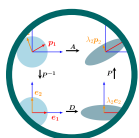


## Matrix Decompositions



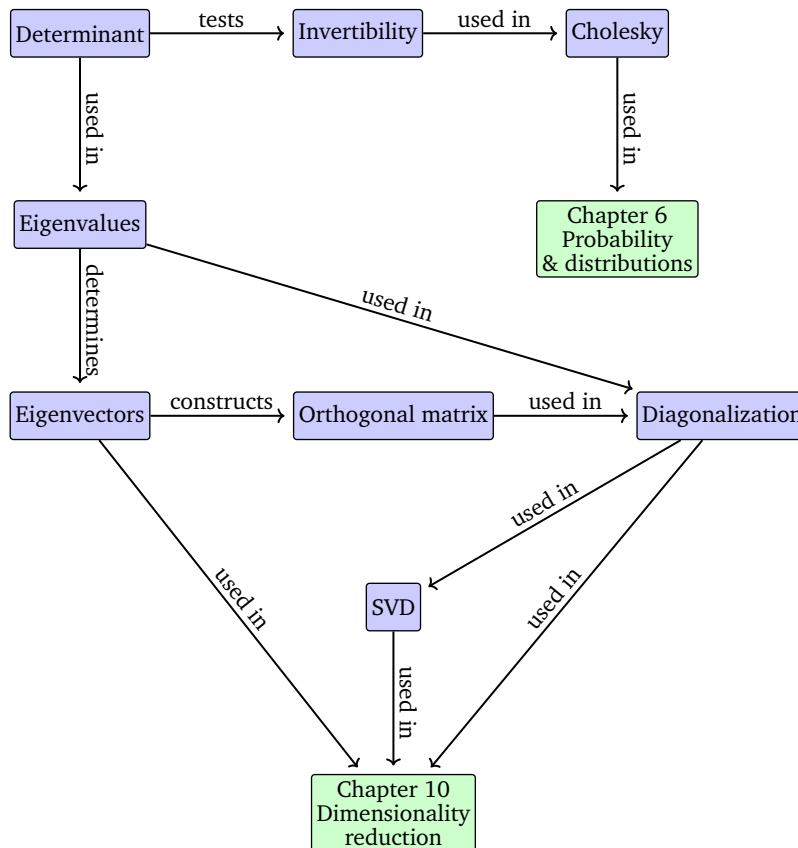
matrix factorization

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.

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

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  $\mathbf{A} \in \mathbb{R}^{n \times n}$ , i.e., matrices with the same number of rows and columns. In this book, we write the determinant as  $\det(\mathbf{A})$  or sometimes as  $|\mathbf{A}|$  so that

$$\det(\mathbf{A}) = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}. \quad (4.1)$$

The determinant notation  $|\mathbf{A}|$  must not be confused with the absolute value.

The *determinant* of a square matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is a function that maps  $\mathbf{A}$  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  $\mathbf{A}$  is invertible (see Section 2.2.2). For the smallest cases, we already know when a matrix is invertible. If  $\mathbf{A}$  is a  $1 \times 1$  matrix, i.e., it is a scalar number, then  $\mathbf{A} = a \implies \mathbf{A}^{-1} = \frac{1}{a}$ . Thus  $a \frac{1}{a} = 1$  holds, if and only if  $a \neq 0$ .

For  $2 \times 2$  matrices, by the definition of the inverse (Definition 2.3), we know that  $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$ . Then, with (2.24), the inverse of  $\mathbf{A}$  is

$$\mathbf{A}^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}. \quad (4.2)$$

Hence,  $\mathbf{A}$  is invertible if and only if

$$a_{11}a_{22} - a_{12}a_{21} \neq 0. \quad (4.3)$$

This quantity is the determinant of  $\mathbf{A} \in \mathbb{R}^{2 \times 2}$ , i.e.,

$$\det(\mathbf{A}) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}. \quad (4.4)$$

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  $\mathbf{A} \in \mathbb{R}^{n \times n}$  it holds that  $\mathbf{A}$  is invertible if and only if  $\det(\mathbf{A}) \neq 0$ .*

We have explicit (closed-form) expressions for determinants of small matrices in terms of the elements of the matrix. For  $n = 1$ ,

$$\det(\mathbf{A}) = \det(a_{11}) = a_{11}. \quad (4.5)$$

For  $n = 2$ ,

$$\det(\mathbf{A}) = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}, \quad (4.6)$$

which we have observed in the preceding example.

For  $n = 3$  (known as Sarrus' rule),

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} + a_{21}a_{32}a_{13} + a_{31}a_{12}a_{23} \\ - a_{31}a_{22}a_{13} - a_{11}a_{32}a_{23} - a_{21}a_{12}a_{33}. \quad (4.7)$$

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  $T$  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  $T \in \mathbb{R}^{n \times n}$ , the determinant is the product of the diagonal elements, i.e.,

$$\det(T) = \prod_{i=1}^n T_{ii}. \quad (4.8)$$

#### 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  $\det(A)$  is the signed volume of an  $n$ -dimensional parallelepiped formed by columns of the matrix  $A$ .

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  $\mathbf{b}, \mathbf{g}$  that form the columns of a matrix  $A = [\mathbf{b}, \mathbf{g}]$ . Then, the absolute value of the determinant of  $A$  is the area of the parallelogram with vertices  $0, \mathbf{b}, \mathbf{g}, \mathbf{b} + \mathbf{g}$ . In particular, if  $\mathbf{b}, \mathbf{g}$  are linearly dependent so that  $\mathbf{b} = \lambda \mathbf{g}$  for some  $\lambda \in \mathbb{R}$ , they no longer form a two-dimensional parallelogram. Therefore, the corresponding area is 0. On the contrary, if  $\mathbf{b}, \mathbf{g}$  are linearly independent and are multiples of the canonical basis vectors  $e_1, e_2$  then they can be written as  $\mathbf{b} = \begin{bmatrix} b \\ 0 \end{bmatrix}$  and

$\mathbf{g} = \begin{bmatrix} 0 \\ g \end{bmatrix}$ , 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  $\mathbf{b}, \mathbf{g}$  with respect to the standard basis  $(e_1, e_2)$ . In our figure, flipping the order to  $\mathbf{g}, \mathbf{b}$  swaps the columns of  $A$  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  $\mathbf{r}, \mathbf{b}, \mathbf{g} \in \mathbb{R}^3$  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  $3 \times 3$  matrix  $[\mathbf{r}, \mathbf{b}, \mathbf{g}]$  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  $\mathbf{r}, \mathbf{g}, \mathbf{b} \in \mathbb{R}^3$  given as

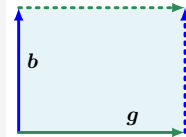
$$\mathbf{r} = \begin{bmatrix} 2 \\ 0 \\ -8 \end{bmatrix}, \quad \mathbf{g} = \begin{bmatrix} 6 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 4 \\ -1 \end{bmatrix}. \quad (4.9)$$

upper-triangular matrix

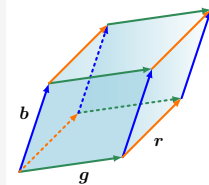
lower-triangular matrix

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  $\mathbf{b}$  and  $\mathbf{g}$  is  $|\det([\mathbf{b}, \mathbf{g}])|$ .



**Figure 4.3** The volume of the parallelepiped (shaded volume) spanned by vectors  $\mathbf{r}, \mathbf{b}, \mathbf{g}$  is  $|\det([\mathbf{r}, \mathbf{b}, \mathbf{g}])|$ .



The sign of the determinant indicates the orientation of the spanning vectors.

Writing these vectors as the columns of a matrix

$$\mathbf{A} = [\mathbf{r}, \mathbf{g}, \mathbf{b}] = \begin{bmatrix} 2 & 6 & 1 \\ 0 & 1 & 4 \\ -8 & 0 & -1 \end{bmatrix} \quad (4.10)$$

allows us to compute the desired volume as

$$V = |\det(\mathbf{A})| = 186. \quad (4.11)$$

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  $2 \times 2$  matrices.

Laplace expansion

**Theorem 4.2** (Laplace Expansion). *Consider a matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$ . Then, for all  $j = 1, \dots, n$ :*

1. *Expansion along column  $j$*

$$\det(\mathbf{A}) = \sum_{k=1}^n (-1)^{k+j} a_{kj} \det(\mathbf{A}_{k,j}). \quad (4.12)$$

2. *Expansion along row  $j$*

$$\det(\mathbf{A}) = \sum_{k=1}^n (-1)^{k+j} a_{jk} \det(\mathbf{A}_{j,k}). \quad (4.13)$$

Here  $\mathbf{A}_{k,j} \in \mathbb{R}^{(n-1) \times (n-1)}$  is the submatrix of  $\mathbf{A}$  that we obtain when deleting row  $k$  and column  $j$ .

### Example 4.3 (Laplace Expansion)

Let us compute the determinant of

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.14)$$

using the Laplace expansion along the first row. Applying (4.13) yields

$$\begin{aligned} \begin{vmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 0 & 0 & 1 \end{vmatrix} &= (-1)^{1+1} \cdot 1 \begin{vmatrix} 1 & 2 \\ 0 & 1 \end{vmatrix} \\ &+ (-1)^{1+2} \cdot 2 \begin{vmatrix} 3 & 2 \\ 0 & 1 \end{vmatrix} + (-1)^{1+3} \cdot 3 \begin{vmatrix} 3 & 1 \\ 0 & 0 \end{vmatrix}. \end{aligned} \quad (4.15)$$

$\det(\mathbf{A}_{k,j})$  is called a *minor* and  $(-1)^{k+j} \det(\mathbf{A}_{k,j})$  a *cofactor*.

We use (4.6) to compute the determinants of all  $2 \times 2$  matrices and obtain

$$\det(\mathbf{A}) = 1(1 - 0) - 2(3 - 0) + 3(0 - 0) = -5. \quad (4.16)$$

For completeness we can compare this result to computing the determinant using Sarrus' rule (4.7):

$$\det(\mathbf{A}) = 1 \cdot 1 \cdot 1 + 3 \cdot 0 \cdot 3 + 0 \cdot 2 \cdot 2 - 0 \cdot 1 \cdot 3 - 1 \cdot 0 \cdot 2 - 3 \cdot 2 \cdot 1 = 1 - 6 = -5. \quad (4.17)$$

For  $\mathbf{A} \in \mathbb{R}^{n \times n}$  the determinant exhibits the following properties:

- The determinant of a matrix product is the product of the corresponding determinants,  $\det(\mathbf{AB}) = \det(\mathbf{A})\det(\mathbf{B})$ .
- Determinants are invariant to transposition, i.e.,  $\det(\mathbf{A}) = \det(\mathbf{A}^\top)$ .
- If  $\mathbf{A}$  is regular (invertible), then  $\det(\mathbf{A}^{-1}) = \frac{1}{\det(\mathbf{A})}$ .
- Similar matrices (Definition 2.22) possess the same determinant. Therefore, for a linear mapping  $\Phi : V \rightarrow V$  all transformation matrices  $\mathbf{A}_\Phi$  of  $\Phi$  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  $\det(\mathbf{A})$ .
- Multiplication of a column/row with  $\lambda \in \mathbb{R}$  scales  $\det(\mathbf{A})$  by  $\lambda$ . In particular,  $\det(\lambda \mathbf{A}) = \lambda^n \det(\mathbf{A})$ .
- Swapping two rows/columns changes the sign of  $\det(\mathbf{A})$ .

Because of the last three properties, we can use Gaussian elimination (see Section 2.1) to compute  $\det(\mathbf{A})$  by bringing  $\mathbf{A}$  into row-echelon form. We can stop Gaussian elimination when we have  $\mathbf{A}$  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  $\mathbf{A} \in \mathbb{R}^{n \times n}$  has  $\det(\mathbf{A}) \neq 0$  if and only if  $\text{rk}(\mathbf{A}) = n$ . In other words,  $\mathbf{A}$  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  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is defined as

trace

$$\text{tr}(\mathbf{A}) := \sum_{i=1}^n a_{ii}, \quad (4.18)$$

i.e., the trace is the sum of the diagonal elements of  $\mathbf{A}$ .

The trace satisfies the following properties:

- $\text{tr}(\mathbf{A} + \mathbf{B}) = \text{tr}(\mathbf{A}) + \text{tr}(\mathbf{B})$  for  $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$
- $\text{tr}(\alpha \mathbf{A}) = \alpha \text{tr}(\mathbf{A})$ ,  $\alpha \in \mathbb{R}$  for  $\mathbf{A} \in \mathbb{R}^{n \times n}$
- $\text{tr}(\mathbf{I}_n) = n$
- $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$  for  $\mathbf{A} \in \mathbb{R}^{n \times k}$ ,  $\mathbf{B} \in \mathbb{R}^{k \times n}$

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.,

$$\text{tr}(\mathbf{AKL}) = \text{tr}(\mathbf{KLA}) \quad (4.19)$$

for matrices  $\mathbf{A} \in \mathbb{R}^{a \times k}$ ,  $\mathbf{K} \in \mathbb{R}^{k \times l}$ ,  $\mathbf{L} \in \mathbb{R}^{l \times a}$ . This property generalizes to products of an arbitrary number of matrices. As a special case of (4.19), it follows that for two vectors  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$

$$\text{tr}(\mathbf{xy}^\top) = \text{tr}(\mathbf{y}^\top \mathbf{x}) = \mathbf{y}^\top \mathbf{x} \in \mathbb{R}. \quad (4.20)$$

Given a linear mapping  $\Phi : V \rightarrow V$ , where  $V$  is a vector space, we define the trace of this map by using the trace of matrix representation of  $\Phi$ . For a given basis of  $V$ , we can describe  $\Phi$  by means of the transformation matrix  $\mathbf{A}$ . Then the trace of  $\Phi$  is the trace of  $\mathbf{A}$ . For a different basis of  $V$ , it holds that the corresponding transformation matrix  $\mathbf{B}$  of  $\Phi$  can be obtained by a basis change of the form  $\mathbf{S}^{-1} \mathbf{AS}$  for suitable  $\mathbf{S}$  (see Section 2.7.2). For the corresponding trace of  $\Phi$ , this means

$$\text{tr}(\mathbf{B}) = \text{tr}(\mathbf{S}^{-1} \mathbf{AS}) \stackrel{(4.19)}{=} \text{tr}(\mathbf{ASS}^{-1}) = \text{tr}(\mathbf{A}). \quad (4.21)$$

Hence, while matrix representations of linear mappings are basis dependent the trace of a linear mapping  $\Phi$  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  $\mathbf{A}$  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  $\mathbf{A} \in \mathbb{R}^{n \times n}$

$$p_{\mathbf{A}}(\lambda) := \det(\mathbf{A} - \lambda \mathbf{I}) \quad (4.22a)$$

$$= c_0 + c_1 \lambda + c_2 \lambda^2 + \cdots + c_{n-1} \lambda^{n-1} + (-1)^n \lambda^n, \quad (4.22b)$$

$c_0, \dots, c_{n-1} \in \mathbb{R}$ , is the *characteristic polynomial* of  $\mathbf{A}$ . In particular,

The trace is  
invariant under  
cyclic permutations.

characteristic  
polynomial

$$c_0 = \det(\mathbf{A}), \quad (4.23)$$

$$c_{n-1} = (-1)^{n-1} \text{tr}(\mathbf{A}). \quad (4.24)$$

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  $\mathbf{A} \in \mathbb{R}^{n \times n}$  be a square matrix. Then  $\lambda \in \mathbb{R}$  is an *eigenvalue* of  $\mathbf{A}$  and  $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$  is the corresponding *eigenvector* of  $\mathbf{A}$  if

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}. \quad (4.25)$$

eigenvalue  
eigenvector

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.  $\diamond$

The following statements are equivalent:

- $\lambda$  is an eigenvalue of  $\mathbf{A} \in \mathbb{R}^{n \times n}$ .
- There exists an  $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$  with  $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ , or equivalently,  $(\mathbf{A} - \lambda\mathbf{I}_n)\mathbf{x} = \mathbf{0}$  can be solved non-trivially, i.e.,  $\mathbf{x} \neq \mathbf{0}$ .
- $\text{rk}(\mathbf{A} - \lambda\mathbf{I}_n) < n$ .
- $\det(\mathbf{A} - \lambda\mathbf{I}_n) = 0$ .

**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  $\mathbf{x}$  is an eigenvector of  $\mathbf{A}$  associated with eigenvalue  $\lambda$ , then for any  $c \in \mathbb{R} \setminus \{0\}$  it holds that  $c\mathbf{x}$  is an eigenvector of  $\mathbf{A}$  with the same eigenvalue since

$$\mathbf{A}(c\mathbf{x}) = c\mathbf{A}\mathbf{x} = c\lambda\mathbf{x} = \lambda(c\mathbf{x}). \quad (4.26)$$

Thus, all vectors that are collinear to  $\mathbf{x}$  are also eigenvectors of  $\mathbf{A}$ .  $\diamond$



**Theorem 4.8.**  $\lambda \in \mathbb{R}$  is an eigenvalue of  $\mathbf{A} \in \mathbb{R}^{n \times n}$  if and only if  $\lambda$  is a root of the characteristic polynomial  $p_{\mathbf{A}}(\lambda)$  of  $\mathbf{A}$ .

algebraic  
multiplicity

**Definition 4.9.** Let a square matrix  $\mathbf{A}$  have an eigenvalue  $\lambda_i$ . The *algebraic multiplicity* of  $\lambda_i$  is the number of times the root appears in the characteristic polynomial.

eigenspace  
eigenspectrum  
spectrum

**Definition 4.10** (Eigenspace and Eigenspectrum). For  $\mathbf{A} \in \mathbb{R}^{n \times n}$ , the set of all eigenvectors of  $\mathbf{A}$  associated with an eigenvalue  $\lambda$  spans a subspace of  $\mathbb{R}^n$ , which is called the *eigenspace* of  $\mathbf{A}$  with respect to  $\lambda$  and is denoted by  $E_\lambda$ . The set of all eigenvalues of  $\mathbf{A}$  is called the *eigenspectrum*, or just *spectrum*, of  $\mathbf{A}$ .

If  $\lambda$  is an eigenvalue of  $\mathbf{A} \in \mathbb{R}^{n \times n}$ , then the corresponding eigenspace  $E_\lambda$  is the solution space of the homogeneous system of linear equations  $(\mathbf{A} - \lambda \mathbf{I})\mathbf{x} = \mathbf{0}$ . 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  $\mathbf{I} \in \mathbb{R}^{n \times n}$  has characteristic polynomial  $p_{\mathbf{I}}(\lambda) = \det(\mathbf{I} - \lambda \mathbf{I}) = (1 - \lambda)^n = 0$ , which has only one eigenvalue  $\lambda = 1$  that occurs  $n$  times. Moreover,  $\mathbf{I}\mathbf{x} = \lambda\mathbf{x} = 1\mathbf{x}$  holds for all vectors  $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ . 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  $\mathbf{I}$ .

Useful properties regarding eigenvalues and eigenvectors include the following:

- A matrix  $\mathbf{A}$  and its transpose  $\mathbf{A}^\top$  possess the same eigenvalues, but not necessarily the same eigenvectors.
- The eigenspace  $E_\lambda$  is the null space of  $\mathbf{A} - \lambda \mathbf{I}$  since

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \iff \mathbf{A}\mathbf{x} - \lambda\mathbf{x} = \mathbf{0} \tag{4.27a}$$

$$\iff (\mathbf{A} - \lambda \mathbf{I})\mathbf{x} = \mathbf{0} \iff \mathbf{x} \in \ker(\mathbf{A} - \lambda \mathbf{I}). \tag{4.27b}$$

- Similar matrices (see Definition 2.22) possess the same eigenvalues. Therefore, a linear mapping  $\Phi$  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  $2 \times 2$  matrix

$$\mathbf{A} = \begin{bmatrix} 4 & 2 \\ 1 & 3 \end{bmatrix}. \quad (4.28)$$

**Step 1: Characteristic Polynomial.** From our definition of the eigenvector  $\mathbf{x} \neq \mathbf{0}$  and eigenvalue  $\lambda$  of  $\mathbf{A}$ , there will be a vector such that  $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ , i.e.,  $(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = \mathbf{0}$ . Since  $\mathbf{x} \neq \mathbf{0}$ , this requires that the kernel (null space) of  $\mathbf{A} - \lambda\mathbf{I}$  contains more elements than just  $\mathbf{0}$ . This means that  $\mathbf{A} - \lambda\mathbf{I}$  is not invertible and therefore  $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$ . Hence, we need to compute the roots of the characteristic polynomial (4.22a) to find the eigenvalues.

**Step 2: Eigenvalues.** The characteristic polynomial is

$$p_{\mathbf{A}}(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I}) \quad (4.29a)$$

$$= \det \left( \begin{bmatrix} 4 & 2 \\ 1 & 3 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \right) = \begin{vmatrix} 4 - \lambda & 2 \\ 1 & 3 - \lambda \end{vmatrix} \quad (4.29b)$$

$$= (4 - \lambda)(3 - \lambda) - 2 \cdot 1. \quad (4.29c)$$

We factorize the characteristic polynomial and obtain

$$p(\lambda) = (4 - \lambda)(3 - \lambda) - 2 \cdot 1 = 10 - 7\lambda + \lambda^2 = (2 - \lambda)(5 - \lambda) \quad (4.30)$$

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  $\mathbf{x}$  such that

$$\begin{bmatrix} 4 - \lambda & 2 \\ 1 & 3 - \lambda \end{bmatrix} \mathbf{x} = \mathbf{0}. \quad (4.31)$$

For  $\lambda = 5$  we obtain

$$\begin{bmatrix} 4 - 5 & 2 \\ 1 & 3 - 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -1 & 2 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \mathbf{0}. \quad (4.32)$$

We solve this homogeneous system and obtain a solution space

$$E_5 = \text{span} \left[ \begin{bmatrix} 2 \\ 1 \end{bmatrix} \right]. \quad (4.33)$$

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

$$\begin{bmatrix} 4 - 2 & 2 \\ 1 & 3 - 2 \end{bmatrix} \mathbf{x} = \begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix} \mathbf{x} = \mathbf{0}. \quad (4.34)$$

This means any vector  $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ , 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

$$E_2 = \text{span}\left[\begin{bmatrix} 1 \\ -1 \end{bmatrix}\right]. \quad (4.35)$$

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  $\lambda_i$  be an eigenvalue of a square matrix  $\mathbf{A}$ . Then the *geometric multiplicity* of  $\lambda_i$  is the number of linearly independent eigenvectors associated with  $\lambda_i$ . In other words, it is the dimensionality of the eigenspace spanned by the eigenvectors associated with  $\lambda_i$ .

*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.  $\diamond$

#### Example 4.6

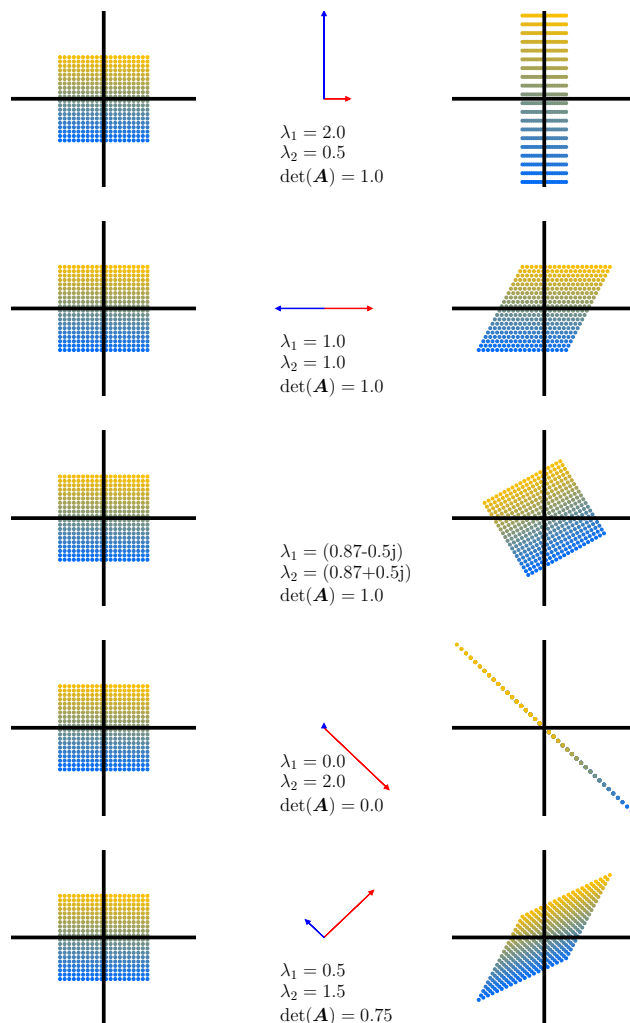
The matrix  $\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix}$  has two repeated eigenvalues  $\lambda_1 = \lambda_2 = 2$  and an algebraic multiplicity of 2. The eigenvalue has, however, only one distinct unit eigenvector  $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$  and, thus, geometric 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  $\mathbf{A}_1, \dots, \mathbf{A}_5$  and their impact on a square grid of points, centered at the origin:

- $\mathbf{A}_1 = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 2 \end{bmatrix}$ . 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 ( $\det(\mathbf{A}_1) = 1 = 2 \cdot \frac{1}{2}$ ).
- $\mathbf{A}_2 = \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix}$  corresponds to a shearing mapping, i.e., it shears the points along the horizontal axis to the right if they are on the positive

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).



**Figure 4.4**  
Determinants and eigenspaces. Overview of five linear mappings and their associated transformation matrices  $A_i \in \mathbb{R}^{2 \times 2}$  projecting 400 color-coded points  $x \in \mathbb{R}^2$  (left column) onto target points  $A_i x$  (right column). The central column depicts the **first eigenvector**, stretched by its associated eigenvalue  $\lambda_1$ , and the **second eigenvector** stretched by its eigenvalue  $\lambda_2$ . Each row depicts the effect of one of five transformation matrices  $A_i$  with respect to the standard basis.

half of the vertical axis, and to the left vice versa. This mapping is area preserving ( $\det(A_2) = 1$ ). 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).

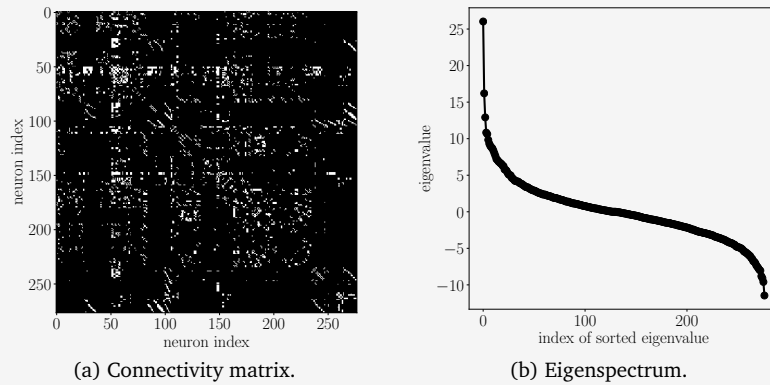
- $A_3 = \begin{bmatrix} \cos(\frac{\pi}{6}) & -\sin(\frac{\pi}{6}) \\ \sin(\frac{\pi}{6}) & \cos(\frac{\pi}{6}) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \sqrt{3} & -1 \\ 1 & \sqrt{3} \end{bmatrix}$  The matrix  $A_3$  rotates the points by  $\frac{\pi}{6}$  rad =  $30^\circ$  counter-clockwise and 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.
- $A_4 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$  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.

- $\mathbf{A}_5 = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix}$  is a shear-and-stretch mapping that scales space by 75% since  $|\det(\mathbf{A}_5)| = \frac{3}{4}$ . It stretches space along the (red) eigenvector of  $\lambda_2$  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)

**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  $\mathbf{A} \in \mathbb{R}^{277 \times 277}$  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  $\mathbf{A}$  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  $\mathbf{A}_{sym} := \mathbf{A} + \mathbf{A}^\top$ . This new matrix  $\mathbf{A}_{sym}$  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  $\mathbf{A}_{sym}$ . 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  $\mathbf{x}_1, \dots, \mathbf{x}_n$  of a matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  with  $n$  distinct eigenvalues  $\lambda_1, \dots, \lambda_n$  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  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is *defective* if it possesses fewer than  $n$  linearly independent eigenvectors. defective

A non-defective matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  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  $\lambda_i$  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).  $\diamond$

**Theorem 4.14.** Given a matrix  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , we can always obtain a symmetric, positive semidefinite matrix  $\mathbf{S} \in \mathbb{R}^{n \times n}$  by defining

$$\mathbf{S} := \mathbf{A}^\top \mathbf{A}. \quad (4.36)$$

*Remark.* If  $\text{rk}(\mathbf{A}) = n$ , then  $\mathbf{S} := \mathbf{A}^\top \mathbf{A}$  is symmetric, positive definite.  $\diamond$

Understanding why Theorem 4.14 holds is insightful for how we can use symmetrized matrices: Symmetry requires  $\mathbf{S} = \mathbf{S}^\top$ , and by inserting (4.36) we obtain  $\mathbf{S} = \mathbf{A}^\top \mathbf{A} = \mathbf{A}^\top (\mathbf{A}^\top)^\top = (\mathbf{A}^\top \mathbf{A})^\top = \mathbf{S}^\top$ . Moreover, positive semidefiniteness (Section 3.2.3) requires that  $\mathbf{x}^\top \mathbf{S} \mathbf{x} \geq 0$  and inserting (4.36) we obtain  $\mathbf{x}^\top \mathbf{S} \mathbf{x} = \mathbf{x}^\top \mathbf{A}^\top \mathbf{A} \mathbf{x} = (\mathbf{x}^\top \mathbf{A}^\top)(\mathbf{A} \mathbf{x}) = (\mathbf{A} \mathbf{x})^\top (\mathbf{A} \mathbf{x}) \geq 0$ , because the dot product computes a sum of squares (which are themselves non-negative).

spectral theorem

**Theorem 4.15** (Spectral Theorem). If  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is symmetric, there exists an orthonormal basis of the corresponding vector space  $V$  consisting of eigenvectors of  $\mathbf{A}$ , and each eigenvalue is real.

A direct implication of the spectral theorem is that the eigendecomposition of a symmetric matrix  $\mathbf{A}$  exists (with real eigenvalues), and that we can find an ONB of eigenvectors so that  $\mathbf{A} = \mathbf{P} \mathbf{D} \mathbf{P}^\top$ , where  $\mathbf{D}$  is diagonal and the columns of  $\mathbf{P}$  contain the eigenvectors.

#### Example 4.8

Consider the matrix

$$\mathbf{A} = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & 2 \\ 2 & 2 & 3 \end{bmatrix}. \quad (4.37)$$

The characteristic polynomial of  $\mathbf{A}$  is

$$p_{\mathbf{A}}(\lambda) = -(\lambda - 1)^2(\lambda - 7), \quad (4.38)$$

so that we obtain the eigenvalues  $\lambda_1 = 1$  and  $\lambda_2 = 7$ , where  $\lambda_1$  is a repeated eigenvalue. Following our standard procedure for computing eigenvectors, we obtain the eigenspaces

$$E_1 = \text{span}\left[\underbrace{\begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}}_{=: \mathbf{x}_1}, \underbrace{\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}}_{=: \mathbf{x}_2}\right], \quad E_7 = \text{span}\left[\underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}}_{=: \mathbf{x}_3}\right]. \quad (4.39)$$

We see that  $\mathbf{x}_3$  is orthogonal to both  $\mathbf{x}_1$  and  $\mathbf{x}_2$ . However, since  $\mathbf{x}_1^\top \mathbf{x}_2 = 1 \neq 0$ , 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  $\mathbf{x}_1, \mathbf{x}_2$  are eigenvectors associated with the same eigenvalue  $\lambda$ . Therefore, for any  $\alpha, \beta \in \mathbb{R}$  it holds that

$$\mathbf{A}(\alpha \mathbf{x}_1 + \beta \mathbf{x}_2) = \mathbf{A}\mathbf{x}_1\alpha + \mathbf{A}\mathbf{x}_2\beta = \lambda(\alpha \mathbf{x}_1 + \beta \mathbf{x}_2), \quad (4.40)$$

i.e., any linear combination of  $\mathbf{x}_1$  and  $\mathbf{x}_2$  is also an eigenvector of  $\mathbf{A}$  associated with  $\lambda$ . 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  $\mathbf{x}_1$  and  $\mathbf{x}_2$  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  $\mathbf{x}_3$ ). In our example, we will obtain

$$\mathbf{x}'_1 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{x}'_2 = \frac{1}{2} \begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix}, \quad (4.41)$$

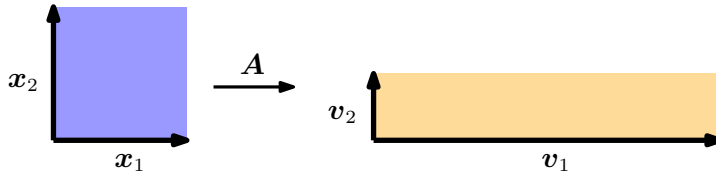
which are orthogonal to each other, orthogonal to  $\mathbf{x}_3$ , and eigenvectors of  $\mathbf{A}$  associated with  $\lambda_1 = 1$ .

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  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is the product of its eigenvalues, i.e.,*

$$\det(\mathbf{A}) = \prod_{i=1}^n \lambda_i, \quad (4.42)$$

where  $\lambda_i \in \mathbb{C}$  are (possibly repeated) eigenvalues of  $\mathbf{A}$ .



**Figure 4.6**  
Geometric interpretation of eigenvalues. The eigenvectors of  $A$  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  $A \in \mathbb{R}^{n \times n}$  is the sum of its eigenvalues, i.e.,*

$$\text{tr}(A) = \sum_{i=1}^n \lambda_i, \quad (4.43)$$

where  $\lambda_i \in \mathbb{C}$  are (possibly repeated) eigenvalues of  $A$ .

Let us provide a geometric intuition of these two theorems. Consider a matrix  $A \in \mathbb{R}^{2 \times 2}$  that possesses two linearly independent eigenvectors  $x_1, x_2$ . For this example, we assume  $(x_1, x_2)$  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  $A$ . In this example, we can compute the change of area explicitly: Mapping the eigenvectors using  $A$  gives us vectors  $v_1 = Ax_1 = \lambda_1 x_1$  and  $v_2 = Ax_2 = \lambda_2 x_2$ , i.e., the new vectors  $v_i$  are scaled versions of the eigenvectors  $x_i$ , and the scaling factors are the corresponding eigenvalues  $\lambda_i$ .  $v_1, v_2$  are still orthogonal, and the area of the rectangle they span is  $|\lambda_1 \lambda_2|$ .

Given that  $x_1, x_2$  (in our example) are orthonormal, we can directly compute the perimeter of the unit square as  $2(1 + 1)$ . Mapping the eigenvectors using  $A$  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  $A$ .

#### Example 4.9 (Google's PageRank – Webpages as Eigenvectors)

Google uses the eigenvector corresponding to the maximal eigenvalue of a matrix  $A$  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)  $x_i \geq 0$  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  $A$  of this graph that tells us with what (click) probability somebody will end up



PageRank

on a different web site. The matrix  $\mathbf{A}$  has the property that for any initial rank/importance vector  $\mathbf{x}$  of a web site the sequence  $\mathbf{x}, \mathbf{A}\mathbf{x}, \mathbf{A}^2\mathbf{x}, \dots$  converges to a vector  $\mathbf{x}^*$ . This vector is called the *PageRank* and satisfies  $\mathbf{A}\mathbf{x}^* = \mathbf{x}^*$ , i.e., it is an eigenvector (with corresponding eigenvalue 1) of  $\mathbf{A}$ . After normalizing  $\mathbf{x}^*$ , such that  $\|\mathbf{x}^*\| = 1$ , 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).

### 4.3 Cholesky Decomposition

Cholesky  
decomposition  
Cholesky  
factorization

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.

**Theorem 4.18** (Cholesky Decomposition). *A symmetric, positive definite matrix  $\mathbf{A}$  can be factorized into a product  $\mathbf{A} = \mathbf{L}\mathbf{L}^\top$ , where  $\mathbf{L}$  is a lower-triangular matrix with positive diagonal elements:*

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ l_{n1} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & \cdots & l_{n1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & l_{nn} \end{bmatrix}. \quad (4.44)$$

Cholesky factor

$\mathbf{L}$  is called the *Cholesky factor* of  $\mathbf{A}$ , and  $\mathbf{L}$  is unique.

#### Example 4.10 (Cholesky Factorization)

Consider a symmetric, positive definite matrix  $\mathbf{A} \in \mathbb{R}^{3 \times 3}$ . We are interested in finding its Cholesky factorization  $\mathbf{A} = \mathbf{L}\mathbf{L}^\top$ , i.e.,

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{21} & a_{22} & a_{32} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \mathbf{L}\mathbf{L}^\top = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} l_{11} & l_{21} & l_{31} \\ 0 & l_{22} & l_{32} \\ 0 & 0 & l_{33} \end{bmatrix}. \quad (4.45)$$

Multiplying out the right-hand side yields

$$\mathbf{A} = \begin{bmatrix} l_{11}^2 & l_{21}l_{11} & l_{31}l_{11} \\ l_{21}l_{11} & l_{21}^2 + l_{22}^2 & l_{31}l_{21} + l_{32}l_{22} \\ l_{31}l_{11} & l_{31}l_{21} + l_{32}l_{22} & l_{31}^2 + l_{32}^2 + l_{33}^2 \end{bmatrix}. \quad (4.46)$$

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  $l_{ii}$ :

$$l_{11} = \sqrt{a_{11}}, \quad l_{22} = \sqrt{a_{22} - l_{21}^2}, \quad l_{33} = \sqrt{a_{33} - (l_{31}^2 + l_{32}^2)}. \quad (4.47)$$

Similarly for the elements below the diagonal ( $l_{ij}$ , where  $i > j$ ), there is also a repeating pattern:

$$l_{21} = \frac{1}{l_{11}} a_{21}, \quad l_{31} = \frac{1}{l_{11}} a_{31}, \quad l_{32} = \frac{1}{l_{22}} (a_{32} - l_{31} l_{21}). \quad (4.48)$$

Thus, we constructed the Cholesky decomposition for any symmetric, positive definite  $3 \times 3$  matrix. The key realization is that we can backward calculate what the components  $l_{ij}$  for the  $\mathbf{L}$  should be, given the values  $a_{ij}$  for  $\mathbf{A}$  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  $\mathbf{A} = \mathbf{L}\mathbf{L}^\top$ , we know that  $\det(\mathbf{A}) = \det(\mathbf{L})\det(\mathbf{L}^\top) = \det(\mathbf{L})^2$ . Since  $\mathbf{L}$  is a triangular matrix, the determinant is simply the product of its diagonal entries so that  $\det(\mathbf{A}) = \prod_i l_{ii}^2$ . 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

$$\mathbf{D} = \begin{bmatrix} c_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & c_n \end{bmatrix}. \quad (4.49)$$

They allow fast computation of determinants, powers, and inverses. The determinant is the product of its diagonal entries, a matrix power  $\mathbf{D}^k$  is given by each diagonal element raised to the power  $k$ , and the inverse  $\mathbf{D}^{-1}$  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  $A, D$  are similar (Definition 2.22) if there exists an invertible matrix  $P$ , such that  $D = P^{-1}AP$ . More specifically, we will look at matrices  $A$  that are similar to diagonal matrices  $D$  that contain the eigenvalues of  $A$  on the diagonal.

diagonalizable

**Definition 4.19** (Diagonalizable). A matrix  $A \in \mathbb{R}^{n \times n}$  is *diagonalizable* if it is similar to a diagonal matrix, i.e., if there exists an invertible matrix  $P \in \mathbb{R}^{n \times n}$  such that  $D = P^{-1}AP$ .

In the following, we will see that diagonalizing a matrix  $A \in \mathbb{R}^{n \times n}$  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  $A$ .

Let  $A \in \mathbb{R}^{n \times n}$ , let  $\lambda_1, \dots, \lambda_n$  be a set of scalars, and let  $p_1, \dots, p_n$  be a set of vectors in  $\mathbb{R}^n$ . We define  $P := [p_1, \dots, p_n]$  and let  $D \in \mathbb{R}^{n \times n}$  be a diagonal matrix with diagonal entries  $\lambda_1, \dots, \lambda_n$ . Then we can show that

$$AP = PD \quad (4.50)$$

if and only if  $\lambda_1, \dots, \lambda_n$  are the eigenvalues of  $A$  and  $p_1, \dots, p_n$  are corresponding eigenvectors of  $A$ .

We can see that this statement holds because

$$AP = A[p_1, \dots, p_n] = [Ap_1, \dots, Ap_n], \quad (4.51)$$

$$PD = [p_1, \dots, p_n] \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} = [\lambda_1 p_1, \dots, \lambda_n p_n]. \quad (4.52)$$

Thus, (4.50) implies that

$$Ap_1 = \lambda_1 p_1 \quad (4.53)$$

$$\vdots$$

$$Ap_n = \lambda_n p_n. \quad (4.54)$$

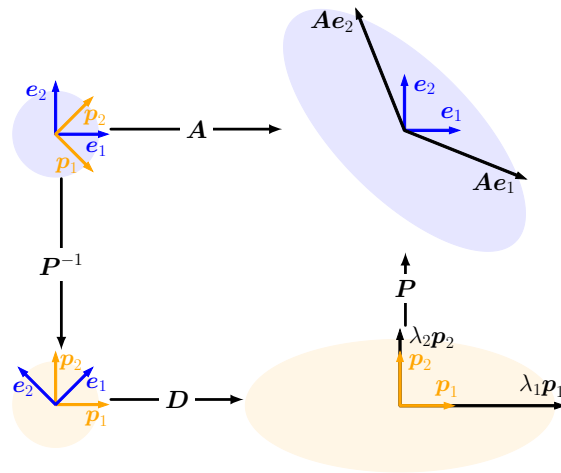
Therefore, the columns of  $P$  must be eigenvectors of  $A$ .

Our definition of diagonalization requires that  $P \in \mathbb{R}^{n \times n}$  is invertible, i.e.,  $P$  has full rank (Theorem 4.3). This requires us to have  $n$  linearly independent eigenvectors  $p_1, \dots, p_n$ , i.e., the  $p_i$  form a basis of  $\mathbb{R}^n$ .

**Theorem 4.20** (Eigendecomposition). A square matrix  $A \in \mathbb{R}^{n \times n}$  can be factored into

$$A = PDP^{-1}, \quad (4.55)$$

where  $P \in \mathbb{R}^{n \times n}$  and  $D$  is a diagonal matrix whose diagonal entries are the eigenvalues of  $A$ , if and only if the eigenvectors of  $A$  form a basis of  $\mathbb{R}^n$ .



**Figure 4.7** Intuition behind the eigendecomposition as sequential transformations. Top-left to bottom-left:  $P^{-1}$  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:  $D$  performs a scaling along the remapped orthogonal eigenvectors, depicted here by a circle being stretched to an ellipse. Bottom-right to top-right:  $P$  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  $P$  are the  $n$  eigenvectors of  $A$ . For symmetric matrices we can obtain even stronger outcomes for the eigenvalue decomposition.

**Theorem 4.21.** *A symmetric matrix  $S \in \mathbb{R}^{n \times n}$  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  $P$  an orthogonal matrix so that  $D = P^\top A P$ .

*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.  $\diamond$

### Geometric Intuition for the Eigendecomposition

We can interpret the eigendecomposition of a matrix as follows (see also Figure 4.7): Let  $A$  be the transformation matrix of a linear mapping with respect to the standard basis  $e_i$  (blue arrows).  $P^{-1}$  performs a basis change from the standard basis into the eigenbasis. Then, the diagonal  $D$  scales the vectors along these axes by the eigenvalues  $\lambda_i$ . Finally,  $P$  transforms these scaled vectors back into the standard/canonical coordinates yielding  $\lambda_i p_i$ .

#### Example 4.11 (Eigendecomposition)

Let us compute the eigendecomposition of  $A = \frac{1}{2} \begin{bmatrix} 5 & -2 \\ -2 & 5 \end{bmatrix}$ .

**Step 1: Compute eigenvalues and eigenvectors.** The characteristic

polynomial of  $\mathbf{A}$  is

$$\det(\mathbf{A} - \lambda \mathbf{I}) = \det \left( \begin{bmatrix} \frac{5}{2} - \lambda & -1 \\ -1 & \frac{5}{2} - \lambda \end{bmatrix} \right) \quad (4.56a)$$

$$= \left(\frac{5}{2} - \lambda\right)^2 - 1 = \lambda^2 - 5\lambda + \frac{21}{4} = \left(\lambda - \frac{7}{2}\right)\left(\lambda - \frac{3}{2}\right). \quad (4.56b)$$

Therefore, the eigenvalues of  $\mathbf{A}$  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

$$\mathbf{A}\mathbf{p}_1 = \frac{7}{2}\mathbf{p}_1, \quad \mathbf{A}\mathbf{p}_2 = \frac{3}{2}\mathbf{p}_2. \quad (4.57)$$

This yields

$$\mathbf{p}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad \mathbf{p}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \quad (4.58)$$

**Step 2: Check for existence.** The eigenvectors  $\mathbf{p}_1, \mathbf{p}_2$  form a basis of  $\mathbb{R}^2$ . Therefore,  $\mathbf{A}$  can be diagonalized.

**Step 3: Construct the matrix  $\mathbf{P}$  to diagonalize  $\mathbf{A}$ .** We collect the eigenvectors of  $\mathbf{A}$  in  $\mathbf{P}$  so that

$$\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2] = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}. \quad (4.59)$$

We then obtain

$$\mathbf{P}^{-1}\mathbf{A}\mathbf{P} = \begin{bmatrix} \frac{7}{2} & 0 \\ 0 & \frac{3}{2} \end{bmatrix} = \mathbf{D}. \quad (4.60)$$

Equivalently, we get (exploiting that  $\mathbf{P}^{-1} = \mathbf{P}^\top$  since the eigenvectors  $\mathbf{p}_1$  and  $\mathbf{p}_2$  in this example form an ONB)

$$\underbrace{\frac{1}{2} \begin{bmatrix} 5 & -2 \\ -2 & 5 \end{bmatrix}}_{\mathbf{A}} = \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} \frac{7}{2} & 0 \\ 0 & \frac{3}{2} \end{bmatrix}}_{\mathbf{D}} \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}}_{\mathbf{P}^{-1}}. \quad (4.61)$$

Figure 4.7 visualizes the eigendecomposition of  $\mathbf{A} = \begin{bmatrix} 5 & -2 \\ -2 & 5 \end{bmatrix}$  as a sequence of linear transformations.

- Diagonal matrices  $\mathbf{D}$  can efficiently be raised to a power. Therefore, we can find a matrix power for a matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  via the eigenvalue decomposition (if it exists) so that

$$\mathbf{A}^k = (\mathbf{P}\mathbf{D}\mathbf{P}^{-1})^k = \mathbf{P}\mathbf{D}^k\mathbf{P}^{-1}. \quad (4.62)$$

Computing  $\mathbf{D}^k$  is efficient because we apply this operation individually to any diagonal element.

- Assume that the eigendecomposition  $\mathbf{A} = \mathbf{P}\mathbf{D}\mathbf{P}^{-1}$  exists. Then,

$$\det(\mathbf{A}) = \det(\mathbf{P}\mathbf{D}\mathbf{P}^{-1}) = \det(\mathbf{P})\det(\mathbf{D})\det(\mathbf{P}^{-1}) \quad (4.63a)$$

$$= \det(\mathbf{D}) = \prod_i d_{ii} \quad (4.63b)$$

allows for an efficient computation of the determinant of  $\mathbf{A}$ .

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  $\mathbf{A}$ , which represents a linear mapping  $\Phi : V \rightarrow W$ , 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.

**Theorem 4.22 (SVD Theorem).** *Let  $\mathbf{A} \in \mathbb{R}^{m \times n}$  be a rectangular matrix of rank  $r \in [0, \min(m, n)]$ . The SVD of  $\mathbf{A}$  is a decomposition of the form*

$$\begin{matrix} n \\ \boxed{\mathbf{A}} \\ m \end{matrix} = \begin{matrix} m \\ \boxed{\mathbf{U}} \\ m \end{matrix} \begin{matrix} m \\ \boxed{\mathbf{\Sigma}} \\ n \end{matrix} \begin{matrix} n \\ \boxed{\mathbf{V}^\top} \\ n \end{matrix} \quad (4.64)$$

SVD theorem

SVD  
singular value  
decomposition

with an orthogonal matrix  $\mathbf{U} \in \mathbb{R}^{m \times m}$  with column vectors  $\mathbf{u}_i$ ,  $i = 1, \dots, m$ , and an orthogonal matrix  $\mathbf{V} \in \mathbb{R}^{n \times n}$  with column vectors  $\mathbf{v}_j$ ,  $j = 1, \dots, n$ . Moreover,  $\mathbf{\Sigma}$  is an  $m \times n$  matrix with  $\Sigma_{ii} = \sigma_i \geq 0$  and  $\Sigma_{ij} = 0$ ,  $i \neq j$ .

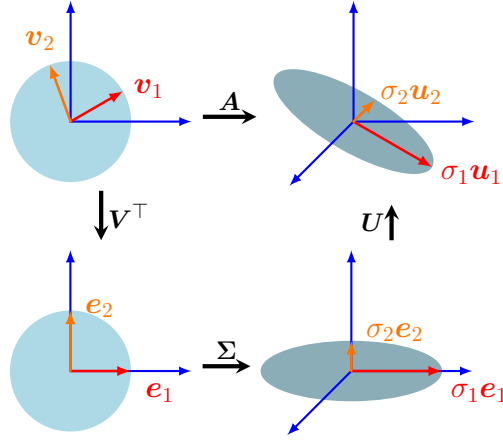
The diagonal entries  $\sigma_i$ ,  $i = 1, \dots, r$ , of  $\mathbf{\Sigma}$  are called the *singular values*,  $\mathbf{u}_i$  are called the *left-singular vectors*, and  $\mathbf{v}_j$  are called the *right-singular vectors*. By convention, the singular values are ordered, i.e.,  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r \geq 0$ .

singular values  
left-singular vectors  
right-singular  
vectors

The *singular value matrix*  $\mathbf{\Sigma}$  is unique, but it requires some attention. Observe that the  $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$  is rectangular. In particular,  $\mathbf{\Sigma}$  is of the same size as  $\mathbf{A}$ . This means that  $\mathbf{\Sigma}$  has a diagonal submatrix that contains the singular values and needs additional zero padding. Specifically, if  $m > n$ , then the matrix  $\mathbf{\Sigma}$  has diagonal structure up to row  $n$  and then consists of

singular value  
matrix

**Figure 4.8** Intuition behind the SVD of a matrix  $A \in \mathbb{R}^{3 \times 2}$  as sequential transformations. Top-left to bottom-left:  $V^\top$  performs a basis change in  $\mathbb{R}^2$ . Bottom-left to bottom-right:  $\Sigma$  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:  $U$  performs a basis change within  $\mathbb{R}^3$ .



$0^\top$  row vectors from  $n + 1$  to  $m$  below so that

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_n \\ 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \end{bmatrix}. \quad (4.65)$$

If  $m < n$ , the matrix  $\Sigma$  has a diagonal structure up to column  $m$  and columns that consist of  $0$  from  $m + 1$  to  $n$ :

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 & \dots & 0 \\ 0 & \ddots & 0 & \vdots & & \vdots \\ 0 & 0 & \sigma_m & 0 & \dots & 0 \end{bmatrix}. \quad (4.66)$$

*Remark.* The SVD exists for any matrix  $A \in \mathbb{R}^{m \times n}$ .  $\diamond$

#### 4.5.1 Geometric Intuitions for the SVD

The SVD offers geometric intuitions to describe a transformation matrix  $A$ . 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)  $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}^m$  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  $V^\top$  followed by a scaling and augmentation (or reduction) in dimensionality via the singular

value matrix  $\Sigma$ . Finally, it performs a second basis change via  $U$ . The SVD entails a number of important details and caveats, which is why we will review our intuition in more detail.

Assume we are given a transformation matrix of a linear mapping  $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}^m$  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  $V$  performs a basis change in the domain  $\mathbb{R}^n$  from  $\tilde{B}$  (represented by the red and orange vectors  $v_1$  and  $v_2$  in the top-left of Figure 4.8) to the standard basis  $B$ .  $V^\top = V^{-1}$  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  $\tilde{B}$ ,  $\Sigma$  scales the new coordinates by the singular values  $\sigma_i$  (and adds or deletes dimensions), i.e.,  $\Sigma$  is the transformation matrix of  $\Phi$  with respect to  $\tilde{B}$  and  $\tilde{C}$ , represented by the red and orange vectors being stretched and lying in the  $e_1$ - $e_2$  plane, which is now embedded in a third dimension in the bottom-right of Figure 4.8.
3.  $U$  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  $e_1$ - $e_2$  plane. This is shown in the top-right of Figure 4.8.

It is useful to review basis changes (Section 2.7.2), orthogonal matrices (Definition 3.8) and orthonormal bases (Section 3.5).

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  $\Sigma$ .

#### 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  $2 \times 2$  centered at the origin. Using the standard basis, we map these vectors using

$$A = \begin{bmatrix} 1 & -0.8 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} = U \Sigma V^\top \quad (4.67a)$$

$$= \begin{bmatrix} -0.79 & 0 & -0.62 \\ 0.38 & -0.78 & -0.49 \\ -0.48 & -0.62 & 0.62 \end{bmatrix} \begin{bmatrix} 1.62 & 0 \\ 0 & 1.0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -0.78 & 0.62 \\ -0.62 & -0.78 \end{bmatrix}. \quad (4.67b)$$

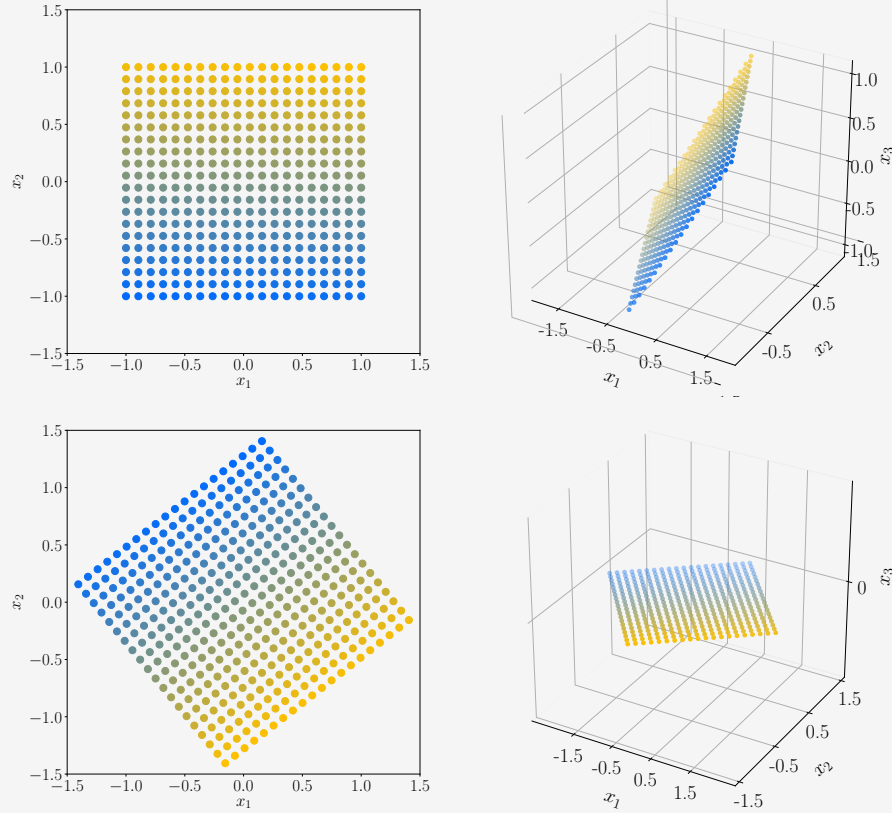
We start with a set of vectors  $\mathcal{X}$  (colored dots; see top-left panel of Figure 4.9) arranged in a grid. We then apply  $V^\top \in \mathbb{R}^{2 \times 2}$ , which rotates  $\mathcal{X}$ . The rotated vectors are shown in the bottom-left panel of Figure 4.9. We now map these vectors using the singular value matrix  $\Sigma$  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  $\mathbf{A}$  to the codomain  $\mathbb{R}^3$  equals the transformation of  $\mathcal{X}$  by  $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ , where  $\mathbf{U}$  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

$$\mathbf{S} = \mathbf{S}^\top = \mathbf{P}\mathbf{D}\mathbf{P}^\top \quad (4.68)$$

with the corresponding SVD

$$S = U\Sigma V^\top. \quad (4.69)$$

If we set

$$U = P = V, \quad D = \Sigma, \quad (4.70)$$

we see that the SVD of SPD matrices is their eigendecomposition.  $\diamond$

In the following, we will explore why Theorem 4.22 holds and how the SVD is constructed. Computing the SVD of  $A \in \mathbb{R}^{m \times n}$  is equivalent to finding two sets of orthonormal bases  $U = (\mathbf{u}_1, \dots, \mathbf{u}_m)$  and  $V = (\mathbf{v}_1, \dots, \mathbf{v}_n)$  of the codomain  $\mathbb{R}^m$  and the domain  $\mathbb{R}^n$ , respectively. From these ordered bases, we will construct the matrices  $U$  and  $V$ .

Our plan is to start with constructing the orthonormal set of right-singular vectors  $\mathbf{v}_1, \dots, \mathbf{v}_n \in \mathbb{R}^n$ . We then construct the orthonormal set of left-singular vectors  $\mathbf{u}_1, \dots, \mathbf{u}_m \in \mathbb{R}^m$ . Thereafter, we will link the two and require that the orthogonality of the  $\mathbf{v}_i$  is preserved under the transformation of  $A$ . This is important because we know that the images  $A\mathbf{v}_i$  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  $A^\top A \in \mathbb{R}^{n \times n}$  from any rectangular matrix  $A \in \mathbb{R}^{m \times n}$ . Thus, we can always diagonalize  $A^\top A$  and obtain

$$A^\top A = PDP^\top = P \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix} P^\top, \quad (4.71)$$

where  $P$  is an orthogonal matrix, which is composed of the orthonormal eigenbasis. The  $\lambda_i \geq 0$  are the eigenvalues of  $A^\top A$ . Let us assume the SVD of  $A$  exists and inject (4.64) into (4.71). This yields

$$A^\top A = (U\Sigma V^\top)^\top (U\Sigma V^\top) = V\Sigma^\top U^\top U\Sigma V^\top, \quad (4.72)$$

where  $U, V$  are orthogonal matrices. Therefore, with  $U^\top U = I$  we obtain

$$A^\top A = V\Sigma^\top \Sigma V^\top = V \begin{bmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_n^2 \end{bmatrix} V^\top. \quad (4.73)$$

Comparing now (4.71) and (4.73), we identify

$$V^\top = P^\top, \quad (4.74)$$

$$\sigma_i^2 = \lambda_i. \quad (4.75)$$

Therefore, the eigenvectors of  $A^\top A$  that compose  $P$  are the right-singular vectors  $V$  of  $A$  (see (4.74)). The eigenvalues of  $A^\top A$  are the squared singular values of  $\Sigma$  (see (4.75)).

To obtain the left-singular vectors  $U$ , we follow a similar procedure. We start by computing the SVD of the symmetric matrix  $AA^\top \in \mathbb{R}^{m \times m}$  (instead of the previous  $A^\top A \in \mathbb{R}^{n \times n}$ ). The SVD of  $A$  yields

$$AA^\top = (U\Sigma V^\top)(U\Sigma V^\top)^\top = U\Sigma V^\top V\Sigma^\top U^\top \quad (4.76a)$$

$$= U \begin{bmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_m^2 \end{bmatrix} U^\top. \quad (4.76b)$$

The spectral theorem tells us that  $AA^\top = SDS^\top$  can be diagonalized and we can find an ONB of eigenvectors of  $AA^\top$ , which are collected in  $S$ . The orthonormal eigenvectors of  $AA^\top$  are the left-singular vectors  $U$  and form an orthonormal basis in the codomain of the SVD.

This leaves the question of the structure of the matrix  $\Sigma$ . Since  $AA^\top$  and  $A^\top A$  have the same nonzero eigenvalues (see page 106), the nonzero entries of the  $\Sigma$  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  $V$ . To finish the construction of the SVD, we connect them with the orthonormal vectors  $U$ . To reach this goal, we use the fact the images of the  $v_i$  under  $A$  have to be orthogonal, too. We can show this by using the results from Section 3.4. We require that the inner product between  $Av_i$  and  $Av_j$  must be 0 for  $i \neq j$ . For any two orthogonal eigenvectors  $v_i, v_j, i \neq j$ , it holds that

$$(Av_i)^\top (Av_j) = v_i^\top (A^\top A) v_j = v_i^\top (\lambda_j v_j) = \lambda_j v_i^\top v_j = 0. \quad (4.77)$$

For the case  $m \geq r$ , it holds that  $\{Av_1, \dots, Av_r\}$  is a basis of an  $r$ -dimensional subspace of  $\mathbb{R}^m$ .

To complete the SVD construction, we need left-singular vectors that are *orthonormal*: We normalize the images of the right-singular vectors  $Av_i$  and obtain

$$u_i := \frac{Av_i}{\|Av_i\|} = \frac{1}{\sqrt{\lambda_i}} Av_i = \frac{1}{\sigma_i} Av_i, \quad (4.78)$$

where the last equality was obtained from (4.75) and (4.76b), showing us that the eigenvalues of  $AA^\top$  are such that  $\sigma_i^2 = \lambda_i$ .

Therefore, the eigenvectors of  $A^\top A$ , which we know are the right-singular vectors  $v_i$ , and their normalized images under  $A$ , the left-singular vectors  $u_i$ , form two self-consistent ONBs that are connected through the singular value matrix  $\Sigma$ .

Let us rearrange (4.78) to obtain the *singular value equation*

$$Av_i = \sigma_i u_i, \quad i = 1, \dots, r. \quad (4.79)$$

singular value  
equation

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  $i \leq n$ , but (4.79) says nothing about the  $\mathbf{u}_i$  for  $i > n$ . However, we know by construction that they are orthonormal. Conversely, for  $m < n$ , (4.79) holds only for  $i \leq m$ . For  $i > m$ , we have  $\mathbf{A}\mathbf{v}_i = \mathbf{0}$  and we still know that the  $\mathbf{v}_i$  form an orthonormal set. This means that the SVD also supplies an orthonormal basis of the kernel (null space) of  $\mathbf{A}$ , the set of vectors  $\mathbf{x}$  with  $\mathbf{A}\mathbf{x} = \mathbf{0}$  (see Section 2.7.3).

Concatenating the  $\mathbf{v}_i$  as the columns of  $\mathbf{V}$  and the  $\mathbf{u}_i$  as the columns of  $\mathbf{U}$  yields

$$\mathbf{A}\mathbf{V} = \mathbf{U}\mathbf{\Sigma}, \quad (4.80)$$

where  $\mathbf{\Sigma}$  has the same dimensions as  $\mathbf{A}$  and a diagonal structure for rows  $1, \dots, r$ . Hence, right-multiplying with  $\mathbf{V}^\top$  yields  $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ , which is the SVD of  $\mathbf{A}$ .

#### Example 4.13 (Computing the SVD)

Let us find the singular value decomposition of

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix}. \quad (4.81)$$

The SVD requires us to compute the right-singular vectors  $\mathbf{v}_j$ , the singular values  $\sigma_k$ , and the left-singular vectors  $\mathbf{u}_i$ .

##### Step 1: Right-singular vectors as the eigenbasis of $\mathbf{A}^\top \mathbf{A}$ .

We start by computing

$$\mathbf{A}^\top \mathbf{A} = \begin{bmatrix} 1 & -2 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 5 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \quad (4.82)$$

We compute the singular values and right-singular vectors  $\mathbf{v}_j$  through the eigenvalue decomposition of  $\mathbf{A}^\top \mathbf{A}$ , which is given as

$$\mathbf{A}^\top \mathbf{A} = \begin{bmatrix} \frac{5}{\sqrt{30}} & 0 & \frac{-1}{\sqrt{6}} \\ \frac{-2}{\sqrt{30}} & \frac{1}{\sqrt{5}} & \frac{-2}{\sqrt{6}} \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{6}} \end{bmatrix} \begin{bmatrix} 6 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{5}{\sqrt{30}} & \frac{-2}{\sqrt{30}} & \frac{1}{\sqrt{30}} \\ 0 & \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{-1}{\sqrt{6}} & \frac{-2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \end{bmatrix} = \mathbf{P}\mathbf{D}\mathbf{P}^\top, \quad (4.83)$$

and we obtain the right-singular vectors as the columns of  $\mathbf{P}$  so that

$$\mathbf{V} = \mathbf{P} = \begin{bmatrix} \frac{5}{\sqrt{30}} & 0 & \frac{-1}{\sqrt{6}} \\ \frac{-2}{\sqrt{30}} & \frac{1}{\sqrt{5}} & \frac{-2}{\sqrt{6}} \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{6}} \end{bmatrix}. \quad (4.84)$$

##### Step 2: Singular-value matrix.

As the singular values  $\sigma_i$  are the square roots of the eigenvalues of

$\mathbf{A}^\top \mathbf{A}$  we obtain them straight from  $\mathbf{D}$ . Since  $\text{rk}(\mathbf{A}) = 2$ , 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  $\mathbf{A}$ , and we obtain

$$\mathbf{\Sigma} = \begin{bmatrix} \sqrt{6} & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \quad (4.85)$$

**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  $\mathbf{A}$  and normalizing them by dividing them by their corresponding singular value. We obtain

$$\mathbf{u}_1 = \frac{1}{\sigma_1} \mathbf{A} \mathbf{v}_1 = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{5}{\sqrt{30}} \\ \frac{-2}{\sqrt{30}} \\ \frac{1}{\sqrt{30}} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{5}} \\ -\frac{2}{\sqrt{5}} \end{bmatrix}, \quad (4.86)$$

$$\mathbf{u}_2 = \frac{1}{\sigma_2} \mathbf{A} \mathbf{v}_2 = \frac{1}{1} \begin{bmatrix} 1 & 0 & 1 \\ -2 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ \frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{bmatrix} = \begin{bmatrix} \frac{2}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{bmatrix}, \quad (4.87)$$

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2] = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 & 2 \\ -2 & 1 \end{bmatrix}. \quad (4.88)$$

Note that on a computer the approach illustrated here has poor numerical behavior, and the SVD of  $\mathbf{A}$  is normally computed without resorting to the eigenvalue decomposition of  $\mathbf{A}^\top \mathbf{A}$ .

### 4.5.3 Eigenvalue Decomposition vs. Singular Value Decomposition

Let us consider the eigendecomposition  $\mathbf{A} = \mathbf{P} \mathbf{D} \mathbf{P}^{-1}$  and the SVD  $\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^\top$  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  $\mathbf{P}$  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  $\mathbf{U}$  and  $\mathbf{V}$  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

$$\begin{array}{c}
 \text{Star Wars} \\
 \text{Blade Runner} \\
 \text{Amelie} \\
 \text{Delicatessen}
 \end{array}
 \begin{array}{c}
 \text{Ali} \\
 \text{Beatrix} \\
 \text{Chandra}
 \end{array}
 \begin{bmatrix}
 5 & 4 & 1 \\
 5 & 5 & 0 \\
 0 & 0 & 5 \\
 1 & 0 & 4
 \end{bmatrix}
 =
 \begin{bmatrix}
 -0.6710 & 0.0236 & 0.4647 & -0.5774 \\
 -0.7197 & 0.2054 & -0.4759 & 0.4619 \\
 -0.0939 & -0.7705 & -0.5268 & -0.3464 \\
 -0.1515 & -0.6030 & 0.5293 & -0.5774
 \end{bmatrix}
 \begin{bmatrix}
 9.6438 & 0 & 0 \\
 0 & 6.3639 & 0 \\
 0 & 0 & 0.7056 \\
 0 & 0 & 0
 \end{bmatrix}
 \begin{bmatrix}
 -0.7367 & -0.6515 & -0.1811 \\
 0.0852 & 0.1762 & -0.9807 \\
 0.6708 & -0.7379 & -0.0743
 \end{bmatrix}$$

**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  $U$  and  $V$  are generally not inverse of each other (they perform basis changes in different vector spaces). In the eigendecomposition, the basis change matrices  $P$  and  $P^{-1}$  are inverses of each other.
- In the SVD, the entries in the diagonal matrix  $\Sigma$  are all real and non-negative, 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  $A$  are eigenvectors of  $AA^T$
  - The right-singular vectors of  $A$  are eigenvectors of  $A^T A$ .
  - The nonzero singular values of  $A$  are the square roots of the nonzero eigenvalues of both  $AA^T$  and  $A^T A$ .
- For symmetric matrices  $A \in \mathbb{R}^{n \times n}$ , 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  $A \in \mathbb{R}^{4 \times 3}$  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  $\mathbf{x}_{\text{Ali}}$ ,  $\mathbf{x}_{\text{Beatrix}}$ ,  $\mathbf{x}_{\text{Chandra}}$ .

Factoring  $\mathbf{A}$  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  $\mathbf{A}$  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  $\mathbf{u}_i$  as stereotypical movies and the right-singular vectors  $\mathbf{v}_j$  as stereotypical viewers.

We then make the assumption that any viewer's specific movie preferences can be expressed as a linear combination of the  $\mathbf{v}_j$ . Similarly, any movie's like-ability can be expressed as a linear combination of the  $\mathbf{u}_i$ . 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  $\mathbf{u}_1$  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  $\mathbf{v}_1$  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  $\mathbf{v}_1$  reflects the notion of a science fiction lover.

Similarly,  $\mathbf{u}_2$ , seems to capture a French art house film theme, and  $\mathbf{v}_2$  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  $\mathbf{v}_1$  gives a rating of zero to everything but science fiction themed—this logic is implied by the diagonal substructure for the singular value matrix  $\Sigma$ . 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.

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.

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*.
- Some authors define the SVD a bit differently and focus on square singular matrices. Then, for  $\mathbf{A} \in \mathbb{R}^{m \times n}$  and  $m \geq n$ ,

$$\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^T. \quad (4.89)$$

$m \times n$        $m \times n$     $n \times n$     $n \times n$

full SVD

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  $\Sigma$  is diagonal, as in the eigenvalue decomposition.

- In Section 4.6, we will learn about matrix approximation techniques using the SVD, which is also called the *truncated SVD*.
- It is possible to define the SVD of a rank- $r$  matrix  $\mathbf{A}$  so that  $\mathbf{U}$  is an  $m \times r$  matrix,  $\Sigma$  a diagonal matrix of size  $r \times r$ , and  $\mathbf{V}$  an  $n \times r$  matrix. This construction is very similar to our definition and ensures that the diagonal matrix  $\Sigma$  has only nonzero entries along the diagonal. The main convenience of this alternative notation is that  $\Sigma$  is diagonal, as in the eigenvalue decomposition.
- A restriction that the SVD for  $\mathbf{A}$  only applies to  $m \times n$  matrices with  $m > n$  is practically unnecessary. When  $m < n$ , the SVD decomposition will yield  $\Sigma$  with more zero columns than rows and, consequently, the singular values  $\sigma_{m+1}, \dots, \sigma_n$  are 0.

reduced SVD

truncated SVD

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  $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^\top \in \mathbb{R}^{m \times n}$  into the product of three matrices, where  $\mathbf{U} \in \mathbb{R}^{m \times m}$  and  $\mathbf{V} \in \mathbb{R}^{n \times n}$  are orthogonal and  $\Sigma$  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  $\mathbf{A}$  as a sum of simpler (low-rank) matrices  $\mathbf{A}_i$ , which lends itself to a matrix approximation scheme that is cheaper to compute than the full SVD.

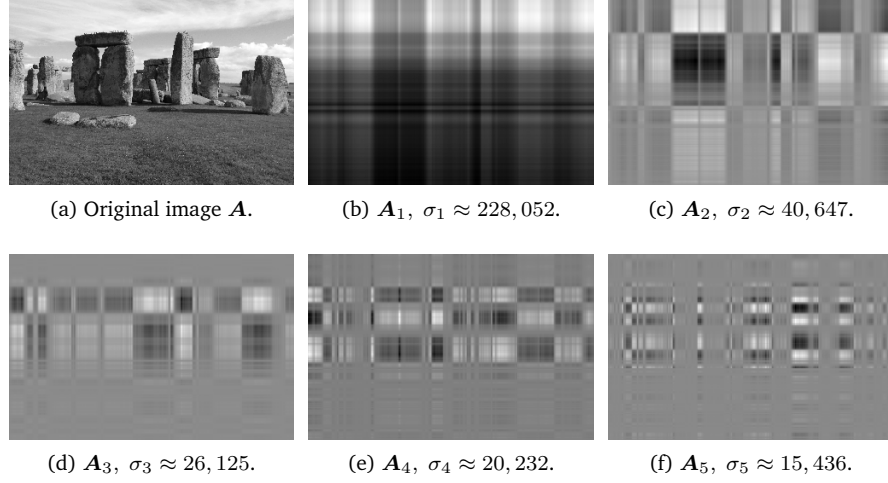
We construct a rank-1 matrix  $\mathbf{A}_i \in \mathbb{R}^{m \times n}$  as

$$\mathbf{A}_i := \mathbf{u}_i \mathbf{v}_i^\top, \quad (4.90)$$

which is formed by the outer product of the  $i$ th orthogonal column vector of  $\mathbf{U}$  and  $\mathbf{V}$ . Figure 4.11 shows an image of Stonehenge, which can be represented by a matrix  $\mathbf{A} \in \mathbb{R}^{1432 \times 1910}$ , and some outer products  $\mathbf{A}_i$ , as defined in (4.90).



**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  $\mathbf{A}_1, \dots, \mathbf{A}_5$  and their corresponding singular values  $\sigma_1, \dots, \sigma_5$ . The grid-like structure of each rank-1 matrix is imposed by the outer-product of the left and right-singular vectors.



A matrix  $\mathbf{A} \in \mathbb{R}^{m \times n}$  of rank  $r$  can be written as a sum of rank-1 matrices  $\mathbf{A}_i$  so that

$$\mathbf{A} = \sum_{i=1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^\top = \sum_{i=1}^r \sigma_i \mathbf{A}_i, \quad (4.91)$$

where the outer-product matrices  $\mathbf{A}_i$  are weighted by the  $i$ th singular value  $\sigma_i$ . We can see why (4.91) holds: The diagonal structure of the singular value matrix  $\Sigma$  multiplies only matching left- and right-singular vectors  $\mathbf{u}_i \mathbf{v}_i^\top$  and scales them by the corresponding singular value  $\sigma_i$ . All terms  $\Sigma_{ij} \mathbf{u}_i \mathbf{v}_j^\top$  vanish for  $i \neq j$  because  $\Sigma$  is a diagonal matrix. Any terms  $i > r$  vanish because the corresponding singular values are 0.

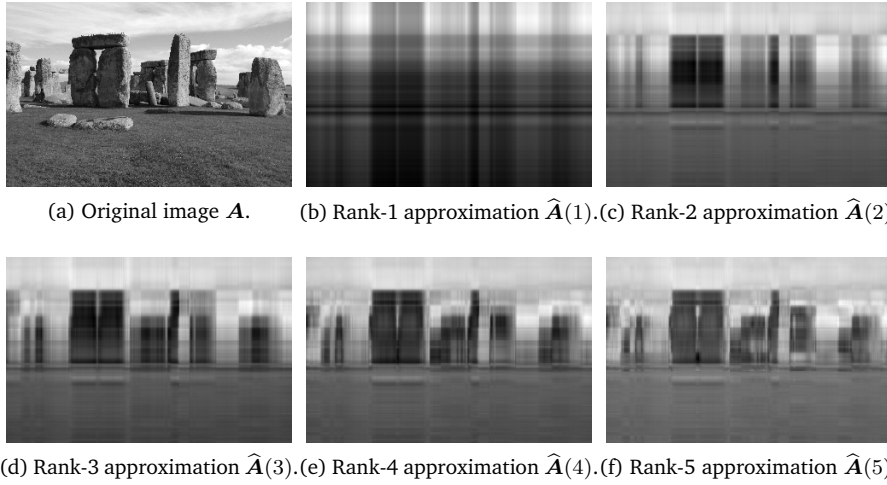
In (4.90), we introduced rank-1 matrices  $\mathbf{A}_i$ . We summed up the  $r$  individual rank-1 matrices to obtain a rank- $r$  matrix  $\mathbf{A}$ ; see (4.91). If the sum does not run over all matrices  $\mathbf{A}_i$ ,  $i = 1, \dots, r$ , but only up to an intermediate value  $k < r$ , we obtain a *rank- $k$  approximation*

$$\hat{\mathbf{A}}(k) := \sum_{i=1}^k \sigma_i \mathbf{u}_i \mathbf{v}_i^\top = \sum_{i=1}^k \sigma_i \mathbf{A}_i \quad (4.92)$$

of  $\mathbf{A}$  with  $\text{rk}(\hat{\mathbf{A}}(k)) = k$ . Figure 4.12 shows low-rank approximations  $\hat{\mathbf{A}}(k)$  of an original image  $\mathbf{A}$  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  $\mathbf{A}$  and its rank- $k$  approximation  $\hat{\mathbf{A}}(k)$ , we need the notion of a norm. In Section 3.1, we already used

rank- $k$   
approximation



**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  $\hat{\mathbf{A}}(k) = \sum_{i=1}^k \sigma_i \mathbf{A}_i$ .

norms on vectors that measure the length of a vector. By analogy we can also define norms on matrices.

**Definition 4.23** (Spectral Norm of a Matrix). For  $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ , the *spectral norm* of a matrix  $\mathbf{A} \in \mathbb{R}^{m \times n}$  is defined as

$$\|\mathbf{A}\|_2 := \max_{\mathbf{x}} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2}. \quad (4.93)$$

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  $\mathbf{x}$  can at most become when multiplied by  $\mathbf{A}$ .

**Theorem 4.24.** The spectral norm of  $\mathbf{A}$  is its largest singular value  $\sigma_1$ .

We leave the proof of this theorem as an exercise.

**Theorem 4.25** (Eckart-Young Theorem (Eckart and Young, 1936)). Consider a matrix  $\mathbf{A} \in \mathbb{R}^{m \times n}$  of rank  $r$  and let  $\mathbf{B} \in \mathbb{R}^{m \times n}$  be a matrix of rank  $k$ . For any  $k \leq r$  with  $\hat{\mathbf{A}}(k) = \sum_{i=1}^k \sigma_i \mathbf{u}_i \mathbf{v}_i^\top$  it holds that

$$\hat{\mathbf{A}}(k) = \operatorname{argmin}_{\operatorname{rk}(\mathbf{B})=k} \|\mathbf{A} - \mathbf{B}\|_2, \quad (4.94)$$

$$\|\mathbf{A} - \hat{\mathbf{A}}(k)\|_2 = \sigma_{k+1}. \quad (4.95)$$

The Eckart-Young theorem states explicitly how much error we introduce by approximating  $\mathbf{A}$  using a rank- $k$  approximation. We can interpret the rank- $k$  approximation obtained with the SVD as a projection of the full-rank matrix  $\mathbf{A}$  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  $\mathbf{A}$  and any rank- $k$  approximation.

We can retrace some of the steps to understand why (4.95) should hold.

spectral norm

Eckart-Young theorem

We observe that the difference between  $\mathbf{A} - \hat{\mathbf{A}}(k)$  is a matrix containing the sum of the remaining rank-1 matrices

$$\mathbf{A} - \hat{\mathbf{A}}(k) = \sum_{i=k+1}^r \sigma_i \mathbf{u}_i \mathbf{v}_i^\top. \quad (4.96)$$

By Theorem 4.24, we immediately obtain  $\sigma_{k+1}$  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  $\mathbf{B}$  with  $\text{rk}(\mathbf{B}) \leq k$ , such that

$$\|\mathbf{A} - \mathbf{B}\|_2 < \|\mathbf{A} - \hat{\mathbf{A}}(k)\|_2, \quad (4.97)$$

then there exists an at least  $(n - k)$ -dimensional null space  $Z \subseteq \mathbb{R}^n$ , such that  $\mathbf{x} \in Z$  implies that  $\mathbf{B}\mathbf{x} = \mathbf{0}$ . Then it follows that

$$\|\mathbf{A}\mathbf{x}\|_2 = \|(\mathbf{A} - \mathbf{B})\mathbf{x}\|_2, \quad (4.98)$$

and by using a version of the Cauchy-Schwartz inequality (3.17) that encompasses norms of matrices, we obtain

$$\|\mathbf{A}\mathbf{x}\|_2 \leq \|\mathbf{A} - \mathbf{B}\|_2 \|\mathbf{x}\|_2 < \sigma_{k+1} \|\mathbf{x}\|_2. \quad (4.99)$$

However, there exists a  $(k + 1)$ -dimensional subspace where  $\|\mathbf{A}\mathbf{x}\|_2 \geq \sigma_{k+1} \|\mathbf{x}\|_2$ , which is spanned by the right-singular vectors  $\mathbf{v}_j, j \leq k + 1$  of  $\mathbf{A}$ . 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  $\mathbf{A}$  to a rank- $k$  matrix  $\hat{\mathbf{A}}$  in a principled, optimal (in the spectral norm sense) manner. We can interpret the approximation of  $\mathbf{A}$  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

$$\mathbf{A}_1 = \mathbf{u}_1 \mathbf{v}_1^\top = \begin{bmatrix} -0.6710 \\ -0.7197 \\ -0.0939 \\ -0.1515 \end{bmatrix} \begin{bmatrix} -0.7367 & -0.6515 & -0.1811 \end{bmatrix} \quad (4.100a)$$

$$= \begin{bmatrix} 0.4943 & 0.4372 & 0.1215 \\ 0.5302 & 0.4689 & 0.1303 \\ 0.0692 & 0.0612 & 0.0170 \\ 0.1116 & 0.0987 & 0.0274 \end{bmatrix}. \quad (4.100b)$$

This first rank-1 approximation  $\mathbf{A}_1$  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:

$$\mathbf{A}_2 = \mathbf{u}_2 \mathbf{v}_2^\top = \begin{bmatrix} 0.0236 \\ 0.2054 \\ -0.7705 \\ -0.6030 \end{bmatrix} [0.0852 \quad 0.1762 \quad -0.9807] \quad (4.101a)$$

$$= \begin{bmatrix} 0.0020 & 0.0042 & -0.0231 \\ 0.0175 & 0.0362 & -0.2014 \\ -0.0656 & -0.1358 & 0.7556 \\ -0.0514 & -0.1063 & 0.5914 \end{bmatrix}. \quad (4.101b)$$

In this second rank-1 approximation  $\mathbf{A}_2$ , we capture Chandra's ratings and movie types well, but not the science fiction movies. This leads us to consider the rank-2 approximation  $\hat{\mathbf{A}}(2)$ , where we combine the first two rank-1 approximations

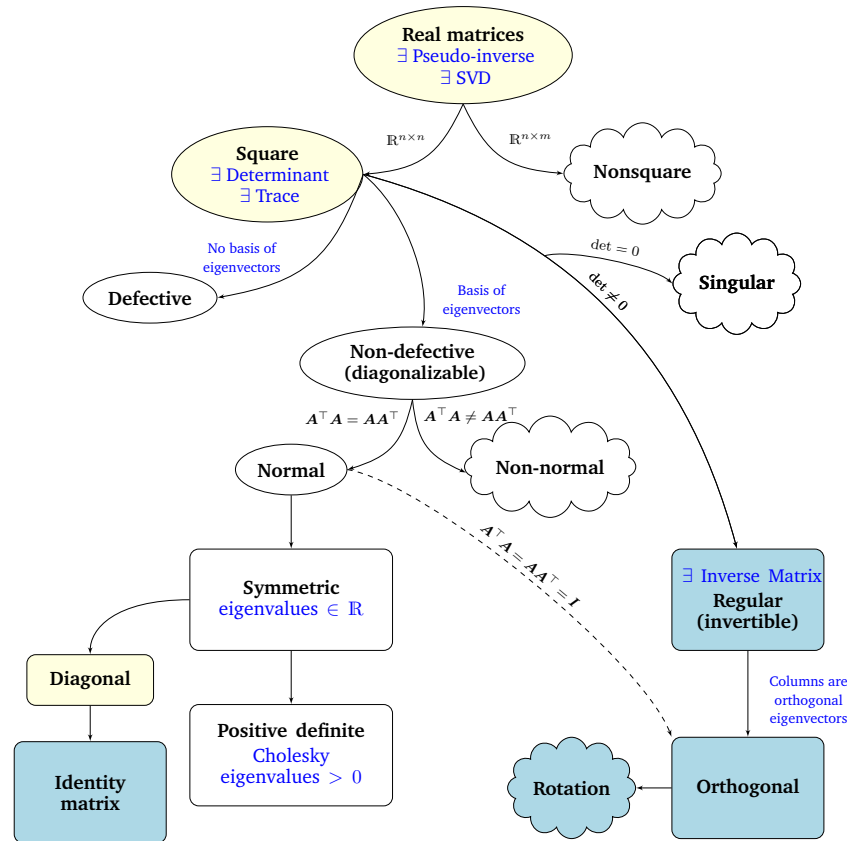
$$\hat{\mathbf{A}}(2) = \sigma_1 \mathbf{A}_1 + \sigma_2 \mathbf{A}_2 = \begin{bmatrix} 4.7801 & 4.2419 & 1.0244 \\ 5.2252 & 4.7522 & -0.0250 \\ 0.2493 & -0.2743 & 4.9724 \\ 0.7495 & 0.2756 & 4.0278 \end{bmatrix}. \quad (4.102)$$

$\hat{\mathbf{A}}(2)$  is similar to the original movie ratings table

$$\mathbf{A} = \begin{bmatrix} 5 & 4 & 1 \\ 5 & 5 & 0 \\ 0 & 0 & 5 \\ 1 & 0 & 4 \end{bmatrix}, \quad (4.103)$$

and this suggests that we can ignore the contribution of  $\mathbf{A}_3$ . 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

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*  $A \in \mathbb{R}^{n \times m}$ . For non-square matrices (where  $n \neq m$ ), the SVD always exists, as we saw in this chapter. Focusing on *square matrices*  $A \in \mathbb{R}^{n \times n}$ , 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).

The word “phylogenetic” describes how we capture the relationships among individuals or groups and derived from the Greek words for “tribe” and “source”.

We dive further into the branch of non-defective square  $n \times n$  matrices.  $\mathbf{A}$  is *normal* if the condition  $\mathbf{A}^\top \mathbf{A} = \mathbf{A} \mathbf{A}^\top$  holds. Moreover, if the more restrictive condition holds that  $\mathbf{A}^\top \mathbf{A} = \mathbf{A} \mathbf{A}^\top = \mathbf{I}$ , then  $\mathbf{A}$  is called *orthogonal* (see Definition 3.8). The set of orthogonal matrices is a subset of the regular (invertible) matrices and satisfies  $\mathbf{A}^\top = \mathbf{A}^{-1}$ .

Normal matrices have a frequently encountered subset, the symmetric matrices  $\mathbf{S} \in \mathbb{R}^{n \times n}$ , which satisfy  $\mathbf{S} = \mathbf{S}^\top$ . Symmetric matrices have only real eigenvalues. A subset of the symmetric matrices consists of the positive definite matrices  $\mathbf{P}$  that satisfy the condition of  $\mathbf{x}^\top \mathbf{P} \mathbf{x} > 0$  for all  $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ . 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*  $\mathbf{D}$ . 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  $\mathbf{I}$ .

## 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

Fisher discriminant analysis

multidimensional scaling

- *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*, which aims to determine a separating hyperplane for data classification (Mika et al., 1999).
- *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

$$\mathbf{A} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \\ 0 & 2 & 4 \end{bmatrix}.$$

- 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.

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

b.

$$\mathbf{B} := \begin{bmatrix} -2 & 2 \\ 2 & 1 \end{bmatrix}$$

- 4.4 Compute all eigenspaces of

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

- 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

$$\mathbf{A} = \begin{bmatrix} 2 & 3 & 0 \\ 1 & 4 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

b. For

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



- 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.

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -8 & 4 \end{bmatrix}$$

b.

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

c.

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

d.

$$\mathbf{A} = \begin{bmatrix} 5 & -6 & -6 \\ -1 & 4 & 2 \\ 3 & -6 & -4 \end{bmatrix}$$

- 4.8 Find the SVD of the matrix

$$\mathbf{A} = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{bmatrix}.$$

- 4.9 Find the singular value decomposition of

$$\mathbf{A} = \begin{bmatrix} 2 & 2 \\ -1 & 1 \end{bmatrix}.$$

- 4.10 Find the rank-1 approximation of

$$\mathbf{A} = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{bmatrix}$$

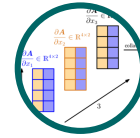
- 4.11 Show that for any  $\mathbf{A} \in \mathbb{R}^{m \times n}$  the matrices  $\mathbf{A}^\top \mathbf{A}$  and  $\mathbf{A} \mathbf{A}^\top$  possess the same nonzero eigenvalues.

- 4.12 Show that for  $\mathbf{x} \neq \mathbf{0}$  Theorem 4.24 holds, i.e., show that

$$\max_{\mathbf{x}} \frac{\|\mathbf{A}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_1,$$

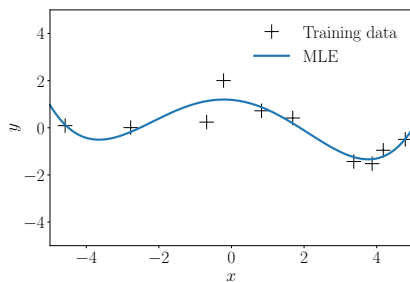
where  $\sigma_1$  is the largest singular value of  $\mathbf{A} \in \mathbb{R}^{m \times n}$ .

## Vector Calculus

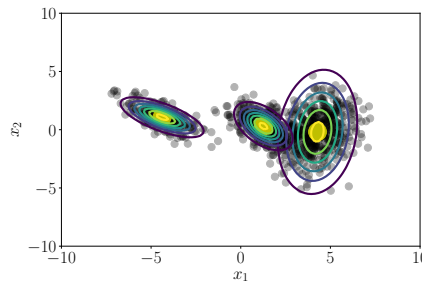


Many algorithms in machine learning optimize an objective function with respect to a set of desired model parameters that control how well a model explains the data: Finding good parameters can be phrased as an optimization problem (see Sections 8.2 and 8.3). Examples include: (i) linear regression (see Chapter 9), where we look at curve-fitting problems and optimize linear weight parameters to maximize the likelihood; (ii) neural-network auto-encoders for dimensionality reduction and data compression, where the parameters are the weights and biases of each layer, and where we minimize a reconstruction error by repeated application of the chain rule; and (iii) Gaussian mixture models (see Chapter 11) for modeling data distributions, where we optimize the location and shape parameters of each mixture component to maximize the likelihood of the model. Figure 5.1 illustrates some of these problems, which we typically solve by using optimization algorithms that exploit gradient information (Section 7.1). Figure 5.2 gives an overview of how concepts in this chapter are related and how they are connected to other chapters of the book.

Central to this chapter is the concept of a function. A function  $f$  is a quantity that relates two quantities to each other. In this book, these quantities are typically inputs  $\mathbf{x} \in \mathbb{R}^D$  and targets (function values)  $f(\mathbf{x})$ , which we assume are real-valued if not stated otherwise. Here  $\mathbb{R}^D$  is the *domain* of  $f$ , and the function values  $f(\mathbf{x})$  are the *image/codomain* of  $f$ .



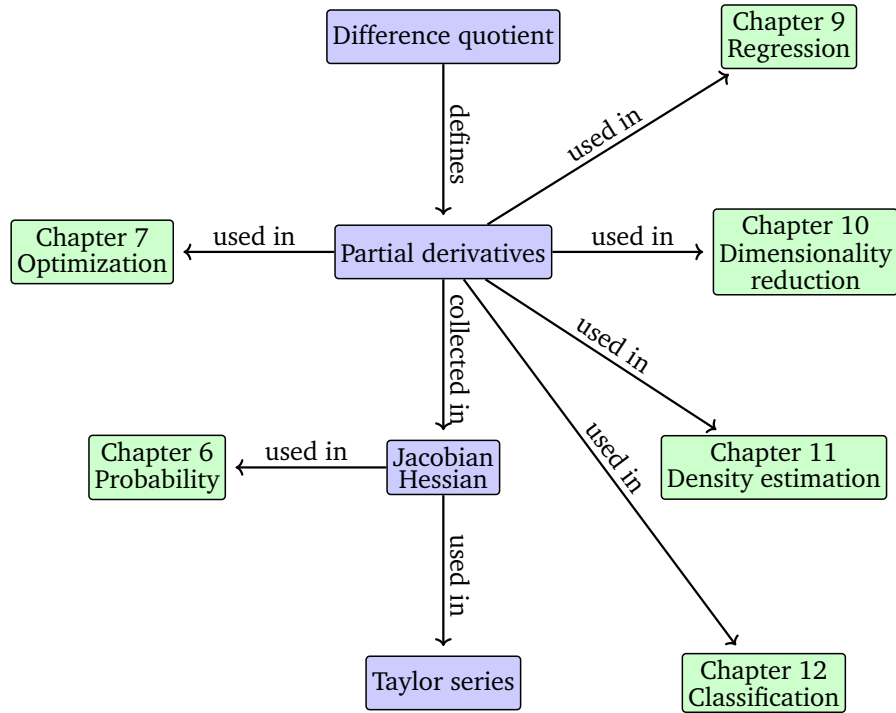
(a) Regression problem: Find parameters, such that the curve explains the observations (crosses) well.



(b) Density estimation with a Gaussian mixture model: Find means and covariances, such that the data (dots) can be explained well.

domain  
image/codomain  
**Figure 5.1** Vector calculus plays a central role in (a) regression (curve fitting) and (b) density estimation, i.e., modeling data distributions.

**Figure 5.2** A mind map of the concepts introduced in this chapter, along with when they are used in other parts of the book.



Section 2.7.3 provides much more detailed discussion in the context of linear functions. We often write

$$f : \mathbb{R}^D \rightarrow \mathbb{R} \quad (5.1a)$$

$$\mathbf{x} \mapsto f(\mathbf{x}) \quad (5.1b)$$

to specify a function, where (5.1a) specifies that  $f$  is a mapping from  $\mathbb{R}^D$  to  $\mathbb{R}$  and (5.1b) specifies the explicit assignment of an input  $\mathbf{x}$  to a function value  $f(\mathbf{x})$ . A function  $f$  assigns every input  $\mathbf{x}$  exactly one function value  $f(\mathbf{x})$ .

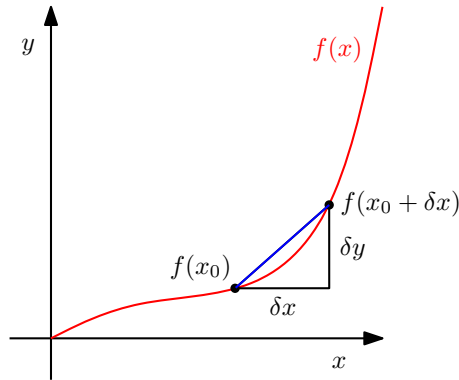
### Example 5.1

Recall the dot product as a special case of an inner product (Section 3.2). In the previous notation, the function  $f(\mathbf{x}) = \mathbf{x}^\top \mathbf{x}$ ,  $\mathbf{x} \in \mathbb{R}^2$ , would be specified as

$$f : \mathbb{R}^2 \rightarrow \mathbb{R} \quad (5.2a)$$

$$\mathbf{x} \mapsto x_1^2 + x_2^2. \quad (5.2b)$$

In this chapter, we will discuss how to compute gradients of functions, which is often essential to facilitate learning in machine learning models since the gradient points in the direction of steepest ascent. Therefore,



**Figure 5.3** The average incline of a function  $f$  between  $x_0$  and  $x_0 + \delta x$  is the incline of the secant (blue) through  $f(x_0)$  and  $f(x_0 + \delta x)$  and given by  $\delta y / \delta x$ .

vector calculus is one of the fundamental mathematical tools we need in machine learning. Throughout this book, we assume that functions are differentiable. With some additional technical definitions, which we do not cover here, many of the approaches presented can be extended to sub-differentials (functions that are continuous but not differentiable at certain points). We will look at an extension to the case of functions with constraints in Chapter 7.

### 5.1 Differentiation of Univariate Functions

In the following, we briefly revisit differentiation of a univariate function, which may be familiar from high school mathematics. We start with the difference quotient of a univariate function  $y = f(x)$ ,  $x, y \in \mathbb{R}$ , which we will subsequently use to define derivatives.

**Definition 5.1** (Difference Quotient). The *difference quotient*

difference quotient

$$\frac{\delta y}{\delta x} := \frac{f(x + \delta x) - f(x)}{\delta x} \quad (5.3)$$

computes the slope of the secant line through two points on the graph of  $f$ . In Figure 5.3, these are the points with  $x$ -coordinates  $x_0$  and  $x_0 + \delta x$ .

The difference quotient can also be considered the average slope of  $f$  between  $x$  and  $x + \delta x$  if we assume  $f$  to be a linear function. In the limit for  $\delta x \rightarrow 0$ , we obtain the tangent of  $f$  at  $x$ , if  $f$  is differentiable. The tangent is then the derivative of  $f$  at  $x$ .

**Definition 5.2** (Derivative). More formally, for  $h > 0$  the *derivative* of  $f$  at  $x$  is defined as the limit

derivative

$$\frac{df}{dx} := \lim_{h \rightarrow 0} \frac{f(x + h) - f(x)}{h}, \quad (5.4)$$

and the secant in Figure 5.3 becomes a tangent.

The derivative of  $f$  points in the direction of steepest ascent of  $f$ .

**Example 5.2 (Derivative of a Polynomial)**

We want to compute the derivative of  $f(x) = x^n$ ,  $n \in \mathbb{N}$ . We may already know that the answer will be  $nx^{n-1}$ , but we want to derive this result using the definition of the derivative as the limit of the difference quotient.

Using the definition of the derivative in (5.4), we obtain

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (5.5a)$$

$$= \lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h} \quad (5.5b)$$

$$= \lim_{h \rightarrow 0} \frac{\sum_{i=0}^n \binom{n}{i} x^{n-i} h^i - x^n}{h}. \quad (5.5c)$$

We see that  $x^n = \binom{n}{0} x^{n-0} h^0$ . By starting the sum at 1, the  $x^n$ -term cancels, and we obtain

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{\sum_{i=1}^n \binom{n}{i} x^{n-i} h^i}{h} \quad (5.6a)$$

$$= \lim_{h \rightarrow 0} \sum_{i=1}^n \binom{n}{i} x^{n-i} h^{i-1} \quad (5.6b)$$

$$= \lim_{h \rightarrow 0} \left( \binom{n}{1} x^{n-1} + \underbrace{\sum_{i=2}^n \binom{n}{i} x^{n-i} h^{i-1}}_{\rightarrow 0 \text{ as } h \rightarrow 0} \right) \quad (5.6c)$$

$$= \frac{n!}{1!(n-1)!} x^{n-1} = nx^{n-1}. \quad (5.6d)$$

**5.1.1 Taylor Series**

The Taylor series is a representation of a function  $f$  as an infinite sum of terms. These terms are determined using derivatives of  $f$  evaluated at  $x_0$ .

Taylor polynomial  
We define  $t^0 := 1$   
for all  $t \in \mathbb{R}$ .

**Definition 5.3** (Taylor Polynomial). The *Taylor polynomial* of degree  $n$  of  $f : \mathbb{R} \rightarrow \mathbb{R}$  at  $x_0$  is defined as

$$T_n(x) := \sum_{k=0}^n \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k, \quad (5.7)$$

where  $f^{(k)}(x_0)$  is the  $k$ th derivative of  $f$  at  $x_0$  (which we assume exists) and  $\frac{f^{(k)}(x_0)}{k!}$  are the coefficients of the polynomial.

**Definition 5.4** (Taylor Series). For a smooth function  $f \in C^\infty$ ,  $f : \mathbb{R} \rightarrow \mathbb{R}$ , the *Taylor series* of  $f$  at  $x_0$  is defined as

Taylor series

$$T_{\infty}(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k. \quad (5.8)$$

For  $x_0 = 0$ , we obtain the *Maclaurin series* as a special instance of the Taylor series. If  $f(x) = T_{\infty}(x)$ , then  $f$  is called *analytic*.

$f \in C^{\infty}$  means that  $f$  is continuously differentiable infinitely many times.  
Maclaurin series analytic

*Remark.* In general, a Taylor polynomial of degree  $n$  is an approximation of a function, which does not need to be a polynomial. The Taylor polynomial is similar to  $f$  in a neighborhood around  $x_0$ . However, a Taylor polynomial of degree  $n$  is an exact representation of a polynomial  $f$  of degree  $k \leq n$  since all derivatives  $f^{(i)}$ ,  $i > k$  vanish.  $\diamond$

### Example 5.3 (Taylor Polynomial)

We consider the polynomial

$$f(x) = x^4 \quad (5.9)$$

and seek the Taylor polynomial  $T_6$ , evaluated at  $x_0 = 1$ . We start by computing the coefficients  $f^{(k)}(1)$  for  $k = 0, \dots, 6$ :

$$f(1) = 1 \quad (5.10)$$

$$f'(1) = 4 \quad (5.11)$$

$$f''(1) = 12 \quad (5.12)$$

$$f^{(3)}(1) = 24 \quad (5.13)$$

$$f^{(4)}(1) = 24 \quad (5.14)$$

$$f^{(5)}(1) = 0 \quad (5.15)$$

$$f^{(6)}(1) = 0 \quad (5.16)$$

Therefore, the desired Taylor polynomial is

$$T_6(x) = \sum_{k=0}^6 \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k \quad (5.17a)$$

$$= 1 + 4(x - 1) + 6(x - 1)^2 + 4(x - 1)^3 + (x - 1)^4 + 0. \quad (5.17b)$$

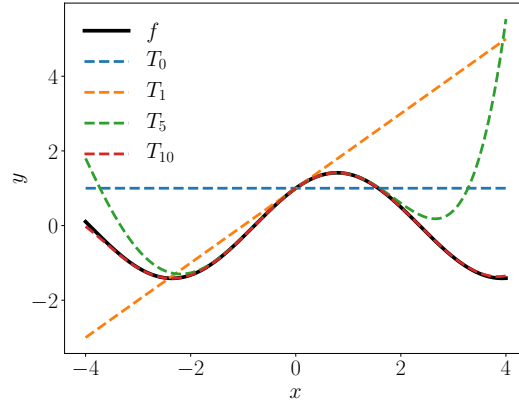
Multiplying out and re-arranging yields

$$T_6(x) = (1 - 4 + 6 - 4 + 1) + x(4 - 12 + 12 - 4) + x^2(6 - 12 + 6) + x^3(4 - 4) + x^4 \quad (5.18a)$$

$$= x^4 = f(x), \quad (5.18b)$$

i.e., we obtain an exact representation of the original function.

**Figure 5.4** Taylor polynomials. The original function  $f(x) = \sin(x) + \cos(x)$  (black, solid) is approximated by Taylor polynomials (dashed) around  $x_0 = 0$ . Higher-order Taylor polynomials approximate the function  $f$  better and more globally.  $T_{10}$  is already similar to  $f$  in  $[-4, 4]$ .



#### Example 5.4 (Taylor Series)

Consider the function in Figure 5.4 given by

$$f(x) = \sin(x) + \cos(x) \in \mathcal{C}^\infty. \quad (5.19)$$

We seek a Taylor series expansion of  $f$  at  $x_0 = 0$ , which is the Maclaurin series expansion of  $f$ . We obtain the following derivatives:

$$f(0) = \sin(0) + \cos(0) = 1 \quad (5.20)$$

$$f'(0) = \cos(0) - \sin(0) = 1 \quad (5.21)$$

$$f''(0) = -\sin(0) - \cos(0) = -1 \quad (5.22)$$

$$f^{(3)}(0) = -\cos(0) + \sin(0) = -1 \quad (5.23)$$

$$f^{(4)}(0) = \sin(0) + \cos(0) = f(0) = 1 \quad (5.24)$$

$\vdots$

We can see a pattern here: The coefficients in our Taylor series are only  $\pm 1$  (since  $\sin(0) = 0$ ), each of which occurs twice before switching to the other one. Furthermore,  $f^{(k+4)}(0) = f^{(k)}(0)$ .

Therefore, the full Taylor series expansion of  $f$  at  $x_0 = 0$  is given by

$$T_\infty(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k \quad (5.25a)$$

$$= 1 + x - \frac{1}{2!}x^2 - \frac{1}{3!}x^3 + \frac{1}{4!}x^4 + \frac{1}{5!}x^5 - \dots \quad (5.25b)$$

$$= 1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \dots + x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \dots \quad (5.25c)$$

$$= \sum_{k=0}^{\infty} (-1)^k \frac{1}{(2k)!} x^{2k} + \sum_{k=0}^{\infty} (-1)^k \frac{1}{(2k+1)!} x^{2k+1} \quad (5.25d)$$

$$= \cos(x) + \sin(x), \quad (5.25e)$$

where we used the *power series representations*

power series  
representation

$$\cos(x) = \sum_{k=0}^{\infty} (-1)^k \frac{1}{(2k)!} x^{2k}, \quad (5.26)$$

$$\sin(x) = \sum_{k=0}^{\infty} (-1)^k \frac{1}{(2k+1)!} x^{2k+1}. \quad (5.27)$$

Figure 5.4 shows the corresponding first Taylor polynomials  $T_n$  for  $n = 0, 1, 5, 10$ .

*Remark.* A Taylor series is a special case of a power series

$$f(x) = \sum_{k=0}^{\infty} a_k (x - c)^k \quad (5.28)$$

where  $a_k$  are coefficients and  $c$  is a constant, which has the special form in Definition 5.4.  $\diamond$

### 5.1.2 Differentiation Rules

In the following, we briefly state basic differentiation rules, where we denote the derivative of  $f$  by  $f'$ .

$$\text{Product rule: } (f(x)g(x))' = f'(x)g(x) + f(x)g'(x) \quad (5.29)$$

$$\text{Quotient rule: } \left( \frac{f(x)}{g(x)} \right)' = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2} \quad (5.30)$$

$$\text{Sum rule: } (f(x) + g(x))' = f'(x) + g'(x) \quad (5.31)$$

$$\text{Chain rule: } (g(f(x)))' = (g \circ f)'(x) = g'(f(x))f'(x) \quad (5.32)$$

Here,  $g \circ f$  denotes function composition  $x \mapsto f(x) \mapsto g(f(x))$ .

#### Example 5.5 (Chain Rule)

Let us compute the derivative of the function  $h(x) = (2x + 1)^4$  using the chain rule. With

$$h(x) = (2x + 1)^4 = g(f(x)), \quad (5.33)$$

$$f(x) = 2x + 1, \quad (5.34)$$

$$g(f) = f^4, \quad (5.35)$$

we obtain the derivatives of  $f$  and  $g$  as

$$f'(x) = 2, \quad (5.36)$$

$$g'(f) = 4f^3, \quad (5.37)$$



such that the derivative of  $h$  is given as

$$h'(x) = g'(f)f'(x) = (4f^3) \cdot 2 \stackrel{(5.34)}{=} 4(2x+1)^3 \cdot 2 = 8(2x+1)^3, \quad (5.38)$$

where we used the chain rule (5.32) and substituted the definition of  $f$  in (5.34) in  $g'(f)$ .

## 5.2 Partial Differentiation and Gradients

Differentiation as discussed in Section 5.1 applies to functions  $f$  of a scalar variable  $x \in \mathbb{R}$ . In the following, we consider the general case where the function  $f$  depends on one or more variables  $\mathbf{x} \in \mathbb{R}^n$ , e.g.,  $f(\mathbf{x}) = f(x_1, x_2)$ . The generalization of the derivative to functions of several variables is the *gradient*.

We find the gradient of the function  $f$  with respect to  $\mathbf{x}$  by *varying one variable at a time* and keeping the others constant. The gradient is then the collection of these *partial derivatives*.

partial derivative

**Definition 5.5** (Partial Derivative). For a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $\mathbf{x} \mapsto f(\mathbf{x})$ ,  $\mathbf{x} \in \mathbb{R}^n$  of  $n$  variables  $x_1, \dots, x_n$  we define the *partial derivatives* as

$$\begin{aligned} \frac{\partial f}{\partial x_1} &= \lim_{h \rightarrow 0} \frac{f(x_1 + h, x_2, \dots, x_n) - f(\mathbf{x})}{h} \\ &\vdots \\ \frac{\partial f}{\partial x_n} &= \lim_{h \rightarrow 0} \frac{f(x_1, \dots, x_{n-1}, x_n + h) - f(\mathbf{x})}{h} \end{aligned} \quad (5.39)$$

and collect them in the row vector

$$\nabla_{\mathbf{x}} f = \text{grad} f = \frac{df}{d\mathbf{x}} = \left[ \frac{\partial f(\mathbf{x})}{\partial x_1} \quad \frac{\partial f(\mathbf{x})}{\partial x_2} \quad \dots \quad \frac{\partial f(\mathbf{x})}{\partial x_n} \right] \in \mathbb{R}^{1 \times n}, \quad (5.40)$$

where  $n$  is the number of variables and 1 is the dimension of the image/range/codomain of  $f$ . Here, we defined the column vector  $\mathbf{x} = [x_1, \dots, x_n]^T \in \mathbb{R}^n$ . The row vector in (5.40) is called the *gradient* of  $f$  or the *Jacobian* and is the generalization of the derivative from Section 5.1.

gradient  
Jacobian

*Remark.* This definition of the Jacobian is a special case of the general definition of the Jacobian for vector-valued functions as the collection of partial derivatives. We will get back to this in Section 5.3.  $\diamond$

We can use results from scalar differentiation: Each partial derivative is a derivative with respect to a scalar.

### Example 5.6 (Partial Derivatives Using the Chain Rule)

For  $f(x, y) = (x + 2y^3)^2$ , we obtain the partial derivatives

$$\frac{\partial f(x, y)}{\partial x} = 2(x + 2y^3) \frac{\partial}{\partial x} (x + 2y^3) = 2(x + 2y^3), \quad (5.41)$$

$$\frac{\partial f(x, y)}{\partial y} = 2(x + 2y^3) \frac{\partial}{\partial y}(x + 2y^3) = 12(x + 2y^3)y^2. \quad (5.42)$$

where we used the chain rule (5.32) to compute the partial derivatives.

*Remark (Gradient as a Row Vector).* It is not uncommon in the literature to define the gradient vector as a column vector, following the convention that vectors are generally column vectors. The reason why we define the gradient vector as a row vector is twofold: First, we can consistently generalize the gradient to vector-valued functions  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  (then the gradient becomes a matrix). Second, we can immediately apply the multi-variate chain rule without paying attention to the dimension of the gradient. We will discuss both points in Section 5.3.  $\diamond$

### Example 5.7 (Gradient)

For  $f(x_1, x_2) = x_1^2 x_2 + x_1 x_2^3 \in \mathbb{R}$ , the partial derivatives (i.e., the derivatives of  $f$  with respect to  $x_1$  and  $x_2$ ) are

$$\frac{\partial f(x_1, x_2)}{\partial x_1} = 2x_1 x_2 + x_2^3 \quad (5.43)$$

$$\frac{\partial f(x_1, x_2)}{\partial x_2} = x_1^2 + 3x_1 x_2^2 \quad (5.44)$$

and the gradient is then

$$\frac{df}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial f(x_1, x_2)}{\partial x_1} & \frac{\partial f(x_1, x_2)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 2x_1 x_2 + x_2^3 & x_1^2 + 3x_1 x_2^2 \end{bmatrix} \in \mathbb{R}^{1 \times 2}. \quad (5.45)$$

### 5.2.1 Basic Rules of Partial Differentiation

In the multivariate case, where  $\mathbf{x} \in \mathbb{R}^n$ , the basic differentiation rules that we know from school (e.g., sum rule, product rule, chain rule; see also Section 5.1.2) still apply. However, when we compute derivatives with respect to vectors  $\mathbf{x} \in \mathbb{R}^n$  we need to pay attention: Our gradients now involve vectors and matrices, and matrix multiplication is not commutative (Section 2.2.1), i.e., the order matters.

Here are the general product rule, sum rule, and chain rule:

$$\text{Product rule:} \quad \frac{\partial}{\partial \mathbf{x}} (f(\mathbf{x})g(\mathbf{x})) = \frac{\partial f}{\partial \mathbf{x}} g(\mathbf{x}) + f(\mathbf{x}) \frac{\partial g}{\partial \mathbf{x}} \quad (5.46)$$

$$\text{Sum rule:} \quad \frac{\partial}{\partial \mathbf{x}} (f(\mathbf{x}) + g(\mathbf{x})) = \frac{\partial f}{\partial \mathbf{x}} + \frac{\partial g}{\partial \mathbf{x}} \quad (5.47)$$

Product rule:

$$(fg)' = f'g + fg',$$

Sum rule:

$$(f + g)' = f' + g',$$

Chain rule:

$$(g(f))' = g'(f)f'$$

Chain rule: 
$$\frac{\partial}{\partial \mathbf{x}}(g \circ f)(\mathbf{x}) = \frac{\partial}{\partial \mathbf{x}}(g(f(\mathbf{x}))) = \frac{\partial g}{\partial f} \frac{\partial f}{\partial \mathbf{x}} \quad (5.48)$$

This is only an intuition, but not mathematically correct since the partial derivative is not a fraction.

Let us have a closer look at the chain rule. The chain rule (5.48) resembles to some degree the rules for matrix multiplication where we said that neighboring dimensions have to match for matrix multiplication to be defined; see Section 2.2.1. If we go from left to right, the chain rule exhibits similar properties:  $\partial f$  shows up in the “denominator” of the first factor and in the “numerator” of the second factor. If we multiply the factors together, multiplication is defined, i.e., the dimensions of  $\partial f$  match, and  $\partial f$  “cancels”, such that  $\partial g / \partial \mathbf{x}$  remains.

### 5.2.2 Chain Rule

Consider a function  $f : \mathbb{R}^2 \rightarrow \mathbb{R}$  of two variables  $x_1, x_2$ . Furthermore,  $x_1(t)$  and  $x_2(t)$  are themselves functions of  $t$ . To compute the gradient of  $f$  with respect to  $t$ , we need to apply the chain rule (5.48) for multivariate functions as

$$\frac{df}{dt} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix} \begin{bmatrix} \frac{\partial x_1(t)}{\partial t} \\ \frac{\partial x_2(t)}{\partial t} \end{bmatrix} = \frac{\partial f}{\partial x_1} \frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2} \frac{\partial x_2}{\partial t}, \quad (5.49)$$

where  $d$  denotes the gradient and  $\partial$  partial derivatives.

#### Example 5.8

Consider  $f(x_1, x_2) = x_1^2 + 2x_2$ , where  $x_1 = \sin t$  and  $x_2 = \cos t$ , then

$$\frac{df}{dt} = \frac{\partial f}{\partial x_1} \frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2} \frac{\partial x_2}{\partial t} \quad (5.50a)$$

$$= 2 \sin t \frac{\partial \sin t}{\partial t} + 2 \frac{\partial \cos t}{\partial t} \quad (5.50b)$$

$$= 2 \sin t \cos t - 2 \sin t = 2 \sin t (\cos t - 1) \quad (5.50c)$$

is the corresponding derivative of  $f$  with respect to  $t$ .

If  $f(x_1, x_2)$  is a function of  $x_1$  and  $x_2$ , where  $x_1(s, t)$  and  $x_2(s, t)$  are themselves functions of two variables  $s$  and  $t$ , the chain rule yields the partial derivatives

$$\frac{\partial f}{\partial s} = \frac{\partial f}{\partial x_1} \frac{\partial x_1}{\partial s} + \frac{\partial f}{\partial x_2} \frac{\partial x_2}{\partial s}, \quad (5.51)$$

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x_1} \frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2} \frac{\partial x_2}{\partial t}, \quad (5.52)$$

and the gradient is obtained by the matrix multiplication

$$\frac{df}{d(s, t)} = \frac{\partial f}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial (s, t)} = \underbrace{\begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix}}_{\frac{\partial f}{\partial \mathbf{x}}} \underbrace{\begin{bmatrix} \frac{\partial x_1}{\partial s} & \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial s} & \frac{\partial x_2}{\partial t} \end{bmatrix}}_{\frac{\partial \mathbf{x}}{\partial (s, t)}}. \quad (5.53)$$

This compact way of writing the chain rule as a matrix multiplication only makes sense if the gradient is defined as a row vector. Otherwise, we will need to start transposing gradients for the matrix dimensions to match. This may still be straightforward as long as the gradient is a vector or a matrix; however, when the gradient becomes a tensor (we will discuss this in the following), the transpose is no longer a triviality.

The chain rule can be written as a matrix multiplication.

*Remark* (Verifying the Correctness of a Gradient Implementation). The definition of the partial derivatives as the limit of the corresponding difference quotient (see (5.39)) can be exploited when numerically checking the correctness of gradients in computer programs: When we compute gradients and implement them, we can use finite differences to numerically test our computation and implementation: We choose the value  $h$  to be small (e.g.,  $h = 10^{-4}$ ) and compare the finite-difference approximation from (5.39) with our (analytic) implementation of the gradient. If the error is small, our gradient implementation is probably correct. “Small” could mean that  $\sqrt{\frac{\sum_i (dh_i - df_i)^2}{\sum_i (dh_i + df_i)^2}} < 10^{-6}$ , where  $dh_i$  is the finite-difference approximation and  $df_i$  is the analytic gradient of  $f$  with respect to the  $i$ th variable  $x_i$ .  $\diamond$

Gradient checking

### 5.3 Gradients of Vector-Valued Functions

Thus far, we discussed partial derivatives and gradients of functions  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  mapping to the real numbers. In the following, we will generalize the concept of the gradient to vector-valued functions (vector fields)  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ , where  $n \geq 1$  and  $m > 1$ .

For a function  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  and a vector  $\mathbf{x} = [x_1, \dots, x_n]^\top \in \mathbb{R}^n$ , the corresponding vector of function values is given as

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{bmatrix} \in \mathbb{R}^m. \quad (5.54)$$

Writing the vector-valued function in this way allows us to view a vector-valued function  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  as a vector of functions  $[f_1, \dots, f_m]^\top$ ,  $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$  that map onto  $\mathbb{R}$ . The differentiation rules for every  $f_i$  are exactly the ones we discussed in Section 5.2.

Therefore, the partial derivative of a vector-valued function  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  with respect to  $x_i \in \mathbb{R}$ ,  $i = 1, \dots, n$ , is given as the vector

$$\frac{\partial \mathbf{f}}{\partial x_i} = \begin{bmatrix} \frac{\partial f_1}{\partial x_i} \\ \vdots \\ \frac{\partial f_m}{\partial x_i} \end{bmatrix} = \begin{bmatrix} \lim_{h \rightarrow 0} \frac{f_1(x_1, \dots, x_{i-1}, x_i+h, x_{i+1}, \dots, x_n) - f_1(\mathbf{x})}{h} \\ \vdots \\ \lim_{h \rightarrow 0} \frac{f_m(x_1, \dots, x_{i-1}, x_i+h, x_{i+1}, \dots, x_n) - f_m(\mathbf{x})}{h} \end{bmatrix} \in \mathbb{R}^m. \quad (5.55)$$

From (5.40), we know that the gradient of  $\mathbf{f}$  with respect to a vector is the row vector of the partial derivatives. In (5.55), every partial derivative  $\partial \mathbf{f} / \partial x_i$  is itself a column vector. Therefore, we obtain the gradient of  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  with respect to  $\mathbf{x} \in \mathbb{R}^n$  by collecting these partial derivatives:

$$\frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}} = \begin{bmatrix} \boxed{\frac{\partial \mathbf{f}(\mathbf{x})}{\partial x_1}} \dots \boxed{\frac{\partial \mathbf{f}(\mathbf{x})}{\partial x_n}} \end{bmatrix} \quad (5.56a)$$

$$= \begin{bmatrix} \boxed{\frac{\partial f_1(\mathbf{x})}{\partial x_1}} \dots \boxed{\frac{\partial f_1(\mathbf{x})}{\partial x_n}} \\ \vdots \\ \boxed{\frac{\partial f_m(\mathbf{x})}{\partial x_1}} \dots \boxed{\frac{\partial f_m(\mathbf{x})}{\partial x_n}} \end{bmatrix} \in \mathbb{R}^{m \times n}. \quad (5.56b)$$

**Definition 5.6** (Jacobian). The collection of all first-order partial derivatives of a vector-valued function  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  is called the *Jacobian*. The Jacobian  $\mathbf{J}$  is an  $m \times n$  matrix, which we define and arrange as follows:

$$\mathbf{J} = \nabla_{\mathbf{x}} \mathbf{f} = \frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial \mathbf{f}(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial \mathbf{f}(\mathbf{x})}{\partial x_n} \end{bmatrix} \quad (5.57)$$

$$= \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial f_1(\mathbf{x})}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial f_m(\mathbf{x})}{\partial x_n} \end{bmatrix}, \quad (5.58)$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad J(i, j) = \frac{\partial f_i}{\partial x_j}. \quad (5.59)$$

As a special case of (5.58), a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}^1$ , which maps a vector  $\mathbf{x} \in \mathbb{R}^n$  onto a scalar (e.g.,  $f(\mathbf{x}) = \sum_{i=1}^n x_i$ ), possesses a Jacobian that is a row vector (matrix of dimension  $1 \times n$ ); see (5.40).

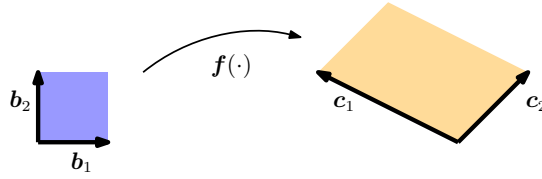
*Remark.* In this book, we use the *numerator layout* of the derivative, i.e., the derivative  $d\mathbf{f}/d\mathbf{x}$  of  $\mathbf{f} \in \mathbb{R}^m$  with respect to  $\mathbf{x} \in \mathbb{R}^n$  is an  $m \times n$  matrix, where the elements of  $\mathbf{f}$  define the rows and the elements of  $\mathbf{x}$  define the columns of the corresponding Jacobian; see (5.58). There

Jacobian

The gradient of a function

$\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  is a matrix of size  $m \times n$ .

numerator layout



**Figure 5.5** The determinant of the Jacobian of  $f$  can be used to compute the magnifier between the blue and orange area.

exists also the *denominator layout*, which is the transpose of the numerator layout. In this book, we will use the numerator layout.  $\diamond$

We will see how the Jacobian is used in the change-of-variable method for probability distributions in Section 6.7. The amount of scaling due to the transformation of a variable is provided by the determinant.

In Section 4.1, we saw that the determinant can be used to compute the area of a parallelogram. If we are given two vectors  $\mathbf{b}_1 = [1, 0]^\top$ ,  $\mathbf{b}_2 = [0, 1]^\top$  as the sides of the unit square (blue; see Figure 5.5), the area of this square is

$$\left| \det \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 1. \quad (5.60)$$

If we take a parallelogram with the sides  $\mathbf{c}_1 = [-2, 1]^\top$ ,  $\mathbf{c}_2 = [1, 1]^\top$  (orange in Figure 5.5), its area is given as the absolute value of the determinant (see Section 4.1)

$$\left| \det \begin{pmatrix} -2 & 1 \\ 1 & 1 \end{pmatrix} \right| = |-3| = 3, \quad (5.61)$$

i.e., the area of this is exactly three times the area of the unit square. We can find this scaling factor by finding a mapping that transforms the unit square into the other square. In linear algebra terms, we effectively perform a variable transformation from  $(\mathbf{b}_1, \mathbf{b}_2)$  to  $(\mathbf{c}_1, \mathbf{c}_2)$ . In our case, the mapping is linear and the absolute value of the determinant of this mapping gives us exactly the scaling factor we are looking for.

We will describe two approaches to identify this mapping. First, we exploit that the mapping is linear so that we can use the tools from Chapter 2 to identify this mapping. Second, we will find the mapping using partial derivatives using the tools we have been discussing in this chapter.

**Approach 1** To get started with the linear algebra approach, we identify both  $\{\mathbf{b}_1, \mathbf{b}_2\}$  and  $\{\mathbf{c}_1, \mathbf{c}_2\}$  as bases of  $\mathbb{R}^2$  (see Section 2.6.1 for a recap). What we effectively perform is a change of basis from  $(\mathbf{b}_1, \mathbf{b}_2)$  to  $(\mathbf{c}_1, \mathbf{c}_2)$ , and we are looking for the transformation matrix that implements the basis change. Using results from Section 2.7.2, we identify the desired basis change matrix as

$$\mathbf{J} = \begin{bmatrix} -2 & 1 \\ 1 & 1 \end{bmatrix}, \quad (5.62)$$

such that  $\mathbf{J}\mathbf{b}_1 = \mathbf{c}_1$  and  $\mathbf{J}\mathbf{b}_2 = \mathbf{c}_2$ . The absolute value of the determi-

denominator layout

nant of  $\mathbf{J}$ , which yields the scaling factor we are looking for, is given as  $|\det(\mathbf{J})| = 3$ , i.e., the area of the square spanned by  $(\mathbf{c}_1, \mathbf{c}_2)$  is three times greater than the area spanned by  $(\mathbf{b}_1, \mathbf{b}_2)$ .

**Approach 2** The linear algebra approach works for linear transformations; for nonlinear transformations (which become relevant in Section 6.7), we follow a more general approach using partial derivatives.

For this approach, we consider a function  $\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  that performs a variable transformation. In our example,  $\mathbf{f}$  maps the coordinate representation of any vector  $\mathbf{x} \in \mathbb{R}^2$  with respect to  $(\mathbf{b}_1, \mathbf{b}_2)$  onto the coordinate representation  $\mathbf{y} \in \mathbb{R}^2$  with respect to  $(\mathbf{c}_1, \mathbf{c}_2)$ . We want to identify the mapping so that we can compute how an area (or volume) changes when it is being transformed by  $\mathbf{f}$ . For this, we need to find out how  $\mathbf{f}(\mathbf{x})$  changes if we modify  $\mathbf{x}$  a bit. This question is exactly answered by the Jacobian matrix  $\frac{d\mathbf{f}}{d\mathbf{x}} \in \mathbb{R}^{2 \times 2}$ . Since we can write

$$y_1 = -2x_1 + x_2 \quad (5.63)$$

$$y_2 = x_1 + x_2 \quad (5.64)$$

we obtain the functional relationship between  $\mathbf{x}$  and  $\mathbf{y}$ , which allows us to get the partial derivatives

$$\frac{\partial y_1}{\partial x_1} = -2, \quad \frac{\partial y_1}{\partial x_2} = 1, \quad \frac{\partial y_2}{\partial x_1} = 1, \quad \frac{\partial y_2}{\partial x_2} = 1 \quad (5.65)$$

and compose the Jacobian as

$$\mathbf{J} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} -2 & 1 \\ 1 & 1 \end{bmatrix}. \quad (5.66)$$

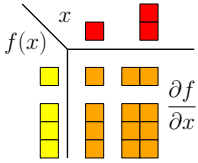
Geometrically, the Jacobian determinant gives the magnification/ scaling factor when we transform an area or volume. Jacobian determinant

The Jacobian represents the coordinate transformation we are looking for. It is exact if the coordinate transformation is linear (as in our case), and (5.66) recovers exactly the basis change matrix in (5.62). If the coordinate transformation is nonlinear, the Jacobian approximates this nonlinear transformation locally with a linear one. The absolute value of the *Jacobian determinant*  $|\det(\mathbf{J})|$  is the factor by which areas or volumes are scaled when coordinates are transformed. Our case yields  $|\det(\mathbf{J})| = 3$ .

The Jacobian determinant and variable transformations will become relevant in Section 6.7 when we transform random variables and probability distributions. These transformations are extremely relevant in machine learning in the context of training deep neural networks using the *reparametrization trick*, also called *infinite perturbation analysis*.

In this chapter, we encountered derivatives of functions. Figure 5.6 summarizes the dimensions of those derivatives. If  $f : \mathbb{R} \rightarrow \mathbb{R}$  the gradient is simply a scalar (top-left entry). For  $f : \mathbb{R}^D \rightarrow \mathbb{R}$  the gradient is a  $1 \times D$  row vector (top-right entry). For  $\mathbf{f} : \mathbb{R} \rightarrow \mathbb{R}^E$ , the gradient is an  $E \times 1$  column vector, and for  $\mathbf{f} : \mathbb{R}^D \rightarrow \mathbb{R}^E$  the gradient is an  $E \times D$  matrix.

**Figure 5.6**  
Dimensionality of  
(partial) derivatives.



**Example 5.9 (Gradient of a Vector-Valued Function)**

We are given

$$\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x}, \quad \mathbf{f}(\mathbf{x}) \in \mathbb{R}^M, \quad \mathbf{A} \in \mathbb{R}^{M \times N}, \quad \mathbf{x} \in \mathbb{R}^N.$$

To compute the gradient  $d\mathbf{f}/d\mathbf{x}$  we first determine the dimension of  $d\mathbf{f}/d\mathbf{x}$ : Since  $\mathbf{f} : \mathbb{R}^N \rightarrow \mathbb{R}^M$ , it follows that  $d\mathbf{f}/d\mathbf{x} \in \mathbb{R}^{M \times N}$ . Second, to compute the gradient we determine the partial derivatives of  $f$  with respect to every  $x_j$ :

$$f_i(\mathbf{x}) = \sum_{j=1}^N A_{ij}x_j \implies \frac{\partial f_i}{\partial x_j} = A_{ij} \quad (5.67)$$

We collect the partial derivatives in the Jacobian and obtain the gradient

$$\frac{d\mathbf{f}}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & & \vdots \\ \frac{\partial f_M}{\partial x_1} & \cdots & \frac{\partial f_M}{\partial x_N} \end{bmatrix} = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & & \vdots \\ A_{M1} & \cdots & A_{MN} \end{bmatrix} = \mathbf{A} \in \mathbb{R}^{M \times N}. \quad (5.68)$$

**Example 5.10 (Chain Rule)**

Consider the function  $h : \mathbb{R} \rightarrow \mathbb{R}$ ,  $h(t) = (f \circ g)(t)$  with

$$f : \mathbb{R}^2 \rightarrow \mathbb{R} \quad (5.69)$$

$$g : \mathbb{R} \rightarrow \mathbb{R}^2 \quad (5.70)$$

$$f(\mathbf{x}) = \exp(x_1 x_2^2), \quad (5.71)$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = g(t) = \begin{bmatrix} t \cos t \\ t \sin t \end{bmatrix} \quad (5.72)$$

and compute the gradient of  $h$  with respect to  $t$ . Since  $f : \mathbb{R}^2 \rightarrow \mathbb{R}$  and  $g : \mathbb{R} \rightarrow \mathbb{R}^2$  we note that

$$\frac{\partial f}{\partial \mathbf{x}} \in \mathbb{R}^{1 \times 2}, \quad \frac{\partial g}{\partial t} \in \mathbb{R}^{2 \times 1}. \quad (5.73)$$

The desired gradient is computed by applying the chain rule:

$$\frac{dh}{dt} = \frac{\partial f}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial t} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix} \begin{bmatrix} \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial t} \end{bmatrix} \quad (5.74a)$$

$$= [\exp(x_1 x_2^2) x_2^2 \quad 2 \exp(x_1 x_2^2) x_1 x_2] \begin{bmatrix} \cos t - t \sin t \\ \sin t + t \cos t \end{bmatrix} \quad (5.74b)$$

$$= \exp(x_1 x_2^2) (x_2^2 (\cos t - t \sin t) + 2x_1 x_2 (\sin t + t \cos t)), \quad (5.74c)$$

where  $x_1 = t \cos t$  and  $x_2 = t \sin t$ ; see (5.72).



We will discuss this model in much more detail in Chapter 9 in the context of linear regression, where we need derivatives of the least-squares loss  $L$  with respect to the parameters  $\theta$ .

least-squares loss

```
dLdtheta =
np.einsum(
'n,nd',
dLde,dedtheta)
```

### Example 5.11 (Gradient of a Least-Squares Loss in a Linear Model)

Let us consider the linear model

$$\mathbf{y} = \Phi \boldsymbol{\theta}, \quad (5.75)$$

where  $\boldsymbol{\theta} \in \mathbb{R}^D$  is a parameter vector,  $\Phi \in \mathbb{R}^{N \times D}$  are input features and  $\mathbf{y} \in \mathbb{R}^N$  are the corresponding observations. We define the functions

$$L(\mathbf{e}) := \|\mathbf{e}\|^2, \quad (5.76)$$

$$\mathbf{e}(\boldsymbol{\theta}) := \mathbf{y} - \Phi \boldsymbol{\theta}. \quad (5.77)$$

We seek  $\frac{\partial L}{\partial \boldsymbol{\theta}}$ , and we will use the chain rule for this purpose.  $L$  is called a *least-squares loss function*.

Before we start our calculation, we determine the dimensionality of the gradient as

$$\frac{\partial L}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{1 \times D}. \quad (5.78)$$

The chain rule allows us to compute the gradient as

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \frac{\partial L}{\partial \mathbf{e}} \frac{\partial \mathbf{e}}{\partial \boldsymbol{\theta}}, \quad (5.79)$$

where the  $d$ th element is given by

$$\frac{\partial L}{\partial \boldsymbol{\theta}}[1, d] = \sum_{n=1}^N \frac{\partial L}{\partial \mathbf{e}}[n] \frac{\partial \mathbf{e}}{\partial \boldsymbol{\theta}}[n, d]. \quad (5.80)$$

We know that  $\|\mathbf{e}\|^2 = \mathbf{e}^\top \mathbf{e}$  (see Section 3.2) and determine

$$\frac{\partial L}{\partial \mathbf{e}} = 2\mathbf{e}^\top \in \mathbb{R}^{1 \times N}. \quad (5.81)$$

Furthermore, we obtain

$$\frac{\partial \mathbf{e}}{\partial \boldsymbol{\theta}} = -\Phi \in \mathbb{R}^{N \times D}, \quad (5.82)$$

such that our desired derivative is

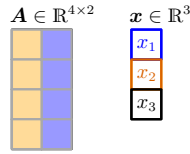
$$\frac{\partial L}{\partial \boldsymbol{\theta}} = -2\mathbf{e}^\top \Phi \stackrel{(5.77)}{=} -\underbrace{2(\mathbf{y}^\top - \boldsymbol{\theta}^\top \Phi^\top)}_{1 \times N} \underbrace{\Phi}_{N \times D} \in \mathbb{R}^{1 \times D}. \quad (5.83)$$

*Remark.* We would have obtained the same result without using the chain rule by immediately looking at the function

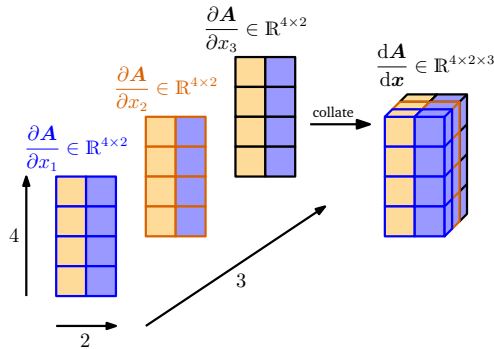
$$L_2(\boldsymbol{\theta}) := \|\mathbf{y} - \Phi \boldsymbol{\theta}\|^2 = (\mathbf{y} - \Phi \boldsymbol{\theta})^\top (\mathbf{y} - \Phi \boldsymbol{\theta}). \quad (5.84)$$

This approach is still practical for simple functions like  $L_2$  but becomes impractical for deep function compositions.  $\diamond$

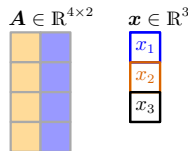
## 5.4 Gradients of Matrices

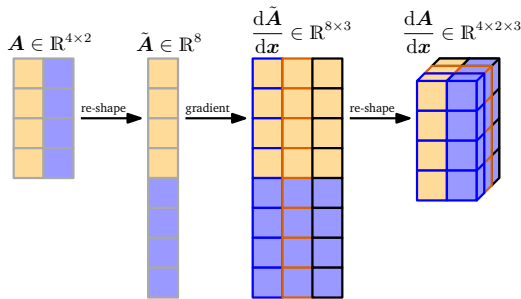
$$\mathbf{A} \in \mathbb{R}^{4 \times 2} \quad \mathbf{x} \in \mathbb{R}^3$$


Partial derivatives:

$$\frac{\partial \mathbf{A}}{\partial x_1} \in \mathbb{R}^{4 \times 2}, \quad \frac{\partial \mathbf{A}}{\partial x_2} \in \mathbb{R}^{4 \times 2}, \quad \frac{\partial \mathbf{A}}{\partial x_3} \in \mathbb{R}^{4 \times 2}$$


(a) Approach 1: We compute the partial derivative  $\frac{\partial \mathbf{A}}{\partial x_1}$ ,  $\frac{\partial \mathbf{A}}{\partial x_2}$ ,  $\frac{\partial \mathbf{A}}{\partial x_3}$ , each of which is a  $4 \times 2$  matrix, and collate them in a  $4 \times 2 \times 3$  tensor.

$$\mathbf{A} \in \mathbb{R}^{4 \times 2} \quad \mathbf{x} \in \mathbb{R}^3$$


$$\mathbf{A} \in \mathbb{R}^{4 \times 2} \rightarrow \tilde{\mathbf{A}} \in \mathbb{R}^8 \xrightarrow{\text{gradient}} \frac{d\tilde{\mathbf{A}}}{d\mathbf{x}} \in \mathbb{R}^{8 \times 3} \xrightarrow{\text{re-shape}} \frac{d\mathbf{A}}{d\mathbf{x}} \in \mathbb{R}^{4 \times 2 \times 3}$$


(b) Approach 2: We re-shape (flatten)  $\mathbf{A} \in \mathbb{R}^{4 \times 2}$  into a vector  $\tilde{\mathbf{A}} \in \mathbb{R}^8$ . Then, we compute the gradient  $\frac{d\tilde{\mathbf{A}}}{d\mathbf{x}} \in \mathbb{R}^{8 \times 3}$ . We obtain the gradient tensor by re-shaping this gradient as illustrated above.

## 5.4 Gradients of Matrices

We will encounter situations where we need to take gradients of matrices with respect to vectors (or other matrices), which results in a multidimensional tensor. We can think of this tensor as a multidimensional array that

155

**Figure 5.7**

Visualization of gradient computation of a matrix with respect to a vector. We are interested in computing the gradient of  $\mathbf{A} \in \mathbb{R}^{4 \times 2}$  with respect to a vector  $\mathbf{x} \in \mathbb{R}^3$ . We know that gradient  $\frac{d\mathbf{A}}{d\mathbf{x}} \in \mathbb{R}^{4 \times 2 \times 3}$ . We follow two equivalent approaches to arrive there: (a) collating partial derivatives into a Jacobian tensor; (b) flattening of the matrix into a vector, computing the Jacobian matrix, re-shaping into a Jacobian tensor.

We can think of a tensor as a multidimensional array.

collects partial derivatives. For example, if we compute the gradient of an  $m \times n$  matrix  $\mathbf{A}$  with respect to a  $p \times q$  matrix  $\mathbf{B}$ , the resulting Jacobian would be  $(m \times n) \times (p \times q)$ , i.e., a four-dimensional tensor  $\mathbf{J}$ , whose entries are given as  $J_{ijkl} = \partial A_{ij} / \partial B_{kl}$ .

Since matrices represent linear mappings, we can exploit the fact that there is a vector-space isomorphism (linear, invertible mapping) between the space  $\mathbb{R}^{m \times n}$  of  $m \times n$  matrices and the space  $\mathbb{R}^{mn}$  of  $mn$  vectors. Therefore, we can re-shape our matrices into vectors of lengths  $mn$  and  $pq$ , respectively. The gradient using these  $mn$  vectors results in a Jacobian of size  $mn \times pq$ . Figure 5.7 visualizes both approaches. In practical applications, it is often desirable to re-shape the matrix into a vector and continue working with this Jacobian matrix: The chain rule (5.48) boils down to simple matrix multiplication, whereas in the case of a Jacobian tensor, we will need to pay more attention to what dimensions we need to sum out.

Matrices can be transformed into vectors by stacking the columns of the matrix (“flattening”).

### Example 5.12 (Gradient of Vectors with Respect to Matrices)

Let us consider the following example, where

$$\mathbf{f} = \mathbf{A}\mathbf{x}, \quad \mathbf{f} \in \mathbb{R}^M, \quad \mathbf{A} \in \mathbb{R}^{M \times N}, \quad \mathbf{x} \in \mathbb{R}^N \quad (5.85)$$

and where we seek the gradient  $d\mathbf{f}/d\mathbf{A}$ . Let us start again by determining the dimension of the gradient as

$$\frac{d\mathbf{f}}{d\mathbf{A}} \in \mathbb{R}^{M \times (M \times N)}. \quad (5.86)$$

By definition, the gradient is the collection of the partial derivatives:

$$\frac{d\mathbf{f}}{d\mathbf{A}} = \begin{bmatrix} \frac{\partial f_1}{\partial \mathbf{A}} \\ \vdots \\ \frac{\partial f_M}{\partial \mathbf{A}} \end{bmatrix}, \quad \frac{\partial f_i}{\partial \mathbf{A}} \in \mathbb{R}^{1 \times (M \times N)}. \quad (5.87)$$

To compute the partial derivatives, it will be helpful to explicitly write out the matrix vector multiplication:

$$f_i = \sum_{j=1}^N A_{ij} x_j, \quad i = 1, \dots, M, \quad (5.88)$$

and the partial derivatives are then given as

$$\frac{\partial f_i}{\partial A_{iq}} = x_q. \quad (5.89)$$

This allows us to compute the partial derivatives of  $f_i$  with respect to a row of  $\mathbf{A}$ , which is given as

$$\frac{\partial f_i}{\partial A_{i,:}} = \mathbf{x}^\top \in \mathbb{R}^{1 \times 1 \times N}, \quad (5.90)$$

$$\frac{\partial f_i}{\partial A_{k \neq i,:}} = \mathbf{0}^\top \in \mathbb{R}^{1 \times 1 \times N} \quad (5.91)$$

where we have to pay attention to the correct dimensionality. Since  $f_i$  maps onto  $\mathbb{R}$  and each row of  $\mathbf{A}$  is of size  $1 \times N$ , we obtain a  $1 \times 1 \times N$ -sized tensor as the partial derivative of  $f_i$  with respect to a row of  $\mathbf{A}$ .

We stack the partial derivatives (5.91) and get the desired gradient in (5.87) via

$$\frac{\partial f_i}{\partial \mathbf{A}} = \begin{bmatrix} \mathbf{0}^\top \\ \vdots \\ \mathbf{0}^\top \\ \mathbf{x}^\top \\ \mathbf{0}^\top \\ \vdots \\ \mathbf{0}^\top \end{bmatrix} \in \mathbb{R}^{1 \times (M \times N)}. \quad (5.92)$$

#### Example 5.13 (Gradient of Matrices with Respect to Matrices)

Consider a matrix  $\mathbf{R} \in \mathbb{R}^{M \times N}$  and  $\mathbf{f} : \mathbb{R}^{M \times N} \rightarrow \mathbb{R}^{N \times N}$  with

$$\mathbf{f}(\mathbf{R}) = \mathbf{R}^\top \mathbf{R} =: \mathbf{K} \in \mathbb{R}^{N \times N}, \quad (5.93)$$

where we seek the gradient  $d\mathbf{K}/d\mathbf{R}$ .

To solve this hard problem, let us first write down what we already know: The gradient has the dimensions

$$\frac{d\mathbf{K}}{d\mathbf{R}} \in \mathbb{R}^{(N \times N) \times (M \times N)}, \quad (5.94)$$

which is a tensor. Moreover,

$$\frac{dK_{pq}}{d\mathbf{R}} \in \mathbb{R}^{1 \times M \times N} \quad (5.95)$$

for  $p, q = 1, \dots, N$ , where  $K_{pq}$  is the  $(p, q)$ th entry of  $\mathbf{K} = \mathbf{f}(\mathbf{R})$ . Denoting the  $i$ th column of  $\mathbf{R}$  by  $\mathbf{r}_i$ , every entry of  $\mathbf{K}$  is given by the dot product of two columns of  $\mathbf{R}$ , i.e.,

$$K_{pq} = \mathbf{r}_p^\top \mathbf{r}_q = \sum_{m=1}^M R_{mp} R_{mq}. \quad (5.96)$$

When we now compute the partial derivative  $\frac{\partial K_{pq}}{\partial R_{ij}}$  we obtain

$$\frac{\partial K_{pq}}{\partial R_{ij}} = \sum_{m=1}^M \frac{\partial}{\partial R_{ij}} R_{mp} R_{mq} = \partial_{pqij}, \quad (5.97)$$

$$\partial_{pqij} = \begin{cases} R_{iq} & \text{if } j = p, p \neq q \\ R_{ip} & \text{if } j = q, p \neq q \\ 2R_{iq} & \text{if } j = p, p = q \\ 0 & \text{otherwise} \end{cases} \quad (5.98)$$

From (5.94), we know that the desired gradient has the dimension  $(N \times N) \times (M \times N)$ , and every single entry of this tensor is given by  $\partial_{pqij}$  in (5.98), where  $p, q, j = 1, \dots, N$  and  $i = 1, \dots, M$ .

### 5.5 Useful Identities for Computing Gradients

In the following, we list some useful gradients that are frequently required in a machine learning context (Petersen and Pedersen, 2012). Here, we use  $\text{tr}(\cdot)$  as the trace (see Definition 4.4),  $\det(\cdot)$  as the determinant (see Section 4.1) and  $\mathbf{f}(\mathbf{X})^{-1}$  as the inverse of  $\mathbf{f}(\mathbf{X})$ , assuming it exists.

$$\frac{\partial}{\partial \mathbf{X}} \mathbf{f}(\mathbf{X})^\top = \left( \frac{\partial \mathbf{f}(\mathbf{X})}{\partial \mathbf{X}} \right)^\top \quad (5.99)$$

$$\frac{\partial}{\partial \mathbf{X}} \text{tr}(\mathbf{f}(\mathbf{X})) = \text{tr} \left( \frac{\partial \mathbf{f}(\mathbf{X})}{\partial \mathbf{X}} \right) \quad (5.100)$$

$$\frac{\partial}{\partial \mathbf{X}} \det(\mathbf{f}(\mathbf{X})) = \det(\mathbf{f}(\mathbf{X})) \text{tr} \left( \mathbf{f}(\mathbf{X})^{-1} \frac{\partial \mathbf{f}(\mathbf{X})}{\partial \mathbf{X}} \right) \quad (5.101)$$

$$\frac{\partial}{\partial \mathbf{X}} \mathbf{f}(\mathbf{X})^{-1} = -\mathbf{f}(\mathbf{X})^{-1} \frac{\partial \mathbf{f}(\mathbf{X})}{\partial \mathbf{X}} \mathbf{f}(\mathbf{X})^{-1} \quad (5.102)$$

$$\frac{\partial \mathbf{a}^\top \mathbf{X}^{-1} \mathbf{b}}{\partial \mathbf{X}} = -(\mathbf{X}^{-1})^\top \mathbf{a} \mathbf{b}^\top (\mathbf{X}^{-1})^\top \quad (5.103)$$

$$\frac{\partial \mathbf{x}^\top \mathbf{a}}{\partial \mathbf{x}} = \mathbf{a}^\top \quad (5.104)$$

$$\frac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}^\top \quad (5.105)$$

$$\frac{\partial \mathbf{a}^\top \mathbf{X} \mathbf{b}}{\partial \mathbf{X}} = \mathbf{a} \mathbf{b}^\top \quad (5.106)$$

$$\frac{\partial \mathbf{x}^\top \mathbf{B} \mathbf{x}}{\partial \mathbf{x}} = \mathbf{x}^\top (\mathbf{B} + \mathbf{B}^\top) \quad (5.107)$$

$$\frac{\partial}{\partial \mathbf{s}} (\mathbf{x} - \mathbf{A} \mathbf{s})^\top \mathbf{W} (\mathbf{x} - \mathbf{A} \mathbf{s}) = -2(\mathbf{x} - \mathbf{A} \mathbf{s})^\top \mathbf{W} \mathbf{A} \quad \text{for symmetric } \mathbf{W} \quad (5.108)$$

*Remark.* In this book, we only cover traces and transposes of matrices. However, we have seen that derivatives can be higher-dimensional tensors, in which case the usual trace and transpose are not defined. In these cases, the trace of a  $D \times D \times E \times F$  tensor would be an  $E \times F$ -dimensional matrix. This is a special case of a tensor contraction. Similarly, when we

“transpose” a tensor, we mean swapping the first two dimensions. Specifically, in (5.99) through (5.102), we require tensor-related computations when we work with multivariate functions  $\mathbf{f}(\cdot)$  and compute derivatives with respect to matrices (and choose not to vectorize them as discussed in Section 5.4).  $\diamond$

## 5.6 Backpropagation and Automatic Differentiation

In many machine learning applications, we find good model parameters by performing gradient descent (Section 7.1), which relies on the fact that we can compute the gradient of a learning objective with respect to the parameters of the model. For a given objective function, we can obtain the gradient with respect to the model parameters using calculus and applying the chain rule; see Section 5.2.2. We already had a taste in Section 5.3 when we looked at the gradient of a squared loss with respect to the parameters of a linear regression model.

Consider the function

$$f(x) = \sqrt{x^2 + \exp(x^2)} + \cos(x^2 + \exp(x^2)). \quad (5.109)$$

By application of the chain rule, and noting that differentiation is linear, we compute the gradient

$$\begin{aligned} \frac{df}{dx} &= \frac{2x + 2x \exp(x^2)}{2\sqrt{x^2 + \exp(x^2)}} - \sin(x^2 + \exp(x^2)) (2x + 2x \exp(x^2)) \\ &= 2x \left( \frac{1}{2\sqrt{x^2 + \exp(x^2)}} - \sin(x^2 + \exp(x^2)) \right) (1 + \exp(x^2)). \end{aligned} \quad (5.110)$$

Writing out the gradient in this explicit way is often impractical since it often results in a very lengthy expression for a derivative. In practice, it means that, if we are not careful, the implementation of the gradient could be significantly more expensive than computing the function, which imposes unnecessary overhead. For training deep neural network models, the *backpropagation* algorithm (Kelley, 1960; Bryson, 1961; Dreyfus, 1962; Rumelhart et al., 1986) is an efficient way to compute the gradient of an error function with respect to the parameters of the model.

A good discussion about backpropagation and the chain rule is available at a blog by Tim Vieira at <https://tinyurl.com/yccfm2yrw>.

backpropagation

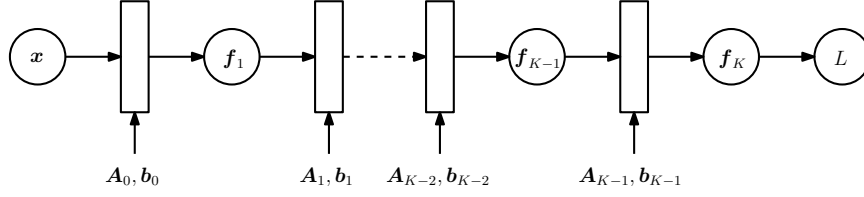
### 5.6.1 Gradients in a Deep Network

An area where the chain rule is used to an extreme is deep learning, where the function value  $\mathbf{y}$  is computed as a many-level function composition

$$\mathbf{y} = (f_K \circ f_{K-1} \circ \cdots \circ f_1)(\mathbf{x}) = f_K(f_{K-1}(\cdots (f_1(\mathbf{x})) \cdots)), \quad (5.111)$$

where  $\mathbf{x}$  are the inputs (e.g., images),  $\mathbf{y}$  are the observations (e.g., class labels), and every function  $f_i$ ,  $i = 1, \dots, K$ , possesses its own parameters.

**Figure 5.8** Forward pass in a multi-layer neural network to compute the loss  $L$  as a function of the inputs  $\mathbf{x}$  and the parameters  $\mathbf{A}_i$ ,  $\mathbf{b}_i$ .



We discuss the case, where the activation functions are identical in each layer to unclutter notation.

In neural networks with multiple layers, we have functions  $f_i(\mathbf{x}_{i-1}) = \sigma(\mathbf{A}_{i-1}\mathbf{x}_{i-1} + \mathbf{b}_{i-1})$  in the  $i$ th layer. Here  $\mathbf{x}_{i-1}$  is the output of layer  $i-1$  and  $\sigma$  an activation function, such as the logistic sigmoid  $\frac{1}{1+e^{-x}}$ , tanh or a rectified linear unit (ReLU). In order to train these models, we require the gradient of a loss function  $L$  with respect to all model parameters  $\mathbf{A}_j$ ,  $\mathbf{b}_j$  for  $j = 1, \dots, K$ . This also requires us to compute the gradient of  $L$  with respect to the inputs of each layer. For example, if we have inputs  $\mathbf{x}$  and observations  $\mathbf{y}$  and a network structure defined by

$$\mathbf{f}_0 := \mathbf{x} \quad (5.112)$$

$$\mathbf{f}_i := \sigma_i(\mathbf{A}_{i-1}\mathbf{f}_{i-1} + \mathbf{b}_{i-1}), \quad i = 1, \dots, K, \quad (5.113)$$

see also Figure 5.8 for a visualization, we may be interested in finding  $\mathbf{A}_j$ ,  $\mathbf{b}_j$  for  $j = 0, \dots, K-1$ , such that the squared loss

$$L(\boldsymbol{\theta}) = \|\mathbf{y} - \mathbf{f}_K(\boldsymbol{\theta}, \mathbf{x})\|^2 \quad (5.114)$$

is minimized, where  $\boldsymbol{\theta} = \{\mathbf{A}_0, \mathbf{b}_0, \dots, \mathbf{A}_{K-1}, \mathbf{b}_{K-1}\}$ .

To obtain the gradients with respect to the parameter set  $\boldsymbol{\theta}$ , we require the partial derivatives of  $L$  with respect to the parameters  $\boldsymbol{\theta}_j = \{\mathbf{A}_j, \mathbf{b}_j\}$  of each layer  $j = 0, \dots, K-1$ . The chain rule allows us to determine the partial derivatives as

$$\frac{\partial L}{\partial \boldsymbol{\theta}_{K-1}} = \frac{\partial L}{\partial \mathbf{f}_K} \frac{\partial \mathbf{f}_K}{\partial \boldsymbol{\theta}_{K-1}} \quad (5.115)$$

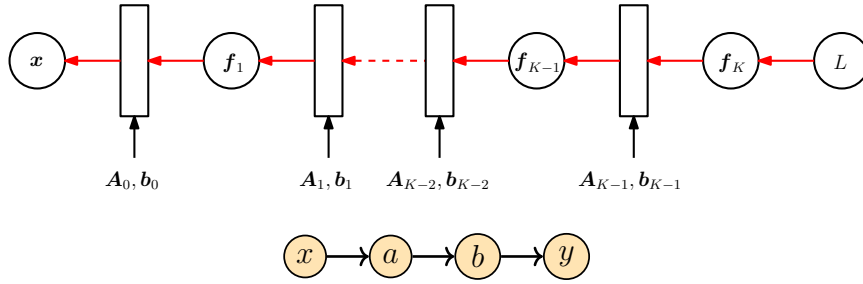
$$\frac{\partial L}{\partial \boldsymbol{\theta}_{K-2}} = \frac{\partial L}{\partial \mathbf{f}_K} \boxed{\frac{\partial \mathbf{f}_K}{\partial \mathbf{f}_{K-1}} \frac{\partial \mathbf{f}_{K-1}}{\partial \boldsymbol{\theta}_{K-2}}} \quad (5.116)$$

$$\frac{\partial L}{\partial \boldsymbol{\theta}_{K-3}} = \frac{\partial L}{\partial \mathbf{f}_K} \frac{\partial \mathbf{f}_K}{\partial \mathbf{f}_{K-1}} \boxed{\frac{\partial \mathbf{f}_{K-1}}{\partial \mathbf{f}_{K-2}} \frac{\partial \mathbf{f}_{K-2}}{\partial \boldsymbol{\theta}_{K-3}}} \quad (5.117)$$

$$\frac{\partial L}{\partial \boldsymbol{\theta}_i} = \frac{\partial L}{\partial \mathbf{f}_K} \frac{\partial \mathbf{f}_K}{\partial \mathbf{f}_{K-1}} \dots \boxed{\frac{\partial \mathbf{f}_{i+2}}{\partial \mathbf{f}_{i+1}} \frac{\partial \mathbf{f}_{i+1}}{\partial \boldsymbol{\theta}_i}} \quad (5.118)$$

The **orange** terms are partial derivatives of the output of a layer with respect to its inputs, whereas the **blue** terms are partial derivatives of the output of a layer with respect to its parameters. Assuming, we have already computed the partial derivatives  $\partial L / \partial \boldsymbol{\theta}_{i+1}$ , then most of the computation can be reused to compute  $\partial L / \partial \boldsymbol{\theta}_i$ . The additional terms that we

A more in-depth discussion about gradients of neural networks can be found in Justin Domke's lecture notes <https://tinyurl.com/yalcxgtv>.



**Figure 5.9** Backward pass in a multi-layer neural network to compute the gradients of the loss function.

**Figure 5.10** Simple graph illustrating the flow of data from  $x$  to  $y$  via some intermediate variables  $a, b$ .

need to compute are indicated by the boxes. Figure 5.9 visualizes that the gradients are passed backward through the network.

### 5.6.2 Automatic Differentiation

It turns out that backpropagation is a special case of a general technique in numerical analysis called *automatic differentiation*. We can think of automatic differentiation as a set of techniques to numerically (in contrast to symbolically) evaluate the exact (up to machine precision) gradient of a function by working with intermediate variables and applying the chain rule. Automatic differentiation applies a series of elementary arithmetic operations, e.g., addition and multiplication and elementary functions, e.g.,  $\sin$ ,  $\cos$ ,  $\exp$ ,  $\log$ . By applying the chain rule to these operations, the gradient of quite complicated functions can be computed automatically. Automatic differentiation applies to general computer programs and has forward and reverse modes. Baydin et al. (2018) give a great overview of automatic differentiation in machine learning.

Figure 5.10 shows a simple graph representing the data flow from inputs  $x$  to outputs  $y$  via some intermediate variables  $a, b$ . If we were to compute the derivative  $dy/dx$ , we would apply the chain rule and obtain

$$\frac{dy}{dx} = \frac{dy}{db} \frac{db}{da} \frac{da}{dx}. \quad (5.119)$$

Intuitively, the forward and reverse mode differ in the order of multiplication. Due to the associativity of matrix multiplication, we can choose between

$$\frac{dy}{dx} = \left( \frac{dy}{db} \frac{db}{da} \right) \frac{da}{dx}, \quad (5.120)$$

$$\frac{dy}{dx} = \frac{dy}{db} \left( \frac{db}{da} \frac{da}{dx} \right). \quad (5.121)$$

Equation (5.120) would be the *reverse mode* because gradients are propagated backward through the graph, i.e., reverse to the data flow. Equation (5.121) would be the *forward mode*, where the gradients flow with the data from left to right through the graph.

automatic differentiation

Automatic differentiation is different from symbolic differentiation and numerical approximations of the gradient, e.g., by using finite differences.

In the general case, we work with Jacobians, which can be vectors, matrices, or tensors.

reverse mode

forward mode



In the following, we will focus on reverse mode automatic differentiation, which is backpropagation. In the context of neural networks, where the input dimensionality is often much higher than the dimensionality of the labels, the reverse mode is computationally significantly cheaper than the forward mode. Let us start with an instructive example.

### Example 5.14

Consider the function

$$f(x) = \sqrt{x^2 + \exp(x^2)} + \cos(x^2 + \exp(x^2)) \quad (5.122)$$

from (5.109). If we were to implement a function  $f$  on a computer, we would be able to save some computation by using *intermediate variables*:

$$a = x^2, \quad (5.123)$$

$$b = \exp(a), \quad (5.124)$$

$$c = a + b, \quad (5.125)$$

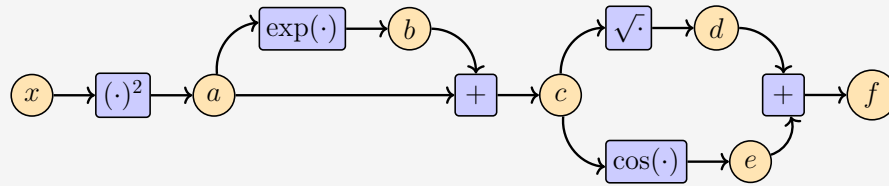
$$d = \sqrt{c}, \quad (5.126)$$

$$e = \cos(c), \quad (5.127)$$

$$f = d + e. \quad (5.128)$$

intermediate  
variables

**Figure 5.11**  
Computation graph  
with inputs  $x$ ,  
function values  $f$ ,  
and intermediate  
variables  $a, b, c, d, e$ .



This is the same kind of thinking process that occurs when applying the chain rule. Note that the preceding set of equations requires fewer operations than a direct implementation of the function  $f(x)$  as defined in (5.109). The corresponding computation graph in Figure 5.11 shows the flow of data and computations required to obtain the function value  $f$ .

The set of equations that include intermediate variables can be thought of as a computation graph, a representation that is widely used in implementations of neural network software libraries. We can directly compute the derivatives of the intermediate variables with respect to their corresponding inputs by recalling the definition of the derivative of elementary functions. We obtain the following:

$$\frac{\partial a}{\partial x} = 2x \quad (5.129)$$

$$\frac{\partial b}{\partial a} = \exp(a) \quad (5.130)$$

$$\frac{\partial c}{\partial a} = 1 = \frac{\partial c}{\partial b} \quad (5.131)$$

$$\frac{\partial d}{\partial c} = \frac{1}{2\sqrt{c}} \quad (5.132)$$

$$\frac{\partial e}{\partial c} = -\sin(c) \quad (5.133)$$

$$\frac{\partial f}{\partial d} = 1 = \frac{\partial f}{\partial e}. \quad (5.134)$$

By looking at the computation graph in Figure 5.11, we can compute  $\partial f/\partial x$  by working backward from the output and obtain

$$\frac{\partial f}{\partial c} = \frac{\partial f}{\partial d} \frac{\partial d}{\partial c} + \frac{\partial f}{\partial e} \frac{\partial e}{\partial c} \quad (5.135)$$

$$\frac{\partial f}{\partial b} = \frac{\partial f}{\partial c} \frac{\partial c}{\partial b} \quad (5.136)$$

$$\frac{\partial f}{\partial a} = \frac{\partial f}{\partial b} \frac{\partial b}{\partial a} + \frac{\partial f}{\partial c} \frac{\partial c}{\partial a} \quad (5.137)$$

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial a} \frac{\partial a}{\partial x}. \quad (5.138)$$

Note that we implicitly applied the chain rule to obtain  $\partial f/\partial x$ . By substituting the results of the derivatives of the elementary functions, we get

$$\frac{\partial f}{\partial c} = 1 \cdot \frac{1}{2\sqrt{c}} + 1 \cdot (-\sin(c)) \quad (5.139)$$

$$\frac{\partial f}{\partial b} = \frac{\partial f}{\partial c} \cdot 1 \quad (5.140)$$

$$\frac{\partial f}{\partial a} = \frac{\partial f}{\partial b} \exp(a) + \frac{\partial f}{\partial c} \cdot 1 \quad (5.141)$$

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial a} \cdot 2x. \quad (5.142)$$

By thinking of each of the derivatives above as a variable, we observe that the computation required for calculating the derivative is of similar complexity as the computation of the function itself. This is quite counter-intuitive since the mathematical expression for the derivative  $\frac{\partial f}{\partial x}$  (5.110) is significantly more complicated than the mathematical expression of the function  $f(x)$  in (5.109).

Automatic differentiation is a formalization of Example 5.14. Let  $x_1, \dots, x_d$  be the input variables to the function,  $x_{d+1}, \dots, x_{D-1}$  be the intermediate variables, and  $x_D$  the output variable. Then the computation graph can be expressed as follows:

$$\text{For } i = d+1, \dots, D: \quad x_i = g_i(x_{\text{Pa}(x_i)}), \quad (5.143)$$

where the  $g_i(\cdot)$  are elementary functions and  $x_{\text{Pa}(x_i)}$  are the parent nodes of the variable  $x_i$  in the graph. Given a function defined in this way, we can use the chain rule to compute the derivative of the function in a step-by-step fashion. Recall that by definition  $f = x_D$  and hence

$$\frac{\partial f}{\partial x_D} = 1. \quad (5.144)$$

For other variables  $x_i$ , we apply the chain rule

$$\frac{\partial f}{\partial x_i} = \sum_{x_j: x_i \in \text{Pa}(x_j)} \frac{\partial f}{\partial x_j} \frac{\partial x_j}{\partial x_i} = \sum_{x_j: x_i \in \text{Pa}(x_j)} \frac{\partial f}{\partial x_j} \frac{\partial g_j}{\partial x_i}, \quad (5.145)$$

where  $\text{Pa}(x_j)$  is the set of parent nodes of  $x_j$  in the computation graph. Equation (5.143) is the forward propagation of a function, whereas (5.145) is the backpropagation of the gradient through the computation graph. For neural network training, we backpropagate the error of the prediction with respect to the label.

The automatic differentiation approach above works whenever we have a function that can be expressed as a computation graph, where the elementary functions are differentiable. In fact, the function may not even be a mathematical function but a computer program. However, not all computer programs can be automatically differentiated, e.g., if we cannot find differential elementary functions. Programming structures, such as for loops and if statements, require more care as well.

Auto-differentiation in reverse mode requires a parse tree.

## 5.7 Higher-Order Derivatives

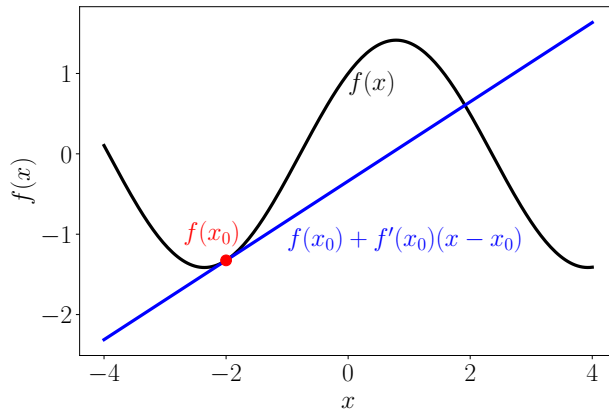
So far, we have discussed gradients, i.e., first-order derivatives. Sometimes, we are interested in derivatives of higher order, e.g., when we want to use Newton's Method for optimization, which requires second-order derivatives (Nocedal and Wright, 2006). In Section 5.1.1, we discussed the Taylor series to approximate functions using polynomials. In the multivariate case, we can do exactly the same. In the following, we will do exactly this. But let us start with some notation.

Consider a function  $f : \mathbb{R}^2 \rightarrow \mathbb{R}$  of two variables  $x, y$ . We use the following notation for higher-order partial derivatives (and for gradients):

- $\frac{\partial^2 f}{\partial x^2}$  is the second partial derivative of  $f$  with respect to  $x$ .
- $\frac{\partial^n f}{\partial x^n}$  is the  $n$ th partial derivative of  $f$  with respect to  $x$ .
- $\frac{\partial^2 f}{\partial y \partial x} = \frac{\partial}{\partial y} \left( \frac{\partial f}{\partial x} \right)$  is the partial derivative obtained by first partial differentiating with respect to  $x$  and then with respect to  $y$ .
- $\frac{\partial^2 f}{\partial x \partial y}$  is the partial derivative obtained by first partial differentiating by  $y$  and then  $x$ .

Hessian

The *Hessian* is the collection of all second-order partial derivatives.



**Figure 5.12** Linear approximation of a function. The original function  $f$  is linearized at  $x_0 = -2$  using a first-order Taylor series expansion.

If  $f(x, y)$  is a twice (continuously) differentiable function, then

$$\frac{\partial^2 f}{\partial x \partial y} = \frac{\partial^2 f}{\partial y \partial x}, \quad (5.146)$$

i.e., the order of differentiation does not matter, and the corresponding *Hessian matrix*

Hessian matrix

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} \quad (5.147)$$

is symmetric. The Hessian is denoted as  $\nabla_{x,y}^2 f(x, y)$ . Generally, for  $\mathbf{x} \in \mathbb{R}^n$  and  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , the Hessian is an  $n \times n$  matrix. The Hessian measures the curvature of the function locally around  $(x, y)$ .

*Remark* (Hessian of a Vector Field). If  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  is a vector field, the Hessian is an  $(m \times n \times n)$ -tensor.  $\diamond$

## 5.8 Linearization and Multivariate Taylor Series

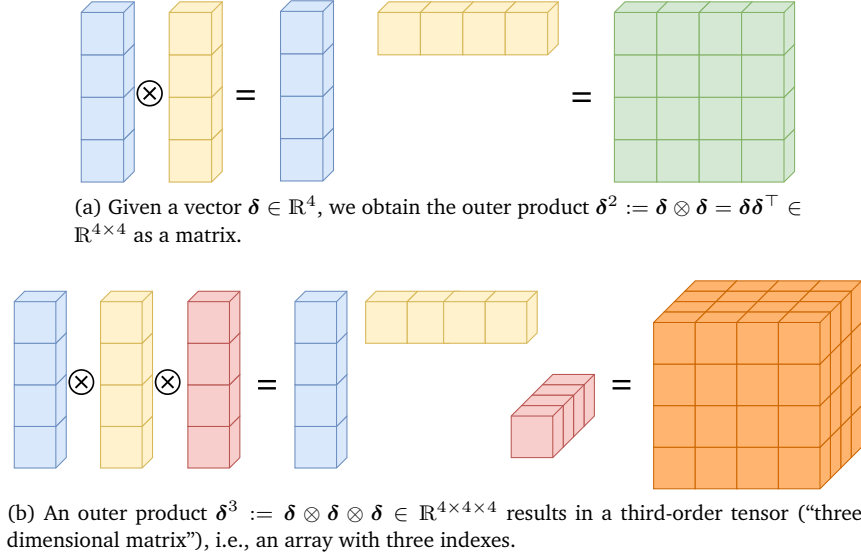
The gradient  $\nabla f$  of a function  $f$  is often used for a locally linear approximation of  $f$  around  $\mathbf{x}_0$ :

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + (\nabla_{\mathbf{x}} f)(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0). \quad (5.148)$$

Here  $(\nabla_{\mathbf{x}} f)(\mathbf{x}_0)$  is the gradient of  $f$  with respect to  $\mathbf{x}$ , evaluated at  $\mathbf{x}_0$ . Figure 5.12 illustrates the linear approximation of a function  $f$  at an input  $x_0$ . The original function is approximated by a straight line. This approximation is locally accurate, but the farther we move away from  $x_0$  the worse the approximation gets. Equation (5.148) is a special case of a multivariate Taylor series expansion of  $f$  at  $\mathbf{x}_0$ , where we consider only the first two terms. We discuss the more general case in the following, which will allow for better approximations.

**Figure 5.13**

Visualizing outer products. Outer products of vectors increase the dimensionality of the array by 1 per term. (a) The outer product of two vectors results in a matrix; (b) the outer product of three vectors yields a third-order tensor.



**Definition 5.7** (Multivariate Taylor Series). We consider a function

$$f : \mathbb{R}^D \rightarrow \mathbb{R} \quad (5.149)$$

$$\mathbf{x} \mapsto f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^D, \quad (5.150)$$

that is smooth at  $\mathbf{x}_0$ . When we define the difference vector  $\delta := \mathbf{x} - \mathbf{x}_0$ , the *multivariate Taylor series* of  $f$  at  $(\mathbf{x}_0)$  is defined as

$$f(\mathbf{x}) = \sum_{k=0}^{\infty} \frac{D_{\mathbf{x}}^k f(\mathbf{x}_0)}{k!} \delta^k, \quad (5.151)$$

where  $D_{\mathbf{x}}^k f(\mathbf{x}_0)$  is the  $k$ -th (total) derivative of  $f$  with respect to  $\mathbf{x}$ , evaluated at  $\mathbf{x}_0$ .

**Definition 5.8** (Taylor Polynomial). The *Taylor polynomial* of degree  $n$  of  $f$  at  $\mathbf{x}_0$  contains the first  $n + 1$  components of the series in (5.151) and is defined as

$$T_n(\mathbf{x}) = \sum_{k=0}^n \frac{D_{\mathbf{x}}^k f(\mathbf{x}_0)}{k!} \delta^k. \quad (5.152)$$

In (5.151) and (5.152), we used the slightly sloppy notation of  $\delta^k$ , which is not defined for vectors  $\mathbf{x} \in \mathbb{R}^D$ ,  $D > 1$ , and  $k > 1$ . Note that both  $D_{\mathbf{x}}^k f$  and  $\delta^k$  are  $k$ -th order tensors, i.e.,  $k$ -dimensional arrays. The

$k$ th-order tensor  $\delta^k \in \mathbb{R}^{\overbrace{D \times D \times \dots \times D}^{k \text{ times}}}$  is obtained as a  $k$ -fold outer product, denoted by  $\otimes$ , of the vector  $\delta \in \mathbb{R}^D$ . For example,

$$\delta^2 := \delta \otimes \delta = \delta \delta^\top, \quad \delta^2[i, j] = \delta[i] \delta[j] \quad (5.153)$$

multivariate Taylor series

Taylor polynomial

A vector can be implemented as a one-dimensional array, a matrix as a two-dimensional array.

$$\delta^3 := \delta \otimes \delta \otimes \delta, \quad \delta^3[i, j, k] = \delta[i]\delta[j]\delta[k]. \quad (5.154)$$

Figure 5.13 visualizes two such outer products. In general, we obtain the terms

$$D_{\mathbf{x}}^k f(\mathbf{x}_0) \delta^k = \sum_{i_1=1}^D \cdots \sum_{i_k=1}^D D_{\mathbf{x}}^k f(\mathbf{x}_0)[i_1, \dots, i_k] \delta[i_1] \cdots \delta[i_k] \quad (5.155)$$

in the Taylor series, where  $D_{\mathbf{x}}^k f(\mathbf{x}_0) \delta^k$  contains  $k$ -th order polynomials.

Now that we defined the Taylor series for vector fields, let us explicitly write down the first terms  $D_{\mathbf{x}}^k f(\mathbf{x}_0) \delta^k$  of the Taylor series expansion for  $k = 0, \dots, 3$  and  $\delta := \mathbf{x} - \mathbf{x}_0$ :

$$k = 0 : D_{\mathbf{x}}^0 f(\mathbf{x}_0) \delta^0 = f(\mathbf{x}_0) \in \mathbb{R} \quad (5.156)$$

$$k = 1 : D_{\mathbf{x}}^1 f(\mathbf{x}_0) \delta^1 = \underbrace{\nabla_{\mathbf{x}} f(\mathbf{x}_0)}_{1 \times D} \underbrace{\delta}_{D \times 1} = \sum_{i=1}^D \nabla_{\mathbf{x}} f(\mathbf{x}_0)[i] \delta[i] \in \mathbb{R} \quad (5.157)$$

$$k = 2 : D_{\mathbf{x}}^2 f(\mathbf{x}_0) \delta^2 = \text{tr} \left( \underbrace{\mathbf{H}(\mathbf{x}_0)}_{D \times D} \underbrace{\delta}_{D \times 1} \underbrace{\delta^{\top}}_{1 \times D} \right) = \delta^{\top} \mathbf{H}(\mathbf{x}_0) \delta \quad (5.158)$$

$$= \sum_{i=1}^D \sum_{j=1}^D H[i, j] \delta[i] \delta[j] \in \mathbb{R} \quad (5.159)$$

$$k = 3 : D_{\mathbf{x}}^3 f(\mathbf{x}_0) \delta^3 = \sum_{i=1}^D \sum_{j=1}^D \sum_{k=1}^D D_{\mathbf{x}}^3 f(\mathbf{x}_0)[i, j, k] \delta[i] \delta[j] \delta[k] \in \mathbb{R} \quad (5.160)$$

Here,  $\mathbf{H}(\mathbf{x}_0)$  is the Hessian of  $f$  evaluated at  $\mathbf{x}_0$ .

#### Example 5.15 (Taylor Series Expansion of a Function with Two Variables)

Consider the function

$$f(x, y) = x^2 + 2xy + y^3. \quad (5.161)$$

We want to compute the Taylor series expansion of  $f$  at  $(x_0, y_0) = (1, 2)$ . Before we start, let us discuss what to expect: The function in (5.161) is a polynomial of degree 3. We are looking for a Taylor series expansion, which itself is a linear combination of polynomials. Therefore, we do not expect the Taylor series expansion to contain terms of fourth or higher order to express a third-order polynomial. This means that it should be sufficient to determine the first four terms of (5.151) for an exact alternative representation of (5.161).

To determine the Taylor series expansion, we start with the constant term and the first-order derivatives, which are given by

$$f(1, 2) = 13 \quad (5.162)$$

```
np.einsum('i,i',Df1,d)
np.einsum('ij,i,j',Df2,d,d)
np.einsum('ijk,i,j,k',Df3,d,d,d)
```

$$\frac{\partial f}{\partial x} = 2x + 2y \implies \frac{\partial f}{\partial x}(1, 2) = 6 \quad (5.163)$$

$$\frac{\partial f}{\partial y} = 2x + 3y^2 \implies \frac{\partial f}{\partial y}(1, 2) = 14. \quad (5.164)$$

Therefore, we obtain

$$D_{x,y}^1 f(1, 2) = \nabla_{x,y} f(1, 2) = \begin{bmatrix} \frac{\partial f}{\partial x}(1, 2) & \frac{\partial f}{\partial y}(1, 2) \end{bmatrix} = \begin{bmatrix} 6 & 14 \end{bmatrix} \in \mathbb{R}^{1 \times 2} \quad (5.165)$$

such that

$$\frac{D_{x,y}^1 f(1, 2)}{1!} \boldsymbol{\delta} = \begin{bmatrix} 6 & 14 \end{bmatrix} \begin{bmatrix} x-1 \\ y-2 \end{bmatrix} = 6(x-1) + 14(y-2). \quad (5.166)$$

Note that  $D_{x,y}^1 f(1, 2) \boldsymbol{\delta}$  contains only linear terms, i.e., first-order polynomials.

The second-order partial derivatives are given by

$$\frac{\partial^2 f}{\partial x^2} = 2 \implies \frac{\partial^2 f}{\partial x^2}(1, 2) = 2 \quad (5.167)$$

$$\frac{\partial^2 f}{\partial y^2} = 6y \implies \frac{\partial^2 f}{\partial y^2}(1, 2) = 12 \quad (5.168)$$

$$\frac{\partial^2 f}{\partial y \partial x} = 2 \implies \frac{\partial^2 f}{\partial y \partial x}(1, 2) = 2 \quad (5.169)$$

$$\frac{\partial^2 f}{\partial x \partial y} = 2 \implies \frac{\partial^2 f}{\partial x \partial y}(1, 2) = 2. \quad (5.170)$$

When we collect the second-order partial derivatives, we obtain the Hessian

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 2 & 12 \end{bmatrix}, \quad (5.171)$$

such that

$$\mathbf{H}(1, 2) = \begin{bmatrix} 2 & 2 \\ 2 & 12 \end{bmatrix} \in \mathbb{R}^{2 \times 2}. \quad (5.172)$$

Therefore, the next term of the Taylor-series expansion is given by

$$\frac{D_{x,y}^2 f(1, 2)}{2!} \boldsymbol{\delta}^2 = \frac{1}{2} \boldsymbol{\delta}^\top \mathbf{H}(1, 2) \boldsymbol{\delta} \quad (5.173a)$$

$$= \frac{1}{2} \begin{bmatrix} x-1 & y-2 \end{bmatrix} \begin{bmatrix} 2 & 2 \\ 2 & 12 \end{bmatrix} \begin{bmatrix} x-1 \\ y-2 \end{bmatrix} \quad (5.173b)$$

$$= (x-1)^2 + 2(x-1)(y-2) + 6(y-2)^2. \quad (5.173c)$$

Here,  $D_{x,y}^2 f(1, 2) \boldsymbol{\delta}^2$  contains only quadratic terms, i.e., second-order polynomials.

The third-order derivatives are obtained as

$$D_{x,y}^3 f = \begin{bmatrix} \frac{\partial \mathbf{H}}{\partial x} & \frac{\partial \mathbf{H}}{\partial y} \end{bmatrix} \in \mathbb{R}^{2 \times 2 \times 2}, \quad (5.174)$$

$$D_{x,y}^3 f[:, :, 1] = \frac{\partial \mathbf{H}}{\partial x} = \begin{bmatrix} \frac{\partial^3 f}{\partial x^3} & \frac{\partial^3 f}{\partial x^2 \partial y} \\ \frac{\partial^3 f}{\partial x \partial y \partial x} & \frac{\partial^3 f}{\partial x \partial y^2} \end{bmatrix}, \quad (5.175)$$

$$D_{x,y}^3 f[:, :, 2] = \frac{\partial \mathbf{H}}{\partial y} = \begin{bmatrix} \frac{\partial^3 f}{\partial y \partial x^2} & \frac{\partial^3 f}{\partial y \partial x \partial y} \\ \frac{\partial^3 f}{\partial y^2 \partial x} & \frac{\partial^3 f}{\partial y^3} \end{bmatrix}. \quad (5.176)$$

Since most second-order partial derivatives in the Hessian in (5.171) are constant, the only nonzero third-order partial derivative is

$$\frac{\partial^3 f}{\partial y^3} = 6 \implies \frac{\partial^3 f}{\partial y^3}(1, 2) = 6. \quad (5.177)$$

Higher-order derivatives and the mixed derivatives of degree 3 (e.g.,  $\frac{\partial^3 f}{\partial x^2 \partial y}$ ) vanish, such that

$$D_{x,y}^3 f[:, :, 1] = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad D_{x,y}^3 f[:, :, 2] = \begin{bmatrix} 0 & 0 \\ 0 & 6 \end{bmatrix} \quad (5.178)$$

and

$$\frac{D_{x,y}^3 f(1, 2)}{3!} \boldsymbol{\delta}^3 = (y - 2)^3, \quad (5.179)$$

which collects all cubic terms of the Taylor series. Overall, the (exact) Taylor series expansion of  $f$  at  $(x_0, y_0) = (1, 2)$  is

$$f(x) = f(1, 2) + D_{x,y}^1 f(1, 2) \boldsymbol{\delta} + \frac{D_{x,y}^2 f(1, 2)}{2!} \boldsymbol{\delta}^2 + \frac{D_{x,y}^3 f(1, 2)}{3!} \boldsymbol{\delta}^3 \quad (5.180a)$$

$$\begin{aligned} &= f(1, 2) + \frac{\partial f(1, 2)}{\partial x}(x - 1) + \frac{\partial f(1, 2)}{\partial y}(y - 2) \\ &\quad + \frac{1}{2!} \left( \frac{\partial^2 f(1, 2)}{\partial x^2}(x - 1)^2 + \frac{\partial^2 f(1, 2)}{\partial y^2}(y - 2)^2 \right. \\ &\quad \left. + 2 \frac{\partial^2 f(1, 2)}{\partial x \partial y}(x - 1)(y - 2) \right) + \frac{1}{6} \frac{\partial^3 f(1, 2)}{\partial y^3}(y - 2)^3 \end{aligned} \quad (5.180b)$$

$$\begin{aligned} &= 13 + 6(x - 1) + 14(y - 2) \\ &\quad + (x - 1)^2 + 6(y - 2)^2 + 2(x - 1)(y - 2) + (y - 2)^3. \end{aligned} \quad (5.180c)$$

In this case, we obtained an exact Taylor series expansion of the polynomial in (5.161), i.e., the polynomial in (5.180c) is identical to the original polynomial in (5.161). In this particular example, this result is not surprising since the original function was a third-order polynomial, which we expressed through a linear combination of constant terms, first-order, second-order, and third-order polynomials in (5.180c).



## 5.9 Further Reading

Further details of matrix differentials, along with a short review of the required linear algebra, can be found in Magnus and Neudecker (2007). Automatic differentiation has had a long history, and we refer to Griewank and Walther (2003), Griewank and Walther (2008), and Elliott (2009) and the references therein.

In machine learning (and other disciplines), we often need to compute expectations, i.e., we need to solve integrals of the form

$$\mathbb{E}_{\mathbf{x}}[f(\mathbf{x})] = \int f(\mathbf{x})p(\mathbf{x})d\mathbf{x}. \quad (5.181)$$

Even if  $p(\mathbf{x})$  is in a convenient form (e.g., Gaussian), this integral generally cannot be solved analytically. The Taylor series expansion of  $f$  is one way of finding an approximate solution: Assuming  $p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  is Gaussian, then the first-order Taylor series expansion around  $\boldsymbol{\mu}$  locally linearizes the nonlinear function  $f$ . For linear functions, we can compute the mean (and the covariance) exactly if  $p(\mathbf{x})$  is Gaussian distributed (see Section 6.5). This property is heavily exploited by the *extended Kalman filter* (Maybeck, 1979) for online state estimation in nonlinear dynamical systems (also called “state-space models”). Other deterministic ways to approximate the integral in (5.181) are the *unscented transform* (Julier and Uhlmann, 1997), which does not require any gradients, or the *Laplace approximation* (MacKay, 2003; Bishop, 2006; Murphy, 2012), which uses a second-order Taylor series expansion (requiring the Hessian) for a local Gaussian approximation of  $p(\mathbf{x})$  around its mode.

extended Kalman  
filter

unscented transform  
Laplace  
approximation

## Exercises

5.1 Compute the derivative  $f'(x)$  for

$$f(x) = \log(x^4) \sin(x^3).$$

5.2 Compute the derivative  $f'(x)$  of the logistic sigmoid

$$f(x) = \frac{1}{1 + \exp(-x)}.$$

5.3 Compute the derivative  $f'(x)$  of the function

$$f(x) = \exp\left(-\frac{1}{2\sigma^2}(x - \mu)^2\right),$$

where  $\mu, \sigma \in \mathbb{R}$  are constants.

5.4 Compute the Taylor polynomials  $T_n$ ,  $n = 0, \dots, 5$  of  $f(x) = \sin(x) + \cos(x)$  at  $x_0 = 0$ .

5.5 Consider the following functions:

$$f_1(\mathbf{x}) = \sin(x_1) \cos(x_2), \quad \mathbf{x} \in \mathbb{R}^2$$

$$f_2(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$$

$$f_3(\mathbf{x}) = \mathbf{x}\mathbf{x}^\top, \quad \mathbf{x} \in \mathbb{R}^n$$

- a. What are the dimensions of  $\frac{\partial f_i}{\partial \mathbf{x}}$ ?
- b. Compute the Jacobians.

5.6 Differentiate  $f$  with respect to  $\mathbf{t}$  and  $g$  with respect to  $\mathbf{X}$ , where

$$\begin{aligned} f(\mathbf{t}) &= \sin(\log(\mathbf{t}^\top \mathbf{t})), & \mathbf{t} &\in \mathbb{R}^D \\ g(\mathbf{X}) &= \text{tr}(\mathbf{A}\mathbf{X}\mathbf{B}), & \mathbf{A} &\in \mathbb{R}^{D \times E}, \mathbf{X} \in \mathbb{R}^{E \times F}, \mathbf{B} \in \mathbb{R}^{F \times D}, \end{aligned}$$

where  $\text{tr}(\cdot)$  denotes the trace.

5.7 Compute the derivatives  $\text{d}f/\text{d}\mathbf{x}$  of the following functions by using the chain rule. Provide the dimensions of every single partial derivative. Describe your steps in detail.

a.

$$f(z) = \log(1 + z), \quad z = \mathbf{x}^\top \mathbf{x}, \quad \mathbf{x} \in \mathbb{R}^D$$

b.

$$f(\mathbf{z}) = \sin(\mathbf{z}), \quad \mathbf{z} = \mathbf{A}\mathbf{x} + \mathbf{b}, \quad \mathbf{A} \in \mathbb{R}^{E \times D}, \mathbf{x} \in \mathbb{R}^D, \mathbf{b} \in \mathbb{R}^E$$

where  $\sin(\cdot)$  is applied to every element of  $\mathbf{z}$ .

5.8 Compute the derivatives  $\text{d}f/\text{d}\mathbf{x}$  of the following functions. Describe your steps in detail.

- a. Use the chain rule. Provide the dimensions of every single partial derivative.

$$\begin{aligned} f(z) &= \exp(-\tfrac{1}{2}z) \\ z &= g(\mathbf{y}) = \mathbf{y}^\top \mathbf{S}^{-1} \mathbf{y} \\ \mathbf{y} &= h(\mathbf{x}) = \mathbf{x} - \boldsymbol{\mu} \end{aligned}$$

where  $\mathbf{x}, \boldsymbol{\mu} \in \mathbb{R}^D$ ,  $\mathbf{S} \in \mathbb{R}^{D \times D}$ .

b.

$$f(\mathbf{x}) = \text{tr}(\mathbf{x}\mathbf{x}^\top + \sigma^2 \mathbf{I}), \quad \mathbf{x} \in \mathbb{R}^D$$

Here  $\text{tr}(\mathbf{A})$  is the trace of  $\mathbf{A}$ , i.e., the sum of the diagonal elements  $A_{ii}$ .  
*Hint: Explicitly write out the outer product.*

- c. Use the chain rule. Provide the dimensions of every single partial derivative. You do not need to compute the product of the partial derivatives explicitly.

$$\begin{aligned} \mathbf{f} &= \tanh(\mathbf{z}) \in \mathbb{R}^M \\ \mathbf{z} &= \mathbf{A}\mathbf{x} + \mathbf{b}, \quad \mathbf{x} \in \mathbb{R}^N, \mathbf{A} \in \mathbb{R}^{M \times N}, \mathbf{b} \in \mathbb{R}^M. \end{aligned}$$

Here,  $\tanh$  is applied to every component of  $\mathbf{z}$ .

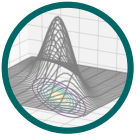
5.9 We define

$$\begin{aligned} g(\mathbf{z}, \boldsymbol{\nu}) &:= \log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}, \boldsymbol{\nu}) \\ \mathbf{z} &:= t(\boldsymbol{\epsilon}, \boldsymbol{\nu}) \end{aligned}$$

for differentiable functions  $p, q, t$ , and  $\mathbf{x} \in \mathbb{R}^D, \mathbf{z} \in \mathbb{R}^E, \boldsymbol{\nu} \in \mathbb{R}^F, \boldsymbol{\epsilon} \in \mathbb{R}^G$ . By using the chain rule, compute the gradient

$$\frac{\text{d}}{\text{d}\boldsymbol{\nu}} g(\mathbf{z}, \boldsymbol{\nu}).$$

## Probability and Distributions



random variable

probability  
distribution

Probability, loosely speaking, concerns the study of uncertainty. Probability can be thought of as the fraction of times an event occurs, or as a degree of belief about an event. We then would like to use this probability to measure the chance of something occurring in an experiment. As mentioned in Chapter 1, we often quantify uncertainty in the data, uncertainty in the machine learning model, and uncertainty in the predictions produced by the model. Quantifying uncertainty requires the idea of a *random variable*, which is a function that maps outcomes of random experiments to a set of properties that we are interested in. Associated with the random variable is a function that measures the probability that a particular outcome (or set of outcomes) will occur; this is called the *probability distribution*.

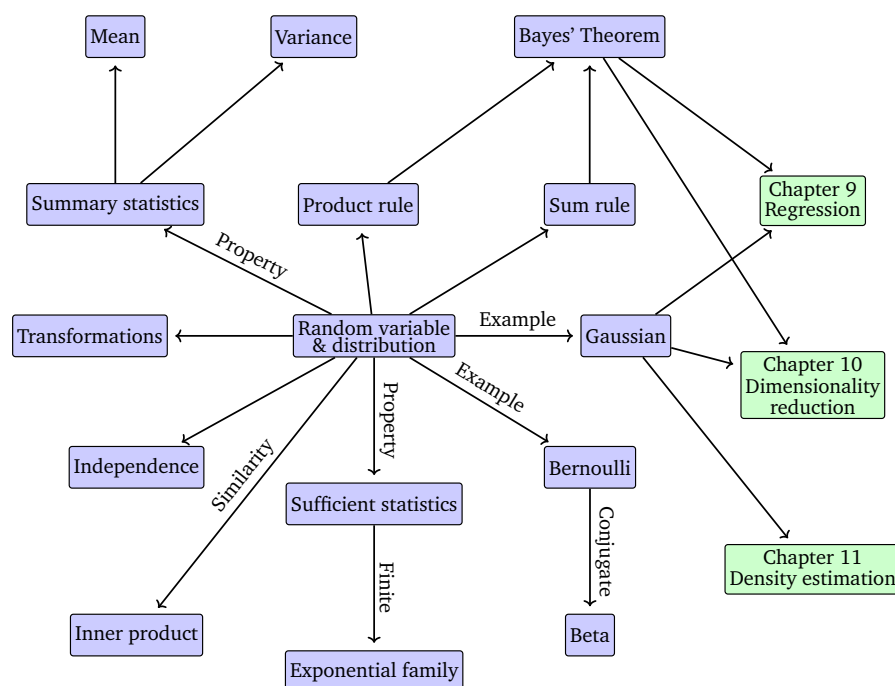
Probability distributions are used as a building block for other concepts, such as probabilistic modeling (Section 8.4), graphical models (Section 8.5), and model selection (Section 8.6). In the next section, we present the three concepts that define a probability space (the sample space, the events, and the probability of an event) and how they are related to a fourth concept called the random variable. The presentation is deliberately slightly hand wavy since a rigorous presentation may occlude the intuition behind the concepts. An outline of the concepts presented in this chapter are shown in Figure 6.1.

### 6.1 Construction of a Probability Space

The theory of probability aims at defining a mathematical structure to describe random outcomes of experiments. For example, when tossing a single coin, we cannot determine the outcome, but by doing a large number of coin tosses, we can observe a regularity in the average outcome. Using this mathematical structure of probability, the goal is to perform automated reasoning, and in this sense, probability generalizes logical reasoning (Jaynes, 2003).

#### 6.1.1 Philosophical Issues

When constructing automated reasoning systems, classical Boolean logic does not allow us to express certain forms of plausible reasoning. Consider



**Figure 6.1** A mind map of the concepts related to random variables and probability distributions, as described in this chapter.

the following scenario: We observe that  $A$  is false. We find  $B$  becomes less plausible, although no conclusion can be drawn from classical logic. We observe that  $B$  is true. It seems  $A$  becomes more plausible. We use this form of reasoning daily. We are waiting for a friend, and consider three possibilities:  $H_1$ , she is on time;  $H_2$ , she has been delayed by traffic; and  $H_3$ , she has been abducted by aliens. When we observe our friend is late, we must logically rule out  $H_1$ . We also tend to consider  $H_2$  to be more likely, though we are not logically required to do so. Finally, we may consider  $H_3$  to be possible, but we continue to consider it quite unlikely. How do we conclude  $H_2$  is the most plausible answer? Seen in this way, probability theory can be considered a generalization of Boolean logic. In the context of machine learning, it is often applied in this way to formalize the design of automated reasoning systems. Further arguments about how probability theory is the foundation of reasoning systems can be found in Pearl (1988).

The philosophical basis of probability and how it should be somehow related to what we think should be true (in the logical sense) was studied by Cox (Jaynes, 2003). Another way to think about it is that if we are precise about our common sense we end up constructing probabilities. E. T. Jaynes (1922–1998) identified three mathematical criteria, which must apply to all plausibilities:

1. The degrees of plausibility are represented by real numbers.
2. These numbers must be based on the rules of common sense.

“For plausible reasoning it is necessary to extend the discrete true and false values of truth to continuous plausibilities” (Jaynes, 2003).

3. The resulting reasoning must be consistent, with the three following meanings of the word “consistent”:
  - (a) Consistency or non-contradiction: When the same result can be reached through different means, the same plausibility value must be found in all cases.
  - (b) Honesty: All available data must be taken into account.
  - (c) Reproducibility: If our state of knowledge about two problems are the same, then we must assign the same degree of plausibility to both of them.

The Cox–Jaynes theorem proves these plausibilities to be sufficient to define the universal mathematical rules that apply to plausibility  $p$ , up to transformation by an arbitrary monotonic function. Crucially, these rules *are* the rules of probability.

*Remark.* In machine learning and statistics, there are two major interpretations of probability: the Bayesian and frequentist interpretations (Bishop, 2006; Efron and Hastie, 2016). The Bayesian interpretation uses probability to specify the degree of uncertainty that the user has about an event. It is sometimes referred to as “subjective probability” or “degree of belief”. The frequentist interpretation considers the relative frequencies of events of interest to the total number of events that occurred. The probability of an event is defined as the relative frequency of the event in the limit when one has infinite data.  $\diamond$

Some machine learning texts on probabilistic models use lazy notation and jargon, which is confusing. This text is no exception. Multiple distinct concepts are all referred to as “probability distribution”, and the reader has to often disentangle the meaning from the context. One trick to help make sense of probability distributions is to check whether we are trying to model something categorical (a discrete random variable) or something continuous (a continuous random variable). The kinds of questions we tackle in machine learning are closely related to whether we are considering categorical or continuous models.

### 6.1.2 Probability and Random Variables

There are three distinct ideas that are often confused when discussing probabilities. First is the idea of a probability space, which allows us to quantify the idea of a probability. However, we mostly do not work directly with this basic probability space. Instead, we work with random variables (the second idea), which transfers the probability to a more convenient (often numerical) space. The third idea is the idea of a distribution or law associated with a random variable. We will introduce the first two ideas in this section and expand on the third idea in Section 6.2.

Modern probability is based on a set of axioms proposed by Kolmogorov

(Grinstead and Snell, 1997; Jaynes, 2003) that introduce the three concepts of sample space, event space, and probability measure. The probability space models a real-world process (referred to as an experiment) with random outcomes.

### The sample space $\Omega$

The *sample space* is the set of all possible outcomes of the experiment, usually denoted by  $\Omega$ . For example, two successive coin tosses have a sample space of {hh, tt, ht, th}, where “h” denotes “heads” and “t” denotes “tails”.

sample space

### The event space $\mathcal{A}$

The *event space* is the space of potential results of the experiment. A subset  $A$  of the sample space  $\Omega$  is in the event space  $\mathcal{A}$  if at the end of the experiment we can observe whether a particular outcome  $\omega \in \Omega$  is in  $A$ . The event space  $\mathcal{A}$  is obtained by considering the collection of subsets of  $\Omega$ , and for discrete probability distributions (Section 6.2.1)  $\mathcal{A}$  is often the power set of  $\Omega$ .

event space

### The probability $P$

With each event  $A \in \mathcal{A}$ , we associate a number  $P(A)$  that measures the probability or degree of belief that the event will occur.  $P(A)$  is called the *probability* of  $A$ .

probability

The probability of a single event must lie in the interval  $[0, 1]$ , and the total probability over all outcomes in the sample space  $\Omega$  must be 1, i.e.,  $P(\Omega) = 1$ . Given a probability space  $(\Omega, \mathcal{A}, P)$ , we want to use it to model some real-world phenomenon. In machine learning, we often avoid explicitly referring to the probability space, but instead refer to probabilities on quantities of interest, which we denote by  $\mathcal{T}$ . In this book, we refer to  $\mathcal{T}$  as the *target space* and refer to elements of  $\mathcal{T}$  as states. We introduce a function  $X : \Omega \rightarrow \mathcal{T}$  that takes an element of  $\Omega$  (an outcome) and returns a particular quantity of interest  $x$ , a value in  $\mathcal{T}$ . This association/mapping from  $\Omega$  to  $\mathcal{T}$  is called a *random variable*. For example, in the case of tossing two coins and counting the number of heads, a random variable  $X$  maps to the three possible outcomes:  $X(\text{hh}) = 2$ ,  $X(\text{ht}) = 1$ ,  $X(\text{th}) = 1$ , and  $X(\text{tt}) = 0$ . In this particular case,  $\mathcal{T} = \{0, 1, 2\}$ , and it is the probabilities on elements of  $\mathcal{T}$  that we are interested in. For a finite sample space  $\Omega$  and finite  $\mathcal{T}$ , the function corresponding to a random variable is essentially a lookup table. For any subset  $S \subseteq \mathcal{T}$ , we associate  $P_X(S) \in [0, 1]$  (the probability) to a particular event occurring corresponding to the random variable  $X$ . Example 6.1 provides a concrete illustration of the terminology.

target space

random variable

The name “random variable” is a great source of misunderstanding as it is neither random nor is it a variable. It is a function.

*Remark.* The aforementioned sample space  $\Omega$  unfortunately is referred to by different names in different books. Another common name for  $\Omega$  is “state space” (Jacod and Protter, 2004), but state space is sometimes reserved for referring to states in a dynamical system (Hasselblatt and

Katok, 2003). Other names sometimes used to describe  $\Omega$  are: “sample description space”, “possibility space,” and “event space”.  $\diamond$

This toy example is essentially a biased coin flip example.

### Example 6.1

We assume that the reader is already familiar with computing probabilities of intersections and unions of sets of events. A gentler introduction to probability with many examples can be found in chapter 2 of Walpole et al. (2011).

Consider a statistical experiment where we model a funfair game consisting of drawing two coins from a bag (with replacement). There are coins from USA (denoted as \$) and UK (denoted as £) in the bag, and since we draw two coins from the bag, there are four outcomes in total. The state space or sample space  $\Omega$  of this experiment is then (\$, \$), (\$, £), (£, \$), (£, £). Let us assume that the composition of the bag of coins is such that a draw returns at random a \$ with probability 0.3.

The event we are interested in is the total number of times the repeated draw returns \$. Let us define a random variable  $X$  that maps the sample space  $\Omega$  to  $\mathcal{T}$ , which denotes the number of times we draw \$ out of the bag. We can see from the preceding sample space we can get zero \$, one \$, or two \$s, and therefore  $\mathcal{T} = \{0, 1, 2\}$ . The random variable  $X$  (a function or lookup table) can be represented as a table like the following:

$$X((\$,\$)) = 2 \quad (6.1)$$

$$X((\$,\pounds)) = 1 \quad (6.2)$$

$$X((\pounds,\$)) = 1 \quad (6.3)$$

$$X((\pounds,\pounds)) = 0. \quad (6.4)$$

Since we return the first coin we draw before drawing the second, this implies that the two draws are independent of each other, which we will discuss in Section 6.4.5. Note that there are two experimental outcomes, which map to the same event, where only one of the draws returns \$. Therefore, the probability mass function (Section 6.2.1) of  $X$  is given by

$$\begin{aligned} P(X = 2) &= P((\$,\$)) \\ &= P(\$) \cdot P(\$) \\ &= 0.3 \cdot 0.3 = 0.09 \end{aligned} \quad (6.5)$$

$$\begin{aligned} P(X = 1) &= P((\$,\pounds) \cup (\pounds,\$)) \\ &= P((\$,\pounds)) + P((\pounds,\$)) \\ &= 0.3 \cdot (1 - 0.3) + (1 - 0.3) \cdot 0.3 = 0.42 \end{aligned} \quad (6.6)$$

$$\begin{aligned} P(X = 0) &= P((\pounds,\pounds)) \\ &= P(\pounds) \cdot P(\pounds) \\ &= (1 - 0.3) \cdot (1 - 0.3) = 0.49. \end{aligned} \quad (6.7)$$

In the calculation, we equated two different concepts, the probability of the output of  $X$  and the probability of the samples in  $\Omega$ . For example, in (6.7) we say  $P(X = 0) = P((\mathcal{L}, \mathcal{L}))$ . Consider the random variable  $X : \Omega \rightarrow \mathcal{T}$  and a subset  $S \subseteq \mathcal{T}$  (for example, a single element of  $\mathcal{T}$ , such as the outcome that one head is obtained when tossing two coins). Let  $X^{-1}(S)$  be the pre-image of  $S$  by  $X$ , i.e., the set of elements of  $\Omega$  that map to  $S$  under  $X$ ;  $\{\omega \in \Omega : X(\omega) \in S\}$ . One way to understand the transformation of probability from events in  $\Omega$  via the random variable  $X$  is to associate it with the probability of the pre-image of  $S$  (Jacod and Protter, 2004). For  $S \subseteq \mathcal{T}$ , we have the notation

$$P_X(S) = P(X \in S) = P(X^{-1}(S)) = P(\{\omega \in \Omega : X(\omega) \in S\}). \quad (6.8)$$

The left-hand side of (6.8) is the probability of the set of possible outcomes (e.g., number of \$ = 1) that we are interested in. Via the random variable  $X$ , which maps states to outcomes, we see in the right-hand side of (6.8) that this is the probability of the set of states (in  $\Omega$ ) that have the property (e.g., \$ $\mathcal{L}$ ,  $\mathcal{L}$ \$). We say that a random variable  $X$  is distributed according to a particular probability distribution  $P_X$ , which defines the probability mapping between the event and the probability of the outcome of the random variable. In other words, the function  $P_X$  or equivalently  $P \circ X^{-1}$  is the *law* or *distribution* of random variable  $X$ .

law  
distribution

*Remark.* The target space, that is, the range  $\mathcal{T}$  of the random variable  $X$ , is used to indicate the kind of probability space, i.e., a  $\mathcal{T}$  random variable. When  $\mathcal{T}$  is finite or countably infinite, this is called a discrete random variable (Section 6.2.1). For continuous random variables (Section 6.2.2), we only consider  $\mathcal{T} = \mathbb{R}$  or  $\mathcal{T} = \mathbb{R}^D$ .  $\diamond$

### 6.1.3 Statistics

Probability theory and statistics are often presented together, but they concern different aspects of uncertainty. One way of contrasting them is by the kinds of problems that are considered. Using probability, we can consider a model of some process, where the underlying uncertainty is captured by random variables, and we use the rules of probability to derive what happens. In statistics, we observe that something has happened and try to figure out the underlying process that explains the observations. In this sense, machine learning is close to statistics in its goals to construct a model that adequately represents the process that generated the data. We can use the rules of probability to obtain a “best-fitting” model for some data.

Another aspect of machine learning systems is that we are interested in generalization error (see Chapter 8). This means that we are actually interested in the performance of our system on instances that we will observe in future, which are not identical to the instances that we have



seen so far. This analysis of future performance relies on probability and statistics, most of which is beyond what will be presented in this chapter. The interested reader is encouraged to look at the books by Boucheron et al. (2013) and Shalev-Shwartz and Ben-David (2014). We will see more about statistics in Chapter 8.

## 6.2 Discrete and Continuous Probabilities

Let us focus our attention on ways to describe the probability of an event as introduced in Section 6.1. Depending on whether the target space is discrete or continuous, the natural way to refer to distributions is different. When the target space  $\mathcal{T}$  is discrete, we can specify the probability that a random variable  $X$  takes a particular value  $x \in \mathcal{T}$ , denoted as  $P(X = x)$ . The expression  $P(X = x)$  for a discrete random variable  $X$  is known as the *probability mass function*. When the target space  $\mathcal{T}$  is continuous, e.g., the real line  $\mathbb{R}$ , it is more natural to specify the probability that a random variable  $X$  is in an interval, denoted by  $P(a \leq X \leq b)$  for  $a < b$ . By convention, we specify the probability that a random variable  $X$  is less than a particular value  $x$ , denoted by  $P(X \leq x)$ . The expression  $P(X \leq x)$  for a continuous random variable  $X$  is known as the *cumulative distribution function*. We will discuss continuous random variables in Section 6.2.2. We will revisit the nomenclature and contrast discrete and continuous random variables in Section 6.2.3.

probability mass  
function

cumulative  
distribution function

univariate

multivariate

*Remark.* We will use the phrase *univariate* distribution to refer to distributions of a single random variable (whose states are denoted by non-bold  $x$ ). We will refer to distributions of more than one random variable as *multivariate* distributions, and will usually consider a vector of random variables (whose states are denoted by bold  $\mathbf{x}$ ).  $\diamond$

### 6.2.1 Discrete Probabilities

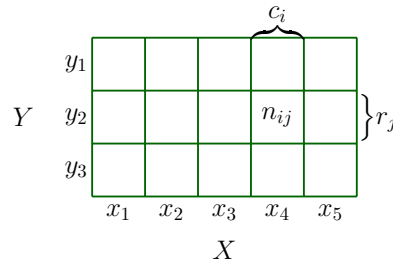
When the target space is discrete, we can imagine the probability distribution of multiple random variables as filling out a (multidimensional) array of numbers. Figure 6.2 shows an example. The target space of the joint probability is the Cartesian product of the target spaces of each of the random variables. We define the *joint probability* as the entry of both values jointly

joint probability

$$P(X = x_i, Y = y_j) = \frac{n_{ij}}{N}, \quad (6.9)$$

where  $n_{ij}$  is the number of events with state  $x_i$  and  $y_j$  and  $N$  the total number of events. The joint probability is the probability of the intersection of both events, that is,  $P(X = x_i, Y = y_j) = P(X = x_i \cap Y = y_j)$ . Figure 6.2 illustrates the *probability mass function* (pmf) of a discrete probability distribution. For two random variables  $X$  and  $Y$ , the probability

probability mass  
function



**Figure 6.2**  
Visualization of a discrete bivariate probability mass function, with random variables  $X$  and  $Y$ . This diagram is adapted from Bishop (2006).

that  $X = x$  and  $Y = y$  is (lazily) written as  $p(x, y)$  and is called the joint probability. One can think of a probability as a function that takes state  $x$  and  $y$  and returns a real number, which is the reason we write  $p(x, y)$ . The *marginal probability* that  $X$  takes the value  $x$  irrespective of the value of random variable  $Y$  is (lazily) written as  $p(x)$ . We write  $X \sim p(x)$  to denote that the random variable  $X$  is distributed according to  $p(x)$ . If we consider only the instances where  $X = x$ , then the fraction of instances (the *conditional probability*) for which  $Y = y$  is written (lazily) as  $p(y | x)$ .

marginal probability

conditional probability

### Example 6.2

Consider two random variables  $X$  and  $Y$ , where  $X$  has five possible states and  $Y$  has three possible states, as shown in Figure 6.2. We denote by  $n_{ij}$  the number of events with state  $X = x_i$  and  $Y = y_j$ , and denote by  $N$  the total number of events. The value  $c_i$  is the sum of the individual frequencies for the  $i$ th column, that is,  $c_i = \sum_{j=1}^3 n_{ij}$ . Similarly, the value  $r_j$  is the row sum, that is,  $r_j = \sum_{i=1}^5 n_{ij}$ . Using these definitions, we can compactly express the distribution of  $X$  and  $Y$ .

The probability distribution of each random variable, the marginal probability, can be seen as the sum over a row or column

$$P(X = x_i) = \frac{c_i}{N} = \frac{\sum_{j=1}^3 n_{ij}}{N} \quad (6.10)$$

and

$$P(Y = y_j) = \frac{r_j}{N} = \frac{\sum_{i=1}^5 n_{ij}}{N}, \quad (6.11)$$

where  $c_i$  and  $r_j$  are the  $i$ th column and  $j$ th row of the probability table, respectively. By convention, for discrete random variables with a finite number of events, we assume that probabilities sum up to one, that is,

$$\sum_{i=1}^5 P(X = x_i) = 1 \quad \text{and} \quad \sum_{j=1}^3 P(Y = y_j) = 1. \quad (6.12)$$

The conditional probability is the fraction of a row or column in a par-

ticular cell. For example, the conditional probability of  $Y$  given  $X$  is

$$P(Y = y_j | X = x_i) = \frac{n_{ij}}{c_i}, \quad (6.13)$$

and the conditional probability of  $X$  given  $Y$  is

$$P(X = x_i | Y = y_j) = \frac{n_{ij}}{r_j}. \quad (6.14)$$

categorical variable

In machine learning, we use discrete probability distributions to model *categorical variables*, i.e., variables that take a finite set of unordered values. They could be categorical features, such as the degree taken at university when used for predicting the salary of a person, or categorical labels, such as letters of the alphabet when doing handwriting recognition. Discrete distributions are also often used to construct probabilistic models that combine a finite number of continuous distributions (Chapter 11).

### 6.2.2 Continuous Probabilities

We consider real-valued random variables in this section, i.e., we consider target spaces that are intervals of the real line  $\mathbb{R}$ . In this book, we pretend that we can perform operations on real random variables as if we have discrete probability spaces with finite states. However, this simplification is not precise for two situations: when we repeat something infinitely often, and when we want to draw a point from an interval. The first situation arises when we discuss generalization errors in machine learning (Chapter 8). The second situation arises when we want to discuss continuous distributions, such as the Gaussian (Section 6.5). For our purposes, the lack of precision allows for a briefer introduction to probability.

measure

Borel  $\sigma$ -algebra

*Remark.* In continuous spaces, there are two additional technicalities, which are counterintuitive. First, the set of all subsets (used to define the event space  $\mathcal{A}$  in Section 6.1) is not well behaved enough.  $\mathcal{A}$  needs to be restricted to behave well under set complements, set intersections, and set unions. Second, the size of a set (which in discrete spaces can be obtained by counting the elements) turns out to be tricky. The size of a set is called its *measure*. For example, the cardinality of discrete sets, the length of an interval in  $\mathbb{R}$ , and the volume of a region in  $\mathbb{R}^d$  are all measures. Sets that behave well under set operations and additionally have a topology are called a *Borel  $\sigma$ -algebra*. Betancourt details a careful construction of probability spaces from set theory without being bogged down in technicalities; see <https://tinyurl.com/yb3t6mfd>. For a more precise construction, we refer to Billingsley (1995) and Jacod and Protter (2004).

In this book, we consider real-valued random variables with their cor-

responding Borel  $\sigma$ -algebra. We consider random variables with values in  $\mathbb{R}^D$  to be a vector of real-valued random variables.  $\diamond$

**Definition 6.1** (Probability Density Function). A function  $f : \mathbb{R}^D \rightarrow \mathbb{R}$  is called a *probability density function* (pdf) if

probability density  
function  
pdf

1.  $\forall \mathbf{x} \in \mathbb{R}^D : f(\mathbf{x}) \geq 0$
2. Its integral exists and

$$\int_{\mathbb{R}^D} f(\mathbf{x}) d\mathbf{x} = 1. \quad (6.15)$$

For probability mass functions (pmf) of discrete random variables, the integral in (6.15) is replaced with a sum (6.12).

Observe that the probability density function is any function  $f$  that is non-negative and integrates to one. We associate a random variable  $X$  with this function  $f$  by

$$P(a \leq X \leq b) = \int_a^b f(x) dx, \quad (6.16)$$

where  $a, b \in \mathbb{R}$  and  $x \in \mathbb{R}$  are outcomes of the continuous random variable  $X$ . States  $\mathbf{x} \in \mathbb{R}^D$  are defined analogously by considering a vector of  $x \in \mathbb{R}$ . This association (6.16) is called the *law* or *distribution* of the random variable  $X$ .

law

*Remark.* In contrast to discrete random variables, the probability of a continuous random variable  $X$  taking a particular value  $P(X = x)$  is zero. This is like trying to specify an interval in (6.16) where  $a = b$ .  $\diamond$

$P(X = x)$  is a set of  
measure zero.

**Definition 6.2** (Cumulative Distribution Function). A *cumulative distribution function* (cdf) of a multivariate real-valued random variable  $X$  with states  $\mathbf{x} \in \mathbb{R}^D$  is given by

cumulative  
distribution function

$$F_X(\mathbf{x}) = P(X_1 \leq x_1, \dots, X_D \leq x_D), \quad (6.17)$$

where  $X = [X_1, \dots, X_D]^\top$ ,  $\mathbf{x} = [x_1, \dots, x_D]^\top$ , and the right-hand side represents the probability that random variable  $X_i$  takes the value smaller than or equal to  $x_i$ .

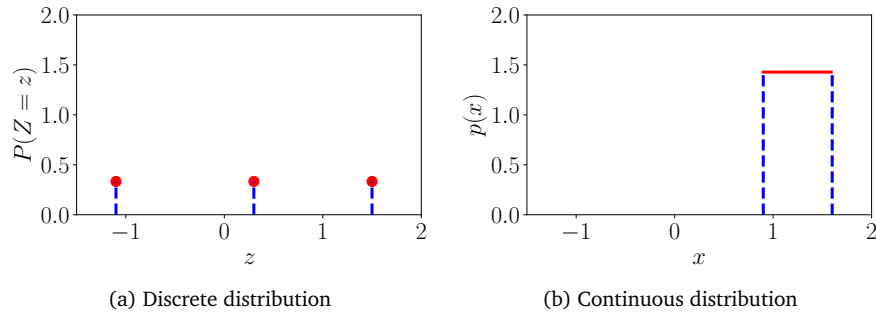
The cdf can be expressed also as the integral of the probability density function  $f(\mathbf{x})$  so that

There are cdfs,  
which do not have  
corresponding pdfs.

$$F_X(\mathbf{x}) = \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_D} f(z_1, \dots, z_D) dz_1 \cdots dz_D. \quad (6.18)$$

*Remark.* We reiterate that there are in fact two distinct concepts when talking about distributions. First is the idea of a pdf (denoted by  $f(x)$ ), which is a nonnegative function that sums to one. Second is the law of a random variable  $X$ , that is, the association of a random variable  $X$  with the pdf  $f(x)$ .  $\diamond$

**Figure 6.3**  
Examples of  
(a) discrete and  
(b) continuous  
uniform  
distributions. See  
Example 6.3 for  
details of the  
distributions.



For most of this book, we will not use the notation  $f(x)$  and  $F_X(x)$  as we mostly do not need to distinguish between the pdf and cdf. However, we will need to be careful about pdfs and cdfs in Section 6.7.

### 6.2.3 Contrasting Discrete and Continuous Distributions

Recall from Section 6.1.2 that probabilities are positive and the total probability sums up to one. For discrete random variables (see (6.12)), this implies that the probability of each state must lie in the interval  $[0, 1]$ . However, for continuous random variables the normalization (see (6.15)) does not imply that the value of the density is less than or equal to 1 for all values. We illustrate this in Figure 6.3 using the *uniform distribution* for both discrete and continuous random variables.

uniform distribution

#### Example 6.3

We consider two examples of the uniform distribution, where each state is equally likely to occur. This example illustrates some differences between discrete and continuous probability distributions.

Let  $Z$  be a discrete uniform random variable with three states  $\{z = -1.1, z = 0.3, z = 1.5\}$ . The probability mass function can be represented as a table of probability values:

$$P(Z = z) \begin{array}{|c|c|c|} \hline z & -1.1 & 0.3 & 1.5 \\ \hline \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \hline \end{array}$$

Alternatively, we can think of this as a graph (Figure 6.3(a)), where we use the fact that the states can be located on the  $x$ -axis, and the  $y$ -axis represents the probability of a particular state. The  $y$ -axis in Figure 6.3(a) is deliberately extended so that it is the same as in Figure 6.3(b).

Let  $X$  be a continuous random variable taking values in the range  $0.9 \leq X \leq 1.6$ , as represented by Figure 6.3(b). Observe that the height of the

The actual values of these states are not meaningful here, and we deliberately chose numbers to drive home the point that we do not want to use (and should ignore) the ordering of the states.

Type	“Point probability”	“Interval probability”
Discrete	$P(X = x)$ Probability mass function	Not applicable
Continuous	$p(x)$ Probability density function	$P(X \leq x)$ Cumulative distribution function

**Table 6.1**  
Nomenclature for  
probability  
distributions.

density can be greater than 1. However, it needs to hold that

$$\int_{0.9}^{1.6} p(x) dx = 1. \quad (6.19)$$

*Remark.* There is an additional subtlety with regards to discrete probability distributions. The states  $z_1, \dots, z_d$  do not in principle have any structure, i.e., there is usually no way to compare them, for example  $z_1 = \text{red}, z_2 = \text{green}, z_3 = \text{blue}$ . However, in many machine learning applications discrete states take numerical values, e.g.,  $z_1 = -1.1, z_2 = 0.3, z_3 = 1.5$ , where we could say  $z_1 < z_2 < z_3$ . Discrete states that assume numerical values are particularly useful because we often consider expected values (Section 6.4.1) of random variables.  $\diamond$

Unfortunately, machine learning literature uses notation and nomenclature that hides the distinction between the sample space  $\Omega$ , the target space  $\mathcal{T}$ , and the random variable  $X$ . For a value  $x$  of the set of possible outcomes of the random variable  $X$ , i.e.,  $x \in \mathcal{T}$ ,  $p(x)$  denotes the probability that random variable  $X$  has the outcome  $x$ . For discrete random variables, this is written as  $P(X = x)$ , which is known as the probability mass function. The pmf is often referred to as the “distribution”. For continuous variables,  $p(x)$  is called the probability density function (often referred to as a density). To muddy things even further, the cumulative distribution function  $P(X \leq x)$  is often also referred to as the “distribution”. In this chapter, we will use the notation  $X$  to refer to both univariate and multivariate random variables, and denote the states by  $x$  and  $\mathbf{x}$  respectively. We summarize the nomenclature in Table 6.1.

We think of the outcome  $x$  as the argument that results in the probability  $p(x)$ .

*Remark.* We will be using the expression “probability distribution” not only for discrete probability mass functions but also for continuous probability density functions, although this is technically incorrect. In line with most machine learning literature, we also rely on context to distinguish the different uses of the phrase probability distribution.  $\diamond$

### 6.3 Sum Rule, Product Rule, and Bayes' Theorem

We think of probability theory as an extension to logical reasoning. As we discussed in Section 6.1.1, the rules of probability presented here follow

naturally from fulfilling the desiderata (Jaynes, 2003, chapter 2). Probabilistic modeling (Section 8.4) provides a principled foundation for designing machine learning methods. Once we have defined probability distributions (Section 6.2) corresponding to the uncertainties of the data and our problem, it turns out that there are only two fundamental rules, the sum rule and the product rule.

Recall from (6.9) that  $p(\mathbf{x}, \mathbf{y})$  is the joint distribution of the two random variables  $\mathbf{x}, \mathbf{y}$ . The distributions  $p(\mathbf{x})$  and  $p(\mathbf{y})$  are the corresponding marginal distributions, and  $p(\mathbf{y} | \mathbf{x})$  is the conditional distribution of  $\mathbf{y}$  given  $\mathbf{x}$ . Given the definitions of the marginal and conditional probability for discrete and continuous random variables in Section 6.2, we can now present the two fundamental rules in probability theory.

The first rule, the *sum rule*, states that

$$p(\mathbf{x}) = \begin{cases} \sum_{\mathbf{y} \in \mathcal{Y}} p(\mathbf{x}, \mathbf{y}) & \text{if } \mathbf{y} \text{ is discrete} \\ \int_{\mathcal{Y}} p(\mathbf{x}, \mathbf{y}) d\mathbf{y} & \text{if } \mathbf{y} \text{ is continuous} \end{cases}, \quad (6.20)$$

where  $\mathcal{Y}$  are the states of the target space of random variable  $Y$ . This means that we sum out (or integrate out) the set of states  $\mathbf{y}$  of the random variable  $Y$ . The sum rule is also known as the *marginalization property*. The sum rule relates the joint distribution to a marginal distribution. In general, when the joint distribution contains more than two random variables, the sum rule can be applied to any subset of the random variables, resulting in a marginal distribution of potentially more than one random variable. More concretely, if  $\mathbf{x} = [x_1, \dots, x_D]^\top$ , we obtain the marginal

$$p(x_i) = \int p(x_1, \dots, x_D) d\mathbf{x}_{\setminus i} \quad (6.21)$$

by repeated application of the sum rule where we integrate/sum out all random variables except  $x_i$ , which is indicated by  $\setminus i$ , which reads “all except  $i$ .”

*Remark.* Many of the computational challenges of probabilistic modeling are due to the application of the sum rule. When there are many variables or discrete variables with many states, the sum rule boils down to performing a high-dimensional sum or integral. Performing high-dimensional sums or integrals is generally computationally hard, in the sense that there is no known polynomial-time algorithm to calculate them exactly.  $\diamond$

The second rule, known as the *product rule*, relates the joint distribution to the conditional distribution via

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{y} | \mathbf{x}) p(\mathbf{x}). \quad (6.22)$$

The product rule can be interpreted as the fact that every joint distribution of two random variables can be factorized (written as a product)

These two rules arise naturally (Jaynes, 2003) from the requirements we discussed in Section 6.1.1. sum rule

marginalization property

product rule

of two other distributions. The two factors are the marginal distribution of the first random variable  $p(\mathbf{x})$ , and the conditional distribution of the second random variable given the first  $p(\mathbf{y} | \mathbf{x})$ . Since the ordering of random variables is arbitrary in  $p(\mathbf{x}, \mathbf{y})$ , the product rule also implies  $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x} | \mathbf{y})p(\mathbf{y})$ . To be precise, (6.22) is expressed in terms of the probability mass functions for discrete random variables. For continuous random variables, the product rule is expressed in terms of the probability density functions (Section 6.2.3).

In machine learning and Bayesian statistics, we are often interested in making inferences of unobserved (latent) random variables given that we have observed other random variables. Let us assume we have some prior knowledge  $p(\mathbf{x})$  about an unobserved random variable  $\mathbf{x}$  and some relationship  $p(\mathbf{y} | \mathbf{x})$  between  $\mathbf{x}$  and a second random variable  $\mathbf{y}$ , which we can observe. If we observe  $\mathbf{y}$ , we can use Bayes' theorem to draw some conclusions about  $\mathbf{x}$  given the observed values of  $\mathbf{y}$ . *Bayes' theorem* (also *Bayes' rule* or *Bayes' law*)

Bayes' theorem  
Bayes' rule  
Bayes' law

$$\underbrace{p(\mathbf{x} | \mathbf{y})}_{\text{posterior}} = \frac{\overbrace{p(\mathbf{y} | \mathbf{x})}^{\text{likelihood}} \overbrace{p(\mathbf{x})}^{\text{prior}}}{\underbrace{p(\mathbf{y})}_{\text{evidence}}} \quad (6.23)$$

is a direct consequence of the product rule in (6.22) since

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x} | \mathbf{y})p(\mathbf{y}) \quad (6.24)$$

and

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{y} | \mathbf{x})p(\mathbf{x}) \quad (6.25)$$

so that

$$p(\mathbf{x} | \mathbf{y})p(\mathbf{y}) = p(\mathbf{y} | \mathbf{x})p(\mathbf{x}) \iff p(\mathbf{x} | \mathbf{y}) = \frac{p(\mathbf{y} | \mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}. \quad (6.26)$$

In (6.23),  $p(\mathbf{x})$  is the *prior*, which encapsulates our subjective prior knowledge of the unobserved (latent) variable  $\mathbf{x}$  before observing any data. We can choose any prior that makes sense to us, but it is critical to ensure that the prior has a nonzero pdf (or pmf) on all plausible  $\mathbf{x}$ , even if they are very rare.

prior

The *likelihood*  $p(\mathbf{y} | \mathbf{x})$  describes how  $\mathbf{x}$  and  $\mathbf{y}$  are related, and in the case of discrete probability distributions, it is the probability of the data  $\mathbf{y}$  if we were to know the latent variable  $\mathbf{x}$ . Note that the likelihood is not a distribution in  $\mathbf{x}$ , but only in  $\mathbf{y}$ . We call  $p(\mathbf{y} | \mathbf{x})$  either the “likelihood of  $\mathbf{x}$  (given  $\mathbf{y}$ )” or the “probability of  $\mathbf{y}$  given  $\mathbf{x}$ ” but never the likelihood of  $\mathbf{y}$  (MacKay, 2003).

likelihood  
The likelihood is sometimes also called the “measurement model”.

The *posterior*  $p(\mathbf{x} | \mathbf{y})$  is the quantity of interest in Bayesian statistics because it expresses exactly what we are interested in, i.e., what we know about  $\mathbf{x}$  after having observed  $\mathbf{y}$ .

posterior



The quantity

$$p(\mathbf{y}) := \int p(\mathbf{y} | \mathbf{x}) p(\mathbf{x}) d\mathbf{x} = \mathbb{E}_{\mathbf{x}}[p(\mathbf{y} | \mathbf{x})] \quad (6.27)$$

marginal likelihood  
evidence

is the *marginal likelihood/evidence*. The right-hand side of (6.27) uses the expectation operator which we define in Section 6.4.1. By definition, the marginal likelihood integrates the numerator of (6.23) with respect to the latent variable  $\mathbf{x}$ . Therefore, the marginal likelihood is independent of  $\mathbf{x}$ , and it ensures that the posterior  $p(\mathbf{x} | \mathbf{y})$  is normalized. The marginal likelihood can also be interpreted as the expected likelihood where we take the expectation with respect to the prior  $p(\mathbf{x})$ . Beyond normalization of the posterior, the marginal likelihood also plays an important role in Bayesian model selection, as we will discuss in Section 8.6. Due to the integration in (8.44), the evidence is often hard to compute.

Bayes' theorem is  
also called the  
"probabilistic  
inverse."  
probabilistic inverse

Bayes' theorem (6.23) allows us to invert the relationship between  $\mathbf{x}$  and  $\mathbf{y}$  given by the likelihood. Therefore, Bayes' theorem is sometimes called the *probabilistic inverse*. We will discuss Bayes' theorem further in Section 8.4.

*Remark.* In Bayesian statistics, the posterior distribution is the quantity of interest as it encapsulates all available information from the prior and the data. Instead of carrying the posterior around, it is possible to focus on some statistic of the posterior, such as the maximum of the posterior, which we will discuss in Section 8.3. However, focusing on some statistic of the posterior leads to loss of information. If we think in a bigger context, then the posterior can be used within a decision-making system, and having the full posterior can be extremely useful and lead to decisions that are robust to disturbances. For example, in the context of model-based reinforcement learning, Deisenroth et al. (2015) show that using the full posterior distribution of plausible transition functions leads to very fast (data/sample efficient) learning, whereas focusing on the maximum of the posterior leads to consistent failures. Therefore, having the full posterior can be very useful for a downstream task. In Chapter 9, we will continue this discussion in the context of linear regression.  $\diamond$

## 6.4 Summary Statistics and Independence

We are often interested in summarizing sets of random variables and comparing pairs of random variables. A statistic of a random variable is a deterministic function of that random variable. The summary statistics of a distribution provide one useful view of how a random variable behaves, and as the name suggests, provide numbers that summarize and characterize the distribution. We describe the mean and the variance, two well-known summary statistics. Then we discuss two ways to compare a pair of random variables: first, how to say that two random variables are independent; and second, how to compute an inner product between them.

### 6.4.1 Means and Covariances

Mean and (co)variance are often useful to describe properties of probability distributions (expected values and spread). We will see in Section 6.6 that there is a useful family of distributions (called the exponential family), where the statistics of the random variable capture all possible information.

The concept of the expected value is central to machine learning, and the foundational concepts of probability itself can be derived from the expected value (Whittle, 2000).

**Definition 6.3** (Expected Value). The *expected value* of a function  $g : \mathbb{R} \rightarrow \mathbb{R}$  of a univariate continuous random variable  $X \sim p(x)$  is given by

$$\mathbb{E}_X[g(x)] = \int_{\mathcal{X}} g(x)p(x)dx. \quad (6.28)$$

Correspondingly, the expected value of a function  $g$  of a discrete random variable  $X \sim p(x)$  is given by

$$\mathbb{E}_X[g(x)] = \sum_{x \in \mathcal{X}} g(x)p(x), \quad (6.29)$$

where  $\mathcal{X}$  is the set of possible outcomes (the target space) of the random variable  $X$ .

In this section, we consider discrete random variables to have numerical outcomes. This can be seen by observing that the function  $g$  takes real numbers as inputs.

*Remark.* We consider multivariate random variables  $X$  as a finite vector of univariate random variables  $[X_1, \dots, X_D]^\top$ . For multivariate random variables, we define the expected value element wise

$$\mathbb{E}_X[g(\mathbf{x})] = \begin{bmatrix} \mathbb{E}_{X_1}[g(x_1)] \\ \vdots \\ \mathbb{E}_{X_D}[g(x_D)] \end{bmatrix} \in \mathbb{R}^D, \quad (6.30)$$

where the subscript  $\mathbb{E}_{X_d}$  indicates that we are taking the expected value with respect to the  $d$ th element of the vector  $\mathbf{x}$ .  $\diamond$

Definition 6.3 defines the meaning of the notation  $\mathbb{E}_X$  as the operator indicating that we should take the integral with respect to the probability density (for continuous distributions) or the sum over all states (for discrete distributions). The definition of the mean (Definition 6.4), is a special case of the expected value, obtained by choosing  $g$  to be the identity function.

**Definition 6.4** (Mean). The *mean* of a random variable  $X$  with states

expected value

The expected value of a function of a random variable is sometimes referred to as the law of the unconscious statistician (Casella and Berger, 2002, Section 2.2).

mean

$\mathbf{x} \in \mathbb{R}^D$  is an average and is defined as

$$\mathbb{E}_X[\mathbf{x}] = \begin{bmatrix} \mathbb{E}_{X_1}[x_1] \\ \vdots \\ \mathbb{E}_{X_D}[x_D] \end{bmatrix} \in \mathbb{R}^D, \quad (6.31)$$

where

$$\mathbb{E}_{X_d}[x_d] := \begin{cases} \int_{\mathcal{X}} x_d p(x_d) dx_d & \text{if } X \text{ is a continuous random variable} \\ \sum_{x_i \in \mathcal{X}} x_i p(x_d = x_i) & \text{if } X \text{ is a discrete random variable} \end{cases} \quad (6.32)$$

for  $d = 1, \dots, D$ , where the subscript  $d$  indicates the corresponding dimension of  $\mathbf{x}$ . The integral and sum are over the states  $\mathcal{X}$  of the target space of the random variable  $X$ .

median

In one dimension, there are two other intuitive notions of “average”, which are the *median* and the *mode*. The *median* is the “middle” value if we sort the values, i.e., 50% of the values are greater than the median and 50% are smaller than the median. This idea can be generalized to continuous values by considering the value where the cdf (Definition 6.2) is 0.5. For distributions, which are asymmetric or have long tails, the median provides an estimate of a typical value that is closer to human intuition than the mean value. Furthermore, the median is more robust to outliers than the mean. The generalization of the median to higher dimensions is non-trivial as there is no obvious way to “sort” in more than one dimension (Hallin et al., 2010; Kong and Mizera, 2012). The *mode* is the most frequently occurring value. For a discrete random variable, the mode is defined as the value of  $x$  having the highest frequency of occurrence. For a continuous random variable, the mode is defined as a peak in the density  $p(\mathbf{x})$ . A particular density  $p(\mathbf{x})$  may have more than one mode, and furthermore there may be a very large number of modes in high-dimensional distributions. Therefore, finding all the modes of a distribution can be computationally challenging.

mode

#### Example 6.4

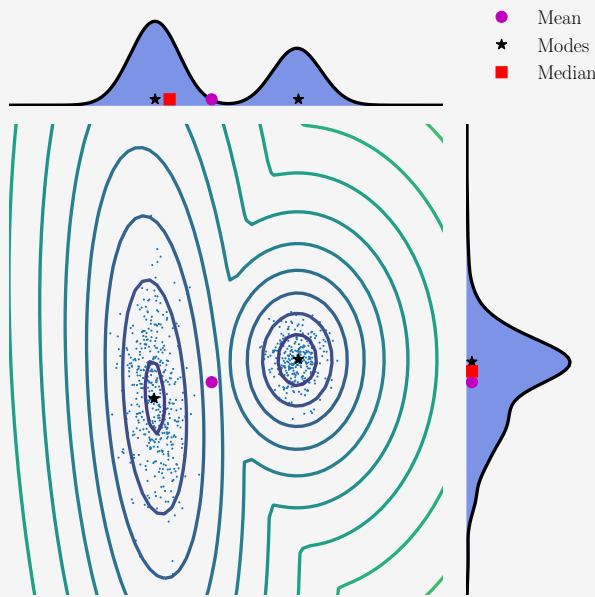
Consider the two-dimensional distribution illustrated in Figure 6.4:

$$p(\mathbf{x}) = 0.4 \mathcal{N}\left(\mathbf{x} \mid \begin{bmatrix} 10 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) + 0.6 \mathcal{N}\left(\mathbf{x} \mid \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 8.4 & 2.0 \\ 2.0 & 1.7 \end{bmatrix}\right). \quad (6.33)$$

We will define the Gaussian distribution  $\mathcal{N}(\mu, \sigma^2)$  in Section 6.5. Also shown is its corresponding marginal distribution in each dimension. Observe that the distribution is bimodal (has two modes), but one of the

marginal distributions is unimodal (has one mode). The horizontal bi-modal univariate distribution illustrates that the mean and median can be different from each other. While it is tempting to define the two-dimensional median to be the concatenation of the medians in each dimension, the fact that we cannot define an ordering of two-dimensional points makes it difficult. When we say “cannot define an ordering”, we mean that there is more than one way to define the relation  $<$  so that

$$\begin{bmatrix} 3 \\ 0 \end{bmatrix} < \begin{bmatrix} 2 \\ 3 \end{bmatrix}.$$



**Figure 6.4**  
Illustration of the mean, mode, and median for a two-dimensional dataset, as well as its marginal densities.

*Remark.* The expected value (Definition 6.3) is a linear operator. For example, given a real-valued function  $f(\mathbf{x}) = ag(\mathbf{x}) + bh(\mathbf{x})$  where  $a, b \in \mathbb{R}$  and  $\mathbf{x} \in \mathbb{R}^D$ , we obtain

$$\mathbb{E}_X[f(\mathbf{x})] = \int f(\mathbf{x})p(\mathbf{x})d\mathbf{x} \quad (6.34a)$$

$$= \int [ag(\mathbf{x}) + bh(\mathbf{x})]p(\mathbf{x})d\mathbf{x} \quad (6.34b)$$

$$= a \int g(\mathbf{x})p(\mathbf{x})d\mathbf{x} + b \int h(\mathbf{x})p(\mathbf{x})d\mathbf{x} \quad (6.34c)$$

$$= a\mathbb{E}_X[g(\mathbf{x})] + b\mathbb{E}_X[h(\mathbf{x})]. \quad (6.34d)$$

◇

For two random variables, we may wish to characterize their correspon-

dence to each other. The covariance intuitively represents the notion of how dependent random variables are to one another.

covariance

**Definition 6.5** (Covariance (Univariate)). The *covariance* between two univariate random variables  $X, Y \in \mathbb{R}$  is given by the expected product of their deviations from their respective means, i.e.,

$$\text{Cov}_{X,Y}[x, y] := \mathbb{E}_{X,Y}[(x - \mathbb{E}_X[x])(y - \mathbb{E}_Y[y])]. \quad (6.35)$$

Terminology: The covariance of multivariate random variables  $\text{Cov}[x, y]$  is sometimes referred to as cross-covariance, with covariance referring to  $\text{Cov}[x, x]$ .

*Remark.* When the random variable associated with the expectation or covariance is clear by its arguments, the subscript is often suppressed (for example,  $\mathbb{E}_X[x]$  is often written as  $\mathbb{E}[x]$ ).  $\diamond$

By using the linearity of expectations, the expression in Definition 6.5 can be rewritten as the expected value of the product minus the product of the expected values, i.e.,

$$\text{Cov}[x, y] = \mathbb{E}[xy] - \mathbb{E}[x]\mathbb{E}[y]. \quad (6.36)$$

variance  
standard deviation

The covariance of a variable with itself  $\text{Cov}[x, x]$  is called the *variance* and is denoted by  $\mathbb{V}_X[x]$ . The square root of the variance is called the *standard deviation* and is often denoted by  $\sigma(x)$ . The notion of covariance can be generalized to multivariate random variables.

**Definition 6.6** (Covariance (Multivariate)). If we consider two multivariate random variables  $X$  and  $Y$  with states  $\mathbf{x} \in \mathbb{R}^D$  and  $\mathbf{y} \in \mathbb{R}^E$  respectively, the *covariance* between  $X$  and  $Y$  is defined as

covariance

$$\text{Cov}[\mathbf{x}, \mathbf{y}] = \mathbb{E}[\mathbf{x}\mathbf{y}^\top] - \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{y}]^\top = \text{Cov}[\mathbf{y}, \mathbf{x}]^\top \in \mathbb{R}^{D \times E}. \quad (6.37)$$

Definition 6.6 can be applied with the same multivariate random variable in both arguments, which results in a useful concept that intuitively captures the “spread” of a random variable. For a multivariate random variable, the variance describes the relation between individual dimensions of the random variable.

variance

**Definition 6.7** (Variance). The *variance* of a random variable  $X$  with states  $\mathbf{x} \in \mathbb{R}^D$  and a mean vector  $\boldsymbol{\mu} \in \mathbb{R}^D$  is defined as

$$\mathbb{V}_X[\mathbf{x}] = \text{Cov}_X[\mathbf{x}, \mathbf{x}] \quad (6.38a)$$

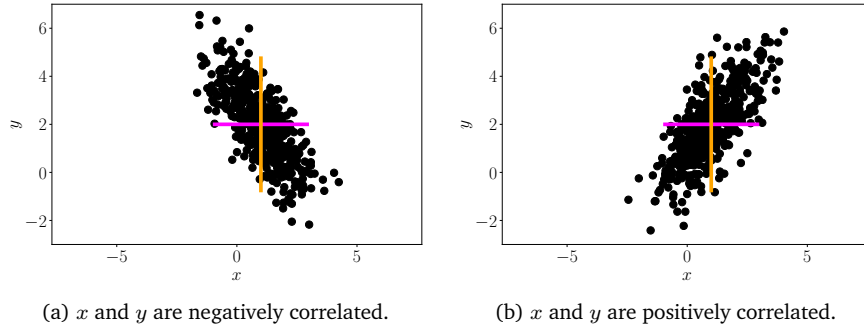
$$= \mathbb{E}_X[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^\top] = \mathbb{E}_X[\mathbf{x}\mathbf{x}^\top] - \mathbb{E}_X[\mathbf{x}]\mathbb{E}_X[\mathbf{x}]^\top \quad (6.38b)$$

$$= \begin{bmatrix} \text{Cov}[x_1, x_1] & \text{Cov}[x_1, x_2] & \dots & \text{Cov}[x_1, x_D] \\ \text{Cov}[x_2, x_1] & \text{Cov}[x_2, x_2] & \dots & \text{Cov}[x_2, x_D] \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}[x_D, x_1] & \dots & \dots & \text{Cov}[x_D, x_D] \end{bmatrix}. \quad (6.38c)$$

covariance matrix

The  $D \times D$  matrix in (6.38c) is called the *covariance matrix* of the multivariate random variable  $X$ . The covariance matrix is symmetric and positive semidefinite and tells us something about the spread of the data. On its diagonal, the covariance matrix contains the variances of the *marginals*

marginal



**Figure 6.5**  
Two-dimensional datasets with identical means and variances along each axis (colored lines) but with different covariances.

$$p(x_i) = \int p(x_1, \dots, x_D) dx_{\setminus i}, \quad (6.39)$$

where “ $\setminus i$ ” denotes “all variables but  $i$ ”. The off-diagonal entries are the *cross-covariance* terms  $\text{Cov}[x_i, x_j]$  for  $i, j = 1, \dots, D$ ,  $i \neq j$ .

cross-covariance

*Remark.* In this book, we generally assume that covariance matrices are positive definite to enable better intuition. We therefore do not discuss corner cases that result in positive semidefinite (low-rank) covariance matrices.  $\diamond$

When we want to compare the covariances between different pairs of random variables, it turns out that the variance of each random variable affects the value of the covariance. The normalized version of covariance is called the correlation.

**Definition 6.8** (Correlation). The *correlation* between two random variables  $X, Y$  is given by

correlation

$$\text{corr}[x, y] = \frac{\text{Cov}[x, y]}{\sqrt{\mathbb{V}[x]\mathbb{V}[y]}} \in [-1, 1]. \quad (6.40)$$

The correlation matrix is the covariance matrix of standardized random variables,  $x/\sigma(x)$ . In other words, each random variable is divided by its standard deviation (the square root of the variance) in the correlation matrix.

The covariance (and correlation) indicate how two random variables are related; see Figure 6.5. Positive correlation  $\text{corr}[x, y]$  means that when  $x$  grows, then  $y$  is also expected to grow. Negative correlation means that as  $x$  increases, then  $y$  decreases.

### 6.4.2 Empirical Means and Covariances

The definitions in Section 6.4.1 are often also called the *population mean and covariance*, as it refers to the true statistics for the population. In machine learning, we need to learn from empirical observations of data. Consider a random variable  $X$ . There are two conceptual steps to go from

population mean and covariance

population statistics to the realization of empirical statistics. First, we use the fact that we have a finite dataset (of size  $N$ ) to construct an empirical statistic that is a function of a finite number of identical random variables,  $X_1, \dots, X_N$ . Second, we observe the data, that is, we look at the realization  $x_1, \dots, x_N$  of each of the random variables and apply the empirical statistic.

empirical mean  
sample mean

Specifically, for the mean (Definition 6.4), given a particular dataset we can obtain an estimate of the mean, which is called the *empirical mean* or *sample mean*. The same holds for the empirical covariance.

empirical mean

**Definition 6.9** (Empirical Mean and Covariance). The *empirical mean* vector is the arithmetic average of the observations for each variable, and it is defined as

$$\bar{\mathbf{x}} := \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n, \quad (6.41)$$

where  $\mathbf{x}_n \in \mathbb{R}^D$ .

empirical covariance

Similar to the empirical mean, the *empirical covariance* matrix is a  $D \times D$  matrix

$$\Sigma := \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^\top. \quad (6.42)$$

Throughout the book, we use the empirical covariance, which is a biased estimate. The unbiased (sometimes called corrected) covariance has the factor  $N - 1$  in the denominator instead of  $N$ . The derivations are exercises at the end of this chapter.

To compute the statistics for a particular dataset, we would use the realizations (observations)  $\mathbf{x}_1, \dots, \mathbf{x}_N$  and use (6.41) and (6.42). Empirical covariance matrices are symmetric, positive semidefinite (see Section 3.2.3).

### 6.4.3 Three Expressions for the Variance

We now focus on a single random variable  $X$  and use the preceding empirical formulas to derive three possible expressions for the variance. The following derivation is the same for the population variance, except that we need to take care of integrals. The standard definition of variance, corresponding to the definition of covariance (Definition 6.5), is the expectation of the squared deviation of a random variable  $X$  from its expected value  $\mu$ , i.e.,

$$\mathbb{V}_X[x] := \mathbb{E}_X[(x - \mu)^2]. \quad (6.43)$$

The expectation in (6.43) and the mean  $\mu = \mathbb{E}_X(x)$  are computed using (6.32), depending on whether  $X$  is a discrete or continuous random variable. The variance as expressed in (6.43) is the mean of a new random variable  $Z := (X - \mu)^2$ .

When estimating the variance in (6.43) empirically, we need to resort to a two-pass algorithm: one pass through the data to calculate the mean  $\mu$  using (6.41), and then a second pass using this estimate  $\hat{\mu}$  calculate the

variance. It turns out that we can avoid two passes by rearranging the terms. The formula in (6.43) can be converted to the so-called *raw-score formula for variance*:

$$\mathbb{V}_X[x] = \mathbb{E}_X[x^2] - (\mathbb{E}_X[x])^2. \quad (6.44)$$

raw-score formula  
for variance

The expression in (6.44) can be remembered as “the mean of the square minus the square of the mean”. It can be calculated empirically in one pass through data since we can accumulate  $x_i$  (to calculate the mean) and  $x_i^2$  simultaneously, where  $x_i$  is the  $i$ th observation. Unfortunately, if implemented in this way, it can be numerically unstable. The raw-score version of the variance can be useful in machine learning, e.g., when deriving the bias–variance decomposition (Bishop, 2006).

If the two terms in (6.44) are huge and approximately equal, we may suffer from an unnecessary loss of numerical precision in floating-point arithmetic.

A third way to understand the variance is that it is a sum of pairwise differences between all pairs of observations. Consider a sample  $x_1, \dots, x_N$  of realizations of random variable  $X$ , and we compute the squared difference between pairs of  $x_i$  and  $x_j$ . By expanding the square, we can show that the sum of  $N^2$  pairwise differences is the empirical variance of the observations:

$$\frac{1}{N^2} \sum_{i,j=1}^N (x_i - x_j)^2 = 2 \left[ \frac{1}{N} \sum_{i=1}^N x_i^2 - \left( \frac{1}{N} \sum_{i=1}^N x_i \right)^2 \right]. \quad (6.45)$$

We see that (6.45) is twice the raw-score expression (6.44). This means that we can express the sum of pairwise distances (of which there are  $N^2$  of them) as a sum of deviations from the mean (of which there are  $N$ ). Geometrically, this means that there is an equivalence between the pairwise distances and the distances from the center of the set of points. From a computational perspective, this means that by computing the mean ( $N$  terms in the summation), and then computing the variance (again  $N$  terms in the summation), we can obtain an expression (left-hand side of (6.45)) that has  $N^2$  terms.

#### 6.4.4 Sums and Transformations of Random Variables

We may want to model a phenomenon that cannot be well explained by textbook distributions (we introduce some in Sections 6.5 and 6.6), and hence may perform simple manipulations of random variables (such as adding two random variables).

Consider two random variables  $X, Y$  with states  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^D$ . Then:

$$\mathbb{E}[\mathbf{x} + \mathbf{y}] = \mathbb{E}[\mathbf{x}] + \mathbb{E}[\mathbf{y}] \quad (6.46)$$

$$\mathbb{E}[\mathbf{x} - \mathbf{y}] = \mathbb{E}[\mathbf{x}] - \mathbb{E}[\mathbf{y}] \quad (6.47)$$

$$\mathbb{V}[\mathbf{x} + \mathbf{y}] = \mathbb{V}[\mathbf{x}] + \mathbb{V}[\mathbf{y}] + \text{Cov}[\mathbf{x}, \mathbf{y}] + \text{Cov}[\mathbf{y}, \mathbf{x}] \quad (6.48)$$

$$\mathbb{V}[\mathbf{x} - \mathbf{y}] = \mathbb{V}[\mathbf{x}] + \mathbb{V}[\mathbf{y}] - \text{Cov}[\mathbf{x}, \mathbf{y}] - \text{Cov}[\mathbf{y}, \mathbf{x}]. \quad (6.49)$$



Mean and (co)variance exhibit some useful properties when it comes to affine transformation of random variables. Consider a random variable  $X$  with mean  $\boldsymbol{\mu}$  and covariance matrix  $\boldsymbol{\Sigma}$  and a (deterministic) affine transformation  $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}$  of  $\mathbf{x}$ . Then  $\mathbf{y}$  is itself a random variable whose mean vector and covariance matrix are given by

$$\mathbb{E}_Y[\mathbf{y}] = \mathbb{E}_X[\mathbf{A}\mathbf{x} + \mathbf{b}] = \mathbf{A}\mathbb{E}_X[\mathbf{x}] + \mathbf{b} = \mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \quad (6.50)$$

$$\mathbb{V}_Y[\mathbf{y}] = \mathbb{V}_X[\mathbf{A}\mathbf{x} + \mathbf{b}] = \mathbb{V}_X[\mathbf{A}\mathbf{x}] = \mathbf{A}\mathbb{V}_X[\mathbf{x}]\mathbf{A}^\top = \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^\top, \quad (6.51)$$

This can be shown directly by using the definition of the mean and covariance.

respectively. Furthermore,

$$\text{Cov}[\mathbf{x}, \mathbf{y}] = \mathbb{E}[\mathbf{x}(\mathbf{A}\mathbf{x} + \mathbf{b})^\top] - \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{A}\mathbf{x} + \mathbf{b}]^\top \quad (6.52a)$$

$$= \mathbb{E}[\mathbf{x}]\mathbf{b}^\top + \mathbb{E}[\mathbf{x}\mathbf{x}^\top]\mathbf{A}^\top - \boldsymbol{\mu}\mathbf{b}^\top - \boldsymbol{\mu}\boldsymbol{\mu}^\top\mathbf{A}^\top \quad (6.52b)$$

$$= \boldsymbol{\mu}\mathbf{b}^\top - \boldsymbol{\mu}\mathbf{b}^\top + (\mathbb{E}[\mathbf{x}\mathbf{x}^\top] - \boldsymbol{\mu}\boldsymbol{\mu}^\top)\mathbf{A}^\top \quad (6.52c)$$

$$\stackrel{(6.38b)}{=} \boldsymbol{\Sigma}\mathbf{A}^\top, \quad (6.52d)$$

where  $\boldsymbol{\Sigma} = \mathbb{E}[\mathbf{x}\mathbf{x}^\top] - \boldsymbol{\mu}\boldsymbol{\mu}^\top$  is the covariance of  $X$ .

#### 6.4.5 Statistical Independence

statistical independence

**Definition 6.10** (Independence). Two random variables  $X, Y$  are *statistically independent* if and only if

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y}). \quad (6.53)$$

Intuitively, two random variables  $X$  and  $Y$  are independent if the value of  $\mathbf{y}$  (once known) does not add any additional information about  $\mathbf{x}$  (and vice versa). If  $X, Y$  are (statistically) independent, then

- $p(\mathbf{y} | \mathbf{x}) = p(\mathbf{y})$
- $p(\mathbf{x} | \mathbf{y}) = p(\mathbf{x})$
- $\mathbb{V}_{X,Y}[\mathbf{x} + \mathbf{y}] = \mathbb{V}_X[\mathbf{x}] + \mathbb{V}_Y[\mathbf{y}]$
- $\text{Cov}_{X,Y}[\mathbf{x}, \mathbf{y}] = \mathbf{0}$

The last point may not hold in converse, i.e., two random variables can have covariance zero but are not statistically independent. To understand why, recall that covariance measures only linear dependence. Therefore, random variables that are nonlinearly dependent could have covariance zero.

#### Example 6.5

Consider a random variable  $X$  with zero mean ( $\mathbb{E}_X[x] = 0$ ) and also  $\mathbb{E}_X[x^3] = 0$ . Let  $y = x^2$  (hence,  $Y$  is dependent on  $X$ ) and consider the covariance (6.36) between  $X$  and  $Y$ . But this gives

$$\text{Cov}[x, y] = \mathbb{E}[xy] - \mathbb{E}[x]\mathbb{E}[y] = \mathbb{E}[x^3] = 0. \quad (6.54)$$

In machine learning, we often consider problems that can be modeled as *independent and identically distributed* (i.i.d.) random variables,  $X_1, \dots, X_N$ . For more than two random variables, the word “independent” (Definition 6.10) usually refers to mutually independent random variables, where all subsets are independent (see Pollard (2002, chapter 4) and Jacod and Protter (2004, chapter 3)). The phrase “identically distributed” means that all the random variables are from the same distribution.

independent and  
identically  
distributed  
i.i.d.

Another concept that is important in machine learning is conditional independence.

**Definition 6.11** (Conditional Independence). Two random variables  $X$  and  $Y$  are *conditionally independent* given  $Z$  if and only if

conditionally  
independent

$$p(\mathbf{x}, \mathbf{y} | \mathbf{z}) = p(\mathbf{x} | \mathbf{z})p(\mathbf{y} | \mathbf{z}) \quad \text{for all } \mathbf{z} \in \mathcal{Z}, \quad (6.55)$$

where  $\mathcal{Z}$  is the set of states of random variable  $Z$ . We write  $X \perp\!\!\!\perp Y | Z$  to denote that  $X$  is conditionally independent of  $Y$  given  $Z$ .

Definition 6.11 requires that the relation in (6.55) must hold true for every value of  $\mathbf{z}$ . The interpretation of (6.55) can be understood as “given knowledge about  $\mathbf{z}$ , the distribution of  $\mathbf{x}$  and  $\mathbf{y}$  factorizes”. Independence can be cast as a special case of conditional independence if we write  $X \perp\!\!\!\perp Y | \emptyset$ . By using the product rule of probability (6.22), we can expand the left-hand side of (6.55) to obtain

$$p(\mathbf{x}, \mathbf{y} | \mathbf{z}) = p(\mathbf{x} | \mathbf{y}, \mathbf{z})p(\mathbf{y} | \mathbf{z}). \quad (6.56)$$

By comparing the right-hand side of (6.55) with (6.56), we see that  $p(\mathbf{y} | \mathbf{z})$  appears in both of them so that

$$p(\mathbf{x} | \mathbf{y}, \mathbf{z}) = p(\mathbf{x} | \mathbf{z}). \quad (6.57)$$

Equation (6.57) provides an alternative definition of conditional independence, i.e.,  $X \perp\!\!\!\perp Y | Z$ . This alternative presentation provides the interpretation “given that we know  $\mathbf{z}$ , knowledge about  $\mathbf{y}$  does not change our knowledge of  $\mathbf{x}$ ”.

#### 6.4.6 Inner Products of Random Variables

Recall the definition of inner products from Section 3.2. We can define an inner product between random variables, which we briefly describe in this section. If we have two uncorrelated random variables  $X, Y$ , then

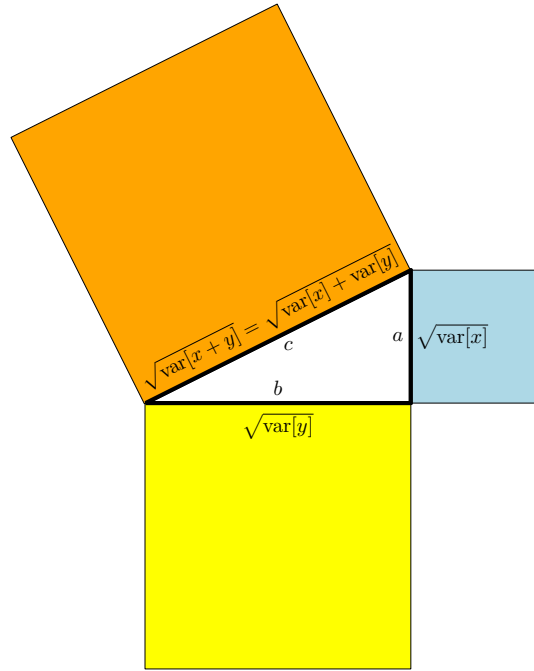
$$\mathbb{V}[x + y] = \mathbb{V}[x] + \mathbb{V}[y]. \quad (6.58)$$

Since variances are measured in squared units, this looks very much like the Pythagorean theorem for right triangles  $c^2 = a^2 + b^2$ .

In the following, we see whether we can find a geometric interpretation of the variance relation of uncorrelated random variables in (6.58).

Inner products  
between  
multivariate random  
variables can be  
treated in a similar  
fashion

**Figure 6.6**  
Geometry of random variables. If random variables  $X$  and  $Y$  are uncorrelated, they are orthogonal vectors in a corresponding vector space, and the Pythagorean theorem applies.



Random variables can be considered vectors in a vector space, and we can define inner products to obtain geometric properties of random variables (Eaton, 2007). If we define

$$\langle X, Y \rangle := \text{Cov}[x, y] \quad (6.59)$$

for zero mean random variables  $X$  and  $Y$ , we obtain an inner product. We see that the covariance is symmetric, positive definite, and linear in either argument. The length of a random variable is

$$\|X\| = \sqrt{\text{Cov}[x, x]} = \sqrt{\mathbb{V}[x]} = \sigma[x], \quad (6.60)$$

i.e., its standard deviation. The “longer” the random variable, the more uncertain it is; and a random variable with length 0 is deterministic.

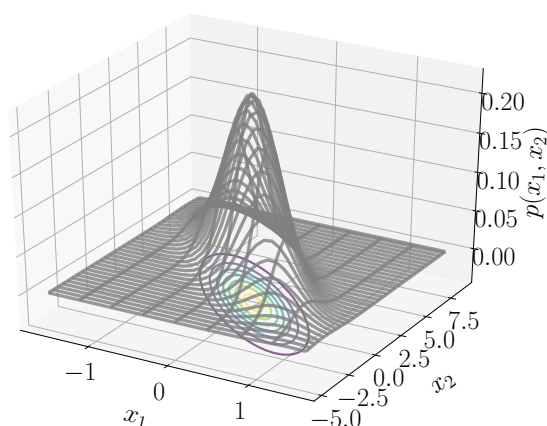
If we look at the angle  $\theta$  between two random variables  $X, Y$ , we get

$$\cos \theta = \frac{\langle X, Y \rangle}{\|X\| \|Y\|} = \frac{\text{Cov}[x, y]}{\sqrt{\mathbb{V}[x] \mathbb{V}[y]}}, \quad (6.61)$$

which is the correlation (Definition 6.8) between the two random variables. This means that we can think of correlation as the cosine of the angle between two random variables when we consider them geometrically. We know from Definition 3.7 that  $X \perp Y \iff \langle X, Y \rangle = 0$ . In our case, this means that  $X$  and  $Y$  are orthogonal if and only if  $\text{Cov}[x, y] = 0$ , i.e., they are uncorrelated. Figure 6.6 illustrates this relationship.

*Remark.* While it is tempting to use the Euclidean distance (constructed

$\text{Cov}[x, x] = 0 \iff x = 0$   
 $\text{Cov}[\alpha x + z, y] = \alpha \text{Cov}[x, y] + \text{Cov}[z, y]$  for  $\alpha \in \mathbb{R}$ .



**Figure 6.7**  
Gaussian  
distribution of two  
random variables  $x_1$   
and  $x_2$ .

from the preceding definition of inner products) to compare probability distributions, it is unfortunately not the best way to obtain distances between distributions. Recall that the probability mass (or density) is positive and needs to add up to 1. These constraints mean that distributions live on something called a statistical manifold. The study of this space of probability distributions is called information geometry. Computing distances between distributions are often done using Kullback-Leibler divergence, which is a generalization of distances that account for properties of the statistical manifold. Just like the Euclidean distance is a special case of a metric (Section 3.3), the Kullback-Leibler divergence is a special case of two more general classes of divergences called Bregman divergences and  $f$ -divergences. The study of divergences is beyond the scope of this book, and we refer for more details to the recent book by Amari (2016), one of the founders of the field of information geometry.  $\diamond$

## 6.5 Gaussian Distribution

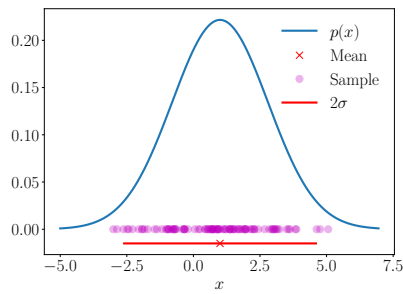
The Gaussian distribution is the most well-studied probability distribution for continuous-valued random variables. It is also referred to as the *normal distribution*. Its importance originates from the fact that it has many computationally convenient properties, which we will be discussing in the following. In particular, we will use it to define the likelihood and prior for linear regression (Chapter 9), and consider a mixture of Gaussians for density estimation (Chapter 11).

There are many other areas of machine learning that also benefit from using a Gaussian distribution, for example Gaussian processes, variational inference, and reinforcement learning. It is also widely used in other application areas such as signal processing (e.g., Kalman filter), control (e.g., linear quadratic regulator), and statistics (e.g., hypothesis testing).

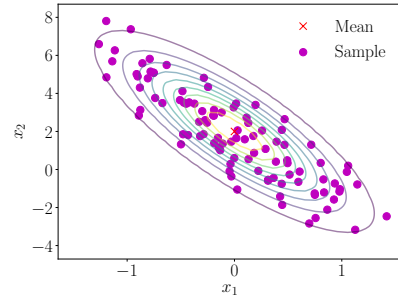
normal distribution

The Gaussian distribution arises naturally when we consider sums of independent and identically distributed random variables. This is known as the central limit theorem (Grinstead and Snell, 1997).

**Figure 6.8**  
Gaussian  
distributions  
overlaid with 100  
samples. (a) One-  
dimensional case;  
(b) two-dimensional  
case.



(a) Univariate (one-dimensional) Gaussian; The red cross shows the mean and the red line shows the extent of the variance.



(b) Multivariate (two-dimensional) Gaussian, viewed from top. The red cross shows the mean and the colored lines show the contour lines of the density.

For a univariate random variable, the Gaussian distribution has a density that is given by

$$p(x | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right). \quad (6.62)$$

The *multivariate Gaussian distribution* is fully characterized by a *mean vector*  $\boldsymbol{\mu}$  and a *covariance matrix*  $\boldsymbol{\Sigma}$  and defined as

$$p(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right), \quad (6.63)$$

where  $\mathbf{x} \in \mathbb{R}^D$ . We write  $p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma})$  or  $X \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ . Figure 6.7 shows a bivariate Gaussian (mesh), with the corresponding contour plot. Figure 6.8 shows a univariate Gaussian and a bivariate Gaussian with corresponding samples. The special case of the Gaussian with zero mean and identity covariance, that is,  $\boldsymbol{\mu} = \mathbf{0}$  and  $\boldsymbol{\Sigma} = \mathbf{I}$ , is referred to as the *standard normal distribution*.

Gaussians are widely used in statistical estimation and machine learning as they have closed-form expressions for marginal and conditional distributions. In Chapter 9, we use these closed-form expressions extensively for linear regression. A major advantage of modeling with Gaussian random variables is that variable transformations (Section 6.7) are often not needed. Since the Gaussian distribution is fully specified by its mean and covariance, we often can obtain the transformed distribution by applying the transformation to the mean and covariance of the random variable.

### 6.5.1 Marginals and Conditionals of Gaussians are Gaussians

In the following, we present marginalization and conditioning in the general case of multivariate random variables. If this is confusing at first reading, the reader is advised to consider two univariate random variables instead. Let  $X$  and  $Y$  be two multivariate random variables, that may have

multivariate  
Gaussian  
distribution  
mean vector  
covariance matrix

Also known as a  
multivariate normal  
distribution.

standard normal  
distribution

different dimensions. To consider the effect of applying the sum rule of probability and the effect of conditioning, we explicitly write the Gaussian distribution in terms of the concatenated states  $[\mathbf{x}^\top, \mathbf{y}^\top]$ ,

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_y \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{xx} & \boldsymbol{\Sigma}_{xy} \\ \boldsymbol{\Sigma}_{yx} & \boldsymbol{\Sigma}_{yy} \end{bmatrix}\right). \quad (6.64)$$

where  $\boldsymbol{\Sigma}_{xx} = \text{Cov}[\mathbf{x}, \mathbf{x}]$  and  $\boldsymbol{\Sigma}_{yy} = \text{Cov}[\mathbf{y}, \mathbf{y}]$  are the marginal covariance matrices of  $\mathbf{x}$  and  $\mathbf{y}$ , respectively, and  $\boldsymbol{\Sigma}_{xy} = \text{Cov}[\mathbf{x}, \mathbf{y}]$  is the cross-covariance matrix between  $\mathbf{x}$  and  $\mathbf{y}$ .

The conditional distribution  $p(\mathbf{x} | \mathbf{y})$  is also Gaussian (illustrated in Figure 6.9(c)) and given by (derived in Section 2.3 of Bishop, 2006)

$$p(\mathbf{x} | \mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_{x|y}, \boldsymbol{\Sigma}_{x|y}) \quad (6.65)$$

$$\boldsymbol{\mu}_{x|y} = \boldsymbol{\mu}_x + \boldsymbol{\Sigma}_{xy} \boldsymbol{\Sigma}_{yy}^{-1} (\mathbf{y} - \boldsymbol{\mu}_y) \quad (6.66)$$

$$\boldsymbol{\Sigma}_{x|y} = \boldsymbol{\Sigma}_{xx} - \boldsymbol{\Sigma}_{xy} \boldsymbol{\Sigma}_{yy}^{-1} \boldsymbol{\Sigma}_{yx}. \quad (6.67)$$

Note that in the computation of the mean in (6.66), the  $\mathbf{y}$ -value is an observation and no longer random.

*Remark.* The conditional Gaussian distribution shows up in many places, where we are interested in posterior distributions:

- The Kalman filter (Kalman, 1960), one of the most central algorithms for state estimation in signal processing, does nothing but computing Gaussian conditionals of joint distributions (Deisenroth and Ohlsson, 2011; Särkkä, 2013).
- Gaussian processes (Rasmussen and Williams, 2006), which are a practical implementation of a distribution over functions. In a Gaussian process, we make assumptions of joint Gaussianity of random variables. By (Gaussian) conditioning on observed data, we can determine a posterior distribution over functions.
- Latent linear Gaussian models (Roweis and Ghahramani, 1999; Murphy, 2012), which include probabilistic principal component analysis (PPCA) (Tipping and Bishop, 1999). We will look at PPCA in more detail in Section 10.7.

◇

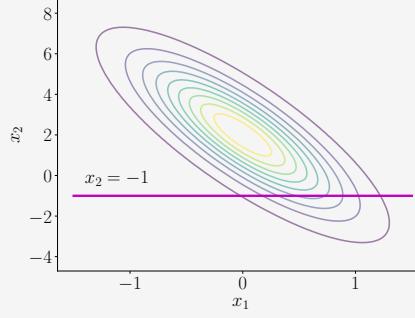
The marginal distribution  $p(\mathbf{x})$  of a joint Gaussian distribution  $p(\mathbf{x}, \mathbf{y})$  (see (6.64)) is itself Gaussian and computed by applying the sum rule (6.20) and given by

$$p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{y}) d\mathbf{y} = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_{xx}). \quad (6.68)$$

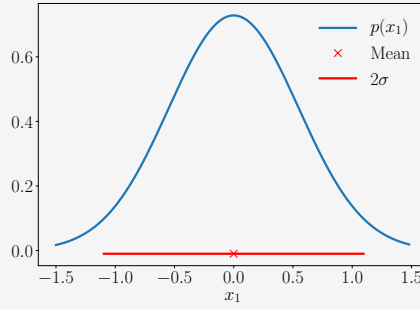
The corresponding result holds for  $p(\mathbf{y})$ , which is obtained by marginalizing with respect to  $\mathbf{x}$ . Intuitively, looking at the joint distribution in (6.64), we ignore (i.e., integrate out) everything we are not interested in. This is illustrated in Figure 6.9(b).

**Example 6.6****Figure 6.9**

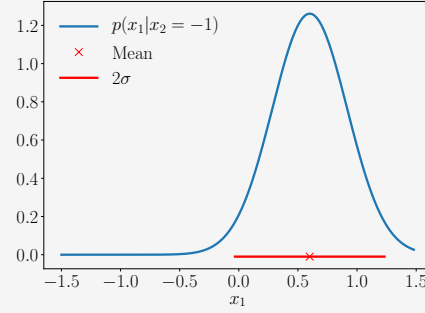
(a) Bivariate Gaussian; (b) marginal of a joint Gaussian distribution is Gaussian; (c) the conditional distribution of a Gaussian is also Gaussian.



(a) Bivariate Gaussian.



(b) Marginal distribution.



(c) Conditional distribution.

Consider the bivariate Gaussian distribution (illustrated in Figure 6.9):

$$p(x_1, x_2) = \mathcal{N} \left( \begin{bmatrix} 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 0.3 & -1 \\ -1 & 5 \end{bmatrix} \right). \quad (6.69)$$

We can compute the parameters of the univariate Gaussian, conditioned on  $x_2 = -1$ , by applying (6.66) and (6.67) to obtain the mean and variance respectively. Numerically, this is

$$\mu_{x_1 | x_2 = -1} = 0 + (-1) \cdot 0.2 \cdot (-1 - 2) = 0.6 \quad (6.70)$$

and

$$\sigma_{x_1 | x_2 = -1}^2 = 0.3 - (-1) \cdot 0.2 \cdot (-1) = 0.1. \quad (6.71)$$

Therefore, the conditional Gaussian is given by

$$p(x_1 | x_2 = -1) = \mathcal{N}(0.6, 0.1). \quad (6.72)$$

The marginal distribution  $p(x_1)$ , in contrast, can be obtained by applying (6.68), which is essentially using the mean and variance of the random variable  $x_1$ , giving us

$$p(x_1) = \mathcal{N}(0, 0.3). \quad (6.73)$$

### 6.5.2 Product of Gaussian Densities

For linear regression (Chapter 9), we need to compute a Gaussian likelihood. Furthermore, we may wish to assume a Gaussian prior (Section 9.3). We apply Bayes' Theorem to compute the posterior, which results in a multiplication of the likelihood and the prior, that is, the multiplication of two Gaussian densities. The *product* of two Gaussians  $\mathcal{N}(\mathbf{x} | \mathbf{a}, \mathbf{A})\mathcal{N}(\mathbf{x} | \mathbf{b}, \mathbf{B})$  is a Gaussian distribution scaled by a  $c \in \mathbb{R}$ , given by  $c\mathcal{N}(\mathbf{x} | \mathbf{c}, \mathbf{C})$  with

$$\mathbf{C} = (\mathbf{A}^{-1} + \mathbf{B}^{-1})^{-1} \quad (6.74)$$

$$\mathbf{c} = \mathbf{C}(\mathbf{A}^{-1}\mathbf{a} + \mathbf{B}^{-1}\mathbf{b}) \quad (6.75)$$

$$c = (2\pi)^{-\frac{D}{2}} |\mathbf{A} + \mathbf{B}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{a} - \mathbf{b})^\top (\mathbf{A} + \mathbf{B})^{-1} (\mathbf{a} - \mathbf{b})\right). \quad (6.76)$$

The scaling constant  $c$  itself can be written in the form of a Gaussian density either in  $\mathbf{a}$  or in  $\mathbf{b}$  with an “inflated” covariance matrix  $\mathbf{A} + \mathbf{B}$ , i.e.,  $c = \mathcal{N}(\mathbf{a} | \mathbf{b}, \mathbf{A} + \mathbf{B}) = \mathcal{N}(\mathbf{b} | \mathbf{a}, \mathbf{A} + \mathbf{B})$ .

*Remark.* For notation convenience, we will sometimes use  $\mathcal{N}(\mathbf{x} | \mathbf{m}, \mathbf{S})$  to describe the functional form of a Gaussian density even if  $\mathbf{x}$  is not a random variable. We have just done this in the preceding demonstration when we wrote

$$c = \mathcal{N}(\mathbf{a} | \mathbf{b}, \mathbf{A} + \mathbf{B}) = \mathcal{N}(\mathbf{b} | \mathbf{a}, \mathbf{A} + \mathbf{B}). \quad (6.77)$$

Here, neither  $\mathbf{a}$  nor  $\mathbf{b}$  are random variables. However, writing  $c$  in this way is more compact than (6.76).  $\diamond$

### 6.5.3 Sums and Linear Transformations

If  $X, Y$  are independent Gaussian random variables (i.e., the joint distribution is given as  $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y})$ ) with  $p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$  and  $p(\mathbf{y}) = \mathcal{N}(\mathbf{y} | \boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$ , then  $\mathbf{x} + \mathbf{y}$  is also Gaussian distributed and given by

$$p(\mathbf{x} + \mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_x + \boldsymbol{\mu}_y, \boldsymbol{\Sigma}_x + \boldsymbol{\Sigma}_y). \quad (6.78)$$

Knowing that  $p(\mathbf{x} + \mathbf{y})$  is Gaussian, the mean and covariance matrix can be determined immediately using the results from (6.46) through (6.49). This property will be important when we consider i.i.d. Gaussian noise acting on random variables, as is the case for linear regression (Chapter 9).

#### Example 6.7

Since expectations are linear operations, we can obtain the weighted sum of independent Gaussian random variables

$$p(a\mathbf{x} + b\mathbf{y}) = \mathcal{N}(a\boldsymbol{\mu}_x + b\boldsymbol{\mu}_y, a^2\boldsymbol{\Sigma}_x + b^2\boldsymbol{\Sigma}_y). \quad (6.79)$$

The derivation is an exercise at the end of this chapter.



*Remark.* A case that will be useful in Chapter 11 is the weighted sum of Gaussian densities. This is different from the weighted sum of Gaussian random variables.  $\diamond$

In Theorem 6.12, the random variable  $x$  is from a density that is a mixture of two densities  $p_1(x)$  and  $p_2(x)$ , weighted by  $\alpha$ . The theorem can be generalized to the multivariate random variable case, since linearity of expectations holds also for multivariate random variables. However, the idea of a squared random variable needs to be replaced by  $\mathbf{x}\mathbf{x}^\top$ .

**Theorem 6.12.** *Consider a mixture of two univariate Gaussian densities*

$$p(x) = \alpha p_1(x) + (1 - \alpha)p_2(x), \quad (6.80)$$

where the scalar  $0 < \alpha < 1$  is the mixture weight, and  $p_1(x)$  and  $p_2(x)$  are univariate Gaussian densities (Equation (6.62)) with different parameters, i.e.,  $(\mu_1, \sigma_1^2) \neq (\mu_2, \sigma_2^2)$ .

Then the mean of the mixture density  $p(x)$  is given by the weighted sum of the means of each random variable:

$$\mathbb{E}[x] = \alpha\mu_1 + (1 - \alpha)\mu_2. \quad (6.81)$$

The variance of the mixture density  $p(x)$  is given by

$$\mathbb{V}[x] = [\alpha\sigma_1^2 + (1 - \alpha)\sigma_2^2] + \left( [\alpha\mu_1^2 + (1 - \alpha)\mu_2^2] - [\alpha\mu_1 + (1 - \alpha)\mu_2]^2 \right). \quad (6.82)$$

*Proof* The mean of the mixture density  $p(x)$  is given by the weighted sum of the means of each random variable. We apply the definition of the mean (Definition 6.4), and plug in our mixture (6.80), which yields

$$\mathbb{E}[x] = \int_{-\infty}^{\infty} xp(x)dx \quad (6.83a)$$

$$= \int_{-\infty}^{\infty} (\alpha xp_1(x) + (1 - \alpha)xp_2(x)) dx \quad (6.83b)$$

$$= \alpha \int_{-\infty}^{\infty} xp_1(x)dx + (1 - \alpha) \int_{-\infty}^{\infty} xp_2(x)dx \quad (6.83c)$$

$$= \alpha\mu_1 + (1 - \alpha)\mu_2. \quad (6.83d)$$

To compute the variance, we can use the raw-score version of the variance from (6.44), which requires an expression of the expectation of the squared random variable. Here we use the definition of an expectation of a function (the square) of a random variable (Definition 6.3),

$$\mathbb{E}[x^2] = \int_{-\infty}^{\infty} x^2 p(x)dx \quad (6.84a)$$

$$= \int_{-\infty}^{\infty} (\alpha x^2 p_1(x) + (1 - \alpha)x^2 p_2(x)) dx \quad (6.84b)$$

$$= \alpha \int_{-\infty}^{\infty} x^2 p_1(x) dx + (1 - \alpha) \int_{-\infty}^{\infty} x^2 p_2(x) dx \quad (6.84c)$$

$$= \alpha(\mu_1^2 + \sigma_1^2) + (1 - \alpha)(\mu_2^2 + \sigma_2^2), \quad (6.84d)$$

where in the last equality, we again used the raw-score version of the variance (6.44) giving  $\sigma^2 = \mathbb{E}[x^2] - \mu^2$ . This is rearranged such that the expectation of a squared random variable is the sum of the squared mean and the variance.

Therefore, the variance is given by subtracting (6.83d) from (6.84d),

$$\mathbb{V}[x] = \mathbb{E}[x^2] - (\mathbb{E}[x])^2 \quad (6.85a)$$

$$= \alpha(\mu_1^2 + \sigma_1^2) + (1 - \alpha)(\mu_2^2 + \sigma_2^2) - (\alpha\mu_1 + (1 - \alpha)\mu_2)^2 \quad (6.85b)$$

$$= [\alpha\sigma_1^2 + (1 - \alpha)\sigma_2^2] + \left( [\alpha\mu_1^2 + (1 - \alpha)\mu_2^2] - [\alpha\mu_1 + (1 - \alpha)\mu_2]^2 \right). \quad (6.85c)$$

□

*Remark.* The preceding derivation holds for any density, but since the Gaussian is fully determined by the mean and variance, the mixture density can be determined in closed form. ◇

For a mixture density, the individual components can be considered to be conditional distributions (conditioned on the component identity). Equation (6.85c) is an example of the conditional variance formula, also known as the *law of total variance*, which generally states that for two random variables  $X$  and  $Y$  it holds that  $\mathbb{V}_X[x] = \mathbb{E}_Y[\mathbb{V}_X[x|y]] + \mathbb{V}_Y[\mathbb{E}_X[x|y]]$ , i.e., the (total) variance of  $X$  is the expected conditional variance plus the variance of a conditional mean.

law of total variance

We consider in Example 6.17 a bivariate standard Gaussian random variable  $X$  and performed a linear transformation  $\mathbf{A}x$  on it. The outcome is a Gaussian random variable with mean zero and covariance  $\mathbf{A}\mathbf{A}^\top$ . Observe that adding a constant vector will change the mean of the distribution, without affecting its variance, that is, the random variable  $x + \mu$  is Gaussian with mean  $\mu$  and identity covariance. Hence, any linear/affine transformation of a Gaussian random variable is Gaussian distributed.

Any linear/affine transformation of a Gaussian random variable is also Gaussian distributed.

Consider a Gaussian distributed random variable  $X \sim \mathcal{N}(\mu, \Sigma)$ . For a given matrix  $\mathbf{A}$  of appropriate shape, let  $Y$  be a random variable such that  $y = \mathbf{A}x$  is a transformed version of  $x$ . We can compute the mean of  $y$  by exploiting that the expectation is a linear operator (6.50) as follows:

$$\mathbb{E}[y] = \mathbb{E}[\mathbf{A}x] = \mathbf{A}\mathbb{E}[x] = \mathbf{A}\mu. \quad (6.86)$$

Similarly the variance of  $y$  can be found by using (6.51):

$$\mathbb{V}[y] = \mathbb{V}[\mathbf{A}x] = \mathbf{A}\mathbb{V}[x]\mathbf{A}^\top = \mathbf{A}\Sigma\mathbf{A}^\top. \quad (6.87)$$

This means that the random variable  $y$  is distributed according to

$$p(y) = \mathcal{N}(y | \mathbf{A}\mu, \mathbf{A}\Sigma\mathbf{A}^\top). \quad (6.88)$$

Let us now consider the reverse transformation: when we know that a random variable has a mean that is a linear transformation of another random variable. For a given full rank matrix  $\mathbf{A} \in \mathbb{R}^{M \times N}$ , where  $M \geq N$ , let  $\mathbf{y} \in \mathbb{R}^M$  be a Gaussian random variable with mean  $\mathbf{Ax}$ , i.e.,

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y} | \mathbf{Ax}, \mathbf{\Sigma}). \quad (6.89)$$

What is the corresponding probability distribution  $p(\mathbf{x})$ ? If  $\mathbf{A}$  is invertible, then we can write  $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$  and apply the transformation in the previous paragraph. However, in general  $\mathbf{A}$  is not invertible, and we use an approach similar to that of the pseudo-inverse (3.57). That is, we pre-multiply both sides with  $\mathbf{A}^\top$  and then invert  $\mathbf{A}^\top \mathbf{A}$ , which is symmetric and positive definite, giving us the relation

$$\mathbf{y} = \mathbf{Ax} \iff (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{y} = \mathbf{x}. \quad (6.90)$$

Hence,  $\mathbf{x}$  is a linear transformation of  $\mathbf{y}$ , and we obtain

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{y}, (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{\Sigma} \mathbf{A} (\mathbf{A}^\top \mathbf{A})^{-1}). \quad (6.91)$$

#### 6.5.4 Sampling from Multivariate Gaussian Distributions

We will not explain the subtleties of random sampling on a computer, and the interested reader is referred to Gentle (2004). In the case of a multivariate Gaussian, this process consists of three stages: first, we need a source of pseudo-random numbers that provide a uniform sample in the interval  $[0,1]$ ; second, we use a non-linear transformation such as the Box-Müller transform (Devroye, 1986) to obtain a sample from a univariate Gaussian; and third, we collate a vector of these samples to obtain a sample from a multivariate standard normal  $\mathcal{N}(\mathbf{0}, \mathbf{I})$ .

For a general multivariate Gaussian, that is, where the mean is non zero and the covariance is not the identity matrix, we use the properties of linear transformations of a Gaussian random variable. Assume we are interested in generating samples  $\mathbf{x}_i, i = 1, \dots, n$ , from a multivariate Gaussian distribution with mean  $\boldsymbol{\mu}$  and covariance matrix  $\mathbf{\Sigma}$ . We would like to construct the sample from a sampler that provides samples from the multivariate standard normal  $\mathcal{N}(\mathbf{0}, \mathbf{I})$ .

To obtain samples from a multivariate normal  $\mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma})$ , we can use the properties of a linear transformation of a Gaussian random variable: If  $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ , then  $\mathbf{y} = \mathbf{Ax} + \boldsymbol{\mu}$ , where  $\mathbf{AA}^\top = \mathbf{\Sigma}$  is Gaussian distributed with mean  $\boldsymbol{\mu}$  and covariance matrix  $\mathbf{\Sigma}$ . One convenient choice of  $\mathbf{A}$  is to use the Cholesky decomposition (Section 4.3) of the covariance matrix  $\mathbf{\Sigma} = \mathbf{AA}^\top$ . The Cholesky decomposition has the benefit that  $\mathbf{A}$  is triangular, leading to efficient computation.

To compute the Cholesky factorization of a matrix, it is required that the matrix is symmetric and positive definite (Section 3.2.3). Covariance matrices possess this property.

## 6.6 Conjugacy and the Exponential Family

Many of the probability distributions “with names” that we find in statistics textbooks were discovered to model particular types of phenomena. For example, we have seen the Gaussian distribution in Section 6.5. The distributions are also related to each other in complex ways (Leemis and McQueston, 2008). For a beginner in the field, it can be overwhelming to figure out which distribution to use. In addition, many of these distributions were discovered at a time that statistics and computation were done by pencil and paper. It is natural to ask what are meaningful concepts in the computing age (Efron and Hastie, 2016). In the previous section, we saw that many of the operations required for inference can be conveniently calculated when the distribution is Gaussian. It is worth recalling at this point the desiderata for manipulating probability distributions in the machine learning context:

1. There is some “closure property” when applying the rules of probability, e.g., Bayes’ theorem. By closure, we mean that applying a particular operation returns an object of the same type.
2. As we collect more data, we do not need more parameters to describe the distribution.
3. Since we are interested in learning from data, we want parameter estimation to behave nicely.

It turns out that the class of distributions called the *exponential family* provides the right balance of generality while retaining favorable computation and inference properties. Before we introduce the exponential family, let us see three more members of “named” probability distributions, the Bernoulli (Example 6.8), Binomial (Example 6.9), and Beta (Example 6.10) distributions.

“Computers” used to be a job description.

exponential family

### Example 6.8

The *Bernoulli distribution* is a distribution for a single binary random variable  $X$  with state  $x \in \{0, 1\}$ . It is governed by a single continuous parameter  $\mu \in [0, 1]$  that represents the probability of  $X = 1$ . The Bernoulli distribution  $\text{Ber}(\mu)$  is defined as

$$p(x | \mu) = \mu^x (1 - \mu)^{1-x}, \quad x \in \{0, 1\}, \quad (6.92)$$

$$\mathbb{E}[x] = \mu, \quad (6.93)$$

$$\mathbb{V}[x] = \mu(1 - \mu), \quad (6.94)$$

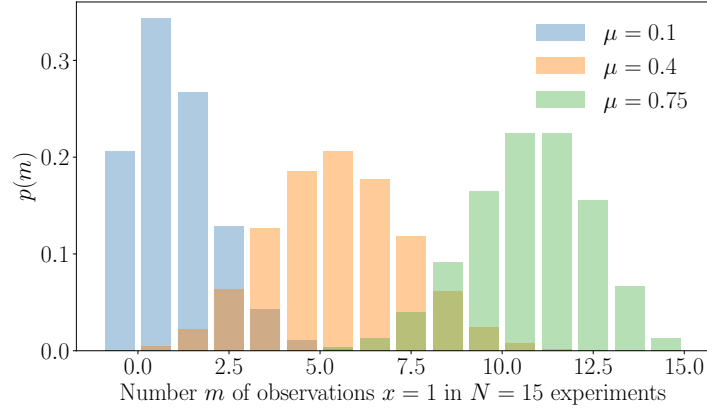
where  $\mathbb{E}[x]$  and  $\mathbb{V}[x]$  are the mean and variance of the binary random variable  $X$ .

Bernoulli distribution



An example where the Bernoulli distribution can be used is when we are interested in modeling the probability of “heads” when flipping a coin.

**Figure 6.10**  
Examples of the  
Binomial  
distribution for  
 $\mu \in \{0.1, 0.4, 0.75\}$   
and  $N = 15$ .



*Remark.* The rewriting above of the Bernoulli distribution, where we use Boolean variables as numerical 0 or 1 and express them in the exponents, is a trick that is often used in machine learning textbooks. Another occurrence of this is when expressing the Multinomial distribution.  $\diamond$

Binomial  
distribution

#### Example 6.9 (Binomial Distribution)

The *Binomial distribution* is a generalization of the Bernoulli distribution to a distribution over integers (illustrated in Figure 6.10). In particular, the Binomial can be used to describe the probability of observing  $m$  occurrences of  $X = 1$  in a set of  $N$  samples from a Bernoulli distribution where  $p(X = 1) = \mu \in [0, 1]$ . The Binomial distribution  $\text{Bin}(N, \mu)$  is defined as

$$p(m | N, \mu) = \binom{N}{m} \mu^m (1 - \mu)^{N-m}, \quad (6.95)$$

$$\mathbb{E}[m] = N\mu, \quad (6.96)$$

$$\mathbb{V}[m] = N\mu(1 - \mu), \quad (6.97)$$

where  $\mathbb{E}[m]$  and  $\mathbb{V}[m]$  are the mean and variance of  $m$ , respectively.

An example where the Binomial could be used is if we want to describe the probability of observing  $m$  “heads” in  $N$  coin-flip experiments if the probability for observing head in a single experiment is  $\mu$ .

Beta distribution

#### Example 6.10 (Beta Distribution)

We may wish to model a continuous random variable on a finite interval. The *Beta distribution* is a distribution over a continuous random variable  $\mu \in [0, 1]$ , which is often used to represent the probability for some binary event (e.g., the parameter governing the Bernoulli distribution). The Beta

distribution  $\text{Beta}(\alpha, \beta)$  (illustrated in Figure 6.11) itself is governed by two parameters  $\alpha > 0$ ,  $\beta > 0$  and is defined as

$$p(\mu | \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \mu^{\alpha-1} (1 - \mu)^{\beta-1} \quad (6.98)$$

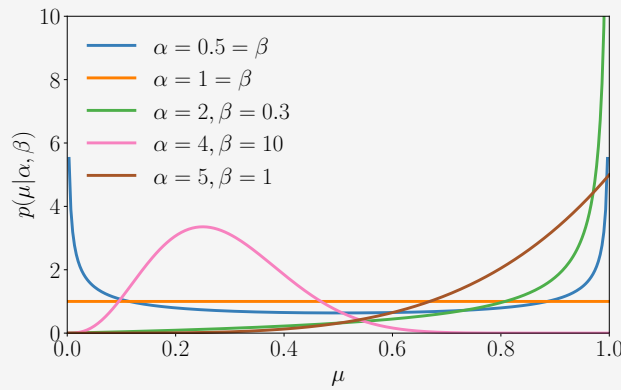
$$\mathbb{E}[\mu] = \frac{\alpha}{\alpha + \beta}, \quad \mathbb{V}[\mu] = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)} \quad (6.99)$$

where  $\Gamma(\cdot)$  is the Gamma function defined as

$$\Gamma(t) := \int_0^\infty x^{t-1} \exp(-x) dx, \quad t > 0. \quad (6.100)$$

$$\Gamma(t + 1) = t\Gamma(t). \quad (6.101)$$

Note that the fraction of Gamma functions in (6.98) normalizes the Beta distribution.



**Figure 6.11**  
Examples of the  
Beta distribution for  
different values of  $\alpha$   
and  $\beta$ .

Intuitively,  $\alpha$  moves probability mass toward 1, whereas  $\beta$  moves probability mass toward 0. There are some special cases (Murphy, 2012):

- For  $\alpha = 1 = \beta$ , we obtain the uniform distribution  $\mathcal{U}[0, 1]$ .
- For  $\alpha, \beta < 1$ , we get a bimodal distribution with spikes at 0 and 1.
- For  $\alpha, \beta > 1$ , the distribution is unimodal.
- For  $\alpha, \beta > 1$  and  $\alpha = \beta$ , the distribution is unimodal, symmetric, and centered in the interval  $[0, 1]$ , i.e., the mode/mean is at  $\frac{1}{2}$ .

*Remark.* There is a whole zoo of distributions with names, and they are related in different ways to each other (Leemis and McQueston, 2008). It is worth keeping in mind that each named distribution is created for a particular reason, but may have other applications. Knowing the reason behind the creation of a particular distribution often allows insight into how to best use it. We introduced the preceding three distributions to be

able to illustrate the concepts of conjugacy (Section 6.6.1) and exponential families (Section 6.6.3).  $\diamond$

### 6.6.1 Conjugacy

According to Bayes' theorem (6.23), the posterior is proportional to the product of the prior and the likelihood. The specification of the prior can be tricky for two reasons: First, the prior should encapsulate our knowledge about the problem before we see any data. This is often difficult to describe. Second, it is often not possible to compute the posterior distribution analytically. However, there are some priors that are computationally convenient: *conjugate priors*.

conjugate prior

conjugate

**Definition 6.13** (Conjugate Prior). A prior is *conjugate* for the likelihood function if the posterior is of the same form/type as the prior.

Conjugacy is particularly convenient because we can algebraically calculate our posterior distribution by updating the parameters of the prior distribution.

*Remark.* When considering the geometry of probability distributions, conjugate priors retain the same distance structure as the likelihood (Agarwal and Daumé III, 2010).  $\diamond$

To introduce a concrete example of conjugate priors, we describe in Example 6.11 the Binomial distribution (defined on discrete random variables) and the Beta distribution (defined on continuous random variables).

#### Example 6.11 (Beta-Binomial Conjugacy)

Consider a Binomial random variable  $x \sim \text{Bin}(N, \mu)$  where

$$p(x | N, \mu) = \binom{N}{x} \mu^x (1 - \mu)^{N-x}, \quad x = 0, 1, \dots, N, \quad (6.102)$$

is the probability of finding  $x$  times the outcome “heads” in  $N$  coin flips, where  $\mu$  is the probability of a “head”. We place a Beta prior on the parameter  $\mu$ , that is,  $\mu \sim \text{Beta}(\alpha, \beta)$ , where

$$p(\mu | \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \mu^{\alpha-1} (1 - \mu)^{\beta-1}. \quad (6.103)$$

If we now observe some outcome  $x = h$ , that is, we see  $h$  heads in  $N$  coin flips, we compute the posterior distribution on  $\mu$  as

$$p(\mu | x = h, N, \alpha, \beta) \propto p(x | N, \mu) p(\mu | \alpha, \beta) \quad (6.104a)$$

$$\propto \mu^h (1 - \mu)^{(N-h)} \mu^{\alpha-1} (1 - \mu)^{\beta-1} \quad (6.104b)$$

$$= \mu^{h+\alpha-1} (1 - \mu)^{(N-h)+\beta-1} \quad (6.104c)$$

Likelihood	Conjugate prior	Posterior
Bernoulli	Beta	Beta
Binomial	Beta	Beta
Gaussian	Gaussian/inverse Gamma	Gaussian/inverse Gamma
Gaussian	Gaussian/inverse Wishart	Gaussian/inverse Wishart
Multinomial	Dirichlet	Dirichlet

**Table 6.2** Examples of conjugate priors for common likelihood functions.

$$\propto \text{Beta}(h + \alpha, N - h + \beta), \quad (6.104d)$$

i.e., the posterior distribution is a Beta distribution as the prior, i.e., the Beta prior is conjugate for the parameter  $\mu$  in the Binomial likelihood function.

In the following example, we will derive a result that is similar to the Beta-Binomial conjugacy result. Here we will show that the Beta distribution is a conjugate prior for the Bernoulli distribution.

#### Example 6.12 (Beta-Bernoulli Conjugacy)

Let  $x \in \{0, 1\}$  be distributed according to the Bernoulli distribution with parameter  $\theta \in [0, 1]$ , that is,  $p(x = 1 | \theta) = \theta$ . This can also be expressed as  $p(x | \theta) = \theta^x (1 - \theta)^{1-x}$ . Let  $\theta$  be distributed according to a Beta distribution with parameters  $\alpha, \beta$ , that is,  $p(\theta | \alpha, \beta) \propto \theta^{\alpha-1} (1 - \theta)^{\beta-1}$ .

Multiplying the Beta and the Bernoulli distributions, we get

$$p(\theta | x, \alpha, \beta) = p(x | \theta) p(\theta | \alpha, \beta) \quad (6.105a)$$

$$\propto \theta^x (1 - \theta)^{1-x} \theta^{\alpha-1} (1 - \theta)^{\beta-1} \quad (6.105b)$$

$$= \theta^{\alpha+x-1} (1 - \theta)^{\beta+(1-x)-1} \quad (6.105c)$$

$$\propto p(\theta | \alpha + x, \beta + (1 - x)). \quad (6.105d)$$

The last line is the Beta distribution with parameters  $(\alpha + x, \beta + (1 - x))$ .

Table 6.2 lists examples for conjugate priors for the parameters of some standard likelihoods used in probabilistic modeling. Distributions such as Multinomial, inverse Gamma, inverse Wishart, and Dirichlet can be found in any statistical text, and are described in Bishop (2006), for example.

The Beta distribution is the conjugate prior for the parameter  $\mu$  in both the Binomial and the Bernoulli likelihood. For a Gaussian likelihood function, we can place a conjugate Gaussian prior on the mean. The reason why the Gaussian likelihood appears twice in the table is that we need to distinguish the univariate from the multivariate case. In the univariate (scalar) case, the inverse Gamma is the conjugate prior for the variance. In the multivariate case, we use a conjugate inverse Wishart distribution as a prior on the covariance matrix. The Dirichlet distribution is the conju-

The Gamma prior is conjugate for the precision (inverse variance) in the univariate Gaussian likelihood, and the Wishart prior is conjugate for the precision matrix (inverse covariance matrix) in the multivariate Gaussian likelihood.



gate prior for the multinomial likelihood function. For further details, we refer to Bishop (2006).

### 6.6.2 Sufficient Statistics

sufficient statistics

Recall that a statistic of a random variable is a deterministic function of that random variable. For example, if  $\mathbf{x} = [x_1, \dots, x_N]^\top$  is a vector of univariate Gaussian random variables, that is,  $x_n \sim \mathcal{N}(\mu, \sigma^2)$ , then the sample mean  $\hat{\mu} = \frac{1}{N}(x_1 + \dots + x_N)$  is a statistic. Sir Ronald Fisher discovered the notion of *sufficient statistics*: the idea that there are statistics that will contain all available information that can be inferred from data corresponding to the distribution under consideration. In other words, sufficient statistics carry all the information needed to make inference about the population, that is, they are the statistics that are sufficient to represent the distribution.

For a set of distributions parametrized by  $\theta$ , let  $X$  be a random variable with distribution  $p(x | \theta_0)$  given an unknown  $\theta_0$ . A vector  $\phi(x)$  of statistics is called sufficient statistics for  $\theta_0$  if they contain all possible information about  $\theta_0$ . To be more formal about “contain all possible information”, this means that the probability of  $x$  given  $\theta$  can be factored into a part that does not depend on  $\theta$ , and a part that depends on  $\theta$  only via  $\phi(x)$ . The Fisher-Neyman factorization theorem formalizes this notion, which we state in Theorem 6.14 without proof.

Fisher-Neyman  
theorem

**Theorem 6.14** (Fisher-Neyman). *[Theorem 6.5 in Lehmann and Casella (1998)] Let  $X$  have probability density function  $p(x | \theta)$ . Then the statistics  $\phi(x)$  are sufficient for  $\theta$  if and only if  $p(x | \theta)$  can be written in the form*

$$p(x | \theta) = h(x)g_\theta(\phi(x)), \quad (6.106)$$

where  $h(x)$  is a distribution independent of  $\theta$  and  $g_\theta$  captures all the dependence on  $\theta$  via sufficient statistics  $\phi(x)$ .

If  $p(x | \theta)$  does not depend on  $\theta$ , then  $\phi(x)$  is trivially a sufficient statistic for any function  $\phi$ . The more interesting case is that  $p(x | \theta)$  is dependent only on  $\phi(x)$  and not  $x$  itself. In this case,  $\phi(x)$  is a sufficient statistic for  $\theta$ .

In machine learning, we consider a finite number of samples from a distribution. One could imagine that for simple distributions (such as the Bernoulli in Example 6.8) we only need a small number of samples to estimate the parameters of the distributions. We could also consider the opposite problem: If we have a set of data (a sample from an unknown distribution), which distribution gives the best fit? A natural question to ask is, as we observe more data, do we need more parameters  $\theta$  to describe the distribution? It turns out that the answer is yes in general, and this is studied in non-parametric statistics (Wasserman, 2007). A converse question is to consider which class of distributions have finite-dimensional

sufficient statistics, that is the number of parameters needed to describe them does not increase arbitrarily. The answer is exponential family distributions, described in the following section.

### 6.6.3 Exponential Family

There are three possible levels of abstraction we can have when considering distributions (of discrete or continuous random variables). At level one (the most concrete end of the spectrum), we have a particular named distribution with fixed parameters, for example a univariate Gaussian  $\mathcal{N}(0, 1)$  with zero mean and unit variance. In machine learning, we often use the second level of abstraction, that is, we fix the parametric form (the univariate Gaussian) and infer the parameters from data. For example, we assume a univariate Gaussian  $\mathcal{N}(\mu, \sigma^2)$  with unknown mean  $\mu$  and unknown variance  $\sigma^2$ , and use a maximum likelihood fit to determine the best parameters  $(\mu, \sigma^2)$ . We will see an example of this when considering linear regression in Chapter 9. A third level of abstraction is to consider families of distributions, and in this book, we consider the exponential family. The univariate Gaussian is an example of a member of the exponential family. Many of the widely used statistical models, including all the “named” models in Table 6.2, are members of the exponential family. They can all be unified into one concept (Brown, 1986).

*Remark.* A brief historical anecdote: Like many concepts in mathematics and science, exponential families were independently discovered at the same time by different researchers. In the years 1935–1936, Edwin Pitman in Tasmania, Georges Darrois in Paris, and Bernard Koopman in New York independently showed that the exponential families are the only families that enjoy finite-dimensional sufficient statistics under repeated independent sampling (Lehmann and Casella, 1998).  $\diamond$

An *exponential family* is a family of probability distributions, parameterized by  $\theta \in \mathbb{R}^D$ , of the form

exponential family

$$p(x | \theta) = h(x) \exp(\langle \theta, \phi(x) \rangle - A(\theta)), \quad (6.107)$$

where  $\phi(x)$  is the vector of sufficient statistics. In general, any inner product (Section 3.2) can be used in (6.107), and for concreteness we will use the standard dot product here ( $\langle \theta, \phi(x) \rangle = \theta^\top \phi(x)$ ). Note that the form of the exponential family is essentially a particular expression of  $g_\theta(\phi(x))$  in the Fisher-Neyman theorem (Theorem 6.14).

The factor  $h(x)$  can be absorbed into the dot product term by adding another entry ( $\log h(x)$ ) to the vector of sufficient statistics  $\phi(x)$ , and constraining the corresponding parameter  $\theta_0 = 1$ . The term  $A(\theta)$  is the normalization constant that ensures that the distribution sums up or integrates to one and is called the *log-partition function*. A good intuitive notion of exponential families can be obtained by ignoring these two terms

log-partition  
function

and considering exponential families as distributions of the form

$$p(\mathbf{x} | \boldsymbol{\theta}) \propto \exp(\boldsymbol{\theta}^\top \boldsymbol{\phi}(\mathbf{x})). \quad (6.108)$$

natural parameters

For this form of parametrization, the parameters  $\boldsymbol{\theta}$  are called the *natural parameters*. At first glance, it seems that exponential families are a mundane transformation by adding the exponential function to the result of a dot product. However, there are many implications that allow for convenient modeling and efficient computation based on the fact that we can capture information about data in  $\boldsymbol{\phi}(\mathbf{x})$ .

### Example 6.13 (Gaussian as Exponential Family)

Consider the univariate Gaussian distribution  $\mathcal{N}(\mu, \sigma^2)$ . Let  $\boldsymbol{\phi}(x) = \begin{bmatrix} x \\ x^2 \end{bmatrix}$ . Then by using the definition of the exponential family,

$$p(x | \boldsymbol{\theta}) \propto \exp(\theta_1 x + \theta_2 x^2). \quad (6.109)$$

Setting

$$\boldsymbol{\theta} = \left[ \frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2} \right]^\top \quad (6.110)$$

and substituting into (6.109), we obtain

$$p(x | \boldsymbol{\theta}) \propto \exp\left(\frac{\mu x}{\sigma^2} - \frac{x^2}{2\sigma^2}\right) \propto \exp\left(-\frac{1}{2\sigma^2}(x - \mu)^2\right). \quad (6.111)$$

Therefore, the univariate Gaussian distribution is a member of the exponential family with sufficient statistic  $\boldsymbol{\phi}(x) = \begin{bmatrix} x \\ x^2 \end{bmatrix}$ , and natural parameters given by  $\boldsymbol{\theta}$  in (6.110).

### Example 6.14 (Bernoulli as Exponential Family)

Recall the Bernoulli distribution from Example 6.8

$$p(x | \mu) = \mu^x (1 - \mu)^{1-x}, \quad x \in \{0, 1\}. \quad (6.112)$$

This can be written in exponential family form

$$p(x | \mu) = \exp[\log(\mu^x (1 - \mu)^{1-x})] \quad (6.113a)$$

$$= \exp[x \log \mu + (1 - x) \log(1 - \mu)] \quad (6.113b)$$

$$= \exp[x \log \mu - x \log(1 - \mu) + \log(1 - \mu)] \quad (6.113c)$$

$$= \exp\left[x \log \frac{\mu}{1 - \mu} + \log(1 - \mu)\right]. \quad (6.113d)$$

The last line (6.113d) can be identified as being in exponential family form (6.107) by observing that

$$h(x) = 1 \quad (6.114)$$

$$\theta = \log \frac{\mu}{1-\mu} \quad (6.115)$$

$$\phi(x) = x \quad (6.116)$$

$$A(\theta) = -\log(1 - \mu) = \log(1 + \exp(\theta)). \quad (6.117)$$

The relationship between  $\theta$  and  $\mu$  is invertible so that

$$\mu = \frac{1}{1 + \exp(-\theta)}. \quad (6.118)$$

The relation (6.118) is used to obtain the right equality of (6.117).

*Remark.* The relationship between the original Bernoulli parameter  $\mu$  and the natural parameter  $\theta$  is known as the *sigmoid* or logistic function. Observe that  $\mu \in (0, 1)$  but  $\theta \in \mathbb{R}$ , and therefore the sigmoid function squeezes a real value into the range  $(0, 1)$ . This property is useful in machine learning, for example it is used in logistic regression (Bishop, 2006, section 4.3.2), as well as as a nonlinear activation functions in neural networks (Goodfellow et al., 2016, chapter 6).  $\diamond$

It is often not obvious how to find the parametric form of the conjugate distribution of a particular distribution (for example, those in Table 6.2). Exponential families provide a convenient way to find conjugate pairs of distributions. Consider the random variable  $X$  is a member of the exponential family (6.107):

$$p(x | \theta) = h(x) \exp(\langle \theta, \phi(x) \rangle - A(\theta)). \quad (6.119)$$

Every member of the exponential family has a conjugate prior (Brown, 1986)

$$p(\theta | \gamma) = h_c(\theta) \exp\left(\left\langle \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix}, \begin{bmatrix} \theta \\ -A(\theta) \end{bmatrix} \right\rangle - A_c(\gamma)\right), \quad (6.120)$$

where  $\gamma = \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix}$  has dimension  $\dim(\theta) + 1$ . The sufficient statistics of the conjugate prior are  $\begin{bmatrix} \theta \\ -A(\theta) \end{bmatrix}$ . By using the knowledge of the general form of conjugate priors for exponential families, we can derive functional forms of conjugate priors corresponding to particular distributions.

### Example 6.15

Recall the exponential family form of the Bernoulli distribution (6.113d)

$$p(x | \mu) = \exp\left[x \log \frac{\mu}{1-\mu} + \log(1 - \mu)\right]. \quad (6.121)$$

The canonical conjugate prior has the form

$$p(\mu | \alpha, \beta) = \frac{\mu}{1 - \mu} \exp \left[ \alpha \log \frac{\mu}{1 - \mu} + (\beta + \alpha) \log(1 - \mu) - A_c(\gamma) \right], \quad (6.122)$$

where we defined  $\gamma := [\alpha, \beta + \alpha]^\top$  and  $h_c(\mu) := \mu/(1 - \mu)$ . Equation (6.122) then simplifies to

$$p(\mu | \alpha, \beta) = \exp [(\alpha - 1) \log \mu + (\beta - 1) \log(1 - \mu) - A_c(\alpha, \beta)]. \quad (6.123)$$

Putting this in non-exponential family form yields

$$p(\mu | \alpha, \beta) \propto \mu^{\alpha-1} (1 - \mu)^{\beta-1}, \quad (6.124)$$

which we identify as the Beta distribution (6.98). In example 6.12, we assumed that the Beta distribution is the conjugate prior of the Bernoulli distribution and showed that it was indeed the conjugate prior. In this example, we derived the form of the Beta distribution by looking at the canonical conjugate prior of the Bernoulli distribution in exponential family form.

As mentioned in the previous section, the main motivation for exponential families is that they have finite-dimensional sufficient statistics. Additionally, conjugate distributions are easy to write down, and the conjugate distributions also come from an exponential family. From an inference perspective, maximum likelihood estimation behaves nicely because empirical estimates of sufficient statistics are optimal estimates of the population values of sufficient statistics (recall the mean and covariance of a Gaussian). From an optimization perspective, the log-likelihood function is concave, allowing for efficient optimization approaches to be applied (Chapter 7).

## 6.7 Change of Variables/Inverse Transform

It may seem that there are very many known distributions, but in reality the set of distributions for which we have names is quite limited. Therefore, it is often useful to understand how transformed random variables are distributed. For example, assuming that  $X$  is a random variable distributed according to the univariate normal distribution  $\mathcal{N}(0, 1)$ , what is the distribution of  $X^2$ ? Another example, which is quite common in machine learning, is, given that  $X_1$  and  $X_2$  are univariate standard normal, what is the distribution of  $\frac{1}{2}(X_1 + X_2)$ ?

One option to work out the distribution of  $\frac{1}{2}(X_1 + X_2)$  is to calculate the mean and variance of  $X_1$  and  $X_2$  and then combine them. As we saw in Section 6.4.4, we can calculate the mean and variance of resulting random variables when we consider affine transformations of random vari-

ables. However, we may not be able to obtain the functional form of the distribution under transformations. Furthermore, we may be interested in nonlinear transformations of random variables for which closed-form expressions are not readily available.

*Remark (Notation).* In this section, we will be explicit about random variables and the values they take. Hence, recall that we use capital letters  $X, Y$  to denote random variables and small letters  $x, y$  to denote the values in the target space  $\mathcal{T}$  that the random variables take. We will explicitly write pmfs of discrete random variables  $X$  as  $P(X = x)$ . For continuous random variables  $X$  (Section 6.2.2), the pdf is written as  $f(x)$  and the cdf is written as  $F_X(x)$ .  $\diamond$

We will look at two approaches for obtaining distributions of transformations of random variables: a direct approach using the definition of a cumulative distribution function and a change-of-variable approach that uses the chain rule of calculus (Section 5.2.2). The change-of-variable approach is widely used because it provides a “recipe” for attempting to compute the resulting distribution due to a transformation. We will explain the techniques for univariate random variables, and will only briefly provide the results for the general case of multivariate random variables.

Transformations of discrete random variables can be understood directly. Suppose that there is a discrete random variable  $X$  with pmf  $P(X = x)$  (Section 6.2.1), and an invertible function  $U(x)$ . Consider the transformed random variable  $Y := U(X)$ , with pmf  $P(Y = y)$ . Then

$$P(Y = y) = P(U(X) = y) \quad \text{transformation of interest} \quad (6.125a)$$

$$= P(X = U^{-1}(y)) \quad \text{inverse} \quad (6.125b)$$

where we can observe that  $x = U^{-1}(y)$ . Therefore, for discrete random variables, transformations directly change the individual events (with the probabilities appropriately transformed).

Moment generating functions can also be used to study transformations of random variables (Casella and Berger, 2002, chapter 2).

### 6.7.1 Distribution Function Technique

The distribution function technique goes back to first principles, and uses the definition of a cdf  $F_X(x) = P(X \leq x)$  and the fact that its differential is the pdf  $f(x)$  (Wasserman, 2004, chapter 2). For a random variable  $X$  and a function  $U$ , we find the pdf of the random variable  $Y := U(X)$  by

1. Finding the cdf:

$$F_Y(y) = P(Y \leq y) \quad (6.126)$$

2. Differentiating the cdf  $F_Y(y)$  to get the pdf  $f(y)$ .

$$f(y) = \frac{d}{dy} F_Y(y). \quad (6.127)$$

We also need to keep in mind that the domain of the random variable may have changed due to the transformation by  $U$ .

**Example 6.16**

Let  $X$  be a continuous random variable with probability density function on  $0 \leq x \leq 1$

$$f(x) = 3x^2. \quad (6.128)$$

We are interested in finding the pdf of  $Y = X^2$ .

The function  $f$  is an increasing function of  $x$ , and therefore the resulting value of  $y$  lies in the interval  $[0, 1]$ . We obtain

$$F_Y(y) = P(Y \leq y) \quad \text{definition of cdf} \quad (6.129a)$$

$$= P(X^2 \leq y) \quad \text{transformation of interest} \quad (6.129b)$$

$$= P(X \leq y^{\frac{1}{2}}) \quad \text{inverse} \quad (6.129c)$$

$$= F_X(y^{\frac{1}{2}}) \quad \text{definition of cdf} \quad (6.129d)$$

$$= \int_0^{y^{\frac{1}{2}}} 3t^2 dt \quad \text{cdf as a definite integral} \quad (6.129e)$$

$$= [t^3]_{t=0}^{t=y^{\frac{1}{2}}} \quad \text{result of integration} \quad (6.129f)$$

$$= y^{\frac{3}{2}}, \quad 0 \leq y \leq 1. \quad (6.129g)$$

Therefore, the cdf of  $Y$  is

$$F_Y(y) = y^{\frac{3}{2}} \quad (6.130)$$

for  $0 \leq y \leq 1$ . To obtain the pdf, we differentiate the cdf

$$f(y) = \frac{d}{dy} F_Y(y) = \frac{3}{2} y^{\frac{1}{2}} \quad (6.131)$$

for  $0 \leq y \leq 1$ .

Functions that have inverses are called bijective functions (Section 2.7).

In Example 6.16, we considered a strictly monotonically increasing function  $f(x) = 3x^2$ . This means that we could compute an inverse function. In general, we require that the function of interest  $y = U(x)$  has an inverse  $x = U^{-1}(y)$ . A useful result can be obtained by considering the cumulative distribution function  $F_X(x)$  of a random variable  $X$ , and using it as the transformation  $U(x)$ . This leads to the following theorem.

**Theorem 6.15.** [Theorem 2.1.10 in Casella and Berger (2002)] Let  $X$  be a continuous random variable with a strictly monotonic cumulative distribution function  $F_X(x)$ . Then the random variable  $Y$  defined as

$$Y := F_X(X) \quad (6.132)$$

has a uniform distribution.

Theorem 6.15 is known as the *probability integral transform*, and it is used to derive algorithms for sampling from distributions by transforming the result of sampling from a uniform random variable (Bishop, 2006). The algorithm works by first generating a sample from a uniform distribution, then transforming it by the inverse cdf (assuming this is available) to obtain a sample from the desired distribution. The probability integral transform is also used for hypothesis testing whether a sample comes from a particular distribution (Lehmann and Romano, 2005). The idea that the output of a cdf gives a uniform distribution also forms the basis of copulas (Nelsen, 2006).

probability integral transform

### 6.7.2 Change of Variables

The distribution function technique in Section 6.7.1 is derived from first principles, based on the definitions of cdfs and using properties of inverses, differentiation, and integration. This argument from first principles relies on two facts:

1. We can transform the cdf of  $Y$  into an expression that is a cdf of  $X$ .
2. We can differentiate the cdf to obtain the pdf.

Let us break down the reasoning step by step, with the goal of understanding the more general change-of-variables approach in Theorem 6.16.

*Remark.* The name “change of variables” comes from the idea of changing the variable of integration when faced with a difficult integral. For univariate functions, we use the substitution rule of integration,

$$\int f(g(x))g'(x)dx = \int f(u)du, \quad \text{where } u = g(x). \quad (6.133)$$

The derivation of this rule is based on the chain rule of calculus (5.32) and by applying twice the fundamental theorem of calculus. The fundamental theorem of calculus formalizes the fact that integration and differentiation are somehow “inverses” of each other. An intuitive understanding of the rule can be obtained by thinking (loosely) about small changes (differentials) to the equation  $u = g(x)$ , that is by considering  $\Delta u = g'(x)\Delta x$  as a differential of  $u = g(x)$ . By substituting  $u = g(x)$ , the argument inside the integral on the right-hand side of (6.133) becomes  $f(g(x))$ . By pretending that the term  $du$  can be approximated by  $du \approx \Delta u = g'(x)\Delta x$ , and that  $dx \approx \Delta x$ , we obtain (6.133).  $\diamond$

Change of variables in probability relies on the change-of-variables method in calculus (Tandra, 2014).

Consider a univariate random variable  $X$ , and an *invertible* function  $U$ , which gives us another random variable  $Y = U(X)$ . We assume that random variable  $X$  has states  $x \in [a, b]$ . By the definition of the cdf, we have

$$F_Y(y) = P(Y \leq y). \quad (6.134)$$



We are interested in a function  $U$  of the random variable

$$P(Y \leq y) = P(U(X) \leq y), \quad (6.135)$$

where we assume that the function  $U$  is invertible. An invertible function on an interval is either strictly increasing or strictly decreasing. In the case that  $U$  is strictly increasing, then its inverse  $U^{-1}$  is also strictly increasing. By applying the inverse  $U^{-1}$  to the arguments of  $P(U(X) \leq y)$ , we obtain

$$P(U(X) \leq y) = P(U^{-1}(U(X)) \leq U^{-1}(y)) = P(X \leq U^{-1}(y)). \quad (6.136)$$

The right-most term in (6.136) is an expression of the cdf of  $X$ . Recall the definition of the cdf in terms of the pdf

$$P(X \leq U^{-1}(y)) = \int_a^{U^{-1}(y)} f(x) dx. \quad (6.137)$$

Now we have an expression of the cdf of  $Y$  in terms of  $x$ :

$$F_Y(y) = \int_a^{U^{-1}(y)} f(x) dx. \quad (6.138)$$

To obtain the pdf, we differentiate (6.138) with respect to  $y$ :

$$f(y) = \frac{d}{dy} F_Y(y) = \frac{d}{dy} \int_a^{U^{-1}(y)} f(x) dx. \quad (6.139)$$

Note that the integral on the right-hand side is with respect to  $x$ , but we need an integral with respect to  $y$  because we are differentiating with respect to  $y$ . In particular, we use (6.133) to get the substitution

$$\int f(U^{-1}(y)) U^{-1'}(y) dy = \int f(x) dx \quad \text{where } x = U^{-1}(y). \quad (6.140)$$

Using (6.140) on the right-hand side of (6.139) gives us

$$f(y) = \frac{d}{dy} \int_a^{U^{-1}(y)} f_x(U^{-1}(y)) U^{-1'}(y) dy. \quad (6.141)$$

We then recall that differentiation is a linear operator and we use the subscript  $x$  to remind ourselves that  $f_x(U^{-1}(y))$  is a function of  $x$  and not  $y$ . Invoking the fundamental theorem of calculus again gives us

$$f(y) = f_x(U^{-1}(y)) \cdot \left( \frac{d}{dy} U^{-1}(y) \right). \quad (6.142)$$

Recall that we assumed that  $U$  is a strictly increasing function. For decreasing functions, it turns out that we have a negative sign when we follow the same derivation. We introduce the absolute value of the differential to have the same expression for both increasing and decreasing  $U$ :

$$f(y) = f_x(U^{-1}(y)) \cdot \left| \frac{d}{dy} U^{-1}(y) \right|. \quad (6.143)$$

This is called the *change-of-variable technique*. The term  $\left| \frac{d}{dy} U^{-1}(y) \right|$  in (6.143) measures how much a unit volume changes when applying  $U$  (see also the definition of the Jacobian in Section 5.3).

change-of-variable  
technique

*Remark.* In comparison to the discrete case in (6.125b), we have an additional factor  $\left| \frac{d}{dy} U^{-1}(y) \right|$ . The continuous case requires more care because  $P(Y = y) = 0$  for all  $y$ . The probability density function  $f(y)$  does not have a description as a probability of an event involving  $y$ .  $\diamond$

So far in this section, we have been studying univariate change of variables. The case for multivariate random variables is analogous, but complicated by fact that the absolute value cannot be used for multivariate functions. Instead, we use the determinant of the Jacobian matrix. Recall from (5.58) that the Jacobian is a matrix of partial derivatives, and that the existence of a nonzero determinant shows that we can invert the Jacobian. Recall the discussion in Section 4.1 that the determinant arises because our differentials (cubes of volume) are transformed into parallelepipeds by the Jacobian. Let us summarize preceding the discussion in the following theorem, which gives us a recipe for multivariate change of variables.

**Theorem 6.16.** [Theorem 17.2 in Billingsley (1995)] Let  $f(\mathbf{x})$  be the value of the probability density of the multivariate continuous random variable  $X$ . If the vector-valued function  $\mathbf{y} = U(\mathbf{x})$  is differentiable and invertible for all values within the domain of  $\mathbf{x}$ , then for corresponding values of  $\mathbf{y}$ , the probability density of  $Y = U(X)$  is given by

$$f(\mathbf{y}) = f_{\mathbf{x}}(U^{-1}(\mathbf{y})) \cdot \left| \det \left( \frac{\partial}{\partial \mathbf{y}} U^{-1}(\mathbf{y}) \right) \right|. \quad (6.144)$$

The theorem looks intimidating at first glance, but the key point is that a change of variable of a multivariate random variable follows the procedure of the univariate change of variable. First we need to work out the inverse transform, and substitute that into the density of  $\mathbf{x}$ . Then we calculate the determinant of the Jacobian and multiply the result. The following example illustrates the case of a bivariate random variable.

#### Example 6.17

Consider a bivariate random variable  $X$  with states  $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$  and probability density function

$$f \left( \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right) = \frac{1}{2\pi} \exp \left( -\frac{1}{2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^\top \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right). \quad (6.145)$$

We use the change-of-variable technique from Theorem 6.16 to derive the

effect of a linear transformation (Section 2.7) of the random variable. Consider a matrix  $\mathbf{A} \in \mathbb{R}^{2 \times 2}$  defined as

$$\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}. \quad (6.146)$$

We are interested in finding the probability density function of the transformed bivariate random variable  $Y$  with states  $\mathbf{y} = \mathbf{A}\mathbf{x}$ .

Recall that for change of variables we require the inverse transformation of  $\mathbf{x}$  as a function of  $\mathbf{y}$ . Since we consider linear transformations, the inverse transformation is given by the matrix inverse (see Section 2.2.2). For  $2 \times 2$  matrices, we can explicitly write out the formula, given by

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \mathbf{A}^{-1} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}. \quad (6.147)$$

Observe that  $ad - bc$  is the determinant (Section 4.1) of  $\mathbf{A}$ . The corresponding probability density function is given by

$$f(\mathbf{x}) = f(\mathbf{A}^{-1}\mathbf{y}) = \frac{1}{2\pi} \exp\left(-\frac{1}{2}\mathbf{y}^\top \mathbf{A}^{-\top} \mathbf{A}^{-1} \mathbf{y}\right). \quad (6.148)$$

The partial derivative of a matrix times a vector with respect to the vector is the matrix itself (Section 5.5), and therefore

$$\frac{\partial}{\partial \mathbf{y}} \mathbf{A}^{-1} \mathbf{y} = \mathbf{A}^{-1}. \quad (6.149)$$

Recall from Section 4.1 that the determinant of the inverse is the inverse of the determinant so that the determinant of the Jacobian matrix is

$$\det\left(\frac{\partial}{\partial \mathbf{y}} \mathbf{A}^{-1} \mathbf{y}\right) = \frac{1}{ad - bc}. \quad (6.150)$$

We are now able to apply the change-of-variable formula from Theorem 6.16 by multiplying (6.148) with (6.150), which yields

$$f(\mathbf{y}) = f(\mathbf{x}) \left| \det\left(\frac{\partial}{\partial \mathbf{y}} \mathbf{A}^{-1} \mathbf{y}\right) \right| \quad (6.151a)$$

$$= \frac{1}{2\pi} \exp\left(-\frac{1}{2}\mathbf{y}^\top \mathbf{A}^{-\top} \mathbf{A}^{-1} \mathbf{y}\right) |ad - bc|^{-1}. \quad (6.151b)$$

While Example 6.17 is based on a bivariate random variable, which allows us to easily compute the matrix inverse, the preceding relation holds for higher dimensions.

*Remark.* We saw in Section 6.5 that the density  $f(\mathbf{x})$  in (6.148) is actually the standard Gaussian distribution, and the transformed density  $f(\mathbf{y})$  is a bivariate Gaussian with covariance  $\mathbf{\Sigma} = \mathbf{A}\mathbf{A}^\top$ .  $\diamond$

We will use the ideas in this chapter to describe probabilistic modeling

in Section 8.4, as well as introduce a graphical language in Section 8.5. We will see direct machine learning applications of these ideas in Chapters 9 and 11.

## 6.8 Further Reading

This chapter is rather terse at times. Grinstead and Snell (1997) and Walpole et al. (2011) provide more relaxed presentations that are suitable for self-study. Readers interested in more philosophical aspects of probability should consider Hacking (2001), whereas an approach that is more related to software engineering is presented by Downey (2014). An overview of exponential families can be found in Barndorff-Nielsen (2014). We will see more about how to use probability distributions to model machine learning tasks in Chapter 8. Ironically, the recent surge in interest in neural networks has resulted in a broader appreciation of probabilistic models. For example, the idea of normalizing flows (Jimenez Rezende and Mohamed, 2015) relies on change of variables for transforming random variables. An overview of methods for variational inference as applied to neural networks is described in chapters 16 to 20 of the book by Goodfellow et al. (2016).

We side stepped a large part of the difficulty in continuous random variables by avoiding measure theoretic questions (Billingsley, 1995; Pollard, 2002), and by assuming without construction that we have real numbers, and ways of defining sets on real numbers as well as their appropriate frequency of occurrence. These details do matter, for example, in the specification of conditional probability  $p(y | x)$  for continuous random variables  $x, y$  (Proschan and Presnell, 1998). The lazy notation hides the fact that we want to specify that  $X = x$  (which is a set of measure zero). Furthermore, we are interested in the probability density function of  $y$ . A more precise notation would have to say  $\mathbb{E}_y[f(y) | \sigma(x)]$ , where we take the expectation over  $y$  of a test function  $f$  conditioned on the  $\sigma$ -algebra of  $x$ . A more technical audience interested in the details of probability theory have many options (Jaynes, 2003; MacKay, 2003; Jacod and Protter, 2004; Grimmett and Welsh, 2014), including some very technical discussions (Shiryayev, 1984; Lehmann and Casella, 1998; Dudley, 2002; Bickel and Doksum, 2006; Çinlar, 2011). An alternative way to approach probability is to start with the concept of expectation, and “work backward” to derive the necessary properties of a probability space (Whittle, 2000). As machine learning allows us to model more intricate distributions on ever more complex types of data, a developer of probabilistic machine learning models would have to understand these more technical aspects. Machine learning texts with a probabilistic modeling focus include the books by MacKay (2003); Bishop (2006); Rasmussen and Williams (2006); Barber (2012); Murphy (2012).

## Exercises

- 6.1 Consider the following bivariate distribution  $p(x, y)$  of two discrete random variables  $X$  and  $Y$ .

$Y$	$y_1$	0.01	0.02	0.03	0.1	0.1
	$y_2$	0.05	0.1	0.05	0.07	0.2
	$y_3$	0.1	0.05	0.03	0.05	0.04
		$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
		$X$				

Compute:

- The marginal distributions  $p(x)$  and  $p(y)$ .
  - The conditional distributions  $p(x|Y = y_1)$  and  $p(y|X = x_3)$ .
- 6.2 Consider a mixture of two Gaussian distributions (illustrated in Figure 6.4),

$$0.4\mathcal{N}\left(\begin{bmatrix} 10 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) + 0.6\mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 8.4 & 2.0 \\ 2.0 & 1.7 \end{bmatrix}\right).$$

- Compute the marginal distributions for each dimension.
  - Compute the mean, mode and median for each marginal distribution.
  - Compute the mean and mode for the two-dimensional distribution.
- 6.3 You have written a computer program that sometimes compiles and sometimes not (code does not change). You decide to model the apparent stochasticity (success vs. no success)  $x$  of the compiler using a Bernoulli distribution with parameter  $\mu$ :

$$p(x|\mu) = \mu^x(1-\mu)^{1-x}, \quad x \in \{0, 1\}.$$

Choose a conjugate prior for the Bernoulli likelihood and compute the posterior distribution  $p(\mu|x_1, \dots, x_N)$ .

- 6.4 There are two bags. The first bag contains four mangos and two apples; the second bag contains four mangos and four apples.

We also have a biased coin, which shows “heads” with probability 0.6 and “tails” with probability 0.4. If the coin shows “heads”, we pick a fruit at random from bag 1; otherwise we pick a fruit at random from bag 2.

Your friend flips the coin (you cannot see the result), picks a fruit at random from the corresponding bag, and presents you a mango.

What is the probability that the mango was picked from bag 2?

*Hint: Use Bayes’ theorem.*

- 6.5 Consider the time-series model

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{A}\mathbf{x}_t + \mathbf{w}, & \mathbf{w} &\sim \mathcal{N}(\mathbf{0}, \mathbf{Q}) \\ \mathbf{y}_t &= \mathbf{C}\mathbf{x}_t + \mathbf{v}, & \mathbf{v} &\sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \end{aligned}$$

where  $\mathbf{w}, \mathbf{v}$  are i.i.d. Gaussian noise variables. Further, assume that  $p(\mathbf{x}_0) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ .

- a. What is the form of  $p(\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_T)$ ? Justify your answer (you do not have to explicitly compute the joint distribution).
- b. Assume that  $p(\mathbf{x}_t | \mathbf{y}_1, \dots, \mathbf{y}_t) = \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ .
  1. Compute  $p(\mathbf{x}_{t+1} | \mathbf{y}_1, \dots, \mathbf{y}_t)$ .
  2. Compute  $p(\mathbf{x}_{t+1}, \mathbf{y}_{t+1} | \mathbf{y}_1, \dots, \mathbf{y}_t)$ .
  3. At time  $t+1$ , we observe the value  $\mathbf{y}_{t+1} = \hat{\mathbf{y}}$ . Compute the conditional distribution  $p(\mathbf{x}_{t+1} | \mathbf{y}_1, \dots, \mathbf{y}_{t+1})$ .
- 6.6 Prove the relationship in (6.44), which relates the standard definition of the variance to the raw-score expression for the variance.
- 6.7 Prove the relationship in (6.45), which relates the pairwise difference between examples in a dataset with the raw-score expression for the variance.
- 6.8 Express the Bernoulli distribution in the natural parameter form of the exponential family, see (6.107).
- 6.9 Express the Binomial distribution as an exponential family distribution. Also express the Beta distribution as an exponential family distribution. Show that the product of the Beta and the Binomial distribution is also a member of the exponential family.
- 6.10 Derive the relationship in Section 6.5.2 in two ways:
  - a. By completing the square
  - b. By expressing the Gaussian in its exponential family form

The product of two Gaussians  $\mathcal{N}(\mathbf{x} | \mathbf{a}, \mathbf{A})\mathcal{N}(\mathbf{x} | \mathbf{b}, \mathbf{B})$  is an unnormalized Gaussian distribution  $c\mathcal{N}(\mathbf{x} | \mathbf{c}, \mathbf{C})$  with

$$\begin{aligned} \mathbf{C} &= (\mathbf{A}^{-1} + \mathbf{B}^{-1})^{-1} \\ \mathbf{c} &= \mathbf{C}(\mathbf{A}^{-1}\mathbf{a} + \mathbf{B}^{-1}\mathbf{b}) \\ c &= (2\pi)^{-\frac{D}{2}} |\mathbf{A} + \mathbf{B}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{a} - \mathbf{b})^\top (\mathbf{A} + \mathbf{B})^{-1}(\mathbf{a} - \mathbf{b})\right). \end{aligned}$$

Note that the normalizing constant  $c$  itself can be considered a (normalized) Gaussian distribution either in  $\mathbf{a}$  or in  $\mathbf{b}$  with an “inflated” covariance matrix  $\mathbf{A} + \mathbf{B}$ , i.e.,  $c = \mathcal{N}(\mathbf{a} | \mathbf{b}, \mathbf{A} + \mathbf{B}) = \mathcal{N}(\mathbf{b} | \mathbf{a}, \mathbf{A} + \mathbf{B})$ .

#### 6.11 Iterated Expectations.

Consider two random variables  $x, y$  with joint distribution  $p(x, y)$ . Show that

$$\mathbb{E}_X[x] = \mathbb{E}_Y[\mathbb{E}_X[x | y]].$$

Here,  $\mathbb{E}_X[x | y]$  denotes the expected value of  $x$  under the conditional distribution  $p(x | y)$ .

#### 6.12 Manipulation of Gaussian Random Variables.

Consider a Gaussian random variable  $\mathbf{x} \sim \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$ , where  $\mathbf{x} \in \mathbb{R}^D$ . Furthermore, we have

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b} + \mathbf{w},$$

where  $\mathbf{y} \in \mathbb{R}^E$ ,  $\mathbf{A} \in \mathbb{R}^{E \times D}$ ,  $\mathbf{b} \in \mathbb{R}^E$ , and  $\mathbf{w} \sim \mathcal{N}(\mathbf{w} | \mathbf{0}, \mathbf{Q})$  is independent Gaussian noise. “Independent” implies that  $\mathbf{x}$  and  $\mathbf{w}$  are independent random variables and that  $\mathbf{Q}$  is diagonal.

- a. Write down the likelihood  $p(\mathbf{y} | \mathbf{x})$ .
- b. The distribution  $p(\mathbf{y}) = \int p(\mathbf{y} | \mathbf{x})p(\mathbf{x})d\mathbf{x}$  is Gaussian. Compute the mean  $\boldsymbol{\mu}_y$  and the covariance  $\boldsymbol{\Sigma}_y$ . Derive your result in detail.

- c. The random variable  $\mathbf{y}$  is being transformed according to the measurement mapping

$$\mathbf{z} = \mathbf{C}\mathbf{y} + \mathbf{v},$$

where  $\mathbf{z} \in \mathbb{R}^F$ ,  $\mathbf{C} \in \mathbb{R}^{F \times E}$ , and  $\mathbf{v} \sim \mathcal{N}(\mathbf{v} | \mathbf{0}, \mathbf{R})$  is independent Gaussian (measurement) noise.

- Write down  $p(\mathbf{z} | \mathbf{y})$ .
  - Compute  $p(\mathbf{z})$ , i.e., the mean  $\boldsymbol{\mu}_z$  and the covariance  $\boldsymbol{\Sigma}_z$ . Derive your result in detail.
- d. Now, a value  $\hat{\mathbf{y}}$  is measured. Compute the posterior distribution  $p(\mathbf{x} | \hat{\mathbf{y}})$ .  
*Hint for solution:* This posterior is also Gaussian, i.e., we need to determine only its mean and covariance matrix. Start by explicitly computing the joint Gaussian  $p(\mathbf{x}, \mathbf{y})$ . This also requires us to compute the cross-covariances  $\text{Cov}_{\mathbf{x}, \mathbf{y}}[\mathbf{x}, \mathbf{y}]$  and  $\text{Cov}_{\mathbf{y}, \mathbf{x}}[\mathbf{y}, \mathbf{x}]$ . Then apply the rules for Gaussian conditioning.

### 6.13 Probability Integral Transformation

Given a continuous random variable  $X$ , with cdf  $F_X(x)$ , show that the random variable  $Y := F_X(X)$  is uniformly distributed (Theorem 6.15).