_U(\boldsymbol{x} - \boldsymbol{x}_0)$, which is illustrated in Figure 3.13(b). This projection can now be translated back into L by adding $\boldsymbol{x}_0$, such that we obtain the orthogonal projection onto an affine space L as

$$\pi_L(\boldsymbol{x}) = \boldsymbol{x}_0 + \pi_U(\boldsymbol{x} - \boldsymbol{x}_0)\,, \tag{3.72}$$

where $\pi_U(\cdot)$ is the orthogonal projection onto the subspace U, i.e., the direction space of L; see Figure 3.13(c).

From Figure 3.13, it is also evident that the distance of $\boldsymbol{x}$ from the affine space L is identical to the distance of $\boldsymbol{x} - \boldsymbol{x}_0$ from U, i.e.,

$$d(\boldsymbol{x}, L) = \|\boldsymbol{x} - \pi_L(\boldsymbol{x})\| = \|\boldsymbol{x} - (\boldsymbol{x}_0 + \pi_U(\boldsymbol{x} - \boldsymbol{x}_0))\| \tag{3.73a}$$

$$= d(\boldsymbol{x} - \boldsymbol{x}_0, \pi_U(\boldsymbol{x} - \boldsymbol{x}_0)) = d(\boldsymbol{x} - \boldsymbol{x}_0, U)\,. \tag{3.73b}$$

We will use projections onto an affine subspace to derive the concept of a separating hyperplane in Section 12.1.

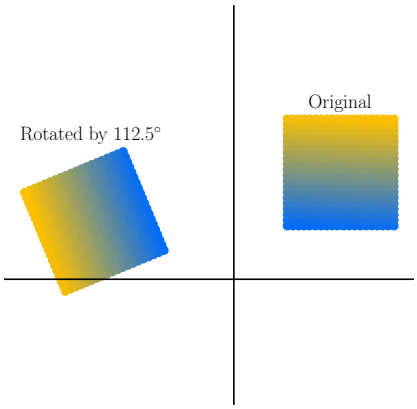


Figure 3.14 A rotation rotates objects in a plane about the origin. If the rotation angle is positive, we rotate counterclockwise.

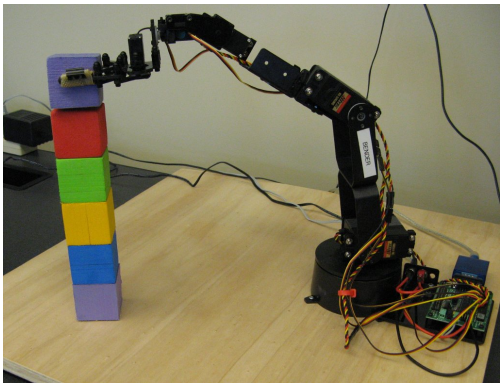


Figure 3.15 The robotic arm needs to rotate its joints in order to pick up objects or to place them correctly. Figure taken from (Deisenroth et al., 2015).

3.9 Rotations

Length and angle preservation, as discussed in Section 3.4, are the two characteristics of linear mappings with orthogonal transformation matrices. In the following, we will have a closer look at specific orthogonal transformation matrices, which describe rotations.

A *rotation* is a linear mapping (more specifically, an automorphism of a Euclidean vector space) that rotates a plane by an angle θ about the origin, i.e., the origin is a fixed point. For a positive angle $\theta > 0$, by common convention, we rotate in a counterclockwise direction. An example is shown in Figure 3.14, where the transformation matrix is

rotation

$$\boldsymbol{R} = \begin{bmatrix} -0.38 & -0.92 \\ 0.92 & -0.38 \end{bmatrix} . \tag{3.74}$$

Important application areas of rotations include computer graphics and robotics. For example, in robotics, it is often important to know how to rotate the joints of a robotic arm in order to pick up or place an object, see Figure 3.15.

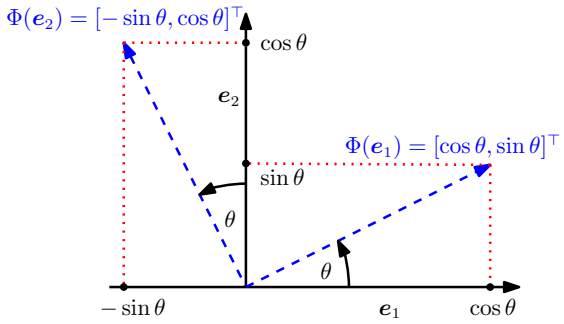


Figure 3.16 Rotation of the standard basis in $\mathbb{R}^2$ by an angle θ.

3.9.1 Rotations in $\mathbb{R}^2$

Consider the standard basis $\left\{\boldsymbol{e}_1 = \begin{bmatrix}1\\0\end{bmatrix}, \boldsymbol{e}_2 = \begin{bmatrix}0\\1\end{bmatrix}\right\}$ of $\mathbb{R}^2$, which defines the standard coordinate system in $\mathbb{R}^2$. We aim to rotate this coordinate system by an angle θ as illustrated in Figure 3.16. Note that the rotated vectors are still linearly independent and, therefore, are a basis of $\mathbb{R}^2$. This means that the rotation performs a basis change.

rotation matrix

Rotations Φ are linear mappings so that we can express them by a *rotation matrix* $\boldsymbol{R}(\theta)$. Trigonometry (see Figure 3.16) allows us to determine the coordinates of the rotated axes (the image of Φ) with respect to the standard basis in $\mathbb{R}^2$. We obtain

$$\Phi(\boldsymbol{e}_1) = \begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix}, \quad \Phi(\boldsymbol{e}_2) = \begin{bmatrix}-\sin\theta\\ \cos\theta\end{bmatrix}. \tag{3.75}$$

Therefore, the rotation matrix that performs the basis change into the rotated coordinates $\boldsymbol{R}(\theta)$ is given as

$$\boldsymbol{R}(\theta) = \begin{bmatrix}\Phi(\boldsymbol{e}_1) & \Phi(\boldsymbol{e}_2)\end{bmatrix} = \begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}. \tag{3.76}$$

3.9.2 Rotations in $\mathbb{R}^3$

In contrast to the $\mathbb{R}^2$ case, in $\mathbb{R}^3$ we can rotate any two-dimensional plane about a one-dimensional axis. The easiest way to specify the general rotation matrix is to specify how the images of the standard basis $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3$ are supposed to be rotated, and making sure these images $\boldsymbol{R}\boldsymbol{e}_1, \boldsymbol{R}\boldsymbol{e}_2, \boldsymbol{R}\boldsymbol{e}_3$ are orthonormal to each other. We can then obtain a general rotation matrix $\boldsymbol{R}$ by combining the images of the standard basis.

To have a meaningful rotation angle, we have to define what "counterclockwise" means when we operate in more than two dimensions. We use the convention that a "counterclockwise" (planar) rotation about an axis refers to a rotation about an axis when we look at the axis "head on, from the end toward the origin". In $\mathbb{R}^3$, there are therefore three (planar) rotations about the three standard basis vectors (see Figure 3.17):

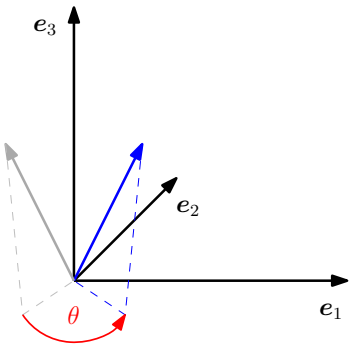


Figure 3.17 Rotation of a vector (gray) in $\mathbb{R}^3$ by an angle θ about the e_3-axis. The rotated vector is shown in blue.

- Rotation about the e_1-axis

$$\boldsymbol{R}_1(\theta) = \begin{bmatrix}\Phi(\boldsymbol{e}_1) & \Phi(\boldsymbol{e}_2) & \Phi(\boldsymbol{e}_3)\end{bmatrix} = \begin{bmatrix}1 & 0 & 0\\ 0 & \cos\theta & -\sin\theta\\ 0 & \sin\theta & \cos\theta\end{bmatrix}. \tag{3.77}$$

  Here, the e_1 coordinate is fixed, and the counterclockwise rotation is performed in the e_2e_3 plane.
- Rotation about the e_2-axis

$$\boldsymbol{R}_2(\theta) = \begin{bmatrix}\cos\theta & 0 & \sin\theta\\ 0 & 1 & 0\\ -\sin\theta & 0 & \cos\theta\end{bmatrix}. \tag{3.78}$$

  If we rotate the e_1e_3 plane about the e_2 axis, we need to look at the e_2 axis from its "tip" toward the origin.
- Rotation about the e_3-axis

$$\boldsymbol{R}_3(\theta) = \begin{bmatrix}\cos\theta & -\sin\theta & 0\\ \sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}. \tag{3.79}$$

  Figure 3.17 illustrates this.

3.9.3 Rotations in n Dimensions

The generalization of rotations from 2D and 3D to n-dimensional Euclidean vector spaces can be intuitively described as fixing $n-2$ dimensions and restrict the rotation to a two-dimensional plane in the n-dimensional space. As in the three-dimensional case, we can rotate any plane (two-dimensional subspace of $\mathbb{R}^n$).

Definition 3.11 (Givens Rotation)**.** Let V be an n-dimensional Euclidean vector space and $\Phi : V \to V$ an automorphism with transformation matrix

$$\boldsymbol{R}_{ij}(\theta) := \begin{bmatrix} \boldsymbol{I}_{i-1} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \cos\theta & \mathbf{0} & -\sin\theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{I}_{j-i-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sin\theta & \mathbf{0} & \cos\theta & \mathbf{0} \\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \boldsymbol{I}_{n-j} \end{bmatrix} \in \mathbb{R}^{n\times n}, \tag{3.80}$$

Givens rotation

for $1 \leqslant i < j \leqslant n$ and $\theta \in \mathbb{R}$. Then $\boldsymbol{R}_{ij}(\theta)$ is called a *Givens rotation*. Essentially, $\boldsymbol{R}_{ij}(\theta)$ is the identity matrix $\boldsymbol{I}_n$ with

$$r_{ii} = \cos\theta\,, \quad r_{ij} = -\sin\theta\,, \quad r_{ji} = \sin\theta\,, \quad r_{jj} = \cos\theta\,. \tag{3.81}$$

In two dimensions (i.e., $n = 2$), we obtain (3.76) as a special case.

3.9.4 Properties of Rotations

Rotations exhibit a number of useful properties, which can be derived by considering them as orthogonal matrices (Definition 3.8):

- Rotations preserve distances, i.e., $\|\boldsymbol{x}-\boldsymbol{y}\| = \|\boldsymbol{R}_\theta(\boldsymbol{x})-\boldsymbol{R}_\theta(\boldsymbol{y})\|$. In other words, rotations leave the distance between any two points unchanged after the transformation.
- Rotations preserve angles, i.e., the angle between $\boldsymbol{R}_\theta\boldsymbol{x}$ and $\boldsymbol{R}_\theta\boldsymbol{y}$ equals the angle between $\boldsymbol{x}$ and $\boldsymbol{y}$.
- Rotations in three (or more) dimensions are generally not commutative. Therefore, the order in which rotations are applied is important, even if they rotate about the same point. Only in two dimensions vector rotations are commutative, such that $\boldsymbol{R}(\phi)\boldsymbol{R}(\theta) = \boldsymbol{R}(\theta)\boldsymbol{R}(\phi)$ for all $\phi, \theta \in [0, 2\pi)$. They form an Abelian group (with multiplication) only if they rotate about the same point (e.g., the origin).

3.10 Further Reading

In this chapter, we gave a brief overview of some of the important concepts of analytic geometry, which we will use in later chapters of the book. For a broader and more in-depth overview of some of the concepts we presented, we refer to the following excellent books: Axler (2015) and Boyd and Vandenberghe (2018).

Inner products allow us to determine specific bases of vector (sub)spaces, where each vector is orthogonal to all others (orthogonal bases) using the Gram-Schmidt method. These bases are important in optimization and numerical algorithms for solving linear equation systems. For instance, Krylov subspace methods, such as conjugate gradients or the generalized minimal residual method (GMRES), minimize residual errors that are orthogonal to each other (Stoer and Burlirsch, 2002).

In machine learning, inner products are important in the context of

kernel methods (Schölkopf and Smola, 2002). Kernel methods exploit the fact that many linear algorithms can be expressed purely by inner product computations. Then, the "kernel trick" allows us to compute these inner products implicitly in a (potentially infinite-dimensional) feature space, without even knowing this feature space explicitly. This allowed the "non-linearization" of many algorithms used in machine learning, such as kernel-PCA (Schölkopf et al., 1997) for dimensionality reduction. Gaussian processes (Rasmussen and Williams, 2006) also fall into the category of kernel methods and are the current state of the art in probabilistic regression (fitting curves to data points). The idea of kernels is explored further in Chapter 12.

Projections are often used in computer graphics, e.g., to generate shadows. In optimization, orthogonal projections are often used to (iteratively) minimize residual errors. This also has applications in machine learning, e.g., in linear regression where we want to find a (linear) function that minimizes the residual errors, i.e., the lengths of the orthogonal projections of the data onto the linear function (Bishop, 2006). We will investigate this further in Chapter 9. PCA (Pearson, 1901; Hotelling, 1933) also uses projections to reduce the dimensionality of high-dimensional data. We will discuss this in more detail in Chapter 10.

Exercises

3.1 Show that $\langle \cdot, \cdot \rangle$ defined for all $\boldsymbol{x} = [x_1, x_2]^\top \in \mathbb{R}^2$ and $\boldsymbol{y} = [y_1, y_2]^\top \in \mathbb{R}^2$ by

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle := x_1 y_1 - (x_1 y_2 + x_2 y_1) + 2(x_2 y_2)$$

is an inner product.

3.2 Consider $\mathbb{R}^2$ with $\langle \cdot, \cdot \rangle$ defined for all $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{R}^2$ as

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\top \underbrace{\begin{bmatrix} 2 & 0 \\ 1 & 2 \end{bmatrix}}_{=:\boldsymbol{A}} \boldsymbol{y} .$$

Is $\langle \cdot, \cdot \rangle$ an inner product?

3.3 Compute the distance between

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix}$$

using

a. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\top \boldsymbol{y}$

b. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\top \boldsymbol{A} \boldsymbol{y}, \quad \boldsymbol{A} := \begin{bmatrix} 2 & 1 & 0 \\ 1 & 3 & -1 \\ 0 & -1 & 2 \end{bmatrix}$

3.4 Compute the angle between

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

using

a. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\top \boldsymbol{y}$

b. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\top \boldsymbol{B} \boldsymbol{y}, \quad \boldsymbol{B} := \begin{bmatrix} 2 & 1 \\ 1 & 3 \end{bmatrix}$

3.5 Consider the Euclidean vector space $\mathbb{R}^5$ with the dot product. A subspace $U \subseteq \mathbb{R}^5$ and $\boldsymbol{x} \in \mathbb{R}^5$ are given by

$$U = \operatorname{span}[\begin{bmatrix} 0 \\ -1 \\ 2 \\ 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 \\ -3 \\ 1 \\ -1 \\ 2 \end{bmatrix}, \begin{bmatrix} -3 \\ 4 \\ 1 \\ 2 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ -3 \\ 5 \\ 0 \\ 7 \end{bmatrix}], \quad \boldsymbol{x} = \begin{bmatrix} -1 \\ -9 \\ -1 \\ 4 \\ 1 \end{bmatrix} .$$

a. Determine the orthogonal projection $\pi_U(\boldsymbol{x})$ of $\boldsymbol{x}$ onto U

b. Determine the distance $d(\boldsymbol{x}, U)$

3.6 Consider $\mathbb{R}^3$ with the inner product

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \boldsymbol{x}^\top \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \boldsymbol{y} .$$

Furthermore, we define $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3$ as the standard/canonical basis in $\mathbb{R}^3$.

a. Determine the orthogonal projection $\pi_U(\boldsymbol{e}_2)$ of $\boldsymbol{e}_2$ onto

$$U = \operatorname{span}[\boldsymbol{e}_1, \boldsymbol{e}_3]\,.$$

Hint: Orthogonality is defined through the inner product.

b. Compute the distance $d(\boldsymbol{e}_2, U)$.

c. Draw the scenario: standard basis vectors and $\pi_U(\boldsymbol{e}_2)$

3.7 Let V be a vector space and π an endomorphism of V.

a. Prove that π is a projection if and only if $\mathrm{id}_V - \pi$ is a projection, where id_V is the identity endomorphism on V.

b. Assume now that π is a projection. Calculate $\mathrm{Im}(\mathrm{id}_V - \pi)$ and $\ker(\mathrm{id}_V - \pi)$ as a function of $\mathrm{Im}(\pi)$ and $\ker(\pi)$.

3.8 Using the Gram-Schmidt method, turn the basis $B = (\boldsymbol{b}_1, \boldsymbol{b}_2)$ of a two-dimensional subspace $U \subseteq \mathbb{R}^3$ into an ONB $C = (\boldsymbol{c}_1, \boldsymbol{c}_2)$ of U, where

$$\boldsymbol{b}_1 := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \boldsymbol{b}_2 := \begin{bmatrix} -1 \\ 2 \\ 0 \end{bmatrix}.$$

3.9 Let $n \in \mathbb{N}$ and let $x_1, \ldots, x_n > 0$ be n positive real numbers so that $x_1 + \ldots + x_n = 1$. Use the Cauchy-Schwarz inequality and show that

a. $\sum_{i=1}^{n} x_i^2 \geqslant \frac{1}{n}$

b. $\sum_{i=1}^{n} \frac{1}{x_i} \geqslant n^2$

Hint: Think about the dot product on $\mathbb{R}^n$. Then, choose specific vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ and apply the Cauchy-Schwarz inequality.

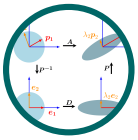
3.10 Rotate the vectors

$$\boldsymbol{x}_1 := \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \quad \boldsymbol{x}_2 := \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

by $30°$.

4

Matrix Decompositions



In Chapters 2 and 3, we studied ways to manipulate and measure vectors, projections of vectors, and linear mappings. Mappings and transformations of vectors can be conveniently described as operations performed by matrices. Moreover, data is often represented in matrix form as well, e.g., where the rows of the matrix represent different people and the columns describe different features of the people, such as weight, height, and socioeconomic status. In this chapter, we present three aspects of matrices: how to summarize matrices, how matrices can be decomposed, and how these decompositions can be used for matrix approximations.

We first consider methods that allow us to describe matrices with just a few numbers that characterize the overall properties of matrices. We will do this in the sections on determinants (Section 4.1) and eigenvalues (Section 4.2) for the important special case of square matrices. These characteristic numbers have important mathematical consequences and allow us to quickly grasp what useful properties a matrix has. From here we will proceed to matrix decomposition methods: An analogy for matrix decomposition is the factoring of numbers, such as the factoring of 21 into prime numbers $7 \cdot 3$. For this reason matrix decomposition is also often referred to as *matrix factorization*. Matrix decompositions are used to describe a matrix by means of a different representation using factors of interpretable matrices.

matrix factorization

We will first cover a square-root-like operation for symmetric, positive definite matrices, the Cholesky decomposition (Section 4.3). From here we will look at two related methods for factorizing matrices into canonical forms. The first one is known as matrix diagonalization (Section 4.4), which allows us to represent the linear mapping using a diagonal transformation matrix if we choose an appropriate basis. The second method, singular value decomposition (Section 4.5), extends this factorization to non-square matrices, and it is considered one of the fundamental concepts in linear algebra. These decompositions are helpful, as matrices representing numerical data are often very large and hard to analyze. We conclude the chapter with a systematic overview of the types of matrices and the characteristic properties that distinguish them in the form of a matrix taxonomy (Section 4.7).

The methods that we cover in this chapter will become important in

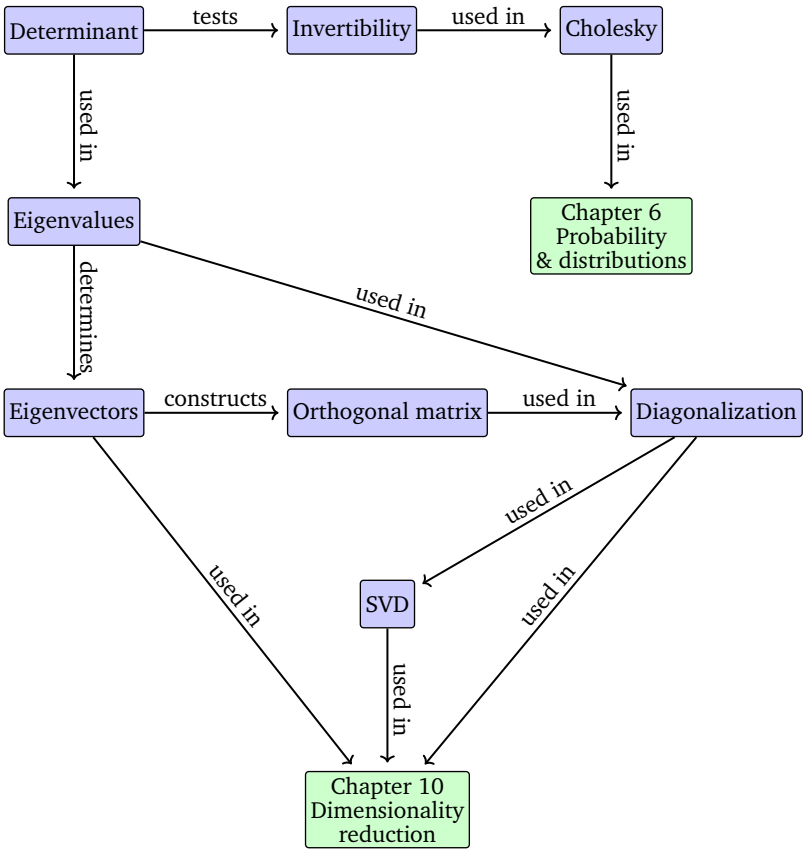


Figure 4.1 A mind map of the concepts introduced in this chapter, along with where they are used in other parts of the book.

both subsequent mathematical chapters, such as Chapter 6, but also in applied chapters, such as dimensionality reduction in Chapters 10 or density estimation in Chapter 11. This chapter's overall structure is depicted in the mind map of Figure 4.1.

4.1 Determinant and Trace

The determinant notation $|\boldsymbol{A}|$ must not be confused with the absolute value.

Determinants are important concepts in linear algebra. A determinant is a mathematical object in the analysis and solution of systems of linear equations. Determinants are only defined for square matrices $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, i.e., matrices with the same number of rows and columns. In this book, we write the determinant as $\det(\boldsymbol{A})$ or sometimes as $|\boldsymbol{A}|$ so that

$$\det(\boldsymbol{A}) = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} . \tag{4.1}$$

The *determinant* of a square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is a function that maps $\boldsymbol{A}$ determinant

onto a real number. Before providing a definition of the determinant for general $n \times n$ matrices, let us have a look at some motivating examples, and define determinants for some special matrices.

Example 4.1 (Testing for Matrix Invertibility)
Let us begin with exploring if a square matrix $\boldsymbol{A}$ is invertible (see Section 2.2.2). For the smallest cases, we already know when a matrix is invertible. If $\boldsymbol{A}$ is a 1×1 matrix, i.e., it is a scalar number, then $\boldsymbol{A} = a \implies \boldsymbol{A}^{-1} = \frac{1}{a}$. Thus $a\,\frac{1}{a} = 1$ holds, if and only if $a \neq 0$.

For 2×2 matrices, by the definition of the inverse (Definition 2.3), we know that $\boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I}$. Then, with (2.24), the inverse of $\boldsymbol{A}$ is

$$\boldsymbol{A}^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}. \tag{4.2}$$

Hence, $\boldsymbol{A}$ is invertible if and only if

$$a_{11}a_{22} - a_{12}a_{21} \neq 0\,. \tag{4.3}$$

This quantity is the determinant of $\boldsymbol{A} \in \mathbb{R}^{2\times 2}$, i.e.,

$$\det(\boldsymbol{A}) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}\,. \tag{4.4}$$

Example 4.1 points already at the relationship between determinants and the existence of inverse matrices. The next theorem states the same result for $n \times n$ matrices.

Theorem 4.1. *For any square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ it holds that $\boldsymbol{A}$ is invertible if and only if* $\det(\boldsymbol{A}) \neq 0$.

We have explicit (closed-form) expressions for determinants of small matrices in terms of the elements of the matrix. For $n = 1$,

$$\det(\boldsymbol{A}) = \det(a_{11}) = a_{11}\,. \tag{4.5}$$

For $n = 2$,

$$\det(\boldsymbol{A}) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}\,, \tag{4.6}$$

which we have observed in the preceding example.

For $n = 3$ (known as Sarrus' rule),

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} + a_{21}a_{32}a_{13} + a_{31}a_{12}a_{23} \\ - a_{31}a_{22}a_{13} - a_{11}a_{32}a_{23} - a_{21}a_{12}a_{33}\,. \tag{4.7}$$

For a memory aid of the product terms in Sarrus' rule, try tracing the elements of the triple products in the matrix.

We call a square matrix $\boldsymbol{T}$ an *upper-triangular matrix* if $T_{ij} = 0$ for $i > j$, i.e., the matrix is zero below its diagonal. Analogously, we define a *lower-triangular matrix* as a matrix with zeros above its diagonal. For a triangular matrix $\boldsymbol{T} \in \mathbb{R}^{n\times n}$, the determinant is the product of the diagonal elements, i.e.,

upper-triangular matrix

lower-triangular matrix

$$\det(\boldsymbol{T}) = \prod_{i=1}^{n} T_{ii}\,. \tag{4.8}$$

Example 4.2 (Determinants as Measures of Volume)

The notion of a determinant is natural when we consider it as a mapping from a set of n vectors spanning an object in $\mathbb{R}^n$. It turns out that the determinant $\det(\boldsymbol{A})$ is the signed volume of an n-dimensional parallelepiped formed by columns of the matrix $\boldsymbol{A}$.

For $n = 2$, the columns of the matrix form a parallelogram; see Figure 4.2. As the angle between vectors gets smaller, the area of a parallelogram shrinks, too. Consider two vectors $\boldsymbol{b}, \boldsymbol{g}$ that form the columns of a matrix $\boldsymbol{A} = [\boldsymbol{b}, \boldsymbol{g}]$. Then, the absolute value of the determinant of $\boldsymbol{A}$ is the area of the parallelogram with vertices $\mathbf{0}, \boldsymbol{b}, \boldsymbol{g}, \boldsymbol{b} + \boldsymbol{g}$. In particular, if $\boldsymbol{b}, \boldsymbol{g}$ are linearly dependent so that $\boldsymbol{b} = \lambda\boldsymbol{g}$ for some $\lambda \in \mathbb{R}$, they no longer form a two-dimensional parallelogram. Therefore, the corresponding area is 0. On the contrary, if $\boldsymbol{b}, \boldsymbol{g}$ are linearly independent and are multiples of the canonical basis vectors $\boldsymbol{e}_1, \boldsymbol{e}_2$ then they can be written as $\boldsymbol{b} = \begin{bmatrix} b \\ 0 \end{bmatrix}$ and $\boldsymbol{g} = \begin{bmatrix} 0 \\ g \end{bmatrix}$, and the determinant is $\begin{vmatrix} b & 0 \\ 0 & g \end{vmatrix} = bg - 0 = bg$.

The sign of the determinant indicates the orientation of the spanning vectors $\boldsymbol{b}, \boldsymbol{g}$ with respect to the standard basis $(\boldsymbol{e}_1, \boldsymbol{e}_2)$. In our figure, flipping the order to $\boldsymbol{g}, \boldsymbol{b}$ swaps the columns of $\boldsymbol{A}$ and reverses the orientation of the shaded area. This becomes the familiar formula: area = height $\times$ length. This intuition extends to higher dimensions. In $\mathbb{R}^3$, we consider three vectors $\boldsymbol{r}, \boldsymbol{b}, \boldsymbol{g} \in \mathbb{R}^3$ spanning the edges of a parallelepiped, i.e., a solid with faces that are parallel parallelograms (see Figure 4.3). The absolute value of the determinant of the 3×3 matrix $[\boldsymbol{r},\ \boldsymbol{b},\ \boldsymbol{g}]$ is the volume of the solid. Thus, the determinant acts as a function that measures the signed volume formed by column vectors composed in a matrix.

Consider the three linearly independent vectors $\boldsymbol{r}, \boldsymbol{g}, \boldsymbol{b} \in \mathbb{R}^3$ given as

$$\boldsymbol{r} = \begin{bmatrix} 2 \\ 0 \\ -8 \end{bmatrix}, \quad \boldsymbol{g} = \begin{bmatrix} 6 \\ 1 \\ 0 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 1 \\ 4 \\ -1 \end{bmatrix}. \tag{4.9}$$

The determinant is the signed volume of the parallelepiped formed by the columns of the matrix.

Figure 4.2 The area of the parallelogram (shaded region) spanned by the vectors $\boldsymbol{b}$ and $\boldsymbol{g}$ is $|\det([\boldsymbol{b}, \boldsymbol{g}])|$.

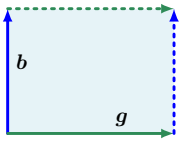
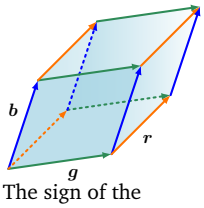


Figure 4.3 The volume of the parallelepiped (shaded volume) spanned by vectors $\boldsymbol{r}, \boldsymbol{b}, \boldsymbol{g}$ is $|\det([\boldsymbol{r}, \boldsymbol{b}, \boldsymbol{g}])|$.



The sign of the determinant indicates the orientation of the spanning vectors.

Writing these vectors as the columns of a matrix

$$\boldsymbol{A} = [\boldsymbol{r},\ \boldsymbol{g},\ \boldsymbol{b}] = \begin{bmatrix} 2 & 6 & 1 \\ 0 & 1 & 4 \\ -8 & 0 & -1 \end{bmatrix} \tag{4.10}$$

allows us to compute the desired volume as

$$V = |\det(\boldsymbol{A})| = 186\,. \tag{4.11}$$

Computing the determinant of an $n \times n$ matrix requires a general algorithm to solve the cases for $n > 3$, which we are going to explore in the following. Theorem 4.2 below reduces the problem of computing the determinant of an $n \times n$ matrix to computing the determinant of $(n-1) \times (n-1)$ matrices. By recursively applying the Laplace expansion (Theorem 4.2), we can therefore compute determinants of $n \times n$ matrices by ultimately computing determinants of 2×2 matrices.

Laplace expansion

Theorem 4.2 (Laplace Expansion)**.** *Consider a matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$. Then, for all $j = 1, \ldots, n$:*

$\det(\boldsymbol{A}_{k,j})$ is called a *minor* and $(-1)^{k+j} \det(\boldsymbol{A}_{k,j})$ a *cofactor*.

1. *Expansion along column j*

$$\det(\boldsymbol{A}) = \sum_{k=1}^{n} (-1)^{k+j} a_{kj} \det(\boldsymbol{A}_{k,j})\,. \tag{4.12}$$

2. *Expansion along row j*

$$\det(\boldsymbol{A}) = \sum_{k=1}^{n} (-1)^{k+j} a_{jk} \det(\boldsymbol{A}_{j,k})\,. \tag{4.13}$$

Here $\boldsymbol{A}_{k,j} \in \mathbb{R}^{(n-1)\times(n-1)}$ is the submatrix of $\boldsymbol{A}$ that we obtain when deleting row k and column j.

Example 4.3 (Laplace Expansion)
Let us compute the determinant of

$$\boldsymbol{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.14}$$

using the Laplace expansion along the first row. Applying (4.13) yields

$$\begin{aligned} \begin{vmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 0 & 0 & 1 \end{vmatrix} &= (-1)^{1+1} \cdot 1 \begin{vmatrix} 1 & 2 \\ 0 & 1 \end{vmatrix} \\ &\quad + (-1)^{1+2} \cdot 2 \begin{vmatrix} 3 & 2 \\ 0 & 1 \end{vmatrix} + (-1)^{1+3} \cdot 3 \begin{vmatrix} 3 & 1 \\ 0 & 0 \end{vmatrix}\,. \end{aligned} \tag{4.15}$$

We use (4.6) to compute the determinants of all 2×2 matrices and obtain

$$\det(\boldsymbol{A}) = 1(1-0) - 2(3-0) + 3(0-0) = -5\,. \tag{4.16}$$

For completeness we can compare this result to computing the determinant using Sarrus' rule (4.7):

$$\det(\boldsymbol{A}) = 1{\cdot}1{\cdot}1 + 3{\cdot}0{\cdot}3 + 0{\cdot}2{\cdot}2 - 0{\cdot}1{\cdot}3 - 1{\cdot}0{\cdot}2 - 3{\cdot}2{\cdot}1 = 1 - 6 = -5\,. \tag{4.17}$$

For $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ the determinant exhibits the following properties:

- The determinant of a matrix product is the product of the corresponding determinants, $\det(\boldsymbol{AB}) = \det(\boldsymbol{A})\det(\boldsymbol{B})$.
- Determinants are invariant to transposition, i.e., $\det(\boldsymbol{A}) = \det(\boldsymbol{A}^\top)$.
- If $\boldsymbol{A}$ is regular (invertible), then $\det(\boldsymbol{A}^{-1}) = \frac{1}{\det(\boldsymbol{A})}$.
- Similar matrices (Definition 2.22) possess the same determinant. Therefore, for a linear mapping $\Phi : V \to V$ all transformation matrices $\boldsymbol{A}_\Phi$ of Φ have the same determinant. Thus, the determinant is invariant to the choice of basis of a linear mapping.
- Adding a multiple of a column/row to another one does not change $\det(\boldsymbol{A})$.
- Multiplication of a column/row with $\lambda \in \mathbb{R}$ scales $\det(\boldsymbol{A})$ by λ. In particular, $\det(\lambda\boldsymbol{A}) = \lambda^n \det(\boldsymbol{A})$.
- Swapping two rows/columns changes the sign of $\det(\boldsymbol{A})$.

Because of the last three properties, we can use Gaussian elimination (see Section 2.1) to compute $\det(\boldsymbol{A})$ by bringing $\boldsymbol{A}$ into row-echelon form. We can stop Gaussian elimination when we have $\boldsymbol{A}$ in a triangular form where the elements below the diagonal are all 0. Recall from (4.8) that the determinant of a triangular matrix is the product of the diagonal elements.

Theorem 4.3. *A square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ has $\det(\boldsymbol{A}) \neq 0$ if and only if $\mathrm{rk}(\boldsymbol{A}) = n$. In other words, $\boldsymbol{A}$ is invertible if and only if it is full rank.*

When mathematics was mainly performed by hand, the determinant calculation was considered an essential way to analyze matrix invertibility. However, contemporary approaches in machine learning use direct numerical methods that superseded the explicit calculation of the determinant. For example, in Chapter 2, we learned that inverse matrices can be computed by Gaussian elimination. Gaussian elimination can thus be used to compute the determinant of a matrix.

Determinants will play an important theoretical role for the following sections, especially when we learn about eigenvalues and eigenvectors (Section 4.2) through the characteristic polynomial.

Definition 4.4. The *trace* of a square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is defined as trace

$$\text{tr}(\boldsymbol{A}) := \sum_{i=1}^{n} a_{ii}\,, \tag{4.18}$$

i.e. , the trace is the sum of the diagonal elements of $\boldsymbol{A}$.

The trace satisfies the following properties:

- $\text{tr}(\boldsymbol{A}+\boldsymbol{B}) = \text{tr}(\boldsymbol{A}) + \text{tr}(\boldsymbol{B})$ for $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$
- $\text{tr}(\alpha\boldsymbol{A}) = \alpha\text{tr}(\boldsymbol{A})\,, \alpha \in \mathbb{R}$ for $\boldsymbol{A} \in \mathbb{R}^{n\times n}$
- $\text{tr}(\boldsymbol{I}_n) = n$
- $\text{tr}(\boldsymbol{A}\boldsymbol{B}) = \text{tr}(\boldsymbol{B}\boldsymbol{A})$ for $\boldsymbol{A} \in \mathbb{R}^{n\times k}, \boldsymbol{B} \in \mathbb{R}^{k\times n}$

It can be shown that only one function satisfies these four properties together – the trace (Gohberg et al., 2012).

The properties of the trace of matrix products are more general. Specifically, the trace is invariant under cyclic permutations, i.e.,

The trace is invariant under cyclic permutations.

$$\text{tr}(\boldsymbol{A}\boldsymbol{K}\boldsymbol{L}) = \text{tr}(\boldsymbol{K}\boldsymbol{L}\boldsymbol{A}) \tag{4.19}$$

for matrices $\boldsymbol{A} \in \mathbb{R}^{a\times k}, \boldsymbol{K} \in \mathbb{R}^{k\times l}, \boldsymbol{L} \in \mathbb{R}^{l\times a}$. This property generalizes to products of an arbitrary number of matrices. As a special case of (4.19), it follows that for two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$

$$\text{tr}(\boldsymbol{x}\boldsymbol{y}^\top) = \text{tr}(\boldsymbol{y}^\top\boldsymbol{x}) = \boldsymbol{y}^\top\boldsymbol{x} \in \mathbb{R}\,. \tag{4.20}$$

Given a linear mapping $\Phi : V \to V$, where V is a vector space, we define the trace of this map by using the trace of matrix representation of Φ. For a given basis of V, we can describe Φ by means of the transformation matrix $\boldsymbol{A}$. Then the trace of Φ is the trace of $\boldsymbol{A}$. For a different basis of V, it holds that the corresponding transformation matrix $\boldsymbol{B}$ of Φ can be obtained by a basis change of the form $\boldsymbol{S}^{-1}\boldsymbol{A}\boldsymbol{S}$ for suitable $\boldsymbol{S}$ (see Section 2.7.2). For the corresponding trace of Φ, this means

$$\text{tr}(\boldsymbol{B}) = \text{tr}(\boldsymbol{S}^{-1}\boldsymbol{A}\boldsymbol{S}) \overset{(4.19)}{=} \text{tr}(\boldsymbol{A}\boldsymbol{S}\boldsymbol{S}^{-1}) = \text{tr}(\boldsymbol{A})\,. \tag{4.21}$$

Hence, while matrix representations of linear mappings are basis dependent the trace of a linear mapping Φ is independent of the basis.

In this section, we covered determinants and traces as functions characterizing a square matrix. Taking together our understanding of determinants and traces we can now define an important equation describing a matrix $\boldsymbol{A}$ in terms of a polynomial, which we will use extensively in the following sections.

Definition 4.5 (Characteristic Polynomial)**.** For $\lambda \in \mathbb{R}$ and a square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$

$$p_{\boldsymbol{A}}(\lambda) := \det(\boldsymbol{A} - \lambda\boldsymbol{I}) \tag{4.22a}$$

$$= c_0 + c_1\lambda + c_2\lambda^2 + \cdots + c_{n-1}\lambda^{n-1} + (-1)^n\lambda^n\,, \tag{4.22b}$$

characteristic polynomial

$c_0, \ldots, c_{n-1} \in \mathbb{R}$, is the *characteristic polynomial* of $\boldsymbol{A}$. In particular,

$$c_0 = \det(\boldsymbol{A})\,, \tag{4.23}$$
$$c_{n-1} = (-1)^{n-1}\mathrm{tr}(\boldsymbol{A})\,. \tag{4.24}$$

The characteristic polynomial (4.22a) will allow us to compute eigenvalues and eigenvectors, covered in the next section.

4.2 Eigenvalues and Eigenvectors

We will now get to know a new way to characterize a matrix and its associated linear mapping. Recall from Section 2.7.1 that every linear mapping has a unique transformation matrix given an ordered basis. We can interpret linear mappings and their associated transformation matrices by performing an "eigen" analysis. As we will see, the eigenvalues of a linear mapping will tell us how a special set of vectors, the eigenvectors, is transformed by the linear mapping.

Eigen is a German word meaning "characteristic", "self", or "own".

Definition 4.6. Let $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ be a square matrix. Then $\lambda \in \mathbb{R}$ is an *eigenvalue* of $\boldsymbol{A}$ and $\boldsymbol{x} \in \mathbb{R}^n\backslash\{\boldsymbol{0}\}$ is the corresponding *eigenvector* of $\boldsymbol{A}$ if

eigenvalue
eigenvector

$$\boldsymbol{A}\boldsymbol{x} = \lambda\boldsymbol{x}\,. \tag{4.25}$$

We call (4.25) the *eigenvalue equation*.

eigenvalue equation

Remark. In the linear algebra literature and software, it is often a convention that eigenvalues are sorted in descending order, so that the largest eigenvalue and associated eigenvector are called the first eigenvalue and its associated eigenvector, and the second largest called the second eigenvalue and its associated eigenvector, and so on. However, textbooks and publications may have different or no notion of orderings. We do not want to presume an ordering in this book if not stated explicitly. ♢

The following statements are equivalent:

- λ is an eigenvalue of $\boldsymbol{A} \in \mathbb{R}^{n\times n}$.
- There exists an $\boldsymbol{x} \in \mathbb{R}^n\backslash\{\boldsymbol{0}\}$ with $\boldsymbol{A}\boldsymbol{x} = \lambda\boldsymbol{x}$, or equivalently, $(\boldsymbol{A} - \lambda\boldsymbol{I}_n)\boldsymbol{x} = \boldsymbol{0}$ can be solved non-trivially, i.e., $\boldsymbol{x} \neq \boldsymbol{0}$.
- $\mathrm{rk}(\boldsymbol{A} - \lambda\boldsymbol{I}_n) < n$.
- $\det(\boldsymbol{A} - \lambda\boldsymbol{I}_n) = 0$.

Definition 4.7 (Collinearity and Codirection)**.** Two vectors that point in the same direction are called *codirected*. Two vectors are *collinear* if they point in the same or the opposite direction.

codirected
collinear

Remark (Non-uniqueness of eigenvectors). If $\boldsymbol{x}$ is an eigenvector of $\boldsymbol{A}$ associated with eigenvalue λ, then for any $c \in \mathbb{R}\backslash\{0\}$ it holds that $c\boldsymbol{x}$ is an eigenvector of $\boldsymbol{A}$ with the same eigenvalue since

$$\boldsymbol{A}(c\boldsymbol{x}) = c\boldsymbol{A}\boldsymbol{x} = c\lambda\boldsymbol{x} = \lambda(c\boldsymbol{x})\,. \tag{4.26}$$

Thus, all vectors that are collinear to $\boldsymbol{x}$ are also eigenvectors of $\boldsymbol{A}$.

♢

Theorem 4.8. *$\lambda \in \mathbb{R}$ is an eigenvalue of $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ if and only if λ is a root of the characteristic polynomial $p_{\boldsymbol{A}}(\lambda)$ of $\boldsymbol{A}$.*

algebraic multiplicity

Definition 4.9. Let a square matrix $\boldsymbol{A}$ have an eigenvalue λ_i. The *algebraic multiplicity* of λ_i is the number of times the root appears in the characteristic polynomial.

Definition 4.10 (Eigenspace and Eigenspectrum)**.** For $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, the set of all eigenvectors of $\boldsymbol{A}$ associated with an eigenvalue λ spans a subspace of $\mathbb{R}^n$, which is called the *eigenspace* of $\boldsymbol{A}$ with respect to λ and is denoted by E_λ. The set of all eigenvalues of $\boldsymbol{A}$ is called the *eigenspectrum*, or just *spectrum*, of $\boldsymbol{A}$.

eigenspace
eigenspectrum
spectrum

If λ is an eigenvalue of $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, then the corresponding eigenspace E_λ is the solution space of the homogeneous system of linear equations $(\boldsymbol{A}-\lambda\boldsymbol{I})\boldsymbol{x} = \boldsymbol{0}$. Geometrically, the eigenvector corresponding to a nonzero eigenvalue points in a direction that is stretched by the linear mapping. The eigenvalue is the factor by which it is stretched. If the eigenvalue is negative, the direction of the stretching is flipped.

Example 4.4 (The Case of the Identity Matrix)
The identity matrix $\boldsymbol{I} \in \mathbb{R}^{n\times n}$ has characteristic polynomial $p_{\boldsymbol{I}}(\lambda) = \det(\boldsymbol{I}-\lambda\boldsymbol{I}) = (1-\lambda)^n = 0$, which has only one eigenvalue $\lambda = 1$ that occurs n times. Moreover, $\boldsymbol{I}\boldsymbol{x} = \lambda\boldsymbol{x} = 1\boldsymbol{x}$ holds for all vectors $\boldsymbol{x} \in \mathbb{R}^n\backslash\{\boldsymbol{0}\}$. Because of this, the sole eigenspace E_1 of the identity matrix spans n dimensions, and all n standard basis vectors of $\mathbb{R}^n$ are eigenvectors of $\boldsymbol{I}$.

Useful properties regarding eigenvalues and eigenvectors include the following:

- A matrix $\boldsymbol{A}$ and its transpose $\boldsymbol{A}^\top$ possess the same eigenvalues, but not necessarily the same eigenvectors.
- The eigenspace E_λ is the null space of $\boldsymbol{A} - \lambda\boldsymbol{I}$ since

$$\boldsymbol{A}\boldsymbol{x} = \lambda\boldsymbol{x} \iff \boldsymbol{A}\boldsymbol{x} - \lambda\boldsymbol{x} = \boldsymbol{0} \tag{4.27a}$$
$$\iff (\boldsymbol{A} - \lambda\boldsymbol{I})\boldsymbol{x} = \boldsymbol{0} \iff \boldsymbol{x} \in \ker(\boldsymbol{A} - \lambda\boldsymbol{I}). \tag{4.27b}$$

- Similar matrices (see Definition 2.22) possess the same eigenvalues. Therefore, a linear mapping Φ has eigenvalues that are independent of the choice of basis of its transformation matrix. This makes eigenvalues, together with the determinant and the trace, key characteristic parameters of a linear mapping as they are all invariant under basis change.
- Symmetric, positive definite matrices always have positive, real eigenvalues.

Example 4.5 (Computing Eigenvalues, Eigenvectors, and Eigenspaces)
Let us find the eigenvalues and eigenvectors of the 2×2 matrix

$$\boldsymbol{A} = \begin{bmatrix} 4 & 2 \\ 1 & 3 \end{bmatrix} . \tag{4.28}$$

Step 1: Characteristic Polynomial. From our definition of the eigenvector $\boldsymbol{x} \neq \boldsymbol{0}$ and eigenvalue λ of $\boldsymbol{A}$, there will be a vector such that $\boldsymbol{A}\boldsymbol{x} = \lambda\boldsymbol{x}$, i.e., $(\boldsymbol{A} - \lambda\boldsymbol{I})\boldsymbol{x} = \boldsymbol{0}$. Since $\boldsymbol{x} \neq \boldsymbol{0}$, this requires that the kernel (null space) of $\boldsymbol{A} - \lambda\boldsymbol{I}$ contains more elements than just $\boldsymbol{0}$. This means that $\boldsymbol{A} - \lambda\boldsymbol{I}$ is not invertible and therefore $\det(\boldsymbol{A} - \lambda\boldsymbol{I}) = 0$. Hence, we need to compute the roots of the characteristic polynomial (4.22a) to find the eigenvalues.

Step 2: Eigenvalues. The characteristic polynomial is

$$\begin{aligned} p_{\boldsymbol{A}}(\lambda) &= \det(\boldsymbol{A} - \lambda\boldsymbol{I}) && \text{(4.29a)} \\ &= \det\left(\begin{bmatrix} 4 & 2 \\ 1 & 3 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}\right) = \begin{vmatrix} 4-\lambda & 2 \\ 1 & 3-\lambda \end{vmatrix} && \text{(4.29b)} \\ &= (4-\lambda)(3-\lambda) - 2 \cdot 1 . && \text{(4.29c)} \end{aligned}$$

We factorize the characteristic polynomial and obtain

$$p(\lambda) = (4-\lambda)(3-\lambda) - 2 \cdot 1 = 10 - 7\lambda + \lambda^2 = (2-\lambda)(5-\lambda) \tag{4.30}$$

giving the roots $\lambda_1 = 2$ and $\lambda_2 = 5$.

Step 3: Eigenvectors and Eigenspaces. We find the eigenvectors that correspond to these eigenvalues by looking at vectors $\boldsymbol{x}$ such that

$$\begin{bmatrix} 4-\lambda & 2 \\ 1 & 3-\lambda \end{bmatrix} \boldsymbol{x} = \boldsymbol{0} . \tag{4.31}$$

For $\lambda = 5$ we obtain

$$\begin{bmatrix} 4-5 & 2 \\ 1 & 3-5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -1 & 2 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \boldsymbol{0} . \tag{4.32}$$

We solve this homogeneous system and obtain a solution space

$$E_5 = \operatorname{span}[\begin{bmatrix} 2 \\ 1 \end{bmatrix}] . \tag{4.33}$$

This eigenspace is one-dimensional as it possesses a single basis vector.

Analogously, we find the eigenvector for $\lambda = 2$ by solving the homogeneous system of equations

$$\begin{bmatrix} 4-2 & 2 \\ 1 & 3-2 \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix} \boldsymbol{x} = \boldsymbol{0} . \tag{4.34}$$

This means any vector $\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, where $x_2 = -x_1$, such as $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$, is an eigenvector with eigenvalue 2. The corresponding eigenspace is given as

$$E_2 = \text{span}[\begin{bmatrix} 1 \\ -1 \end{bmatrix}]\,. \tag{4.35}$$

The two eigenspaces E_5 and E_2 in Example 4.5 are one-dimensional as they are each spanned by a single vector. However, in other cases we may have multiple identical eigenvalues (see Definition 4.9) and the eigenspace may have more than one dimension.

geometric multiplicity

Definition 4.11. Let λ_i be an eigenvalue of a square matrix $\boldsymbol{A}$. Then the *geometric multiplicity* of λ_i is the number of linearly independent eigenvectors associated with λ_i. In other words, it is the dimensionality of the eigenspace spanned by the eigenvectors associated with λ_i.

Remark. A specific eigenvalue's geometric multiplicity must be at least one because every eigenvalue has at least one associated eigenvector. An eigenvalue's geometric multiplicity cannot exceed its algebraic multiplicity, but it may be lower. ♢

Example 4.6

The matrix $\boldsymbol{A} = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix}$ has two repeated eigenvalues $\lambda_1 = \lambda_2 = 2$ and an algebraic multiplicity of 2. The eigenvalue has, however, only one distinct unit eigenvector $\boldsymbol{x}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and, thus, geometric multiplicity 1.

Graphical Intuition in Two Dimensions

Let us gain some intuition for determinants, eigenvectors, and eigenvalues using different linear mappings. Figure 4.4 depicts five transformation matrices $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_5$ and their impact on a square grid of points, centered at the origin:

In geometry, the area-preserving properties of this type of shearing parallel to an axis is also known as Cavalieri's principle of equal areas for parallelograms (Katz, 2004).

- $\boldsymbol{A}_1 = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 2 \end{bmatrix}$. The direction of the two eigenvectors correspond to the canonical basis vectors in $\mathbb{R}^2$, i.e., to two cardinal axes. The vertical axis is extended by a factor of 2 (eigenvalue $\lambda_1 = 2$), and the horizontal axis is compressed by factor $\frac{1}{2}$ (eigenvalue $\lambda_2 = \frac{1}{2}$). The mapping is area preserving ($\det(\boldsymbol{A}_1) = 1 = 2 \cdot \frac{1}{2}$).
- $\boldsymbol{A}_2 = \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix}$ corresponds to a shearing mapping , i.e., it shears the points along the horizontal axis to the right if they are on the positive

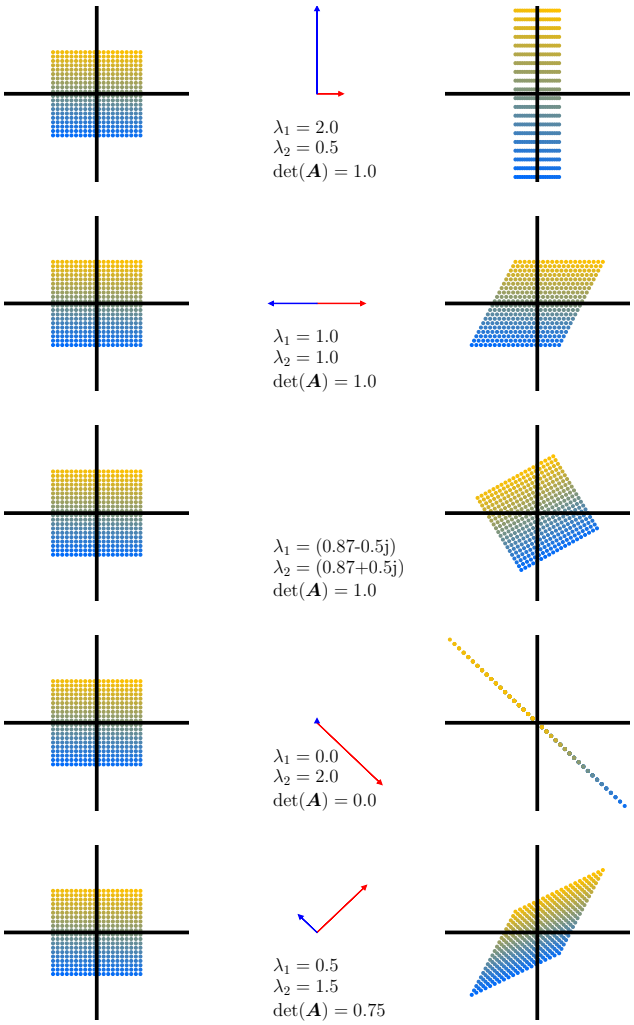


Figure 4.4 Determinants and eigenspaces. Overview of five linear mappings and their associated transformation matrices $\boldsymbol{A}_i \in \mathbb{R}^{2\times 2}$ projecting 400 color-coded points $\boldsymbol{x} \in \mathbb{R}^2$ (left column) onto target points $\boldsymbol{A}_i\boldsymbol{x}$ (right column). The central column depicts the first eigenvector, stretched by its associated eigenvalue λ_1, and the second eigenvector stretched by its eigenvalue λ_2. Each row depicts the effect of one of five transformation matrices $\boldsymbol{A}_i$ with respect to the standard basis.

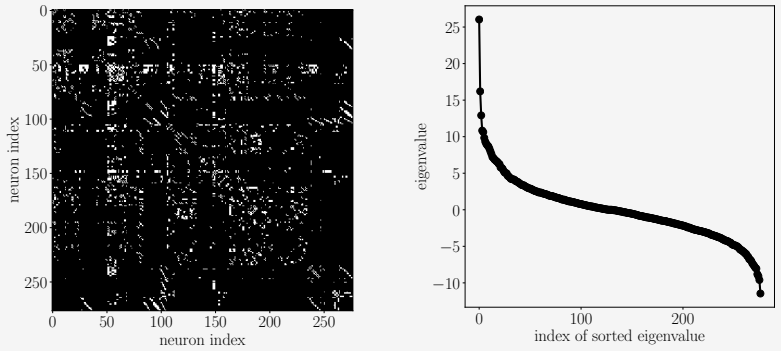
half of the vertical axis, and to the left vice versa. This mapping is area preserving ($\det(\boldsymbol{A}_2) = 1$). The eigenvalue $\lambda_1 = 1 = \lambda_2$ is repeated and the eigenvectors are collinear (drawn here for emphasis in two opposite directions). This indicates that the mapping acts only along one direction (the horizontal axis).

- $\boldsymbol{A}_3 = \begin{bmatrix} \cos(\frac{\pi}{6}) & -\sin(\frac{\pi}{6}) \\ \sin(\frac{\pi}{6}) & \cos(\frac{\pi}{6}) \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \sqrt{3} & -1 \\ 1 & \sqrt{3} \end{bmatrix}$ The matrix $\boldsymbol{A}_3$ rotates the points by $\frac{\pi}{6}\,\text{rad} = 30°$ counter-clockwise and has only complex eigenvalues, reflecting that the mapping is a rotation (hence, no eigenvectors are drawn). A rotation has to be volume preserving, and so the determinant is 1. For more details on rotations, we refer to Section 3.9.
- $\boldsymbol{A}_4 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$ represents a mapping in the standard basis that collapses a two-dimensional domain onto one dimension. Since one eigen-

value is 0, the space in direction of the (blue) eigenvector corresponding to $\lambda_1 = 0$ collapses, while the orthogonal (red) eigenvector stretches space by a factor $\lambda_2 = 2$. Therefore, the area of the image is 0.

- $\boldsymbol{A}_5 = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix}$ is a shear-and-stretch mapping that scales space by 75% since $|\det(\boldsymbol{A}_5)| = \frac{3}{4}$. It stretches space along the (red) eigenvector of λ_2 by a factor 1.5 and compresses it along the orthogonal (blue) eigenvector by a factor 0.5.

Example 4.7 (Eigenspectrum of a Biological Neural Network)



(a) Connectivity matrix. (b) Eigenspectrum.

Figure 4.5 Caenorhabditis elegans neural network (Kaiser and Hilgetag, 2006).(a) Symmetrized connectivity matrix; (b) Eigenspectrum.

Methods to analyze and learn from network data are an essential component of machine learning methods. The key to understanding networks is the connectivity between network nodes, especially if two nodes are connected to each other or not. In data science applications, it is often useful to study the matrix that captures this connectivity data.

We build a connectivity/adjacency matrix $\boldsymbol{A} \in \mathbb{R}^{277 \times 277}$ of the complete neural network of the worm *C.Elegans*. Each row/column represents one of the 277 neurons of this worm's brain. The connectivity matrix $\boldsymbol{A}$ has a value of $a_{ij} = 1$ if neuron i talks to neuron j through a synapse, and $a_{ij} = 0$ otherwise. The connectivity matrix is not symmetric, which implies that eigenvalues may not be real valued. Therefore, we compute a symmetrized version of the connectivity matrix as $\boldsymbol{A}_{sym} := \boldsymbol{A} + \boldsymbol{A}^\top$. This new matrix $\boldsymbol{A}_{sym}$ is shown in Figure 4.5(a) and has a nonzero value a_{ij} if and only if two neurons are connected (white pixels), irrespective of the direction of the connection. In Figure 4.5(b), we show the corresponding eigenspectrum of $\boldsymbol{A}_{sym}$. The horizontal axis shows the index of the eigenvalues, sorted in descending order. The vertical axis shows the corresponding eigenvalue. The S-like shape of this eigenspectrum is typical for many biological neural networks. The underlying mechanism responsible for this is an area of active neuroscience research.

Theorem 4.12. *The eigenvectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ of a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ with n distinct eigenvalues $\lambda_1, \ldots, \lambda_n$ are linearly independent.*

This theorem states that eigenvectors of a matrix with n distinct eigenvalues form a basis of $\mathbb{R}^n$.

Definition 4.13. A square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is *defective* if it possesses fewer than n linearly independent eigenvectors. defective

A non-defective matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ does not necessarily require n distinct eigenvalues, but it does require that the eigenvectors form a basis of $\mathbb{R}^n$. Looking at the eigenspaces of a defective matrix, it follows that the sum of the dimensions of the eigenspaces is less than n. Specifically, a defective matrix has at least one eigenvalue λ_i with an algebraic multiplicity $m > 1$ and a geometric multiplicity of less than m.

Remark. A defective matrix cannot have n distinct eigenvalues, as distinct eigenvalues have linearly independent eigenvectors (Theorem 4.12). ♢

Theorem 4.14. *Given a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, we can always obtain a symmetric, positive semidefinite matrix $\boldsymbol{S} \in \mathbb{R}^{n\times n}$ by defining*

$$\boldsymbol{S} := \boldsymbol{A}^\top \boldsymbol{A}\,. \tag{4.36}$$

Remark. If $\mathrm{rk}(\boldsymbol{A}) = n$, then $\boldsymbol{S} := \boldsymbol{A}^\top \boldsymbol{A}$ is symmetric, positive definite. ♢

Understanding why Theorem 4.14 holds is insightful for how we can use symmetrized matrices: Symmetry requires $\boldsymbol{S} = \boldsymbol{S}^\top$, and by inserting (4.36) we obtain $\boldsymbol{S} = \boldsymbol{A}^\top \boldsymbol{A} = \boldsymbol{A}^\top (\boldsymbol{A}^\top)^\top = (\boldsymbol{A}^\top \boldsymbol{A})^\top = \boldsymbol{S}^\top$. Moreover, positive semidefiniteness (Section 3.2.3) requires that $\boldsymbol{x}^\top \boldsymbol{S}\boldsymbol{x} \geqslant 0$ and inserting (4.36) we obtain $\boldsymbol{x}^\top \boldsymbol{S}\boldsymbol{x} = \boldsymbol{x}^\top \boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{x} = (\boldsymbol{x}^\top \boldsymbol{A}^\top)(\boldsymbol{A}\boldsymbol{x}) = (\boldsymbol{A}\boldsymbol{x})^\top (\boldsymbol{A}\boldsymbol{x}) \geqslant 0$, because the dot product computes a sum of squares (which are themselves non-negative).

spectral theorem

Theorem 4.15 (Spectral Theorem)**.** *If $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is symmetric, there exists an orthonormal basis of the corresponding vector space V consisting of eigenvectors of $\boldsymbol{A}$, and each eigenvalue is real.*

A direct implication of the spectral theorem is that the eigendecomposition of a symmetric matrix $\boldsymbol{A}$ exists (with real eigenvalues), and that we can find an ONB of eigenvectors so that $\boldsymbol{A} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^\top$, where $\boldsymbol{D}$ is diagonal and the columns of $\boldsymbol{P}$ contain the eigenvectors.

Example 4.8
Consider the matrix

$$\boldsymbol{A} = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & 2 \\ 2 & 2 & 3 \end{bmatrix}. \tag{4.37}$$

The characteristic polynomial of $\boldsymbol{A}$ is

$$p_{\boldsymbol{A}}(\lambda) = -(\lambda-1)^2(\lambda-7)\,, \tag{4.38}$$

so that we obtain the eigenvalues $\lambda_1 = 1$ and $\lambda_2 = 7$, where λ_1 is a repeated eigenvalue. Following our standard procedure for computing eigenvectors, we obtain the eigenspaces

$$E_1 = \operatorname{span}[\underbrace{\begin{bmatrix}-1\\1\\0\end{bmatrix}}_{=:\boldsymbol{x}_1}, \underbrace{\begin{bmatrix}-1\\0\\1\end{bmatrix}}_{=:\boldsymbol{x}_2}], \quad E_7 = \operatorname{span}[\underbrace{\begin{bmatrix}1\\1\\1\end{bmatrix}}_{=:\boldsymbol{x}_3}]\,. \tag{4.39}$$

We see that $\boldsymbol{x}_3$ is orthogonal to both $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. However, since $\boldsymbol{x}_1^\top \boldsymbol{x}_2 = 1 \neq 0$, they are not orthogonal. The spectral theorem (Theorem 4.15) states that there exists an orthogonal basis, but the one we have is not orthogonal. However, we can construct one.

To construct such a basis, we exploit the fact that $\boldsymbol{x}_1, \boldsymbol{x}_2$ are eigenvectors associated with the same eigenvalue λ. Therefore, for any $\alpha, \beta \in \mathbb{R}$ it holds that

$$\boldsymbol{A}(\alpha\boldsymbol{x}_1 + \beta\boldsymbol{x}_2) = \boldsymbol{A}\boldsymbol{x}_1\alpha + \boldsymbol{A}\boldsymbol{x}_2\beta = \lambda(\alpha\boldsymbol{x}_1 + \beta\boldsymbol{x}_2)\,, \tag{4.40}$$

i.e., any linear combination of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is also an eigenvector of $\boldsymbol{A}$ associated with λ. The Gram-Schmidt algorithm (Section 3.8.3) is a method for iteratively constructing an orthogonal/orthonormal basis from a set of basis vectors using such linear combinations. Therefore, even if $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are not orthogonal, we can apply the Gram-Schmidt algorithm and find eigenvectors associated with $\lambda_1 = 1$ that are orthogonal to each other (and to $\boldsymbol{x}_3$). In our example, we will obtain

$$\boldsymbol{x}_1' = \begin{bmatrix}-1\\1\\0\end{bmatrix}, \quad \boldsymbol{x}_2' = \frac{1}{2}\begin{bmatrix}-1\\-1\\2\end{bmatrix}, \tag{4.41}$$

which are orthogonal to each other, orthogonal to $\boldsymbol{x}_3$, and eigenvectors of $\boldsymbol{A}$ associated with $\lambda_1 = 1$.

Before we conclude our considerations of eigenvalues and eigenvectors it is useful to tie these matrix characteristics together with the concepts of the determinant and the trace.

Theorem 4.16. *The determinant of a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is the product of its eigenvalues, i.e.,*

$$\det(\boldsymbol{A}) = \prod_{i=1}^{n} \lambda_i\,, \tag{4.42}$$

where $\lambda_i \in \mathbb{C}$ are (possibly repeated) eigenvalues of $\boldsymbol{A}$.

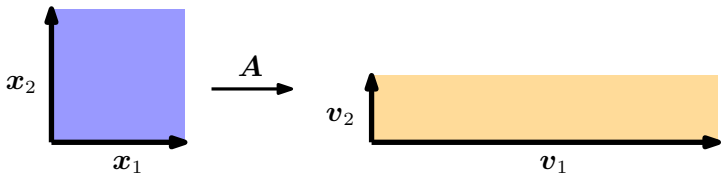


Figure 4.6 Geometric interpretation of eigenvalues. The eigenvectors of $\boldsymbol{A}$ get stretched by the corresponding eigenvalues. The area of the unit square changes by $|\lambda_1\lambda_2|$, the perimeter changes by a factor of $\frac{1}{2}(|\lambda_1| + |\lambda_2|)$.

Theorem 4.17. *The trace of a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is the sum of its eigenvalues, i.e.,*

$$tr(\boldsymbol{A}) = \sum_{i=1}^{n} \lambda_i \,, \tag{4.43}$$

where $\lambda_i \in \mathbb{C}$ are (possibly repeated) eigenvalues of $\boldsymbol{A}$.

Let us provide a geometric intuition of these two theorems. Consider a matrix $\boldsymbol{A} \in \mathbb{R}^{2\times 2}$ that possesses two linearly independent eigenvectors $\boldsymbol{x}_1, \boldsymbol{x}_2$. For this example, we assume $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ are an ONB of $\mathbb{R}^2$ so that they are orthogonal and the area of the square they span is 1; see Figure 4.6. From Section 4.1, we know that the determinant computes the change of area of unit square under the transformation $\boldsymbol{A}$. In this example, we can compute the change of area explicitly: Mapping the eigenvectors using $\boldsymbol{A}$ gives us vectors $\boldsymbol{v}_1 = \boldsymbol{A}\boldsymbol{x}_1 = \lambda_1\boldsymbol{x}_1$ and $\boldsymbol{v}_2 = \boldsymbol{A}\boldsymbol{x}_2 = \lambda_2\boldsymbol{x}_2$, i.e., the new vectors $\boldsymbol{v}_i$ are scaled versions of the eigenvectors $\boldsymbol{x}_i$, and the scaling factors are the corresponding eigenvalues λ_i. $\boldsymbol{v}_1, \boldsymbol{v}_2$ are still orthogonal, and the area of the rectangle they span is $|\lambda_1\lambda_2|$.

Given that $\boldsymbol{x}_1, \boldsymbol{x}_2$ (in our example) are orthonormal, we can directly compute the perimeter of the unit square as $2(1 + 1)$. Mapping the eigenvectors using $\boldsymbol{A}$ creates a rectangle whose perimeter is $2(|\lambda_1| + |\lambda_2|)$. Therefore, the sum of the absolute values of the eigenvalues tells us how the perimeter of the unit square changes under the transformation matrix $\boldsymbol{A}$.

Example 4.9 (Google's PageRank – Webpages as Eigenvectors)
Google uses the eigenvector corresponding to the maximal eigenvalue of a matrix $\boldsymbol{A}$ to determine the rank of a page for search. The idea for the PageRank algorithm, developed at Stanford University by Larry Page and Sergey Brin in 1996, was that the importance of any web page can be approximated by the importance of pages that link to it. For this, they write down all web sites as a huge directed graph that shows which page links to which. PageRank computes the weight (importance) $x_i \geqslant 0$ of a web site a_i by counting the number of pages pointing to a_i. Moreover, PageRank takes into account the importance of the web sites that link to a_i. The navigation behavior of a user is then modeled by a transition matrix $\boldsymbol{A}$ of this graph that tells us with what (click) probability somebody will end up

on a different web site. The matrix $\boldsymbol{A}$ has the property that for any initial rank/importance vector $\boldsymbol{x}$ of a web site the sequence $\boldsymbol{x}, \boldsymbol{A}\boldsymbol{x}, \boldsymbol{A}^2\boldsymbol{x}, \ldots$ converges to a vector $\boldsymbol{x}^*$. This vector is called the *PageRank* and satisfies $\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{x}^*$, i.e., it is an eigenvector (with corresponding eigenvalue 1) of $\boldsymbol{A}$. After normalizing $\boldsymbol{x}^*$, such that $\|\boldsymbol{x}^*\| = 1$, we can interpret the entries as probabilities. More details and different perspectives on PageRank can be found in the original technical report (Page et al., 1999).

PageRank

4.3 Cholesky Decomposition

There are many ways to factorize special types of matrices that we encounter often in machine learning. In the positive real numbers, we have the square-root operation that gives us a decomposition of the number into identical components, e.g., $9 = 3 \cdot 3$. For matrices, we need to be careful that we compute a square-root-like operation on positive quantities. For symmetric, positive definite matrices (see Section 3.2.3), we can choose from a number of square-root equivalent operations. The *Cholesky decomposition/Cholesky factorization* provides a square-root equivalent operation on symmetric, positive definite matrices that is useful in practice.

Cholesky decomposition
Cholesky factorization

Theorem 4.18 (Cholesky Decomposition)**.** *A symmetric, positive definite matrix $\boldsymbol{A}$ can be factorized into a product $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^\top$, where $\boldsymbol{L}$ is a lower-triangular matrix with positive diagonal elements:*

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ l_{n1} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & \cdots & l_{n1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & l_{nn} \end{bmatrix}. \tag{4.44}$$

$\boldsymbol{L}$ is called the Cholesky factor of $\boldsymbol{A}$, and $\boldsymbol{L}$ is unique.

Cholesky factor

Example 4.10 (Cholesky Factorization)
Consider a symmetric, positive definite matrix $\boldsymbol{A} \in \mathbb{R}^{3\times 3}$. We are interested in finding its Cholesky factorization $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^\top$, i.e.,

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{21} & a_{22} & a_{32} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \boldsymbol{L}\boldsymbol{L}^\top = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} l_{11} & l_{21} & l_{31} \\ 0 & l_{22} & l_{32} \\ 0 & 0 & l_{33} \end{bmatrix}. \tag{4.45}$$

Multiplying out the right-hand side yields

$$\boldsymbol{A} = \begin{bmatrix} l_{11}^2 & l_{21}l_{11} & l_{31}l_{11} \\ l_{21}l_{11} & l_{21}^2 + l_{22}^2 & l_{31}l_{21} + l_{32}l_{22} \\ l_{31}l_{11} & l_{31}l_{21} + l_{32}l_{22} & l_{31}^2 + l_{32}^2 + l_{33}^2 \end{bmatrix}. \tag{4.46}$$

Comparing the left-hand side of (4.45) and the right-hand side of (4.46) shows that there is a simple pattern in the diagonal elements l_{ii}:

$$l_{11} = \sqrt{a_{11}}\,, \quad l_{22} = \sqrt{a_{22} - l_{21}^2}\,, \quad l_{33} = \sqrt{a_{33} - (l_{31}^2 + l_{32}^2)}\,. \tag{4.47}$$

Similarly for the elements below the diagonal (l_{ij}, where $i > j$), there is also a repeating pattern:

$$l_{21} = \frac{1}{l_{11}} a_{21}\,, \quad l_{31} = \frac{1}{l_{11}} a_{31}\,, \quad l_{32} = \frac{1}{l_{22}}(a_{32} - l_{31} l_{21})\,. \tag{4.48}$$

Thus, we constructed the Cholesky decomposition for any symmetric, positive definite 3×3 matrix. The key realization is that we can backward calculate what the components l_{ij} for the $\boldsymbol{L}$ should be, given the values a_{ij} for $\boldsymbol{A}$ and previously computed values of l_{ij}.

The Cholesky decomposition is an important tool for the numerical computations underlying machine learning. Here, symmetric positive definite matrices require frequent manipulation, e.g., the covariance matrix of a multivariate Gaussian variable (see Section 6.5) is symmetric, positive definite. The Cholesky factorization of this covariance matrix allows us to generate samples from a Gaussian distribution. It also allows us to perform a linear transformation of random variables, which is heavily exploited when computing gradients in deep stochastic models, such as the variational auto-encoder (Jimenez Rezende et al., 2014; Kingma and Welling, 2014). The Cholesky decomposition also allows us to compute determinants very efficiently. Given the Cholesky decomposition $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^\top$, we know that $\det(\boldsymbol{A}) = \det(\boldsymbol{L})\det(\boldsymbol{L}^\top) = \det(\boldsymbol{L})^2$. Since $\boldsymbol{L}$ is a triangular matrix, the determinant is simply the product of its diagonal entries so that $\det(\boldsymbol{A}) = \prod_i l_{ii}^2$. Thus, many numerical software packages use the Cholesky decomposition to make computations more efficient.

4.4 Eigendecomposition and Diagonalization

A *diagonal matrix* is a matrix that has value zero on all off-diagonal elements, i.e., they are of the form diagonal matrix

$$\boldsymbol{D} = \begin{bmatrix} c_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & c_n \end{bmatrix}. \tag{4.49}$$

They allow fast computation of determinants, powers, and inverses. The determinant is the product of its diagonal entries, a matrix power $\boldsymbol{D}^k$ is given by each diagonal element raised to the power k, and the inverse $\boldsymbol{D}^{-1}$ is the reciprocal of its diagonal elements if all of them are nonzero.

In this section, we will discuss how to transform matrices into diagonal

form. This is an important application of the basis change we discussed in Section 2.7.2 and eigenvalues from Section 4.2.

Recall that two matrices $\boldsymbol{A}, \boldsymbol{D}$ are similar (Definition 2.22) if there exists an invertible matrix $\boldsymbol{P}$, such that $\boldsymbol{D} = \boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P}$. More specifically, we will look at matrices $\boldsymbol{A}$ that are similar to diagonal matrices $\boldsymbol{D}$ that contain the eigenvalues of $\boldsymbol{A}$ on the diagonal.

diagonalizable

Definition 4.19 (Diagonalizable)**.** A matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is *diagonalizable* if it is similar to a diagonal matrix, i.e., if there exists an invertible matrix $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ such that $\boldsymbol{D} = \boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P}$.

In the following, we will see that diagonalizing a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is a way of expressing the same linear mapping but in another basis (see Section 2.6.1), which will turn out to be a basis that consists of the eigenvectors of $\boldsymbol{A}$.

Let $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, let $\lambda_1, \dots, \lambda_n$ be a set of scalars, and let $\boldsymbol{p}_1, \dots, \boldsymbol{p}_n$ be a set of vectors in $\mathbb{R}^n$. We define $\boldsymbol{P} := [\boldsymbol{p}_1, \dots, \boldsymbol{p}_n]$ and let $\boldsymbol{D} \in \mathbb{R}^{n\times n}$ be a diagonal matrix with diagonal entries $\lambda_1, \dots, \lambda_n$. Then we can show that

$$\boldsymbol{A}\boldsymbol{P} = \boldsymbol{P}\boldsymbol{D} \tag{4.50}$$

if and only if $\lambda_1, \dots, \lambda_n$ are the eigenvalues of $\boldsymbol{A}$ and $\boldsymbol{p}_1, \dots, \boldsymbol{p}_n$ are corresponding eigenvectors of $\boldsymbol{A}$.

We can see that this statement holds because

$$\boldsymbol{A}\boldsymbol{P} = \boldsymbol{A}[\boldsymbol{p}_1, \dots, \boldsymbol{p}_n] = [\boldsymbol{A}\boldsymbol{p}_1, \dots, \boldsymbol{A}\boldsymbol{p}_n]\,, \tag{4.51}$$

$$\boldsymbol{P}\boldsymbol{D} = [\boldsymbol{p}_1, \dots, \boldsymbol{p}_n]\begin{bmatrix}\lambda_1 & & 0\\ & \ddots & \\ 0 & & \lambda_n\end{bmatrix} = [\lambda_1\boldsymbol{p}_1, \dots, \lambda_n\boldsymbol{p}_n]\,. \tag{4.52}$$

Thus, (4.50) implies that

$$\boldsymbol{A}\boldsymbol{p}_1 = \lambda_1\boldsymbol{p}_1 \tag{4.53}$$

$$\vdots$$

$$\boldsymbol{A}\boldsymbol{p}_n = \lambda_n\boldsymbol{p}_n\,. \tag{4.54}$$

Therefore, the columns of $\boldsymbol{P}$ must be eigenvectors of $\boldsymbol{A}$.

Our definition of diagonalization requires that $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ is invertible, i.e., $\boldsymbol{P}$ has full rank (Theorem 4.3). This requires us to have n linearly independent eigenvectors $\boldsymbol{p}_1, \dots, \boldsymbol{p}_n$, i.e., the $\boldsymbol{p}_i$ form a basis of $\mathbb{R}^n$.

Theorem 4.20 (Eigendecomposition)**.** *A square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ can be factored into*

$$\boldsymbol{A} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^{-1}\,, \tag{4.55}$$

where $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{D}$ is a diagonal matrix whose diagonal entries are the eigenvalues of $\boldsymbol{A}$, if and only if the eigenvectors of $\boldsymbol{A}$ form a basis of $\mathbb{R}^n$.

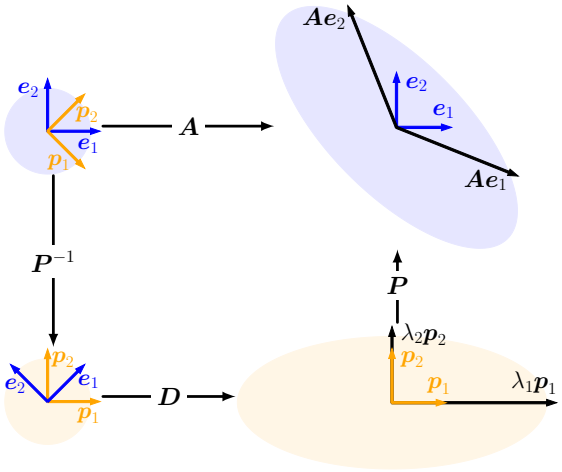


Figure 4.7 Intuition behind the eigendecomposition as sequential transformations. Top-left to bottom-left: $\boldsymbol{P}^{-1}$ performs a basis change (here drawn in $\mathbb{R}^2$ and depicted as a rotation-like operation) from the standard basis into the eigenbasis. Bottom-left to bottom-right: $\boldsymbol{D}$ performs a scaling along the remapped orthogonal eigenvectors, depicted here by a circle being stretched to an ellipse. Bottom-right to top-right: $\boldsymbol{P}$ undoes the basis change (depicted as a reverse rotation) and restores the original coordinate frame.

Theorem 4.20 implies that only non-defective matrices can be diagonalized and that the columns of $\boldsymbol{P}$ are the n eigenvectors of $\boldsymbol{A}$. For symmetric matrices we can obtain even stronger outcomes for the eigenvalue decomposition.

Theorem 4.21. *A symmetric matrix $\boldsymbol{S} \in \mathbb{R}^{n\times n}$ can always be diagonalized.*

Theorem 4.21 follows directly from the spectral theorem 4.15. Moreover, the spectral theorem states that we can find an ONB of eigenvectors of $\mathbb{R}^n$. This makes $\boldsymbol{P}$ an orthogonal matrix so that $\boldsymbol{D} = \boldsymbol{P}^\top \boldsymbol{A}\boldsymbol{P}$.

Remark. The Jordan normal form of a matrix offers a decomposition that works for defective matrices (Lang, 1987) but is beyond the scope of this book. ♢

Geometric Intuition for the Eigendecomposition

We can interpret the eigendecomposition of a matrix as follows (see also Figure 4.7): Let $\boldsymbol{A}$ be the transformation matrix of a linear mapping with respect to the standard basis $\boldsymbol{e}_i$ (blue arrows). $\boldsymbol{P}^{-1}$ performs a basis change from the standard basis into the eigenbasis. Then, the diagonal $\boldsymbol{D}$ scales the vectors along these axes by the eigenvalues λ_i. Finally, $\boldsymbol{P}$ transforms these scaled vectors back into the standard/canonical coordinates yielding $\lambda_i \boldsymbol{p}_i$.

Example 4.11 (Eigendecomposition)

Let us compute the eigendecomposition of $\boldsymbol{A} = \frac{1}{2}\begin{bmatrix} 5 & -2 \\ -2 & 5 \end{bmatrix}$.

Step 1: Compute eigenvalues and eigenvectors. The characteristic

polynomial of $\boldsymbol{A}$ is

$$\det(\boldsymbol{A}-\lambda\boldsymbol{I}) = \det\left(\begin{bmatrix}\frac{5}{2}-\lambda & -1\\ -1 & \frac{5}{2}-\lambda\end{bmatrix}\right) \tag{4.56a}$$

$$= (\tfrac{5}{2}-\lambda)^2 - 1 = \lambda^2 - 5\lambda + \tfrac{21}{4} = (\lambda - \tfrac{7}{2})(\lambda - \tfrac{3}{2})\,. \tag{4.56b}$$

Therefore, the eigenvalues of $\boldsymbol{A}$ are $\lambda_1 = \frac{7}{2}$ and $\lambda_2 = \frac{3}{2}$ (the roots of the characteristic polynomial), and the associated (normalized) eigenvectors are obtained via

$$\boldsymbol{A}\boldsymbol{p}_1 = \frac{7}{2}\boldsymbol{p}_1\,, \quad \boldsymbol{A}\boldsymbol{p}_2 = \frac{3}{2}\boldsymbol{p}_2\,. \tag{4.57}$$

This yields

$$\boldsymbol{p}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}\,, \quad \boldsymbol{p}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}\,. \tag{4.58}$$

Step 2: Check for existence. The eigenvectors $\boldsymbol{p}_1, \boldsymbol{p}_2$ form a basis of $\mathbb{R}^2$. Therefore, $\boldsymbol{A}$ can be diagonalized.
Step 3: Construct the matrix $\boldsymbol{P}$ to diagonalize $\boldsymbol{A}$. We collect the eigenvectors of $\boldsymbol{A}$ in $\boldsymbol{P}$ so that

$$\boldsymbol{P} = [\boldsymbol{p}_1,\ \boldsymbol{p}_2] = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ -1 & 1\end{bmatrix}\,. \tag{4.59}$$

We then obtain

$$\boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{P} = \begin{bmatrix}\frac{7}{2} & 0\\ 0 & \frac{3}{2}\end{bmatrix} = \boldsymbol{D}\,. \tag{4.60}$$

Figure 4.7 visualizes the eigendecomposition of $\boldsymbol{A} = \begin{bmatrix}5 & -2\\ -2 & 5\end{bmatrix}$ as a sequence of linear transformations.

Equivalently, we get (exploiting that $\boldsymbol{P}^{-1} = \boldsymbol{P}^\top$ since the eigenvectors $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ in this example form an ONB)

$$\underbrace{\frac{1}{2}\begin{bmatrix}5 & -2\\ -2 & 5\end{bmatrix}}_{\boldsymbol{A}} = \underbrace{\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ -1 & 1\end{bmatrix}}_{\boldsymbol{P}}\underbrace{\begin{bmatrix}\frac{7}{2} & 0\\ 0 & \frac{3}{2}\end{bmatrix}}_{\boldsymbol{D}}\underbrace{\frac{1}{\sqrt{2}}\begin{bmatrix}1 & -1\\ 1 & 1\end{bmatrix}}_{\boldsymbol{P}^{-1}}\,. \tag{4.61}$$

- Diagonal matrices $\boldsymbol{D}$ can efficiently be raised to a power. Therefore, we can find a matrix power for a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ via the eigenvalue decomposition (if it exists) so that

$$\boldsymbol{A}^k = (\boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^{-1})^k = \boldsymbol{P}\boldsymbol{D}^k\boldsymbol{P}^{-1}\,. \tag{4.62}$$

  Computing $\boldsymbol{D}^k$ is efficient because we apply this operation individually to any diagonal element.
- Assume that the eigendecomposition $\boldsymbol{A} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^{-1}$ exists. Then,

$$\det(\boldsymbol{A}) = \det(\boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^{-1}) = \det(\boldsymbol{P})\det(\boldsymbol{D})\det(\boldsymbol{P}^{-1}) \tag{4.63a}$$

$$= \det(\boldsymbol{D}) = \prod_i d_{ii} \tag{4.63b}$$

allows for an efficient computation of the determinant of $\boldsymbol{A}$.

The eigenvalue decomposition requires square matrices. It would be useful to perform a decomposition on general matrices. In the next section, we introduce a more general matrix decomposition technique, the singular value decomposition.

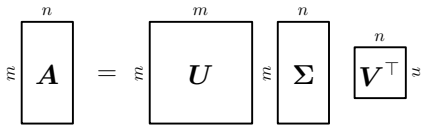
4.5 Singular Value Decomposition

The singular value decomposition (SVD) of a matrix is a central matrix decomposition method in linear algebra. It has been referred to as the "fundamental theorem of linear algebra" (Strang, 1993) because it can be applied to all matrices, not only to square matrices, and it always exists. Moreover, as we will explore in the following, the SVD of a matrix $\boldsymbol{A}$, which represents a linear mapping $\Phi : V \to W$, quantifies the change between the underlying geometry of these two vector spaces. We recommend the work by Kalman (1996) and Roy and Banerjee (2014) for a deeper overview of the mathematics of the SVD.

SVD theorem

Theorem 4.22 (SVD Theorem)**.** *Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ be a rectangular matrix of rank $r \in [0, \min(m, n)]$. The SVD of $\boldsymbol{A}$ is a decomposition of the form*

SVD
singular value decomposition



(4.64)

with an orthogonal matrix $\boldsymbol{U} \in \mathbb{R}^{m \times m}$ with column vectors $\boldsymbol{u}_i$, $i = 1, \ldots, m$, and an orthogonal matrix $\boldsymbol{V} \in \mathbb{R}^{n \times n}$ with column vectors $\boldsymbol{v}_j$, $j = 1, \ldots, n$. Moreover, $\boldsymbol{\Sigma}$ is an $m \times n$ matrix with $\Sigma_{ii} = \sigma_i \geqslant 0$ and $\Sigma_{ij} = 0,\ i \neq j$.

The diagonal entries σ_i, $i = 1, \ldots, r$, of $\boldsymbol{\Sigma}$ are called the *singular values*, $\boldsymbol{u}_i$ are called the *left-singular vectors*, and $\boldsymbol{v}_j$ are called the *right-singular vectors*. By convention, the singular values are ordered, i.e., $\sigma_1 \geqslant \sigma_2 \geqslant \sigma_r \geqslant 0$.

singular values
left-singular vectors
right-singular vectors

The *singular value matrix* $\boldsymbol{\Sigma}$ is unique, but it requires some attention. Observe that the $\boldsymbol{\Sigma} \in \mathbb{R}^{m \times n}$ is rectangular. In particular, $\boldsymbol{\Sigma}$ is of the same size as $\boldsymbol{A}$. This means that $\boldsymbol{\Sigma}$ has a diagonal submatrix that contains the singular values and needs additional zero padding. Specifically, if $m > n$, then the matrix $\boldsymbol{\Sigma}$ has diagonal structure up to row n and then consists of

singular value matrix

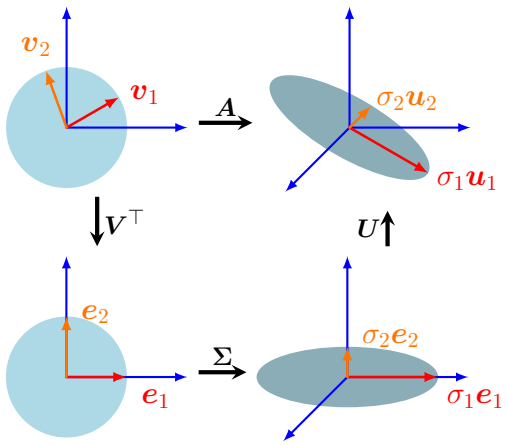


Figure 4.8 Intuition behind the SVD of a matrix $\boldsymbol{A} \in \mathbb{R}^{3\times 2}$ as sequential transformations. Top-left to bottom-left: $\boldsymbol{V}^\top$ performs a basis change in $\mathbb{R}^2$. Bottom-left to bottom-right: $\boldsymbol{\Sigma}$ scales and maps from $\mathbb{R}^2$ to $\mathbb{R}^3$. The ellipse in the bottom-right lives in $\mathbb{R}^3$. The third dimension is orthogonal to the surface of the elliptical disk. Bottom-right to top-right: $\boldsymbol{U}$ performs a basis change within $\mathbb{R}^3$.

$\mathbf{0}^\top$ row vectors from $n+1$ to m below so that

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_n \\ 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \end{bmatrix} . \tag{4.65}$$

If $m < n$, the matrix $\boldsymbol{\Sigma}$ has a diagonal structure up to column m and columns that consist of $\mathbf{0}$ from $m+1$ to n:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 & \dots & 0 \\ 0 & \ddots & 0 & \vdots & & \vdots \\ 0 & 0 & \sigma_m & 0 & \dots & 0 \end{bmatrix} . \tag{4.66}$$

Remark. The SVD exists for any matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$. ♢

4.5.1 Geometric Intuitions for the SVD

The SVD offers geometric intuitions to describe a transformation matrix $\boldsymbol{A}$. In the following, we will discuss the SVD as sequential linear transformations performed on the bases. In Example 4.12, we will then apply transformation matrices of the SVD to a set of vectors in $\mathbb{R}^2$, which allows us to visualize the effect of each transformation more clearly.

The SVD of a matrix can be interpreted as a decomposition of a corresponding linear mapping (recall Section 2.7.1) $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ into three operations; see Figure 4.8. The SVD intuition follows superficially a similar structure to our eigendecomposition intuition, see Figure 4.7: Broadly speaking, the SVD performs a basis change via $\boldsymbol{V}^\top$ followed by a scaling and augmentation (or reduction) in dimensionality via the singular

value matrix $\mathbf{\Sigma}$. Finally, it performs a second basis change via $\boldsymbol{U}$. The SVD entails a number of important details and caveats, which is why we will review our intuition in more detail.

It is useful to review basis changes (Section 2.7.2), orthogonal matrices (Definition 3.8) and orthonormal bases (Section 3.5).

Assume we are given a transformation matrix of a linear mapping $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ with respect to the standard bases B and C of $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively. Moreover, assume a second basis $\tilde{B}$ of $\mathbb{R}^n$ and $\tilde{C}$ of $\mathbb{R}^m$. Then

1. The matrix $\boldsymbol{V}$ performs a basis change in the domain $\mathbb{R}^n$ from $\tilde{B}$ (represented by the red and orange vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ in the top-left of Figure 4.8) to the standard basis B. $\boldsymbol{V}^\top = \boldsymbol{V}^{-1}$ performs a basis change from B to $\tilde{B}$. The red and orange vectors are now aligned with the canonical basis in the bottom-left of Figure 4.8.
2. Having changed the coordinate system to $\tilde{B}$, $\mathbf{\Sigma}$ scales the new coordinates by the singular values σ_i (and adds or deletes dimensions), i.e., $\mathbf{\Sigma}$ is the transformation matrix of Φ with respect to $\tilde{B}$ and $\tilde{C}$, represented by the red and orange vectors being stretched and lying in the $\boldsymbol{e}_1$-$\boldsymbol{e}_2$ plane, which is now embedded in a third dimension in the bottom-right of Figure 4.8.
3. $\boldsymbol{U}$ performs a basis change in the codomain $\mathbb{R}^m$ from $\tilde{C}$ into the canonical basis of $\mathbb{R}^m$, represented by a rotation of the red and orange vectors out of the $\boldsymbol{e}_1$-$\boldsymbol{e}_2$ plane. This is shown in the top-right of Figure 4.8.

The SVD expresses a change of basis in both the domain and codomain. This is in contrast with the eigendecomposition that operates within the same vector space, where the same basis change is applied and then undone. What makes the SVD special is that these two different bases are simultaneously linked by the singular value matrix $\mathbf{\Sigma}$.

Example 4.12 (Vectors and the SVD)
Consider a mapping of a square grid of vectors $\mathcal{X} \in \mathbb{R}^2$ that fit in a box of size 2×2 centered at the origin. Using the standard basis, we map these vectors using

$$\boldsymbol{A} = \begin{bmatrix} 1 & -0.8 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} = \boldsymbol{U}\mathbf{\Sigma}\boldsymbol{V}^\top \tag{4.67a}$$

$$= \begin{bmatrix} -0.79 & 0 & -0.62 \\ 0.38 & -0.78 & -0.49 \\ -0.48 & -0.62 & 0.62 \end{bmatrix} \begin{bmatrix} 1.62 & 0 \\ 0 & 1.0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -0.78 & 0.62 \\ -0.62 & -0.78 \end{bmatrix}. \tag{4.67b}$$

We start with a set of vectors $\mathcal{X}$ (colored dots; see top-left panel of Figure 4.9) arranged in a grid. We then apply $\boldsymbol{V}^\top \in \mathbb{R}^{2\times 2}$, which rotates $\mathcal{X}$. The rotated vectors are shown in the bottom-left panel of Figure 4.9. We now map these vectors using the singular value matrix $\mathbf{\Sigma}$ to the codomain $\mathbb{R}^3$ (see the bottom-right panel in Figure 4.9). Note that all vectors lie in

the x_1-x_2 plane. The third coordinate is always 0. The vectors in the x_1-x_2 plane have been stretched by the singular values.

The direct mapping of the vectors $\mathcal{X}$ by $\boldsymbol{A}$ to the codomain $\mathbb{R}^3$ equals the transformation of $\mathcal{X}$ by $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, where $\boldsymbol{U}$ performs a rotation within the codomain $\mathbb{R}^3$ so that the mapped vectors are no longer restricted to the x_1-x_2 plane; they still are on a plane as shown in the top-right panel of Figure 4.9.

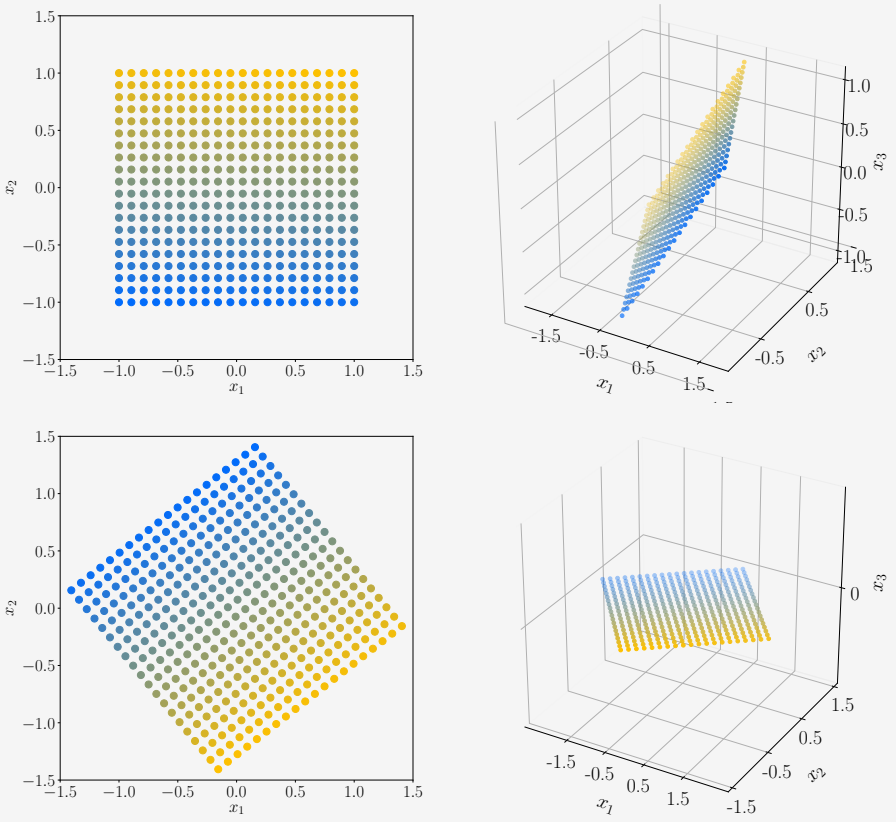


Figure 4.9 SVD and mapping of vectors (represented by discs). The panels follow the same anti-clockwise structure of Figure 4.8.

4.5.2 Construction of the SVD

We will next discuss why the SVD exists and show how to compute it in detail. The SVD of a general matrix shares some similarities with the eigendecomposition of a square matrix.

Remark. Compare the eigendecomposition of an SPD matrix

$$\boldsymbol{S} = \boldsymbol{S}^\top = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^\top \tag{4.68}$$

with the corresponding SVD

$$\boldsymbol{S} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top . \tag{4.69}$$

If we set

$$\boldsymbol{U} = \boldsymbol{P} = \boldsymbol{V} , \quad \boldsymbol{D} = \boldsymbol{\Sigma} , \tag{4.70}$$

we see that the SVD of SPD matrices is their eigendecomposition. ♢

In the following, we will explore why Theorem 4.22 holds and how the SVD is constructed. Computing the SVD of $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is equivalent to finding two sets of orthonormal bases $U = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m)$ and $V = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$ of the codomain $\mathbb{R}^m$ and the domain $\mathbb{R}^n$, respectively. From these ordered bases, we will construct the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$.

Our plan is to start with constructing the orthonormal set of right-singular vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n \in \mathbb{R}^n$. We then construct the orthonormal set of left-singular vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m \in \mathbb{R}^m$. Thereafter, we will link the two and require that the orthogonality of the $\boldsymbol{v}_i$ is preserved under the transformation of $\boldsymbol{A}$. This is important because we know that the images $\boldsymbol{A}\boldsymbol{v}_i$ form a set of orthogonal vectors. We will then normalize these images by scalar factors, which will turn out to be the singular values.

Let us begin with constructing the right-singular vectors. The spectral theorem (Theorem 4.15) tells us that the eigenvectors of a symmetric matrix form an ONB, which also means it can be diagonalized. Moreover, from Theorem 4.14 we can always construct a symmetric, positive semidefinite matrix $\boldsymbol{A}^\top\boldsymbol{A} \in \mathbb{R}^{n\times n}$ from any rectangular matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$. Thus, we can always diagonalize $\boldsymbol{A}^\top\boldsymbol{A}$ and obtain

$$\boldsymbol{A}^\top\boldsymbol{A} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^\top = \boldsymbol{P}\begin{bmatrix}\lambda_1 & \cdots & 0\\ \vdots & \ddots & \vdots\\ 0 & \cdots & \lambda_n\end{bmatrix}\boldsymbol{P}^\top , \tag{4.71}$$

where $\boldsymbol{P}$ is an orthogonal matrix, which is composed of the orthonormal eigenbasis. The $\lambda_i \geqslant 0$ are the eigenvalues of $\boldsymbol{A}^\top\boldsymbol{A}$. Let us assume the SVD of $\boldsymbol{A}$ exists and inject (4.64) into (4.71). This yields

$$\boldsymbol{A}^\top\boldsymbol{A} = (\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top)^\top(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top) = \boldsymbol{V}\boldsymbol{\Sigma}^\top\boldsymbol{U}^\top\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top , \tag{4.72}$$

where $\boldsymbol{U}, \boldsymbol{V}$ are orthogonal matrices. Therefore, with $\boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}$ we obtain

$$\boldsymbol{A}^\top\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Sigma}^\top\boldsymbol{\Sigma}\boldsymbol{V}^\top = \boldsymbol{V}\begin{bmatrix}\sigma_1^2 & 0 & 0\\ 0 & \ddots & 0\\ 0 & 0 & \sigma_n^2\end{bmatrix}\boldsymbol{V}^\top . \tag{4.73}$$

Comparing now (4.71) and (4.73), we identify

$$\boldsymbol{V}^\top = \boldsymbol{P}^\top , \tag{4.74}$$

$$\sigma_i^2 = \lambda_i . \tag{4.75}$$

Therefore, the eigenvectors of $\boldsymbol{A}^\top\boldsymbol{A}$ that compose $\boldsymbol{P}$ are the right-singular vectors $\boldsymbol{V}$ of $\boldsymbol{A}$ (see (4.74)). The eigenvalues of $\boldsymbol{A}^\top\boldsymbol{A}$ are the squared singular values of $\boldsymbol{\Sigma}$ (see (4.75)).

To obtain the left-singular vectors $\boldsymbol{U}$, we follow a similar procedure. We start by computing the SVD of the symmetric matrix $\boldsymbol{A}\boldsymbol{A}^\top \in \mathbb{R}^{m\times m}$ (instead of the previous $\boldsymbol{A}^\top\boldsymbol{A} \in \mathbb{R}^{n\times n}$). The SVD of $\boldsymbol{A}$ yields

$$\boldsymbol{A}\boldsymbol{A}^\top = (\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top)(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top)^\top = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top\boldsymbol{V}\boldsymbol{\Sigma}^\top\boldsymbol{U}^\top \tag{4.76a}$$

$$= \boldsymbol{U}\begin{bmatrix}\sigma_1^2 & 0 & 0\\ 0 & \ddots & 0\\ 0 & 0 & \sigma_m^2\end{bmatrix}\boldsymbol{U}^\top . \tag{4.76b}$$

The spectral theorem tells us that $\boldsymbol{A}\boldsymbol{A}^\top = \boldsymbol{S}\boldsymbol{D}\boldsymbol{S}^\top$ can be diagonalized and we can find an ONB of eigenvectors of $\boldsymbol{A}\boldsymbol{A}^\top$, which are collected in $\boldsymbol{S}$. The orthonormal eigenvectors of $\boldsymbol{A}\boldsymbol{A}^\top$ are the left-singular vectors $\boldsymbol{U}$ and form an orthonormal basis in the codomain of the SVD.

This leaves the question of the structure of the matrix $\boldsymbol{\Sigma}$. Since $\boldsymbol{A}\boldsymbol{A}^\top$ and $\boldsymbol{A}^\top\boldsymbol{A}$ have the same nonzero eigenvalues (see page 106), the nonzero entries of the $\boldsymbol{\Sigma}$ matrices in the SVD for both cases have to be the same.

The last step is to link up all the parts we touched upon so far. We have an orthonormal set of right-singular vectors in $\boldsymbol{V}$. To finish the construction of the SVD, we connect them with the orthonormal vectors $\boldsymbol{U}$. To reach this goal, we use the fact the images of the $\boldsymbol{v}_i$ under $\boldsymbol{A}$ have to be orthogonal, too. We can show this by using the results from Section 3.4. We require that the inner product between $\boldsymbol{A}\boldsymbol{v}_i$ and $\boldsymbol{A}\boldsymbol{v}_j$ must be 0 for $i \neq j$. For any two orthogonal eigenvectors $\boldsymbol{v}_i, \boldsymbol{v}_j$, $i \neq j$, it holds that

$$(\boldsymbol{A}\boldsymbol{v}_i)^\top(\boldsymbol{A}\boldsymbol{v}_j) = \boldsymbol{v}_i^\top(\boldsymbol{A}^\top\boldsymbol{A})\boldsymbol{v}_j = \boldsymbol{v}_i^\top(\lambda_j\boldsymbol{v}_j) = \lambda_j\boldsymbol{v}_i^\top\boldsymbol{v}_j = 0 . \tag{4.77}$$

For the case $m \geqslant r$, it holds that $\{\boldsymbol{A}\boldsymbol{v}_1, \ldots, \boldsymbol{A}\boldsymbol{v}_r\}$ is a basis of an r-dimensional subspace of $\mathbb{R}^m$.

To complete the SVD construction, we need left-singular vectors that are ortho*normal*: We normalize the images of the right-singular vectors $\boldsymbol{A}\boldsymbol{v}_i$ and obtain

$$\boldsymbol{u}_i := \frac{\boldsymbol{A}\boldsymbol{v}_i}{\|\boldsymbol{A}\boldsymbol{v}_i\|} = \frac{1}{\sqrt{\lambda_i}}\boldsymbol{A}\boldsymbol{v}_i = \frac{1}{\sigma_i}\boldsymbol{A}\boldsymbol{v}_i , \tag{4.78}$$

where the last equality was obtained from (4.75) and (4.76b), showing us that the eigenvalues of $\boldsymbol{A}\boldsymbol{A}^\top$ are such that $\sigma_i^2 = \lambda_i$.

Therefore, the eigenvectors of $\boldsymbol{A}^\top\boldsymbol{A}$, which we know are the right-singular vectors $\boldsymbol{v}_i$, and their normalized images under $\boldsymbol{A}$, the left-singular vectors $\boldsymbol{u}_i$, form two self-consistent ONBs that are connected through the singular value matrix $\boldsymbol{\Sigma}$.

singular value equation

Let us rearrange (4.78) to obtain the *singular value equation*

$$\boldsymbol{A}\boldsymbol{v}_i = \sigma_i\boldsymbol{u}_i , \quad i = 1, \ldots, r . \tag{4.79}$$

This equation closely resembles the eigenvalue equation (4.25), but the vectors on the left- and the right-hand sides are not the same.

For $n < m$, (4.79) holds only for $i \leqslant n$, but (4.79) says nothing about the $\boldsymbol{u}_i$ for $i > n$. However, we know by construction that they are orthonormal. Conversely, for $m < n$, (4.79) holds only for $i \leqslant m$. For $i > m$, we have $\boldsymbol{A}\boldsymbol{v}_i = \boldsymbol{0}$ and we still know that the $\boldsymbol{v}_i$ form an orthonormal set. This means that the SVD also supplies an orthonormal basis of the kernel (null space) of $\boldsymbol{A}$, the set of vectors $\boldsymbol{x}$ with $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$ (see Section 2.7.3).

Concatenating the $\boldsymbol{v}_i$ as the columns of $\boldsymbol{V}$ and the $\boldsymbol{u}_i$ as the columns of $\boldsymbol{U}$ yields

$$\boldsymbol{A}\boldsymbol{V} = \boldsymbol{U}\boldsymbol{\Sigma}\,, \tag{4.80}$$

where $\boldsymbol{\Sigma}$ has the same dimensions as $\boldsymbol{A}$ and a diagonal structure for rows $1, \ldots, r$. Hence, right-multiplying with $\boldsymbol{V}^\top$ yields $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, which is the SVD of $\boldsymbol{A}$.

Example 4.13 (Computing the SVD)
Let us find the singular value decomposition of

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix}. \tag{4.81}$$

The SVD requires us to compute the right-singular vectors $\boldsymbol{v}_j$, the singular values σ_k, and the left-singular vectors $\boldsymbol{u}_i$.

Step 1: Right-singular vectors as the eigenbasis of $\boldsymbol{A}^\top\boldsymbol{A}$.
We start by computing

$$\boldsymbol{A}^\top\boldsymbol{A} = \begin{bmatrix} 1 & -2 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 5 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \tag{4.82}$$

We compute the singular values and right-singular vectors $\boldsymbol{v}_j$ through the eigenvalue decomposition of $\boldsymbol{A}^\top\boldsymbol{A}$, which is given as

$$\boldsymbol{A}^\top\boldsymbol{A} = \begin{bmatrix} \frac{5}{\sqrt{30}} & 0 & \frac{-1}{\sqrt{6}} \\ \frac{-2}{\sqrt{30}} & \frac{1}{\sqrt{5}} & \frac{-2}{\sqrt{6}} \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{6}} \end{bmatrix} \begin{bmatrix} 6 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{5}{\sqrt{30}} & \frac{-2}{\sqrt{30}} & \frac{1}{\sqrt{30}} \\ 0 & \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{-1}{\sqrt{6}} & \frac{-2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \end{bmatrix} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^\top, \tag{4.83}$$

and we obtain the right-singular vectors as the columns of $\boldsymbol{P}$ so that

$$\boldsymbol{V} = \boldsymbol{P} = \begin{bmatrix} \frac{5}{\sqrt{30}} & 0 & \frac{-1}{\sqrt{6}} \\ \frac{-2}{\sqrt{30}} & \frac{1}{\sqrt{5}} & \frac{-2}{\sqrt{6}} \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{6}} \end{bmatrix}. \tag{4.84}$$

Step 2: Singular-value matrix.
As the singular values σ_i are the square roots of the eigenvalues of

$\boldsymbol{A}^\top\boldsymbol{A}$ we obtain them straight from $\boldsymbol{D}$. Since $\mathrm{rk}(\boldsymbol{A}) = 2$, there are only two nonzero singular values: $\sigma_1 = \sqrt{6}$ and $\sigma_2 = 1$. The singular value matrix must be the same size as $\boldsymbol{A}$, and we obtain

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sqrt{6} & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \tag{4.85}$$

Step 3: Left-singular vectors as the normalized image of the right-singular vectors.

We find the left-singular vectors by computing the image of the right-singular vectors under $\boldsymbol{A}$ and normalizing them by dividing them by their corresponding singular value. We obtain

$$\boldsymbol{u}_1 = \frac{1}{\sigma_1}\boldsymbol{A}\boldsymbol{v}_1 = \frac{1}{\sqrt{6}}\begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix}\begin{bmatrix} \frac{5}{\sqrt{30}} \\ \frac{-2}{\sqrt{30}} \\ \frac{1}{\sqrt{30}} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{5}} \\ -\frac{2}{\sqrt{5}} \end{bmatrix}, \tag{4.86}$$

$$\boldsymbol{u}_2 = \frac{1}{\sigma_2}\boldsymbol{A}\boldsymbol{v}_2 = \frac{1}{1}\begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ \frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{bmatrix} = \begin{bmatrix} \frac{2}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{bmatrix}, \tag{4.87}$$

$$\boldsymbol{U} = [\boldsymbol{u}_1, \boldsymbol{u}_2] = \frac{1}{\sqrt{5}}\begin{bmatrix} 1 & 2 \\ -2 & 1 \end{bmatrix}. \tag{4.88}$$

Note that on a computer the approach illustrated here has poor numerical behavior, and the SVD of $\boldsymbol{A}$ is normally computed without resorting to the eigenvalue decomposition of $\boldsymbol{A}^\top\boldsymbol{A}$.

4.5.3 Eigenvalue Decomposition vs. Singular Value Decomposition

Let us consider the eigendecomposition $\boldsymbol{A} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^{-1}$ and the SVD $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ and review the core elements of the past sections.

- The SVD always exists for any matrix $\mathbb{R}^{m\times n}$. The eigendecomposition is only defined for square matrices $\mathbb{R}^{n\times n}$ and only exists if we can find a basis of eigenvectors of $\mathbb{R}^n$.
- The vectors in the eigendecomposition matrix $\boldsymbol{P}$ are not necessarily orthogonal, i.e., the change of basis is not a simple rotation and scaling. On the other hand, the vectors in the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ in the SVD are orthonormal, so they do represent rotations.
- Both the eigendecomposition and the SVD are compositions of three linear mappings:
  1. Change of basis in the domain
  2. Independent scaling of each new basis vector and mapping from domain to codomain
  3. Change of basis in the codomain

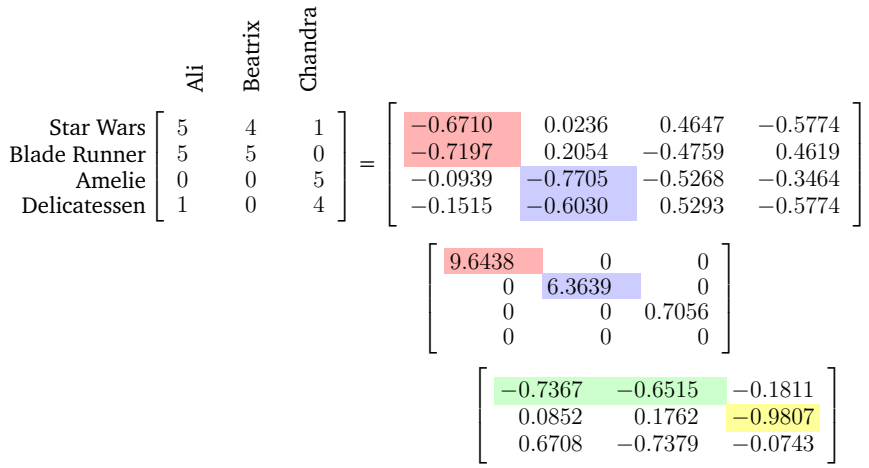


Figure 4.10 Movie ratings of three people for four movies and its SVD decomposition.

A key difference between the eigendecomposition and the SVD is that in the SVD, domain and codomain can be vector spaces of different dimensions.

- In the SVD, the left- and right-singular vector matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are generally not inverse of each other (they perform basis changes in different vector spaces). In the eigendecomposition, the basis change matrices $\boldsymbol{P}$ and $\boldsymbol{P}^{-1}$ are inverses of each other.
- In the SVD, the entries in the diagonal matrix $\boldsymbol{\Sigma}$ are all real and nonnegative, which is not generally true for the diagonal matrix in the eigendecomposition.
- The SVD and the eigendecomposition are closely related through their projections
  - The left-singular vectors of $\boldsymbol{A}$ are eigenvectors of $\boldsymbol{A}\boldsymbol{A}^\top$
  - The right-singular vectors of $\boldsymbol{A}$ are eigenvectors of $\boldsymbol{A}^\top\boldsymbol{A}$.
  - The nonzero singular values of $\boldsymbol{A}$ are the square roots of the nonzero eigenvalues of both $\boldsymbol{A}\boldsymbol{A}^\top$ and $\boldsymbol{A}^\top\boldsymbol{A}$.
- For symmetric matrices $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, the eigenvalue decomposition and the SVD are one and the same, which follows from the spectral theorem 4.15.

Example 4.14 (Finding Structure in Movie Ratings and Consumers)
Let us add a practical interpretation of the SVD by analyzing data on people and their preferred movies. Consider three viewers (Ali, Beatrix, Chandra) rating four different movies (*Star Wars*, *Blade Runner*, *Amelie*, *Delicatessen*). Their ratings are values between 0 (worst) and 5 (best) and encoded in a data matrix $\boldsymbol{A} \in \mathbb{R}^{4\times 3}$ as shown in Figure 4.10. Each row represents a movie and each column a user. Thus, the column vectors of movie ratings, one for each viewer, are $\boldsymbol{x}_{\text{Ali}}$, $\boldsymbol{x}_{\text{Beatrix}}$, $\boldsymbol{x}_{\text{Chandra}}$.

Factoring $\boldsymbol{A}$ using the SVD offers us a way to capture the relationships of how people rate movies, and especially if there is a structure linking which people like which movies. Applying the SVD to our data matrix $\boldsymbol{A}$ makes a number of assumptions:

1. All viewers rate movies consistently using the same linear mapping.
2. There are no errors or noise in the ratings.
3. We interpret the left-singular vectors $\boldsymbol{u}_i$ as stereotypical movies and the right-singular vectors $\boldsymbol{v}_j$ as stereotypical viewers.

We then make the assumption that any viewer's specific movie preferences can be expressed as a linear combination of the $\boldsymbol{v}_j$. Similarly, any movie's like-ability can be expressed as a linear combination of the $\boldsymbol{u}_i$. Therefore, a vector in the domain of the SVD can be interpreted as a viewer in the "space" of stereotypical viewers, and a vector in the codomain of the SVD correspondingly as a movie in the "space" of stereotypical movies. Let us inspect the SVD of our movie-user matrix. The first left-singular vector $\boldsymbol{u}_1$ has large absolute values for the two science fiction movies and a large first singular value (red shading in Figure 4.10). Thus, this groups a type of users with a specific set of movies (science fiction theme). Similarly, the first right-singular $\boldsymbol{v}_1$ shows large absolute values for Ali and Beatrix, who give high ratings to science fiction movies (green shading in Figure 4.10). This suggests that $\boldsymbol{v}_1$ reflects the notion of a science fiction lover.

These two "spaces" are only meaningfully spanned by the respective viewer and movie data if the data itself covers a sufficient diversity of viewers and movies.

Similarly, $\boldsymbol{u}_2$, seems to capture a French art house film theme, and $\boldsymbol{v}_2$ indicates that Chandra is close to an idealized lover of such movies. An idealized science fiction lover is a purist and only loves science fiction movies, so a science fiction lover $\boldsymbol{v}_1$ gives a rating of zero to everything but science fiction themed—this logic is implied by the diagonal substructure for the singular value matrix $\boldsymbol{\Sigma}$. A specific movie is therefore represented by how it decomposes (linearly) into its stereotypical movies. Likewise, a person would be represented by how they decompose (via linear combination) into movie themes.

It is worth to briefly discuss SVD terminology and conventions, as there are different versions used in the literature. While these differences can be confusing, the mathematics remains invariant to them.

- For convenience in notation and abstraction, we use an SVD notation where the SVD is described as having two square left- and right-singular vector matrices, but a non-square singular value matrix. Our definition (4.64) for the SVD is sometimes called the *full SVD*.

full SVD

- Some authors define the SVD a bit differently and focus on square singular matrices. Then, for $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and $m \geqslant n$,

$$\underset{m \times n}{\boldsymbol{A}} = \underset{m \times n}{\boldsymbol{U}} \; \underset{n \times n}{\boldsymbol{\Sigma}} \; \underset{n \times n}{\boldsymbol{V}^\top} . \tag{4.89}$$

Sometimes this formulation is called the *reduced SVD* (e.g., Datta (2010)) or *the* SVD (e.g., Press et al. (2007)). This alternative format changes merely how the matrices are constructed but leaves the mathematical structure of the SVD unchanged. The convenience of this alternative formulation is that $\boldsymbol{\Sigma}$ is diagonal, as in the eigenvalue decomposition. reduced SVD

- In Section 4.6, we will learn about matrix approximation techniques using the SVD, which is also called the *truncated SVD*. truncated SVD
- It is possible to define the SVD of a rank-r matrix $\boldsymbol{A}$ so that $\boldsymbol{U}$ is an $m \times r$ matrix, $\boldsymbol{\Sigma}$ a diagonal matrix $r \times r$, and $\boldsymbol{V}$ an $r \times n$ matrix. This construction is very similar to our definition, and ensures that the diagonal matrix $\boldsymbol{\Sigma}$ has only nonzero entries along the diagonal. The main convenience of this alternative notation is that $\boldsymbol{\Sigma}$ is diagonal, as in the eigenvalue decomposition.
- A restriction that the SVD for $\boldsymbol{A}$ only applies to $m \times n$ matrices with $m > n$ is practically unnecessary. When $m < n$, the SVD decomposition will yield $\boldsymbol{\Sigma}$ with more zero columns than rows and, consequently, the singular values $\sigma_{m+1}, \dots, \sigma_n$ are 0.

The SVD is used in a variety of applications in machine learning from least-squares problems in curve fitting to solving systems of linear equations. These applications harness various important properties of the SVD, its relation to the rank of a matrix, and its ability to approximate matrices of a given rank with lower-rank matrices. Substituting a matrix with its SVD has often the advantage of making calculation more robust to numerical rounding errors. As we will explore in the next section, the SVD's ability to approximate matrices with "simpler" matrices in a principled manner opens up machine learning applications ranging from dimensionality reduction and topic modeling to data compression and clustering.

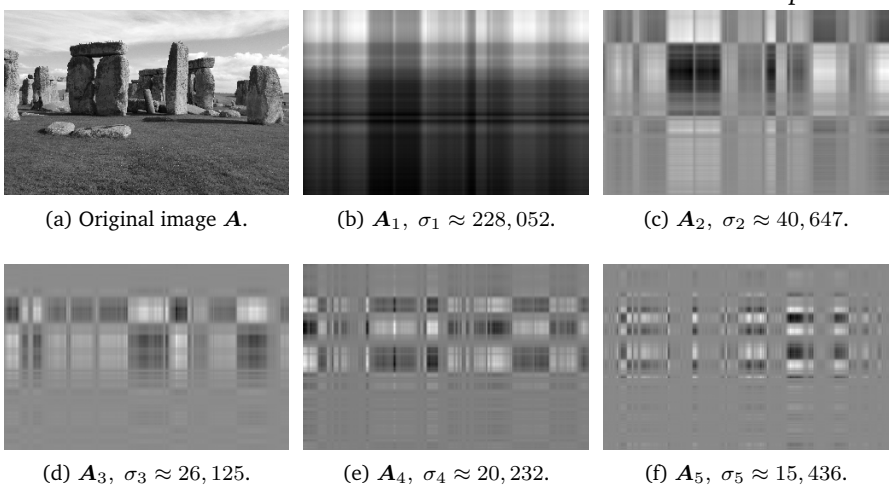
4.6 Matrix Approximation

We considered the SVD as a way to factorize $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top \in \mathbb{R}^{m\times n}$ into the product of three matrices, where $\boldsymbol{U} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{V} \in \mathbb{R}^{n\times n}$ are orthogonal and $\boldsymbol{\Sigma}$ contains the singular values on its main diagonal. Instead of doing the full SVD factorization, we will now investigate how the SVD allows us to represent a matrix $\boldsymbol{A}$ as a sum of simpler (low-rank) matrices $\boldsymbol{A}_i$, which lends itself to a matrix approximation scheme that is cheaper to compute than the full SVD.

We construct a rank-1 matrix $\boldsymbol{A}_i \in \mathbb{R}^{m\times n}$ as

$$\boldsymbol{A}_i := \boldsymbol{u}_i \boldsymbol{v}_i^\top , \tag{4.90}$$

which is formed by the outer product of the ith orthogonal column vector of $\boldsymbol{U}$ and $\boldsymbol{V}$. Figure 4.11 shows an image of Stonehenge, which can be represented by a matrix $\boldsymbol{A} \in \mathbb{R}^{1432\times 1910}$, and some outer products $\boldsymbol{A}_i$, as defined in (4.90).



(a) Original image $\boldsymbol{A}$. (b) $\boldsymbol{A}_1,\ \sigma_1 \approx 228,052$. (c) $\boldsymbol{A}_2,\ \sigma_2 \approx 40,647$.

(d) $\boldsymbol{A}_3,\ \sigma_3 \approx 26,125$. (e) $\boldsymbol{A}_4,\ \sigma_4 \approx 20,232$. (f) $\boldsymbol{A}_5,\ \sigma_5 \approx 15,436$.

Figure 4.11 Image processing with the SVD. (a) The original grayscale image is a $1,432 \times 1,910$ matrix of values between 0 (black) and 1 (white). (b)–(f) Rank-1 matrices $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_5$ and their corresponding singular values $\sigma_1, \ldots, \sigma_5$. The grid-like structure of each rank-1 matrix is imposed by the outer-product of the left and right-singular vectors.

A matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ of rank r can be written as a sum of rank-1 matrices $\boldsymbol{A}_i$ so that

$$\boldsymbol{A} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^\top = \sum_{i=1}^{r} \sigma_i \boldsymbol{A}_i \,, \tag{4.91}$$

where the outer-product matrices $\boldsymbol{A}_i$ are weighted by the ith singular value σ_i. We can see why (4.91) holds: The diagonal structure of the singular value matrix $\boldsymbol{\Sigma}$ multiplies only matching left- and right-singular vectors $\boldsymbol{u}_i \boldsymbol{v}_i^\top$ and scales them by the corresponding singular value σ_i. All terms $\Sigma_{ij} \boldsymbol{u}_i \boldsymbol{v}_j^\top$ vanish for $i \neq j$ because $\boldsymbol{\Sigma}$ is a diagonal matrix. Any terms $i > r$ vanish because the corresponding singular values are 0.

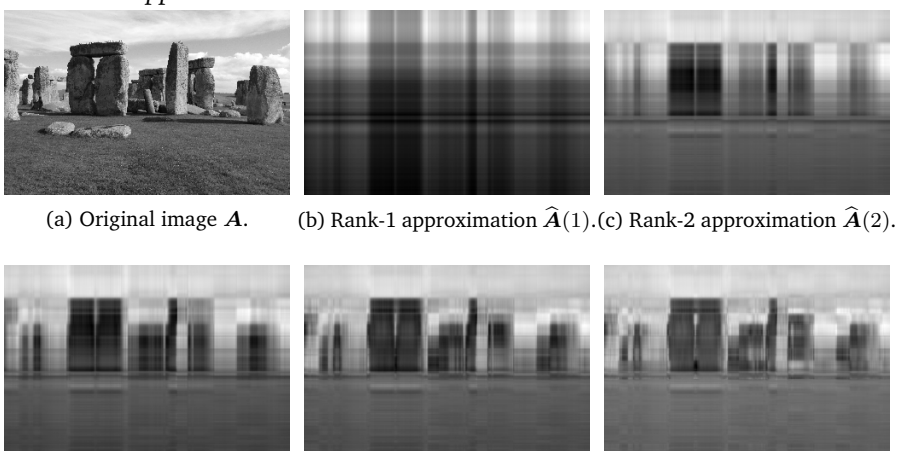
In (4.90), we introduced rank-1 matrices $\boldsymbol{A}_i$. We summed up the r individual rank-1 matrices to obtain a rank-r matrix $\boldsymbol{A}$; see (4.91). If the sum does not run over all matrices $\boldsymbol{A}_i$, $i = 1, \ldots, r$, but only up to an intermediate value $k < r$, we obtain a *rank-k approximation*

rank-k approximation

$$\widehat{\boldsymbol{A}}(k) := \sum_{i=1}^{k} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^\top = \sum_{i=1}^{k} \sigma_i \boldsymbol{A}_i \tag{4.92}$$

of $\boldsymbol{A}$ with $\mathrm{rk}(\widehat{\boldsymbol{A}}(k)) = k$. Figure 4.12 shows low-rank approximations $\widehat{\boldsymbol{A}}(k)$ of an original image $\boldsymbol{A}$ of Stonehenge. The shape of the rocks becomes increasingly visible and clearly recognizable in the rank-5 approximation. While the original image requires $1,432 \cdot 1,910 = 2,735,120$ numbers, the rank-5 approximation requires us only to store the five singular values and the five left- and right-singular vectors ($1,432$ and $1,910$-dimensional each) for a total of $5 \cdot (1,432 + 1,910 + 1) = 16,715$ numbers – just above 0.6% of the original.

To measure the difference (error) between $\boldsymbol{A}$ and its rank-k approximation $\widehat{\boldsymbol{A}}(k)$, we need the notion of a norm. In Section 3.1, we already used



(a) Original image $\boldsymbol{A}$. (b) Rank-1 approximation $\widehat{\boldsymbol{A}}(1)$. (c) Rank-2 approximation $\widehat{\boldsymbol{A}}(2)$.

(d) Rank-3 approximation $\widehat{\boldsymbol{A}}(3)$. (e) Rank-4 approximation $\widehat{\boldsymbol{A}}(4)$. (f) Rank-5 approximation $\widehat{\boldsymbol{A}}(5)$.

Figure 4.12 Image reconstruction with the SVD. (a) Original image. (b)–(f) Image reconstruction using the low-rank approximation of the SVD, where the rank-k approximation is given by $\widehat{\boldsymbol{A}}(k) = \sum_{i=1}^{k} \sigma_i \boldsymbol{A}_i$.

norms on vectors that measure the length of a vector. By analogy we can also define norms on matrices.

spectral norm

Definition 4.23 (Spectral Norm of a Matrix)**.** For $\boldsymbol{x} \in \mathbb{R}^n \backslash \{\boldsymbol{0}\}$, the *spectral norm* of a matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is defined as

$$\|\boldsymbol{A}\|_2 := \max_{\boldsymbol{x}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2} \,. \tag{4.93}$$

We introduce the notation of a subscript in the matrix norm (left-hand side), similar to the Euclidean norm for vectors (right-hand side), which has subscript 2. The spectral norm (4.93) determines how long any vector $\boldsymbol{x}$ can at most become when multiplied by $\boldsymbol{A}$.

Theorem 4.24. *The spectral norm of $\boldsymbol{A}$ is its largest singular value σ_1.*

We leave the proof of this theorem as an exercise.

Eckart-Young theorem

Theorem 4.25 (Eckart-Young Theorem (Eckart and Young, 1936))**.** *Consider a matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ of rank r and let $\boldsymbol{B} \in \mathbb{R}^{m \times n}$ be a matrix of rank k. For any $k \leqslant r$ with $\widehat{\boldsymbol{A}}(k) = \sum_{i=1}^{k} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^\top$ it holds that*

$$\widehat{\boldsymbol{A}}(k) = \operatorname{argmin}_{\mathrm{rk}(\boldsymbol{B})=k} \|\boldsymbol{A} - \boldsymbol{B}\|_2 \,, \tag{4.94}$$

$$\left\|\boldsymbol{A} - \widehat{\boldsymbol{A}}(k)\right\|_2 = \sigma_{k+1} \,. \tag{4.95}$$

The Eckart-Young theorem states explicitly how much error we introduce by approximating $\boldsymbol{A}$ using a rank-k approximation. We can interpret the rank-k approximation obtained with the SVD as a projection of the full-rank matrix $\boldsymbol{A}$ onto a lower-dimensional space of rank-at-most-k matrices. Of all possible projections, the SVD minimizes the error (with respect to the spectral norm) between $\boldsymbol{A}$ and any rank-k approximation.

We can retrace some of the steps to understand why (4.95) should hold.

We observe that the difference between $\boldsymbol{A} - \widehat{\boldsymbol{A}}(k)$ is a matrix containing the sum of the remaining rank-1 matrices

$$\boldsymbol{A} - \widehat{\boldsymbol{A}}(k) = \sum_{i=k+1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^\top \,. \tag{4.96}$$

By Theorem 4.24, we immediately obtain σ_{k+1} as the spectral norm of the difference matrix. Let us have a closer look at (4.94). If we assume that there is another matrix $\boldsymbol{B}$ with $\mathrm{rk}(\boldsymbol{B}) \leqslant k$, such that

$$\|\boldsymbol{A} - \boldsymbol{B}\|_2 < \left\|\boldsymbol{A} - \widehat{\boldsymbol{A}}(k)\right\|_2 \,, \tag{4.97}$$

then there exists an at least $(n-k)$-dimensional null space $Z \subseteq \mathbb{R}^n$, such that $\boldsymbol{x} \in Z$ implies that $\boldsymbol{B}\boldsymbol{x} = \boldsymbol{0}$. Then it follows that

$$\|\boldsymbol{A}\boldsymbol{x}\|_2 = \|(\boldsymbol{A} - \boldsymbol{B})\boldsymbol{x}\|_2 \,, \tag{4.98}$$

and by using a version of the Cauchy-Schwartz inequality (3.17) that encompasses norms of matrices, we obtain

$$\|\boldsymbol{A}\boldsymbol{x}\|_2 \leqslant \|\boldsymbol{A} - \boldsymbol{B}\|_2 \, \|\boldsymbol{x}\|_2 < \sigma_{k+1} \, \|\boldsymbol{x}\|_2 \,. \tag{4.99}$$

However, there exists a $(k+1)$-dimensional subspace where $\|\boldsymbol{A}\boldsymbol{x}\|_2 \geqslant \sigma_{k+1} \, \|\boldsymbol{x}\|_2$, which is spanned by the right-singular vectors $\boldsymbol{v}_j, j \leqslant k+1$ of $\boldsymbol{A}$. Adding up dimensions of these two spaces yields a number greater than n, as there must be a nonzero vector in both spaces. This is a contradiction of the rank-nullity theorem (Theorem 2.24) in Section 2.7.3.

The Eckart-Young theorem implies that we can use SVD to reduce a rank-r matrix $\boldsymbol{A}$ to a rank-k matrix $\widehat{\boldsymbol{A}}$ in a principled, optimal (in the spectral norm sense) manner. We can interpret the approximation of $\boldsymbol{A}$ by a rank-k matrix as a form of lossy compression. Therefore, the low-rank approximation of a matrix appears in many machine learning applications, e.g., image processing, noise filtering, and regularization of ill-posed problems. Furthermore, it plays a key role in dimensionality reduction and principal component analysis, as we will see in Chapter 10.

Example 4.15 (Finding Structure in Movie Ratings and Consumers (continued))
Coming back to our movie-rating example, we can now apply the concept of low-rank approximations to approximate the original data matrix. Recall that our first singular value captures the notion of science fiction theme in movies and science fiction lovers. Thus, by using only the first singular value term in a rank-1 decomposition of the movie-rating matrix, we obtain the predicted ratings

$$\boldsymbol{A}_1 = \boldsymbol{u}_1 \boldsymbol{v}_1^\top = \begin{bmatrix} -0.6710 \\ -0.7197 \\ -0.0939 \\ -0.1515 \end{bmatrix} \begin{bmatrix} -0.7367 & -0.6515 & -0.1811 \end{bmatrix} \tag{4.100a}$$

$$= \begin{bmatrix} 0.4943 & 0.4372 & 0.1215 \\ 0.5302 & 0.4689 & 0.1303 \\ 0.0692 & 0.0612 & 0.0170 \\ 0.1116 & 0.0987 & 0.0274 \end{bmatrix}. \tag{4.100b}$$

This first rank-1 approximation $\boldsymbol{A}_1$ is insightful: it tells us that Ali and Beatrix like science fiction movies, such as *Star Wars* and *Bladerunner* (entries have values > 0.4), but fails to capture the ratings of the other movies by Chandra. This is not surprising, as Chandra's type of movies is not captured by the first singular value. The second singular value gives us a better rank-1 approximation for those movie-theme lovers:

$$\boldsymbol{A}_2 = \boldsymbol{u}_2\boldsymbol{v}_2^\top = \begin{bmatrix} 0.0236 \\ 0.2054 \\ -0.7705 \\ -0.6030 \end{bmatrix} \begin{bmatrix} 0.0852 & 0.1762 & -0.9807 \end{bmatrix} \tag{4.101a}$$

$$= \begin{bmatrix} 0.0020 & 0.0042 & -0.0231 \\ 0.0175 & 0.0362 & -0.2014 \\ -0.0656 & -0.1358 & 0.7556 \\ -0.0514 & -0.1063 & 0.5914 \end{bmatrix}. \tag{4.101b}$$

In this second rank-1 approximation $\boldsymbol{A}_2$, we capture Chandra's ratings and movie types well, but not the science fiction movies. This leads us to consider the rank-2 approximation $\widehat{\boldsymbol{A}}(2)$, where we combine the first two rank-1 approximations

$$\widehat{\boldsymbol{A}}(2) = \sigma_1\boldsymbol{A}_1 + \sigma_2\boldsymbol{A}_2 = \begin{bmatrix} 4.7801 & 4.2419 & 1.0244 \\ 5.2252 & 4.7522 & -0.0250 \\ 0.2493 & -0.2743 & 4.9724 \\ 0.7495 & 0.2756 & 4.0278 \end{bmatrix}. \tag{4.102}$$

$\widehat{\boldsymbol{A}}(2)$ is similar to the original movie ratings table

$$\boldsymbol{A} = \begin{bmatrix} 5 & 4 & 1 \\ 5 & 5 & 0 \\ 0 & 0 & 5 \\ 1 & 0 & 4 \end{bmatrix}, \tag{4.103}$$

and this suggests that we can ignore the contribution of $\boldsymbol{A}_3$. We can interpret this so that in the data table there is no evidence of a third movie-theme/movie-lovers category. This also means that the entire space of movie-themes/movie-lovers in our example is a two-dimensional space spanned by science fiction and French art house movies and lovers.

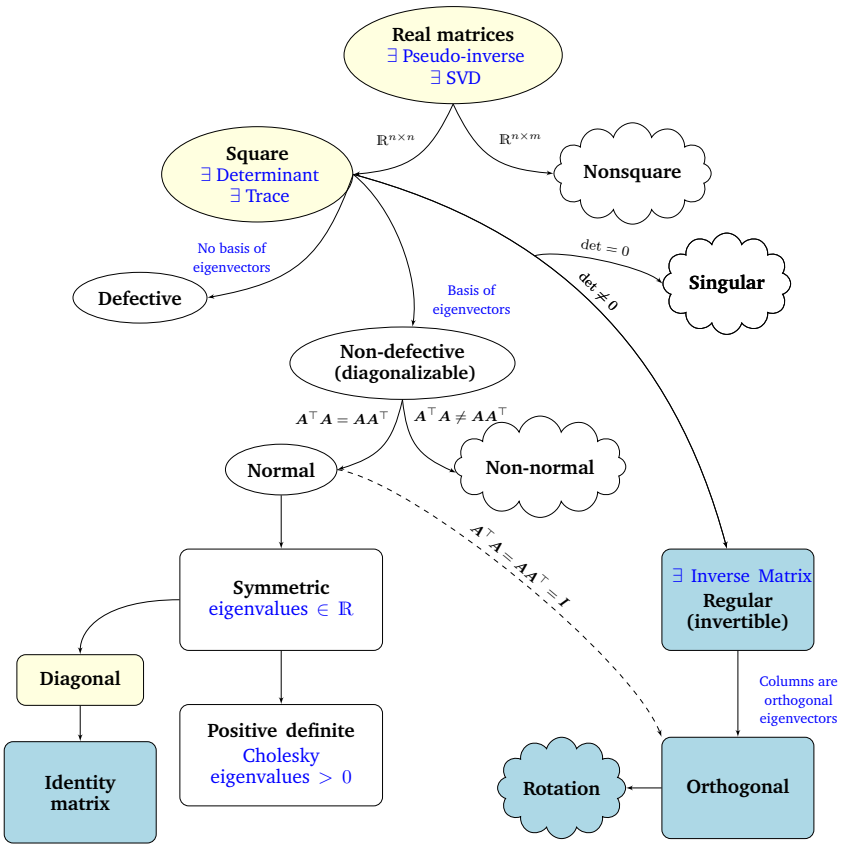


Figure 4.13 A functional phylogeny of matrices encountered in machine learning.

4.7 Matrix Phylogeny

The word "phylogenetic" describes how we capture the relationships among individuals or groups and derived from the Greek words for "tribe" and "source".

In Chapters 2 and 3, we covered the basics of linear algebra and analytic geometry. In this chapter, we looked at fundamental characteristics of matrices and linear mappings. Figure 4.13 depicts the phylogenetic tree of relationships between different types of matrices (black arrows indicating "is a subset of") and the covered operations we can perform on them (in blue). We consider all *real matrices* $\boldsymbol{A} \in \mathbb{R}^{n\times m}$. For non-square matrices (where $n \neq m$), the SVD always exists, as we saw in this chapter. Focusing on *square matrices* $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, the *determinant* informs us whether a square matrix possesses an *inverse matrix*, i.e., whether it belongs to the class of regular, invertible matrices. If the square $n \times n$ matrix possesses n linearly independent eigenvectors, then the matrix is *non-defective* and an *eigendecomposition* exists (Theorem 4.12). We know that repeated eigenvalues may result in defective matrices, which cannot be diagonalized.

Non-singular and non-defective matrices are not the same. For example, a rotation matrix will be invertible (determinant is nonzero) but not diagonalizable in the real numbers (eigenvalues are not guaranteed to be real numbers).

We dive further into the branch of non-defective square $n \times n$ matrices. $\boldsymbol{A}$ is *normal* if the condition $\boldsymbol{A}^\top \boldsymbol{A} = \boldsymbol{A}\boldsymbol{A}^\top$ holds. Moreover, if the more restrictive condition holds that $\boldsymbol{A}^\top \boldsymbol{A} = \boldsymbol{A}\boldsymbol{A}^\top = \boldsymbol{I}$, then $\boldsymbol{A}$ is called *orthogona*l (see Definition 3.8). The set of orthogonal matrices is a subset of the regular (invertible) matrices and satisfies $\boldsymbol{A}^\top = \boldsymbol{A}^{-1}$.

Normal matrices have a frequently encountered subset, the symmetric matrices $\boldsymbol{S} \in \mathbb{R}^{n \times n}$, which satisfy $\boldsymbol{S} = \boldsymbol{S}^\top$. Symmetric matrices have only real eigenvalues. A subset of the symmetric matrices consists of the positive definite matrices $\boldsymbol{P}$ that satisfy the condition of $\boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} > 0$ for all $\boldsymbol{x} \in \mathbb{R}^n \backslash \{\boldsymbol{0}\}$. In this case, a unique *Cholesky decomposition* exists (Theorem 4.18). Positive definite matrices have only positive eigenvalues and are always invertible (i.e., have a nonzero determinant).

Another subset of symmetric matrices consists of the *diagonal matrices* $\boldsymbol{D}$. Diagonal matrices are closed under multiplication and addition, but do not necessarily form a group (this is only the case if all diagonal entries are nonzero so that the matrix is invertible). A special diagonal matrix is the identity matrix $\boldsymbol{I}$.

4.8 Further Reading

Most of the content in this chapter establishes underlying mathematics and connects them to methods for studying mappings, many of which are at the heart of machine learning at the level of underpinning software solutions and building blocks for almost all machine learning theory. Matrix characterization using determinants, eigenspectra, and eigenspaces provides fundamental features and conditions for categorizing and analyzing matrices. This extends to all forms of representations of data and mappings involving data, as well as judging the numerical stability of computational operations on such matrices (Press et al., 2007).

Determinants are fundamental tools in order to invert matrices and compute eigenvalues "by hand". However, for almost all but the smallest instances, numerical computation by Gaussian elimination outperforms determinants (Press et al., 2007). Determinants remain nevertheless a powerful theoretical concept, e.g., to gain intuition about the orientation of a basis based on the sign of the determinant. Eigenvectors can be used to perform basis changes to transform data into the coordinates of meaningful orthogonal, feature vectors. Similarly, matrix decomposition methods, such as the Cholesky decomposition, reappear often when we compute or simulate random events (Rubinstein and Kroese, 2016). Therefore, the Cholesky decomposition enables us to compute the *reparametrization trick* where we want to perform continuous differentiation over random variables, e.g., in variational autoencoders (Jimenez Rezende et al., 2014; Kingma and Welling, 2014).

Eigendecomposition is fundamental in enabling us to extract meaningful and interpretable information that characterizes linear mappings.

Therefore, the eigendecomposition underlies a general class of machine learning algorithms called *spectral methods* that perform eigendecomposition of a positive-definite kernel. These spectral decomposition methods encompass classical approaches to statistical data analysis, such as the following:

principal component analysis

- *Principal component analysis* (PCA (Pearson, 1901), see also Chapter 10), in which a low-dimensional subspace, which explains most of the variability in the data, is sought.

Fisher discriminant analysis

- *Fisher discriminant analysis*, which aims to determine a separating hyperplane for data classification (Mika et al., 1999).

multidimensional scaling

- *Multidimensional scaling* (MDS) (Carroll and Chang, 1970).

The computational efficiency of these methods typically comes from finding the best rank-k approximation to a symmetric, positive semidefinite matrix. More contemporary examples of spectral methods have different origins, but each of them requires the computation of the eigenvectors and eigenvalues of a positive-definite kernel, such as *Isomap* (Tenenbaum et al., 2000), *Laplacian eigenmaps* (Belkin and Niyogi, 2003), *Hessian eigenmaps* (Donoho and Grimes, 2003), and *spectral clustering* (Shi and Malik, 2000). The core computations of these are generally underpinned by low-rank matrix approximation techniques (Belabbas and Wolfe, 2009) as we encountered here via the SVD.

Isomap
Laplacian eigenmaps
Hessian eigenmaps
spectral clustering

The SVD allows us to discover some of the same kind of information as the eigendecomposition. However, the SVD is more generally applicable to non-square matrices and data tables. These matrix factorization methods become relevant whenever we want to identify heterogeneity in data when we want to perform data compression by approximation, e.g., instead of storing $n \times m$ values just storing $(n+m)k$ values, or when we want to perform data pre-processing, e.g., to decorrelate predictor variables of a design matrix (Ormoneit et al., 2001). The SVD operates on matrices, which we can interpret as rectangular arrays with two indices (rows and columns). The extension of matrix-like structure to higher-dimensional arrays are called tensors. It turns out that the SVD is the special case of a more general family of decompositions that operate on such tensors (Kolda and Bader, 2009). SVD-like operations and low-rank approximations on tensors are, for example, the *Tucker decomposition* (Tucker, 1966) or the *CP decomposition* (Carroll and Chang, 1970).

Tucker decomposition
CP decomposition

The SVD low-rank approximation is frequently used in machine learning for computational efficiency reasons. This is because it reduces the amount of memory and operations with nonzero multiplications we need to perform on potentially very large matrices of data (Trefethen and Bau III, 1997). Moreover, low-rank approximations are used to operate on matrices that may contain missing values as well as for purposes of lossy compression and dimensionality reduction (Moonen and De Moor, 1995; Markovsky, 2011).

Exercises

4.1 Compute the determinant using the Laplace expansion (using the first row) and the Sarrus rule for

$$\boldsymbol{A} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \\ 0 & 2 & 4 \end{bmatrix}.$$

4.2 Compute the following determinant efficiently:

$$\begin{bmatrix} 2 & 0 & 1 & 2 & 0 \\ 2 & -1 & 0 & 1 & 1 \\ 0 & 1 & 2 & 1 & 2 \\ -2 & 0 & 2 & -1 & 2 \\ 2 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

4.3 Compute the eigenspaces of

a.

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

b.

$$\boldsymbol{B} := \begin{bmatrix} -2 & 2 \\ 2 & 1 \end{bmatrix}$$

4.4 Compute all eigenspaces of

$$\boldsymbol{A} = \begin{bmatrix} 0 & -1 & 1 & 1 \\ -1 & 1 & -2 & 3 \\ 2 & -1 & 0 & 0 \\ 1 & -1 & 1 & 0 \end{bmatrix}.$$

4.5 Diagonalizability of a matrix is unrelated to its invertibility. Determine for the following four matrices whether they are diagonalizable and/or invertible

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

4.6 Compute the eigenspaces of the following transformation matrices. Are they diagonalizable?

a. For

$$\boldsymbol{A} = \begin{bmatrix} 2 & 3 & 0 \\ 1 & 4 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

b. For

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

4.7 Are the following matrices diagonalizable? If yes, determine their diagonal form and a basis with respect to which the transformation matrices are diagonal. If no, give reasons why they are not diagonalizable.

a.
$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ -8 & 4 \end{bmatrix}$$

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

c.
$$\boldsymbol{A} = \begin{bmatrix} 5 & 4 & 2 & 1 \\ 0 & 1 & -1 & -1 \\ -1 & -1 & 3 & 0 \\ 1 & 1 & -1 & 2 \end{bmatrix}$$

d.
$$\boldsymbol{A} = \begin{bmatrix} 5 & -6 & -6 \\ -1 & 4 & 2 \\ 3 & -6 & -4 \end{bmatrix}$$

4.8 Find the SVD of the matrix
$$\boldsymbol{A} = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{bmatrix}.$$

4.9 Find the singular value decomposition of
$$\boldsymbol{A} = \begin{bmatrix} 2 & 2 \\ -1 & 1 \end{bmatrix}.$$

4.10 Find the rank-1 approximation of
$$\boldsymbol{A} = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{bmatrix}$$

4.11 Show that for any $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ the matrices $\boldsymbol{A}^\top \boldsymbol{A}$ and $\boldsymbol{A}\boldsymbol{A}^\top$ possess the same nonzero eigenvalues.

4.12 Show that for $\boldsymbol{x} \neq \boldsymbol{0}$ Theorem 4.24 holds, i.e., show that
$$\max_{\boldsymbol{x}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2} = \sigma_1 \,,$$
where σ_1 is the largest singular value of $\boldsymbol{A} \in \mathbb{R}^{m \times n}$.