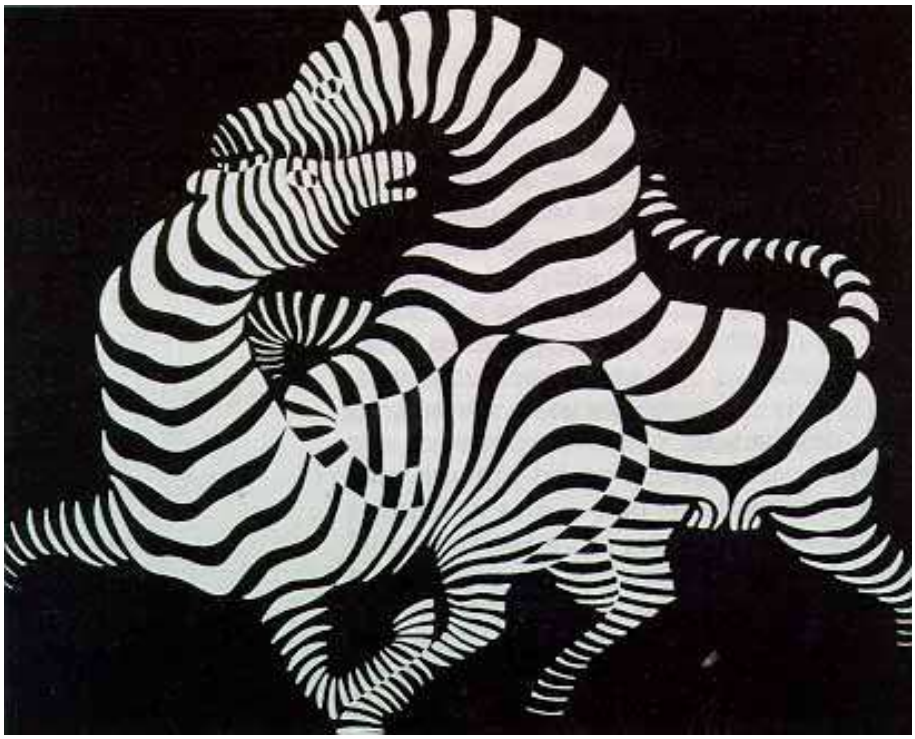
EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATHEMATISCH-
NATURWISSENSCHAFTLICHE FAKULTÄT
INSTITUT FÜR NEUROBIOLOGIE

Hanspeter A. Mallot

Analysing Data with MATLAB

Mathematical Background



July 2014

This text has been developed as lecture notes for the course "Datenanlyse mit MATLAB" held together with Heinz Bendele since summer term 2001. The selection of the material follows the needs of the Cognitive Neuroscience lab, but at the same time should be of more general interest. The text focusses on an intuitive understanding of the mathematical issues, while the actual MATLAB codes and algorithms are discussed in the practical part of the course.

I am grateful to Michaela Mohr for help with the translation of the text.

The title page shows the painting "Zebras" (1950) by Victor Vasarely (1908 – 1997).

Contents

Chapter 1

Numbers and Vectors

The first chapters summarize a number of important fundamentals that we need in order to understand the subsequent procedures for data analysis. The first chapter covers numbers and vectors, then follow matrices and matrix operations as well as functions and elements of calculus[1]. As a first application of these results, we study the presentation of data in various kinds of diagrams and plots. A list of recommended general textbooks about the subject is compiled at the end of each chapter.

1.1 Numbers

The most important type of numbers are the *real numbers* that we denote by the letter $\mathbb{R}$. They include the following more specific sets of numbers

- The integers[2] $(..., -2, -1, 0, 1, 2, ...)$. The positive integers are also called "natural numbers" and are denoted by $\mathbb{N}$.
- The fractions or ratios of integers are the *rational numbers*, $(0, \pm\frac{1}{1}, \pm\frac{2}{1}, \pm\frac{1}{2}, \pm\frac{3}{1}, \pm\frac{2}{2}, \pm\frac{1}{3}, \pm\frac{4}{1}, \pm\frac{3}{2}, \pm\frac{2}{3}, \pm\frac{1}{4}, ...)$. The rational numbers are "countable", i.e. they can be arranged in an exhaustive list as indicated above. This implies that there are just as many rational numbers as natural ones. Note that all integers are also rational,
- The solutions of algebraic equations (equations that contain powers of unknowns; e.g., $x^2 = 2$) lead to the set of the *algebraic numbers*. One example is $\sqrt{2}$ that can not be written as a fraction of integers. Note that all rational numbers are also algebraic.
- Finally, there exist real numbers which are neither fractions nor solutions of algebraic equations; they are called *transcendental numbers*. Examples are the ratio of the circumference and diameter of a sphere, π and Euler's number e.

The set of real numbers is innumerable; thus there are "more" real than natural numbers. The real numbers model the one-dimensional continuum. Between any pair of real numbers, there exists another one and therefore, by iteration, infinitely many. While this property is already satisfied for the rational numbers, the following is characteristic for the real numbers: For every sequence of real numbers that converges, for example for every bounded, but monotonically growing sequence in $\mathbb{R}$, the limit is an element of $\mathbb{R}$ as well.

[1] The German word for calculus is "Analysis".
[2] In German: Ganze Zahlen

Example We consider a *sequence* of numbers a_i which are given by the following equation

$$a_i = \left(1 + \frac{1}{i}\right)^i = \frac{(i+1)^i}{i^i} \quad \text{for} \quad i \in \mathbb{N}. \tag{1.1}$$

Obviously all a_i are rational numbers since enumerator and denominator are always integers. For example, we have:

$$\begin{aligned} a_1 &= \left(1 + \frac{1}{1}\right)^1 = 2, \\ a_2 &= \left(1 + \frac{1}{2}\right)^2 = \frac{9}{4} = 2.25, \\ a_3 &= \left(1 + \frac{1}{3}\right)^3 = \frac{64}{27} \approx 2.37, \\ a_4 &= \left(1 + \frac{1}{4}\right)^4 = \frac{625}{256} \approx 2.44. \end{aligned}$$

With growing i, a_i can be shown to grow ever more slowly and the rest of the sequence can be encased in smaller and smaller intervalls. The sequence converges to a so-called limit which is denoted as $\lim_{i\to\infty} a_i$. In our example, this happens to be Euler's number e, the base of natural logarithm.

$$\lim_{i\to\infty} a_i = e \approx 2.718. \tag{1.2}$$

Although all a_i are rational, e is not a rational number.

In MATLAB we can perform the calculation using the following code:

```
>> for i = 1:5
    a(i) = (1+1/i)^i
end
```

MATLAB creates a vector `a` whose length is incremented by one component in each iteration of this so-called `for`-loop. Simultaneously, the value of `a` is output in each step:

```
a =
    2.0000
a =
    2.0000    2.2500
a =
    2.0000    2.2500    2.3704
a =
    2.0000    2.2500    2.3704    2.4414
a =
    2.0000    2.2500    2.3704    2.4414    2.4883
```

As already mentioned the limit values of all real sequences are again real. This property of $\mathbb{R}$, i.e. that it contains the limits of all sequences in $\mathbb{R}$, is called *completeness*.

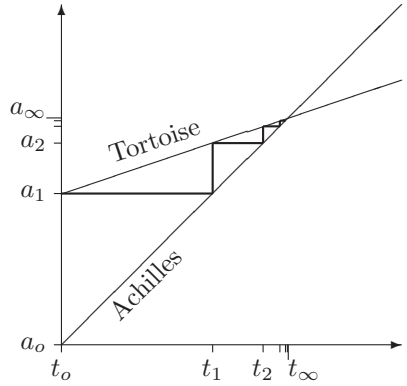


Figure 1.1: Achilles and the tortoise. The slanting lines show the positions of Achilles and the tortoise (with head start) as a function of time. The point of intersection marks the time of overtaking. The staircase function markes the positions a_i and times t_i where and when Achilles reaches the previous position of the tortoise. The sequences are growing monotonically towards their limit values a_∞ and $t_i nfty$, respectively.

Sequences and limits: An example

The concept of the limit can be realized easily with the help of the so-called paradoxon of Zenon[3]. This is a race between Achilles and a tortoise in which the tortoise is given a head start. It is vividly clear that Achilles will overtake the tortoise at some point. If one considers, however, when exactly this is going to happen, one realizes that the tortoise has also advanced in the time that Achilles needs to reach its starting position and therefore again has an, albeit smaller, head start. One can repeat this thought: once more the tortoise has a head start, again Achilles needs time to reach its position and again the tortoise will have advanced during that time. If one regards the times when Achilles arrives at the tortoise's previous position, one obtains a strictly increasing sequence of points in time, represented by a series of numbers in which each is greater than the previous one. This thought seems to show that Achilles never reaches the tortoise.

The concept of the limit, developed in early modern mathematics, is indeed necessary in order to understand that a sequence that grows strictly monotonically does not have to grow beyond all limits. To see this, we back up the example with Achilles and the tortoise with numbers. Assume that Achilles is running at 10m/s, the tortoise at 1m/s and that the head-start would be 10m. We look at those moments when Achilles has reached the former position of the tortoise.

step in sequence i	0	1	2	3	4	...	∞
Position of Achilles a_i	0m	10m	11m	11.1m	11.11 m	...	$11\frac{1}{9}$ m
Head start of the tortoise v_i	10m	1m	$\frac{1}{10}$ m	$\frac{1}{100}$ m	$\frac{1}{1000}$ m	...	0m
Point in time t_i	0s	1s	1.1s	1.11s	1.111s	...	$\frac{10}{9}$s

After $1\frac{1}{9}$ seconds Achilles has caught up with the tortoise; the sequence of points in time t_i converges towards that number; we write

$$\lim_{t\to\infty} t_i = 1\frac{1}{9}. \tag{1.3}$$

In Fig. 1.1, the limit value is denoted t_∞. Obviously, the race will continue after that event, even though the sequence t_i will never go beyond that point t_∞.

In general, one can show that every monotonically growing sequence which is bounded from above has to converge against a limit. An example for a restricted, non-monotonous and not converging sequence is $a_i := (-1)^i$.

Of course we can calculate the moment when Achilles catches up with the tortoise without the theory of limit values, simply by equalizing the equations of motion of both participants,

[3]Zenon of Elea, greek Philosopher, ca. 490-430 BC.

$$\text{Head Start} + \text{Speed Tortoise} \times t = \text{Speed Achilles} \times t,$$

and solving this for t. The relationship between this statement and the limit consideration is shown in Fig. 1.1.

1.2 Psychological Scales

The different types of numbers find an experimental counterpart in the various types of scales used in behavioral and psychological experiments (see for example, Bortz 2002). In a slightly simplified scheme, the following types may be distinguished:

Nominal Scales. Here, a set M of objects or observations is grouped in "classes" with the help of an equivalence relation. An equivalence relation is a relation between two elements of M satisfying the following requirements:

$$\begin{aligned} \text{reflexivity} \quad & a \sim a \\ \text{symmetry:} \quad & a \sim b \Leftrightarrow b \sim a \\ \text{transitivity:} \quad & a \sim b \ \text{ and } b \sim c \Rightarrow a \sim c \end{aligned}$$

The sign "$\sim$" (read "tilde") denotes that the relation holds for the respective pair of elements of M. The simplest example for an equivalent relation is the equality ("=") on number sets. Other examples include "is as tall as" (for humans, say), "carries the same allele as", etc. Through suitable transformation to a set of numbers every equivalent relation can be transferred into the equality relation.

Nominal scale data are integer numbers, i.e. frequencies of cases or specimen in M that fall into each class. The classes themselves can also be numbered, or named (hence the term "nominal" data). If numbers are chosen, numbering itself is arbitrary, i.e. the data does not change if the class numbers are changed.

Ordinal Scales. In the case of nominal data, the numbering of the classes is arbitrary. An ordering can be defined, if a "order relation" $\succeq$ exists on M, with has to satisfy the following requirements:

$$\begin{aligned} \text{antisymmetry} \quad & a \succeq b \text{ and } \ b \succeq a \ \Leftrightarrow a = b \\ \text{transitivity:} \quad & a \succeq b \ \text{ and } \ b \succeq c \Rightarrow a \succeq c. \end{aligned}$$

The simplest example of such an order relation is the greater-equal relation $\geq$ on number sets. If an order relation exists, M is called a partially ordered set. In ordinal data, differences between class numbers have no meaning, i.e. the data does not change if a monotonic transformation (i.e. a transformation preserving the ordering) is applied to the class numbers.

Cardinal Scales. In this case the measured variable is a rational or real number that allows pair comparisons. In psychology one distinguishes mostly interval scales where differences between measured values are well defined and comparable to each other, from ratio scales where multiples and fractions can be formed and compared usefully as well. Physical as well as physiological measurements are usually of this kind. One example of an interval scale that is not a ratio scale is the Celsius scale for temperature. One can say that the difference between 5° and 10° is as great

as the one between 15° and 20°, because the expansion of mercury is the same in both cases. It does not make a lot of sense though to say that 20° would be "'twice as warm"' as 10°, because the zero point of the Celsius scale is arbitrary. A significant difference between interval scales and ratio scales is in fact the existence of a well defined zero to which comparison statements can be related.

In experimental statistics the question of scale types plays a major role. We will not further elaborate on this point here, but will have to occasionally get back to it.

1.3 Vectors

1.3.1 Examples

1. The x-y-coordinates of a plane form a two-dimensional vector space.
2. Multiple simultaneous measurements in one specimen are often composed into a data vector. A five-dimensional data vector for a human subject might contain measurements of body weight, body volume, body height, arm length, and leg length. In this case, each vector component corresponds to one variable.
3. By recording simultaneously the activity of n neurons and dividing up the result in time windows of 10 ms, one obtains for every time window a one-dimensional vector whose components can assume the values 0 (no potential of action in the time window) or 1 (action potential in the time window). This vector is also called the state vector of the observed group of neurons. Here, each vector component corresponds to one neuron.
4. In a behavioural learning experiment, an animal's performance may be measured during a set of subsequent days. For each individual, a learning curve can be represented as a vector where each component corresponds to one training session (time).

Intuitively, vectors should be thought of as ordered lists, not necessarily as points or arrows in space. This avoids the problem that vectors with more than three dimensions can not be geometrically imagined. Dimensions can be thought of as degrees of freedom in respect to which a value can independently vary. The above examples show that there is no reason to limit the number of these degrees of freedom to three. Even in the case of more than three dimensions the total of all vectors with a given dimension n is called a vector space, specifically a n-dimensional vector space.

1.3.2 Vectors in MATLAB

Vectors are the most important type of data in MATLAB which is generelly used when repeated measurements are to be illustrated. Vectors can be considered as rows or columns of numbers (usually: real numbers).

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} = (v_1, v_2, \ldots, v_n)'. \tag{1.4}$$

The conversion of a row vector into a column vector or reverse is called *transposition*, the result also *transposed* vector. Instead of the apostrophe (') used here and also in MATLAB, transposition may also be denoted by the symbol $\top$.

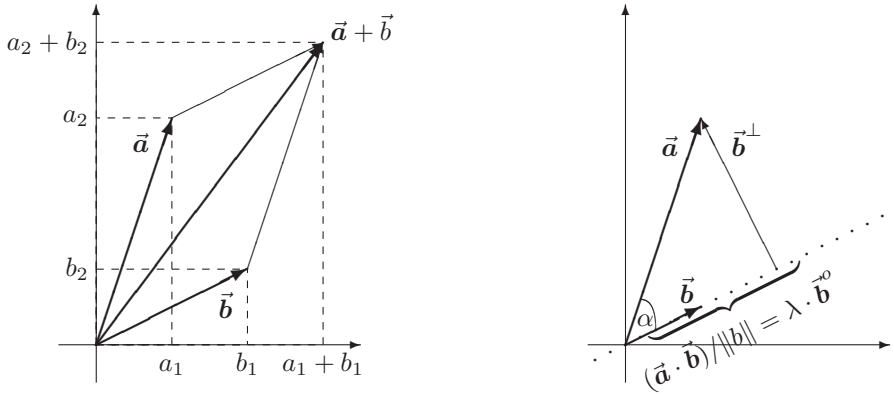


Figure 1.2: **left:** Vector addition. The vector sum $\vec{a} + \vec{b}$ can be reduced to the sum of the component. Geometrically, this amounts to aligning the start point of one vector with the end point of the other and forming the resultant vector ("'parallelogram of forces"'). **right:** Dot product. The dot product of two vectors of equal dimension computes the lenght of the projection of one vector onto the direction defined by the other vector.

The numbers v_i are called *components* of the vector $\vec{v}$, the number n of components is its *dimension*. The set of all vectors with real components and dimension n is called a n-dimensional *vector space*, $\mathbb{R}^n$.

Vectors form the most important structure of data in MATLAB. Row vectors are being arranged as list without comma and in square brackets:

```
>>   v = [ 1 2 3 ]
v =
    1   2   3
```

In column vectors one separates the components with semicolons:

```
>>   u = [ 4; 5; 6 ]
u =
    4
    5
    6
```

The transposition is being marked by an apostrophe like in Eq. 1.4:

```
>>   u'
ans =
    4   5   6
```

1.3.3 Arithmetic Operations

Important arithmetic operations for vectors include:

Vector summation. Two vectors of the same dimension are added up by adding up the components ("'parallelogramm of forces"', Fig. 1.2):

$$\vec{a} + \vec{b} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_n + b_n \end{pmatrix}. \tag{1.5}$$

Multiplication with a number. The multiplication as generalization of the addition of the vector with itself (in the sense of $\vec{a} + \vec{a} = 2\vec{a}$) is explained for arbitrary numbers ("'Scalars"') $\lambda \in \mathbb{R}$:

$$\lambda\vec{a} = \lambda \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} \lambda a_1 \\ \lambda a_2 \\ \vdots \\ \lambda a_n \end{pmatrix} \tag{1.6}$$

Dot product (scalar product, inner product). Two vectors of the same dimension can be multiplied by the following rule in which the result is a number (scalar):

$$(\vec{a} \cdot \vec{b}) := a_1 b_1 + a_2 b_2 + \ldots a_n b_n = \sum_{i=1}^{n} a_i b_i. \tag{1.7}$$

Because of $(\vec{a} \cdot \vec{a}) = \sum a_i^2$ the lenght or *norm* of the vector $\vec{a}$ is $\sqrt{(\vec{a} \cdot \vec{a})} =: \|\vec{a}\|$ (Pythagoras' theorem). A vector with norm 1 is called a unit vector; an arbitrary vector $\vec{a}$ can be converted by division through its norm into a unit vector. We write

$$\vec{a}^{\,o} := \frac{\vec{a}}{\|\vec{a}\|} \tag{1.8}$$

If the scalar product of two vectors is zero, the vectors are orthogonal (perpendicular) to each other. This is apparent immediately for the base vectors of a standard coordinate system which are given for three dimensions by

$$\vec{e}_1 := \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad \vec{e}_2 := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \vec{e}_3 := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{1.9}$$

Obviously, we have $(\vec{e}_i \cdot \vec{e}_j) = 0$ if $i \neq j$, i.e. the base vectors are pairwise orthogonal to each other. In general, the orthogonality of two vectors is defined by a zero dot product.

An intuitive interpretation of the scalar product in the general case is given in Fig. 1.2b. The dotted line marks the direction defined by the vector $\vec{b}$ or its unit vector $\vec{b}^{\,o}$. We denote the vector orthogonal to $\vec{b}$ as $\vec{b}^{\,\perp}$; in the multi-dimensional case we take the one that lies also in the plane spanned by $\vec{a}$ and $\vec{b}$. For suitable constants λ, μ, we then obtain:

$$\lambda\vec{b}^{\,o} + \mu\vec{b}^{\,\perp} = \vec{a}. \tag{1.10}$$

In this equation λ is the length of the projection of $\vec{a}$ onto the direction of $\vec{b}$. Applying the dot product with $\vec{b}$ to both sides of the equation, and observing that $(\vec{b}^{\perp} \cdot \vec{b}) = 0$, we obtain:

$$\lambda(\vec{b}^{o} \cdot \vec{b}) = (\vec{a} \cdot \vec{b}) \tag{1.11}$$

and hence

$$\lambda = \frac{(\vec{a} \cdot \vec{b})}{\|\vec{b}\|} = \|\vec{a}\|(\vec{a}^{o} \cdot \vec{b}^{o}). \tag{1.12}$$

The scalar product thus describes the projection of a vector onto another one. This property can be used to define angles in vector spaces:

$$\cos\alpha = (\vec{a}^{o} \cdot \vec{b}^{o}) = \frac{(\vec{a} \cdot \vec{b})}{\|\vec{a}\|\|\vec{b}\|}. \tag{1.13}$$

In MATLAB the dot product is treated as a special case of matrix multiplication (see below). If

```
>> a = [ 1 2 3 ]
a =
    1   2   3
>> b = [ 10 20 30 ]
b =
    10   20   30
```

are two row vectors, then the operation `a * b` is not defined. The same holds true when both `a` and `b` are defined as column vectors. In this example the scalar product should take the value $1 \times 10 + 2 \times 20 + 3 \times 30 = 140$. It is obtained from the command

```
>> a * b'
ans =
    140
```

Other products. More products defined in MATLAB are the outer product

```
>> a' * b
ans =
    10   20   30
    20   40   60
    30   60   90
```

and the component-wise product

```
>> a .* b
ans =
    10   40   90
```

We will come back to the outer product in matric multiplication and in multivariate statistics. The component-wise product is equivalent to the multiplication of sampled functions.

1.3.4 Circular Data and Unit Vectors

Vectors occur naturally as data formats if multiple variables are measured at each of a sample of specimen. We will come back to this type of data vectors in the sections on bi- and multivariate statistics. Here we consider a special example with considerable relevance in the behavioural sciences.

Examples of circular measurements include pointing movements of human observers to a target currently out of sight ($-180°$ to $180°$), the initial flight direction of a migrating bird when starting, or the time of day at which an event occurs (0 to 24 hours). At first glance, these measurements behave like numbers, but when calculating sums or differences that exceed the limits, strange effects may arise.

The problem becomes clear in a simple averaging operation. Consider an angular measurement yielding the values $\phi_{1,2} = \pm 1°$. The average value of these is $0°$, as expected. If, however, the measurement was $\phi_{1,2} = \pm 179°$, the expected average would be $\bar{\phi} = 180°$ (or $\bar{\phi} = -180°$) but the calulation will again yield $0°$. The reason for this is of course that for angles, $-180°$ is the same as $+180°$, which is clearly not the case for numbers. Mathematically, the problem is related to the singularity of the standard arcus tangent function ($\arctan_1$ below), which is usually mended by the two-dimensional $\arctan_2$ function:

$$\arctan_2(x,y) = \begin{cases} \arctan_1(y/x) & \text{für} \quad x > 0 \\ \pi + \arctan_1(y/x) & \text{für} \quad x \leq 0, y \geq 0 \\ -\pi + \arctan_1(y/x) & \text{für} \quad x \leq 0, y < 0 \end{cases} \tag{1.14}$$

Angles do thus not form a well behaved interval scale as discussed above. A meaningful averaging operation for angles can be defined by unit vectors associated to each angle, i.e.

$$\phi \leftrightarrow \begin{pmatrix} \cos\phi \\ \sin\phi \end{pmatrix}. \tag{1.15}$$

In this step, each angle is replaced by a unit vector in the direction of the angle. Averaging is then defined by averaging the vectors by standard vector summation and calculating the resultant direction with the $\arctan_2$-function applied to the vector average:

$$C = \sum \cos\phi_i \tag{1.16}$$

$$S = \sum \sin\phi_i \tag{1.17}$$

$$\bar{\phi} = \arctan_2(C,S) \tag{1.18}$$

Note that the sum vector $(C,S)'$ will be the longer, the more the individual values in the sample align. Its length cannot exceed the sample size n, which would be obtained if all angles were equal. In circular statistics, the total length of the sum vector,

$$R := \sqrt{C^2 + S^2} \tag{1.19}$$

is therefore used to define a circular variance V by the equation

$$V = 1 - R/n. \tag{1.20}$$

Circular statistics is an important field with applications, e.g. in the behavioral and the geo sciences. For reference see Fisher (1995).

Literature

Anton, H. (1998) *Lineare Algebra. Einführung, Grundlagen, Übungen.* Heidelberg, Berlin: Spektrum Akademischer Verlag

Berens, P. (2009) CircStat: A MATLAB toolbox for circular statistics. *Journal of Statistical Software* 31(10)

Bortz, J. (2005) *Statistik für Human- und Sozialwissenschaftler.* Berlin etc: Springer. 6. Edition

Farin, G., Hansford, D. (2008) *Mathematical Principles for Scientific Computing and Visualization.* Wellesley MA: A K. Peters Ltd.

Fisher N. I. (1995) *Statistical Analysis of Circular Data.* Cambridge UK: Cambridge University Press.

Glaeser G. (2004) *Der mathematische Werkzeugkasten. Anwendungen in Natur und Technik.* München: Elsevier Spektrum. Kapitel 4: Vektorrechnung

Wallisch, P., Lusignan, M., Benayoun, M., Baker, T.I., Dickey, A.S., Hatsopulos, N. (2008) *Matlab for Neuroscientists: An Introduction to Scientific Computing in Matlab.* Academic Press.

Chapter 2

Matrices

2.1 Introduction

Matrices[1] are rectangular arrays of numbers, in our applications always real numbers. Consider for example:

$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix}. \tag{2.1}$$

M is a matrix with 3 rows and 4 columns or short a 3×4-matrix. The components have two indices where the first one describes the row number and the second the column number.

Vectors can be treated as specific matrices with only one column or row. More specifically, a column vector is equivalent to a $n \times 1$-matrix and a row vector to a $1 \times n$-matrix. If the column and row indices are interchanged, the *transposed* matrix M' results, which in the example is given as:

$$M' = \begin{pmatrix} m_{11} & m_{21} & m_{31} \\ m_{12} & m_{22} & m_{32} \\ m_{13} & m_{23} & m_{33} \\ m_{14} & m_{24} & m_{34} \end{pmatrix} \tag{2.2}$$

By transposition, the 3×4-matrix M has been transformed into the 4×3-Matrix M'. The transposition of matrices is therefore completely analogous to that of vectors if vectors are treated as single-column or single-row matrices.

2.2 Matrix Multiplication

Matrices describe mappings between vector spaces. Square matrices (i.e., matrices with as many rows as columns) describe mappings of a vector space onto itself. In this mapping, every vector $\vec{\boldsymbol{v}}$ is assigned another vector $\vec{\boldsymbol{u}}$ by the rule

$$\vec{\boldsymbol{u}} = M\vec{\boldsymbol{v}} \tag{2.3}$$

[1]The term "matrix" (plural: matrices) traces back to the British mathematician James Joseph Sylvester (1814-1897). Determinants were used for solving systems of equation long before.

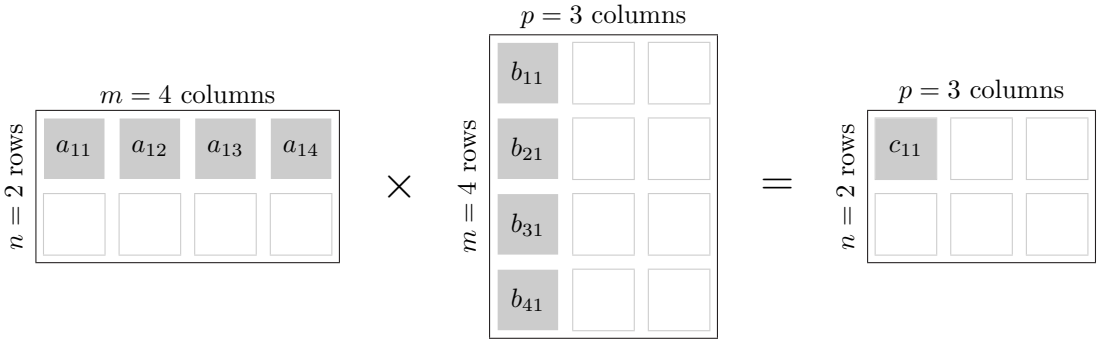


Figure 2.1: Matrix Multiplication. The multiplication of a 2×4-matrix A with a 4×3-matrix B results in a 2×3-Matrix C. The component $c_{11} = a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31} + a_{14}b_{41} = \sum_{j=1}^{m} a_{1j}b_{j1}$ and the components of the matrices A and B contributing to this component are highlighted in gray.

This rule is called the multiplication of a matrix with a vector. Let n be the dimension of the vector $\vec{\boldsymbol{n}}$ and M a square matrix with $n \times n$ componentes. Matrix multiplication than reads:

$$\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} & \dots & m_{1n} \\ m_{21} & m_{22} & \dots & m_{2n} \\ \vdots & \vdots & & \vdots \\ m_{n1} & m_{n2} & \dots & m_{nn} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \tag{2.4}$$

$$= \begin{pmatrix} m_{11}v_1 + m_{12}v_2 + \ldots + m_{1n}v_n \\ m_{21}v_1 + m_{22}v_2 + \ldots + m_{2n}v_n \\ \vdots \\ m_{n1}v_1 + m_{n2}v_2 + \ldots + m_{nn}v_n \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{n} m_{1j}v_j \\ \sum_{j=1}^{n} m_{2j}v_j \\ \vdots \\ \sum_{j=1}^{n} m_{nj}v_j \end{pmatrix} \tag{2.5}$$

To calculate the i-th component of the new vector $\vec{\boldsymbol{u}}$, we start by singling out the i-th row of the matrix; this row by itself can be considered a row vector. We now multiply the first component of this row with the first component of the vector $\vec{\boldsymbol{v}}$, then the second components and so on. Finally, we add up all products and obtain the result $u_i = \sum_{j=1}^{n} m_{ij}v_j$ ("row times column"). This multiplication pattern assumes that the matrix has just as many columns as the vector has components (or rows), since otherwise the pairs for the multiplication do not match.

In general one can multiply two matrices with each other when the first has as many columns as the second has rows (Fig. 2.1). For the multiplication of a $n \times m$-matrix A with a $m \times p$-matrix B one obtains a $n \times p$-matrix with the components

$$c_{ik} = \sum_{j=1}^{m} a_{ij}b_{jk}. \tag{2.6}$$

If $n = p = 1$, we have the multiplication of a row vector with a column vector; the result is a number. It is just the dot product introduced already in Chapter 1. If $m = 1$, a column vector is being multiplied with a row vector resulting in a $n \times p$-matrix (so-called outer or tensor product).

Table 2.1: Rules for matrix multiplication

Associativity	$(AB)C = A(BC)$
Distributivity	$A(B+C) = AB + AC$
Transposition	$(AB)' = B'A'$

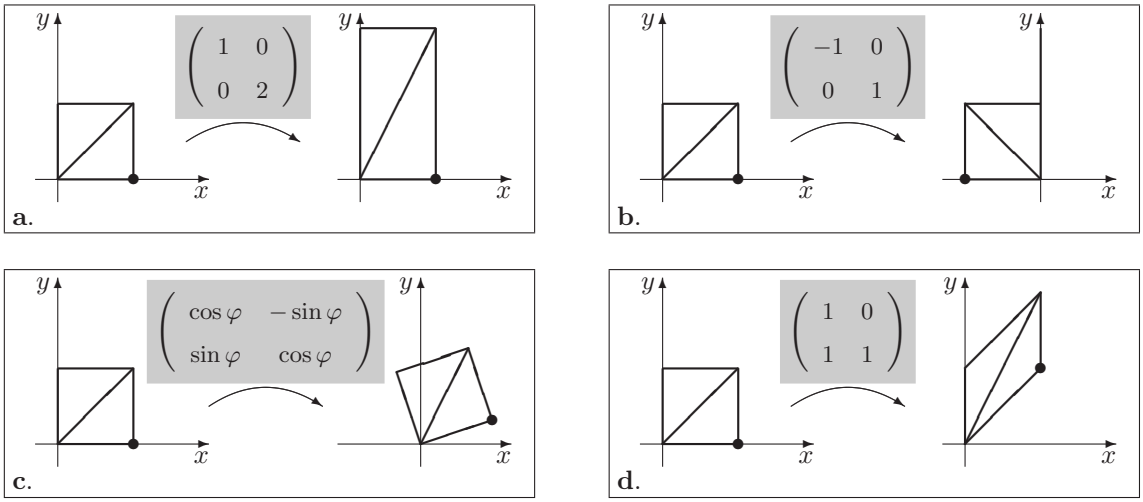


Figure 2.2: Similarity transformations in the plane. **a.** Expansion in direction y. **b.** Reflection at the y-axis. **c.** Rotation around the origin with angle of rotation φ. **d.** Shear. The matrices of the depicted transformations are shown respectively. By concatenation of the basic types all possible similarity transformations of the plane (except for shifts) can be produced.

One can already tell from this example that the matrix multiplication ist not commutative. This is also true for square matrices in general. Some rules for the matrix multiplication are shown in table 2.1.

The square matrix

$$E = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{2.7}$$

is called the 2×2 unit matrix. When a 2-dimensional vector is multiplied with this matrix, it stays the same, $E\vec{\boldsymbol{v}} = \vec{\boldsymbol{v}}$ for all $\vec{\boldsymbol{v}}$. Unit matrices in higher dimensions are defined accordingly, their coefficients along the diagonals have the value 1, all others the value 0.

Matrix multiplication defines a mapping of a vector $\vec{\boldsymbol{x}}$ to a vector $\vec{\boldsymbol{y}} = M\vec{\boldsymbol{x}}$. This mapping is *linear*, i.e. it satisfies the conditions

$$M(\vec{\boldsymbol{u}} + \vec{\boldsymbol{v}}) = M\vec{\boldsymbol{u}} + M\vec{\boldsymbol{v}} \tag{2.8}$$

$$M(\lambda\vec{\boldsymbol{u}}) = \lambda M\vec{\boldsymbol{u}} \quad \text{for } \lambda \in \mathbb{R} \tag{2.9}$$

Conversely one can show that every linear mapping between finite-dimensional vector spaces can be described by a matrix multiplication.

2.3 Examples

2.3.1 Similarity Transformations in the Plane

Similarity transformations in the plane are mappings that map lines to lines and parallel lines into parallels. Examples are rotation, reflection, shear and translation. If one disregards the shift, then the rest of the similarity transformations on the plane are described by 2×2-matrices. The transformation properties for straight lines can easily be derived from the linearity of the matrix-operation:

- *Lines through the origin are mapped to lines through the origin*
  For any vector $\vec{\boldsymbol{g}} \neq 0$ a line through the origin is defined by $\lambda\vec{\boldsymbol{g}}$ where λ varies in the set of real numbers. Linearity yields
  $$M(\lambda\vec{\boldsymbol{g}}) = \lambda(M\vec{\boldsymbol{g}}).$$
  This is again a straight line through the origin, however with a new direction $M\vec{\boldsymbol{g}}$.

- *Parallels to lines through the origin are mapped to parallels*
  Let $\vec{\boldsymbol{g}}$ and $\vec{\boldsymbol{v}}$ be two vectors; then $\lambda\vec{\boldsymbol{g}} + \vec{\boldsymbol{v}}$ is line parallel to $\lambda\vec{\boldsymbol{g}}$ but displaced by the vector $\vec{\boldsymbol{v}}$ (or rather its component orthogonal to $\vec{\boldsymbol{g}}$). The image of the parallels under M is
  $$\lambda(M\vec{\boldsymbol{g}}) + (M\vec{\boldsymbol{v}}).$$
  This is a line parallel to $M\vec{\boldsymbol{g}}$ with the shift of $M\vec{\boldsymbol{v}}$.

A few specific examples together with the corresponding matrices are shown in Fig. 2.2.

Axes that are preserved during transformation are called *eigenvectors* of the matrix. Let $\vec{\boldsymbol{e}}$ be an eigenvector of M, then $M\vec{\boldsymbol{e}} = \lambda\vec{\boldsymbol{e}}$, where λ is a real number called the "eigenvalue" associated with $\vec{\boldsymbol{e}}$. Under the mapping described by M, the eigenvector $\vec{\boldsymbol{e}}$ cannot change its orientation, but only its length or sense of direction; the line through the origin that is defined by all multiples of $\vec{\boldsymbol{e}}$ remains unchanged. The expansion shown in Fig. 2.2a has got the eigenvectors $(0,1)'$ ($\lambda = 1$) and $(1,0)'$ ($\lambda = 2$). The reflection in Fig. 2.2b has got the eigenvectors $(0,1)'$ (eigenvalue $\lambda = -1$) and $(1,0)'$ ($\lambda = 1$). Rotations (Fig. 2.2c) do not possess any eigenvectors for $\varphi \neq n\pi$, and the shear in Fig. 2.2d has a single eigenvector $(0,1)'$ with $\lambda = 1$.

The areal magnification associated with the transformation equals the value of the determinant of the transformation matrix

$$\det M := (m_{11}m_{22} - m_{12}m_{21}). \tag{2.10}$$

In Fig. 2.2a the area expansion is 2, in the other examples it has the value 1.

2.3.2 Systems of Linear Equations

Systems of linear equations can be written as matrix equations where the vector components are the unknowns. We consider a simple example that is motivated by computer tomography.

From Fig. 2.3 one obtains four equations for the four unknowns x_1 to x_4:

$$\begin{array}{ccccccccc} x_1 & + & & & x_3 & & = & d_1 \\ & & x_2 & + & & x_4 & = & d_2 \\ & & x_2 & + & x_3 & & = & d_3 \\ x_1 & + & x_2 & & & & = & d_4 \end{array} \tag{2.11}$$

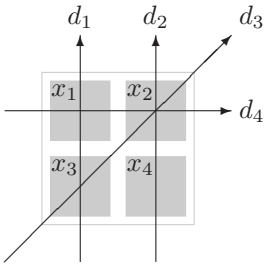


Figure 2.3: Four volume cells ("voxels") of a body are being x-rayed in the indicated directions. The total absorption for each direction, d_1 to d_4, results from summation of the absorption coefficients of the voxels. How big are the single absorptions?

Of course, this equation system can easily be solved for the four unknowns, using elementary algebra. A systematic procedure which can readily be used in programs like MATLAB starts by transforming the system into a matrix equation:

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix}. \tag{2.12}$$

Let M denote the matrix M. The equation would be solved if one could "divide by the matrix M" on both sides. Unfortunately, the division by matrices is not defined. In many cases, however, a so-called *inverse Matrix* M^{-1} can be calculated, satisfying $M^{-1}M = MM^{-1} = E$; here E is the unit matrix (in MATLAB, "eye") defined in Eq. 2.7. With the help of the inverse matrix one can write:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = M^{-1} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix} \tag{2.13}$$

$$= \frac{1}{2} \begin{pmatrix} 1 & 0 & -1 & 1 \\ -1 & 0 & 1 & 1 \\ 1 & 0 & 1 & -1 \\ 1 & 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix} \tag{2.14}$$

The inverse matrix exists only if the original matrix is square and "regular".[2] If, for example, one would use one equation in the above system twice and removed another for it, the inversion would not be possible anymore. In general, this problem arises whenever one equation is predictable through linear combination of the remaining equations. This is also the reason for having chosen a diagonal in Eq. 2.11 instead of another horizontal projection. At least four independent equations are needed in order to determine four unknowns. If there are less equations that unknowns,

[2] A regular matrix is a matrix with non-zero determinant. A matrix which is not regular is also called "singular". In the similarity transformations considered in Fig. 2.2, a singular matrix would collapse the entire (x, y)-plane into a single straight line or even the point $(0, 0)$. Clearly such transformations are not invertible.

there are generally many solutions (under-determined problem). If there are more euqations than unknowns, the system is over-determined and the equations can only be satisfied approximately. Such optimal solutions of over-determined equation systems can be found with the help of pseudo-inverse matrices (see below, Chapter 4.3).

Calculating of the inverse matrix with paper and pencil is not easier than the direct solving of the equation system. The advantage lies in in the availability of numerical routines for inverting matrices in MATLAB, for example through the command "\".

2.3.3 Markov-Chains[3]

A student S may be known for visiting one of three bars $(b_1, ..., b_3)$ every night. The probability with which he chooses one of the three is supposed to depend on his selection of the day before. We assume the following distribution:

- If S visits the "Ammerschlag" (b_1) on day t, we will find him at the "Ammerschlag" again on the next day $(t+1)$ with a probability of 60 % and with a probablity of 20 % respectively at the "Jazzkeller" or the "Stern".
- If S has visited the "Jazzkeller"(b_2) on day t, then one will see him the next day $(t+1)$ at the "Stern" with a 100 % probablity.
- If S has visited the "Stern" (b_3) on day t, the next day $(t+1)$ we will find him at the "Ammerschlag" with a 10 % chance, with a 80 % chance at the "Jazzkeller" and with a 10 % chance again at the "Stern".

Let p_{ij} denote the probability that S visits a bar b_i on one day when he has been at bar b_j the day before (conditional probability). We then obtain the probability matrix

$$P = \begin{pmatrix} 0.6 & 0 & 0.1 \\ 0.2 & 0 & 0.8 \\ 0.2 & 1.0 & 0.1 \end{pmatrix} \tag{2.15}$$

Note that the columns of this matrix add up to one, indicating that we will always find S in one of the three bars.

With q_i^t we label the probability that S visits bar b_i on day t. Since S visits exactly one bar each day, we know that $\sum_{i=1}^{3} q_i^t = 1$ for all t. We can then formulate the behavior of S as a matrix equation:

$$\begin{pmatrix} q_1^{t+1} \\ q_2^{t+1} \\ q_3^{t+1} \end{pmatrix} = \begin{pmatrix} 0.6 & 0 & 0.1 \\ 0.2 & 0 & 0.8 \\ 0.2 & 1.0 & 0.1 \end{pmatrix} \begin{pmatrix} q_1^t \\ q_2^t \\ q_3^t \end{pmatrix} \tag{2.16}$$

or shorter

$$\vec{\boldsymbol{q}}^{t+1} = P\vec{\boldsymbol{q}}^t. \tag{2.17}$$

We now assume that S goes to the "Ammerschlag" on day 1, $\vec{\boldsymbol{q}}^1 = (1, 0, 0)^\top$. Then we obtain the following probability distributions for the subsequent days:

[3] Andrey Andreyevich Markov (1856-1922), Russian Mathematician. The German term (using the German transcription of the Russian name) is "Markoff-Ketten"

	1	2	3	4	...	∞
Ammerschlag	1	0.6	0.38	0.262	...	0.1190
Jazzkeller	0	0.2	0.28	0.348	...	0.4048
Stern	0	0.2	0.34	0.390	...	0.4762

If S starts at the "Stern", one gets:

	1	2	3	4	5	...	∞
Ammerschlag	0	0.1	0.07	0.125	0.0947	...	0.1190
Jazzkeller	0	0.8	0.10	0.678	0.1826	...	0.4048
Stern	1	0.1	0.83	0.197	0.7227	...	0.4762

In this case, S oscillates between Jazzkeller and Stern for a while, but in the end the same steady state is reached which was also reached when starting from the Ammerschlag. This steady state indicates where we find S with the greatest probability if we do not know where he was on the previous day.

We denote the probability distribution in the limit by $\vec{\boldsymbol{q}}^{\infty}$. Since there is no change in the limiting distribution, it follows:

$$\vec{\boldsymbol{q}}^{\infty} = P\vec{\boldsymbol{q}}^{\infty}, \tag{2.18}$$

i.e., $\vec{\boldsymbol{q}}^{\infty}$ is eigenvector of P with eigenvalue 1.

2.3.4 Covariance Matrix

In an experiment, three variables are being collected from each of $n = 6$ subjects, for example body height, body mass, and age. Assume the following (fictitious) data:

Subj.	Height/m (X)	Weight/kg (Y)	Age/Years (Z)
1	1.53	52	13
2	1.98	98	19
3	1.72	65	48
4	0.92	14	3
5	2.01	105	27
6	1.83	89	61
Average	1.665	70.5	28.5

First we subtract the mean values and arrange the data into a 6×3-matrix D:

$$D = \begin{pmatrix} X_1 - \bar{X} & Y_1 - \bar{Y} & Z_1 - \bar{Z} \\ X_2 - \bar{X} & Y_2 - \bar{Y} & Z_2 - \bar{Z} \\ X_3 - \bar{X} & Y_3 - \bar{Y} & Z_3 - \bar{Z} \\ X_4 - \bar{X} & Y_4 - \bar{Y} & Z_4 - \bar{Z} \\ X_5 - \bar{X} & Y_5 - \bar{Y} & Z_5 - \bar{Z} \\ X_6 - \bar{X} & Y_6 - \bar{Y} & Z_6 - \bar{Z} \end{pmatrix} = \begin{pmatrix} -0.135 & -18.5 & -15.5 \\ 0.315 & 27.5 & -9.5 \\ 0.055 & -5.5 & 19.5 \\ -0.745 & -56.5 & -25.5 \\ 0.345 & 34.5 & -1.5 \\ 0.165 & 18.5 & 32.5 \end{pmatrix} \tag{2.19}$$

Then we calculate the matrix multiplication

$$\begin{aligned} C &:= \frac{1}{n} D'D \\ &= \frac{1}{n} \begin{pmatrix} -0.135 & 0.315 & 0.055 & -0.745 & 0.345 & 0.165 \\ -18.5 & 27.5 & -5.5 & -56.5 & 34.5 & 18.5 \\ -15.5 & -9.5 & 19.5 & -25.5 & -1.5 & 32.5 \end{pmatrix} \begin{pmatrix} -0.135 & -18.5 & -15.5 \\ 0.315 & 27.5 & -9.5 \\ 0.055 & -5.5 & 19.5 \\ -0.745 & -56.5 & -25.5 \\ 0.345 & 34.5 & -1.5 \\ 0.165 & 18.5 & 32.5 \end{pmatrix} \\ &= \begin{pmatrix} 0.137 & 11.318 & 4.003 \\ 11.318 & 975.58 & 318.08 \\ 4.003 & 318.08 & 403.25 \end{pmatrix}. \end{aligned} \tag{2.20}$$

C is a 3×3-matrix that contains the sample variance of the three variables (body height, body mass, age) on the diagonal and the respective covariances on the other fields. For n measurement variables one obtains a $n \times n$-matrix. The number of subjects does not enter into the dimensionality of the matrix. In general, we have:

$$C := \frac{1}{n} D'D \tag{2.21}$$

$$= \frac{1}{n} \begin{pmatrix} X_1 - \bar{X} & X_2 - \bar{X} & X_3 - \bar{X} & X_4 - \bar{X} & X_5 - \bar{X} & X_6 - \bar{X} \\ Y_1 - \bar{Y} & Y_2 - \bar{Y} & Y_3 - \bar{Y} & Y_4 - \bar{Y} & Y_5 - \bar{Y} & Y_6 - \bar{Y} \\ Z_1 - \bar{Z} & Z_2 - \bar{Z} & Z_3 - \bar{Z} & Z_4 - \bar{Z} & Z_5 - \bar{Z} & Z_6 - \bar{Z} \end{pmatrix} \begin{pmatrix} X_1 - \bar{X} & Y_1 - \bar{Y} & Z_1 - \bar{Z} \\ X_2 - \bar{X} & Y_2 - \bar{Y} & Z_2 - \bar{Z} \\ X_3 - \bar{X} & Y_3 - \bar{Y} & Z_3 - \bar{Z} \\ X_4 - \bar{X} & Y_4 - \bar{Y} & Z_4 - \bar{Z} \\ X_5 - \bar{X} & Y_5 - \bar{Y} & Z_5 - \bar{Z} \\ X_6 - \bar{X} & Y_6 - \bar{Y} & Z_6 - \bar{Z} \end{pmatrix}$$

If the matrix multiplication is carried out, one obtains the sums of products which correspond to the sample variances or covariances respectively.

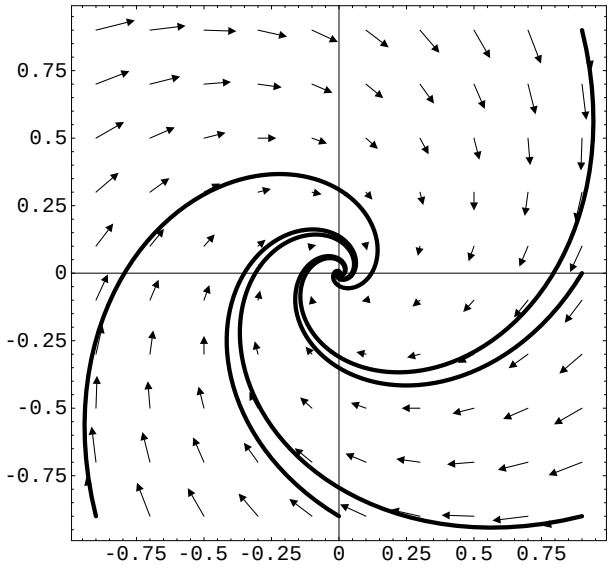


Figure 2.4: Interpretation of a 2×2-matrix A as a vector field, i.e., as a mapping $\vec{x} \to A\vec{x}$. The arrows show the vectors assigned by the mapping to each point in the plane. The curves are trajectories whose local tangents are given by the arrows at the respective position. They are solutions of the two-dimensional differential equation 2.25.

$$C = \frac{1}{n}\begin{pmatrix} \sum_{i=1}^{3}(X_i-\bar{X})^2 & \sum_{i=1}^{3}(X_i-\bar{X})(Y_i-\bar{Y}) & \sum_{i=1}^{3}(X_i-\bar{X})(Z_i-\bar{Z}) \\ \sum_{i=1}^{3}(Y_i-\bar{Y})(X_i-\bar{Y}) & \sum_{i=1}^{3}(Y_i-\bar{Y})^2 & \sum_{i=1}^{3}(Y_i-\bar{Y})(Z_i-\bar{Z}) \\ \sum_{i=1}^{3}(Z_i-\bar{Z})(X_i-\bar{X}) & \sum_{i=1}^{3}(Z_i-\bar{Z})(Y_i-\bar{Y}) & \sum_{i=1}^{3}(Z_i-\bar{Z})^2 \end{pmatrix} \tag{2.22}$$

$$=: \begin{pmatrix} \text{var } X & \text{cov } XY & \text{cov } XZ \\ \text{cov } XY & \text{var } Y & \text{cov } YZ \\ \text{cov } XZ & \text{cov } YZ & \text{var } Z \end{pmatrix} \tag{2.23}$$

The matrix C is symmetrical, i.e., $C = C'$; the covariance of X and Y is equal to that of Y and X.

The big matrices which we have written in this section may look complicated but they simplify the analysis of statistical data with programs like MATLAB enormously. If the data matrix is defined once, one only needs to calculate $\frac{1}{n}D'D$ and all (co)variances are herewith determined. We will meet the covariance matrix again in connection with the multidimensional normal distribution.

2.3.5 Linear Vector Fields

The transformations from example 2.3.1 can also be interpreted as vectorfields, i.e. by assuming that a vector $A\vec{x}$ is assigned to every place $\vec{x}$ of the plane and that this vector is drawn with $\vec{x}$ as a starting point.

An example is given in Figure 2.4 for the matrix

$$A = \begin{pmatrix} -0.3 & 0.5 \\ -0.5 & -0.3 \end{pmatrix} \tag{2.24}$$

In the sense of section 2.3.1 this matrix describes a rotation with a simultaneous contraction. This can be seen from the inwardly directed swirly pattern of the arrows. The continuous lines show trajectories of an imagined movement which locally always runs in the direction of the arrows. Mathematically, these are solutions of the linear two-dimensional differential equation.

$$\begin{pmatrix} x_1' \\ x_2' \end{pmatrix} = A \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \tag{2.25}$$

Literature

Anton, H. (1998) *Lineare Algebra. Einführung, Grundlagen, Übungen.* Heidelberg, Berlin: Spektrum Akademischer Verlag

Janich K. (2013) *Linear Algebra.* 11. Auflage, Heidelbergs etc.: Springer

Lang, S (2004) *Linear Algebra (Undergraduate Texts in Mathematics*, Third Edition. Berlin etc: Springer

Chapter 3

Functions

3.1 Functions and Mappings

The notion of a "function" is central to the whole field of calculus from which most methods for the analysis and modeling of data are taken. Functions are a special case of *relations* which are defined as ordered pairs built from the elements in two sets, called the domain D and range R, respectively. Relations can thus be thought of as subsets of the Cartesian product $D \times R$. *Mappings* and *functions* are specific relations that satisfy a uniqueness constraint:

A *mapping* A is a rule (a relation) that assigns to every element of the domain D one and only one element of the range or value set R. In mathematical notation:

$$M : D \to R; \quad \text{and} \quad M(d) = r.$$

Here D and R are (not necessarily different) sets and $d \in D$; $r \in R$. If D and R are (subsets of) the real numbers, then M is called a *function*. The term 'mapping' is more general and includes multidimensional domain and range sets.

We usually consider functions of time, of spatial variables, or of some other independent variable of a measurement. A few important special cases are shown in table 3.1. The class of polynomial functions contains the three previous functions (identity, line, parabola) as special cases.

Table 3.1: Some important functions

Name	Symbol	Defintion	Derivative
Identity	id	$\mathrm{id}(x) := x$	$\mathrm{id}'(x) \equiv 1$
Line	–	$f(x) = c + sx$	$f'(x) \equiv s$
Parabola	–	$f(x) = x^2$	$f'(x) = 2x$
Polynomial (of degree n)	–	$p(x) = \sum_{i=0}^{n} a_i x^i$	$p'(x) = \sum_{i=1}^{n} i a_i x^{i-1}$
Exponential function	exp	$\exp(x) = e^x := \sum_{i=0}^{\infty} \frac{x^i}{i!}$	$\exp'(x) = \exp(x)$
Sine function	sin	$\sin(x) := \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i+1}}{(2i+1)!}$	$\sin'(x) = \cos(x)$

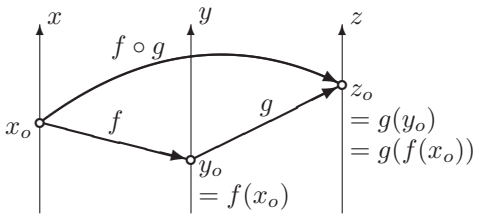


Figure 3.1: Composition (or concatenation) of functions

Calculations defined for numbers can be generalized to functions. For example, one can add two functions by adding their values pointwise. Phrased more exactly: if one has two functions f and g with the same domain, one can define a function $h = f + g$ which assigns to every x the value $h(x) := f(x) + g(x)$. One can proceed in the same way for multiplication, subtraction and, as long as the denominator is different from zero, for the division of functions.

A new operation that is specifically defined for functions is the concatenation or composition. We consider two functions f and g and assume that range of f is contained in the domain of g. The composition $h := f \circ g$ is then the consecutive evaluation of f and than g, $h(x) = g(f(x))$. If, for example, $g(x) = \exp(x)$ and $f(x) = -x^2$, we obtain

$$h(x) = g(f(x)) = \exp\{-x^2\}. \tag{3.1}$$

h is known as the Gaussian Function ('bell-shaped curve').

If for two functions f and g the equation $f \circ g = \mathrm{id}$ holds, f is the inverse function of g and vice versa. (By "id" we here denote the identity function introduced in table 3.1 which assigns every number to itself.) We write $g = f^{-1}$. The inverse function of the identity function id is again the identity Further examples of pairs of functions and their inverses include the quadratic parabola and the root function or the exponential function and the logarithm.

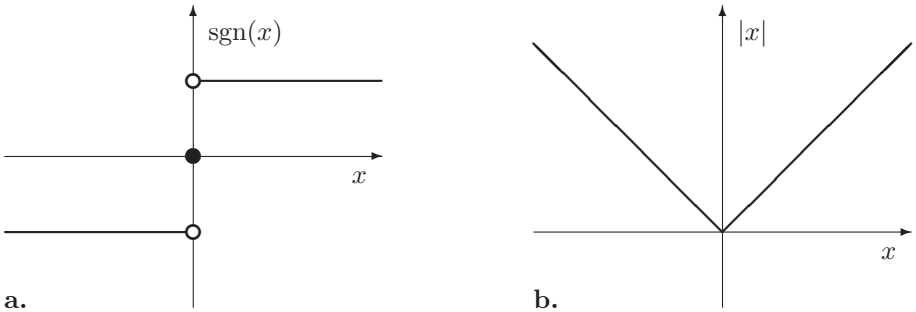


Figure 3.2: **a.** The sign function (sgn). **b.** The absolute value function.

3.2 "Smoothness": Continuity and Differentiability

3.2.1 Continuity

A function is called *continuous* at a point x_o of its domain if for every sequence converging against x_o (in the domain) the sequence of the related function values converges towards the function value

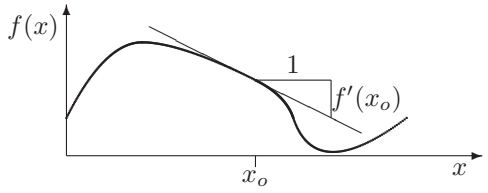


Figure 3.3: The derivative of a function f at the point x_o is the slope of a tangent which touches the graph of the function at the point $(x_o, f(x_o))$.

at x_o, $f(x_o)$:

$$f(\lim_{x\to x_o} x) = \lim_{x\to x_o} f(x). \tag{3.2}$$

Intuitively, a function is continuous if it can be plotted without lifting the pen from the paper.

One example for a discontinuous function is the sign function,

$$\mathrm{sgn}(x) := \begin{cases} -1 & \text{if} \quad x < 0 \\ 0 & \text{if} \quad x = 0 \\ 1 & \text{if} \quad x > 0 \end{cases}. \tag{3.3}$$

Let $x_i = -\frac{1}{i}$ for $i \in \mathbb{N}$ be a sequence in the domain of the function. Clearly, for large i, it will converge to 0: $\lim_{i\to\infty} x_i = 0$. On the other hand, for the sequence of functional values, we have $\mathrm{sgn}(x_i) = -1$ for all $i \in \mathbb{N}$. The sequence of function values therefore converges to -1 while the sequence of related arguments approaches 0:

$$\lim_{i\to\infty} f(x_i) = -1 \;\neq\; 0 = f(\lim_{i\to\infty} x_i). \tag{3.4}$$

Continuity is violated by the two jumps of the function from -1 to 0 and from 0 to $+1$ as can be seen in Fig.3.2a.

All functions listed in table 3.1 are continuous everywhere (i.e., for all x). If f is a continuous function and the inverse function f^{-1} exists, then this is continuous, too. The same holds true for $\frac{1}{f}$ if $f(x) \neq 0$. For two continuous functions f and g the pointwise sum $f + g$ and product $f \cdot g$ as well as the composition $f \circ g$ are again continuous functions.

3.2.2 Derivatives

The derivative of a function of one variable is defined as the limit:

$$\frac{df}{dx}(x) = f'(x) := \lim_{h\to 0} \frac{f(x+h) - f(x)}{h}; \tag{3.5}$$

the derivative exists if and only if such a limit is uniquely defined. In this case, we call the function f differentiable at a point x. The graph of f will have a unique tangent at the point $(x, f(x))$ and the slope of this tangent will be equal to the derivative (Fig. 3.3).

A simple example is the function $f(x) = ax$ for a arbitrary constant $a \in \mathbb{R}$. From Eq. 3.5 we obtain:

$$\begin{aligned} f'(x) &= \lim_{h\to 0} \frac{f(x+h) - f(x)}{h} \\ &= \lim_{h\to 0} \frac{a(x+h) - ax}{h} \\ &= \lim_{h\to 0} \frac{ah}{h} \\ &= a. \end{aligned}$$

This means that f has a constant slope a.

One example for a continuous function which is not differentiable at $x = 0$ is the absolute value function $f(x) = |x|$ (Fig. 3.2b). In this case, the fraction in Eq. 3.5 takes the value $+1$ when choosing $h > 0$ and the value -1 for $h < 0$. We can therefore built two different sequences for h, one approaching zero from below and one from above, for which the fractions $(f(x+h)-f(x))/h$ will be constantly $+1$ or -1, respectively. This means that no *unambiguous* limit exists. Graphically, no single tangent can be drawn at the point $(0, 0)$, because the graph has a kink there. The function is differentiable at all other points.

Every differentiable function is continuous, but not vice versa. The absolute value function is already an example for a continuous function which is non-differentiable. Sums, products, inverse and concatenation of differentiable functions are also differentiable.

For many functions, the derivative can be determined at every point x. Therefore $f'(x)$ will again be a function of x; it is always continuous. A few important rules for the calculation of derivatives are assembled in table 3.2. The derivatives of some special functions are given in table 3.1.

Since derivatives are again functions of x, they can usually be differentiated again. As example, consider the function $f(x) = x^n$. With the help of the power rule from table 3.2 we obtain $f'(x) = nx^{n-1}$. We can now apply the rule once more and obtain the *second derivative* $f''(x) = n(n-1)x^{n-2}$. For $k \leq n$ we obtain the k-th derivation $f^{(k)}(x) = n(n-1)...(n-k+1)x^{n-k}$. All higher derivatives will vanish, i.e., $f^{(k)} \equiv 0$ for all $k > n$. Overall this means that the function f can be differentiated any number of times. All functions listed in table 3.1 show this characteristic.

Derivatives can be used for describing the course of a function qualitatively. Some important characteristics are:

- Sections with growing or decreasing output resp. can be located by evaluating the sign of f'.
- Maxima are points satisfying $f'(x) = 0$ and $f''(x) < 0$.
- Minima are points satisfying $f'(x) = 0$ and $f''(x) > 0$.
- Inflection (turning) points connect regions with positive and negative gradient (e.g., left turn followed by right turn). They satisfy $f''(x) = 0, f'''(x) \neq 0$.

Table 3.2: Derivation Rules

Addition rule:	$(f(x) + g(x))' = f'(x) + g'(x)$
Product rule:	$(f(x)g(x))' = f'(x)g(x) + f(x)g'(x)$
Quotient rule:	$\left(\frac{f(x)}{g(x)}\right)' = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2}$
Power rule:	$(x^n)' = nx^{n-1}$
Chain rule:	$[f(g(x))]' = f'(g(x))g'(x)$
Inverse function:	$(f^{-1}(y))' = \frac{1}{f'(f^{-1}(y))}$

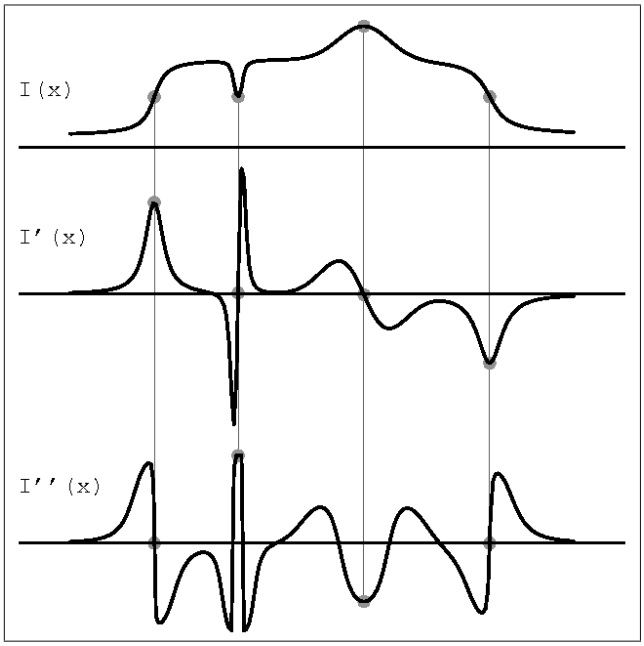


Figure 3.4: A function $I(x)$ and its first and second derivatives. Local extrema of I show up as zero-crossings of I', turning points as zero-crossings of I''.

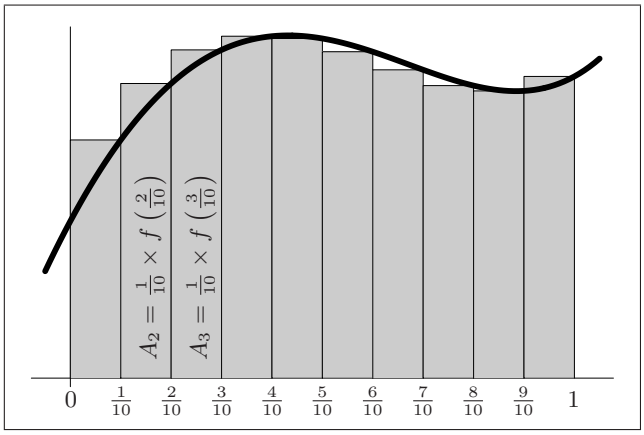


Figure 3.5: Approximation of the area under the function $f(x)$ by ten regularly spaced rectangles. Every rectangle has the side lengths 1/10 and $f(i/10)$ where i varies from 1 to 10. Their areas are therefore $0.1f(i/10)$. The sum of the areas of all rectangles approximates the total area under the curve. See equation 3.6.

3.3 Integrals

3.3.1 Basic Definitions

For a function with positive values, the integral can be interpreted as the area under the graph of that function. It is calculated as the limit of the sum:

$$\int_0^1 f(x)dx = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} f(\frac{i}{n}). \tag{3.6}$$

The result is a number (definite integral). In the sum on the right side of the equation every term $\frac{1}{n} f(\frac{i}{n})$ gives the area of one of the approximating rectangles with width $\frac{1}{n}$ and height $f(\frac{i}{n})$ shown in Fig. 3.2.2. In contrast to the area, the integral is a signed quantity. Evaluating Eq. 3.6 for

functions with negative values will yield negative results. Furthermore, we have

$$\int_a^b f(x)dx = -\int_b^a f(x)dx. \tag{3.7}$$

An interpretation of the integral where the algebraic sign is taken account of is the mean. The definite integral can be understood as mean of the integral, multiplied with the length of the integration interval:

$$\bar{f} = \frac{1}{b-a}\int_a^b f(x)dx. \tag{3.8}$$

Let us now consider integrals with variable limits, for example integrals over the interval from 0 to a variable x. This leads to the notion of *indefinite* integrals which are again functions, namely of the upper limit x. We denote this function by an upper case F:

$$F(x) := \int_0^x f(x')dx'. \tag{3.9}$$

The variable of F is the x which is the upper limit in the integral. In addition, there exists an 'integration variable' x', which runs from 0 to x and does not appear on the left side of the equation anymore.

Integration is the inverse of the derivation. The exact relation of the two operations is described in the *Fundamental Theorem of Calculus:*

$$\int_a^b F'(x)dx = F(b) - F(a). \tag{3.10}$$

In addition to definite and indefinite integrals, we will sometimes also look at integrals without explicitly specified limits. It this then understood that the integration passes through the entire set of real numbers, i.e., from $-\infty$ to $+\infty$. Those integrals are called 'improper':

$$\int_{-\infty}^{\infty} \exp(-x^2)dx := \lim_{b\to\infty}\int_{-b}^{b} \exp(-x^2)dx = \sqrt{\pi}. \tag{3.11}$$

In summary, three types of integrals can be distinguished: definite integrals are calculated over a fixed interval and result in a number. Indefinite integrals are functions of one of the integration limits (usually the upper one). Improper integrals are limits of definite integrals with the limits approaching infinity.

3.3.2 Examples

Distance, speed, acceleration

The relation of derivation and integration can be demonstrated by the following example of a car ride. The speedometer provides current velocity values $v(t)$. If the duration of the journey is given by the interval $[0, \tau]$, one obtains the overall covered distance as the definite integral $\int_0^\tau v(t)dt$. We now denote by $s(t)$ the total distance covered by time t and obtain for this total distance the indefinite integral $s(t) = \int_0^t v(t')dt'$. Clearly, the derivative of s is v; the fundamental theorem of calculus then states

$$\int_0^\tau v(t)dt = s(\tau) - s(0). \tag{3.12}$$

As mentioned before, this integral can also be interpreted as an average. For the average speed, we obtain

$$\bar{v} = \frac{s(\tau) - s(0)}{\tau} = \frac{1}{\tau} \int_o^\tau v(t)dt. \tag{3.13}$$

Similarly, we can formulate the relation between speed and acceleration. In summary, we have the following situation:

$$\begin{array}{ccccc} \boxed{\text{distance}} & & \boxed{\text{speed}} & & \boxed{\text{acceleration}} \\ s(t) & \overset{d/dt}{\longrightarrow} & v(t) = \frac{ds}{dt}(t) & \overset{d/dt}{\longrightarrow} & a(t) = \frac{dv}{dt}(t) \\ s(t) = s_o + \int_0^\tau v(t)dt & \overset{\int dt}{\longleftarrow} & v(t) = v_o + \int_0^\tau a(t)dt & \overset{\int dt}{\longleftarrow} & a(t) \end{array} \tag{3.14}$$

Progessive taxation

For the calculation of "progressive" income tax one uses a variable taxation rate $\rho(x)$ which determines a different tax rate for each income. Usually $\rho(x)$ is zero for small x (below the basic income allowance) and rises up to a maximum. For an income amounting to x_o taxes in the extent of

$$\int_0^{x_o} x\rho(x)dx \tag{3.15}$$

have to be paid.

Probability density

Example 3 Consider a continuous random variable such as the radial distance of a dart thrown to a dartboard. For a given thrower, a so-called distribution density $p(r)$ can be defined describing the probability of hitting the board at a distance r from the center. From this distribution function, the probability of a throw to land in the intervall the interval $[r_1, r_2]$ is given by the definite integral

$$P(r_1 \leq r \leq r_2) = \int_{r_1}^{r_2} p(r)dr. \tag{3.16}$$

If the interval is small, the probability of landing a throw in the interval will also be small. In the limit of a point, the probability itself will approach zero: $P(r = r_1) = \int_{r_1}^{r_1} p(r)dr = 0$. Still, the probability density at this point may be different from zero.

The probability of the error being below some maximum R is

$$D(R) := P(r \leq R) = \int_0^R p(r)dr. \tag{3.17}$$

$D(R)$ is called the distribution function of the process and is the indefinite integral of distribution density.

Note also that the expected value of the random variable r is defined as the improper integral

$$E(r) := \int_0^\infty rp(r)dr. \tag{3.18}$$

3.3.3 Integration Rules

Since the integral is the inverse of the derivative, one can derive rules for the integration of functions from the derivation rules. However, while calculating derivatives is relatively simple, integration often turns out to be difficult and at times not possible analytically. For example, all power functions be derivated according to the power rule given in table 3.2, which leads to the corresponding rule for integration:

$$\int_0^x x^p dx = \frac{1}{p+1} x^{p+1}. \tag{3.19}$$

This rule, however, does not work for $p = -1$, that is for the power function $f(x) = \frac{1}{x} = x^{-1}$, since a division by zero would occur here. Different arguments show that the integral of the function x^{-1} leads onto a whole different class of functions, namely the natural logarithm:

$$\int_1^x \frac{1}{x}\, dx = \ln x. \tag{3.20}$$

Let us finally mention the rule for the integration of concatenated functions, known as "integration by substitution":

$$\int_{g(a)}^{g(b)} f(y)dy = \int_a^b f(g(x))g'(x)dx. \tag{3.21}$$

3.4 Vector-Valued Functions

Curves in the plane or in space can be interpreted as vector-valued functions, i.e. as functions of the form

$$C : \mathbb{R} \to \mathbb{R}^n, \quad C(t) = \begin{pmatrix} k_1(t) \\ k_2(t) \\ \vdots \\ k_n(t) \end{pmatrix}. \tag{3.22}$$

An important application of such functions are paths or trajectories assigning a set of (x, y)- or (x, y, z)-coordinates to every instant of time t. The derivative dK/dt is again a vector, the length (norm) of which is the instantaneous speed of movement along the trajectory.

One example for such a function is shown in Fig. 3.6a for the function

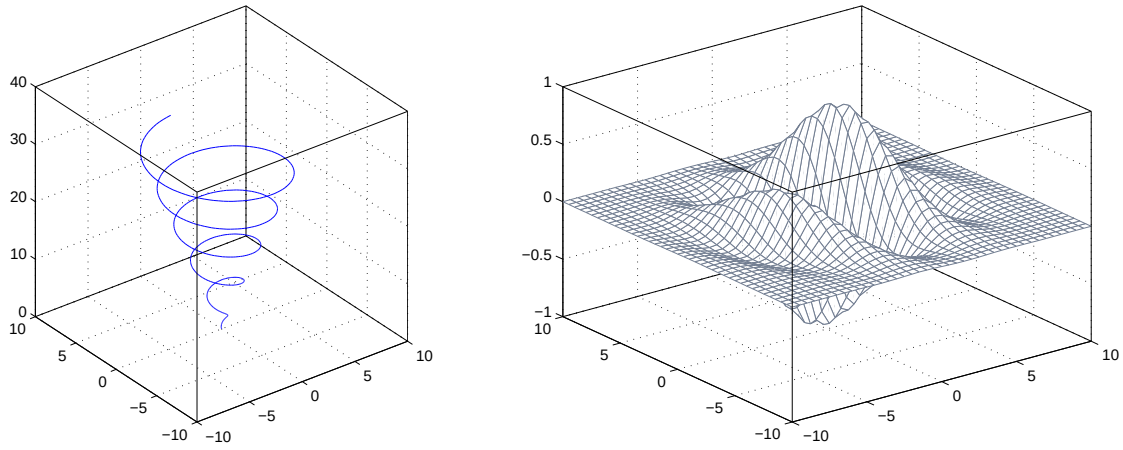
$$C(t) = (\frac{t}{5}\sin t, \frac{t}{5}\cos t, t)' \tag{3.23}$$

and the interval $t \in [0, 10\pi]$. The MATLAB-code for creating this plot is:

```
>> t = 0:pi/50:10*pi;
>> plot3(0.2 t .* sin(t), 0.2 * t .* cos(t), t)
>> grid on; box on; square on
```

3.5 Functions of two and more Variables

Functions of two variables appear in the description of distributions of space-dependent variables in the plane. One frequently occurring example is the distribution of light intensities on the retina



a. **b.**

Figure 3.6: Examples for three-dimensional plots from MATLAB. **a.** Expanding spiral (Eq. 3.23) as example for a vector-valued function $\mathbb{R} \to \mathbb{R}^3$. **b.** Gabor-function (Eq. 3.24) as example for a "landscape" function, $\mathbb{R}^2 \to \mathbb{R}$.

which we will call an image function. Generally the domain of a function of n variables is a subset of the n-dimensional space $\mathbb{R}^n$. As range set, we also consider subsets of the real numbers $\mathbb{R}$. We write $f(x_1, x_2, ..., x_n)$.

As example for a function $\mathbb{R}^2 \to \mathbb{R}$ we consider a so-called Gabor-function

$$f(x, y) = \sin(x + \frac{y}{3}) \exp\{-\frac{x^2 + y^2}{20}\} \tag{3.24}$$

on the interval $-10 < x < 10$, $-10 < y < 10$. Fig. 3.6b shows the graph of the function, generated by the MATLAB-code

```
>> [x,y] = meshgrid(-10:0.5:10);
>> mycolor = zeros(41,41);
>> mesh(x, y, sin(x+y/3) .* exp(-(x .^ 2 + y .^ 2)/20), mycolor);
>> box on; grid on; axes square
```

3.5.1 Derivatives of Functions with Several Variables

Fig. 3.7 shows a function of two variables as a 'landscape', i.e. the functional value $f(x, y)$ is assigned as an elevation to each point (x, y) in the domain of the function. The vectors $\vec{p}$ and $\vec{q}$ are tangential to this surface at the point $(x_o, y_o, f(x_o, y_o))$. Such tangential vectors are graphically obtained by laying an intersection through the graph of f which is orthogonal to the (x, y)-plane and passing through $(x_o, y_o, f(x_o, y_o))$. In the resulting slice, the functional values form a one-dimensional function to which tangents can be defined by taking one-dimensional derivatives. These tangents (or their slope) are called directional derivatives of f. Directional derivatives can be defined for any direction parallel to the ground plane by considering the according slices. For a local characteristic of the surface, it suffices to consider just two directional derivatives, one for each coordinate direction. They are called the partial derivatives of f with respect to x or to y; all other direction- derivations can be calculated from them.

For a function $f(x, y)$, the slices in coordinate direction are given by

$$g_{y_o}(x) := f(x, y_o) \quad \text{and} \quad g_{x_o}(y) := f(x_o, y), \tag{3.25}$$

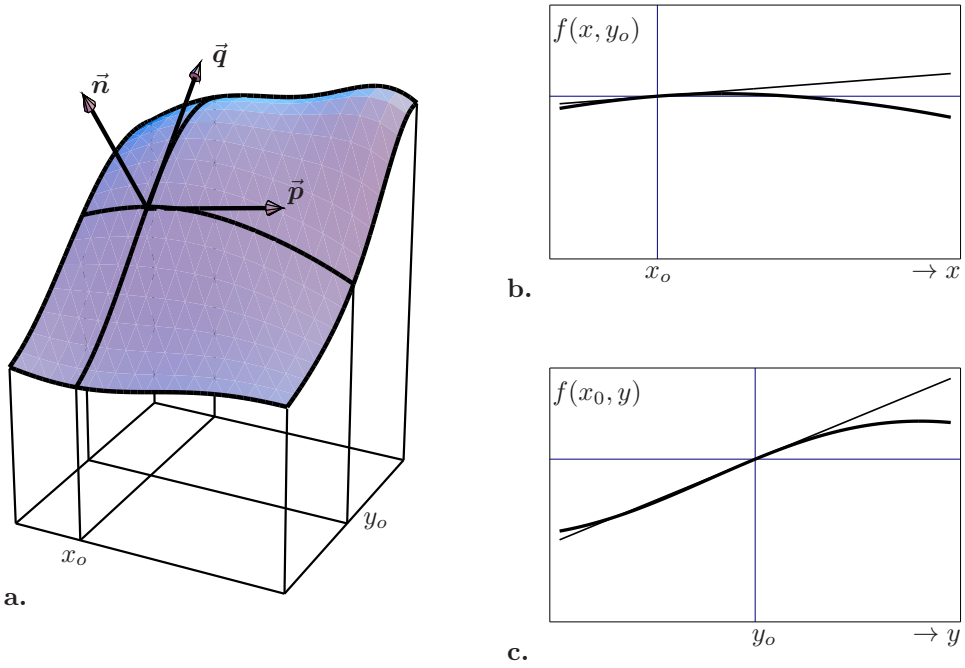


Figure 3.7: **a.** A function of two variables, $F(x, y)$, describes a 'landscape' above its domain. The figure shows the landscape together with the local surface normal $\vec{\boldsymbol{n}}$ at point $(x_o, y_o, f(x_o, y_o))$ and the corresponding partial derivatives $\vec{\boldsymbol{p}}$, $\vec{\boldsymbol{q}}$. **b.**, **c.** show slices cut through the surface in the directions x or y which are needed for the calculation of the partial derivatives.

see Fig. 3.7b,c. The index y_o indicates that there exists an intersection function for every value y_o. The function g_{y_o} is one-dimensional and its derivative can be taken as explained above. We call $g'_{y_o}(x)$ the partial derivative with respect to the first variable of f at the position (x, y_o) and write:

$$g'_{y_o}(x) \quad =: \quad \frac{\partial f}{\partial x}(x, y_o). \tag{3.26}$$

$$g'_{x_o}(y) \quad =: \quad \frac{\partial f}{\partial y}(x_o, y). \tag{3.27}$$

Practically, this means that when taking a partial derivative with respect to one variable, all other variables in the definition of the function are treated as constants.

The partial derivatives are again functions of two variables. At every point (x, y) this function indicates the slope of the tangent to the surface taken in the direction of derivation.

For a function of n variables, n partial derivatives exist, one for each coordinate direction. The vector of this n partial derivation is called the *gradient* of the function. It is:

$$\begin{aligned} &\mathrm{grad} f(x_1, x_2, ..., x_n) := \\ &\quad \left(\frac{\partial f}{\partial x_1}(x_1, x_2, ..., x_n),\ \frac{\partial f}{\partial x_2}(x_1, x_2, ..., x_n),\ \ldots,\ \frac{\partial f}{\partial x_n}(x_1, x_2, ..., x_n) \right). \end{aligned} \tag{3.28}$$

The gradient is a vector in the domain of f; it indicates the direction of the steepest ascent on the graph of the function. The steepness corresponds to the length of the gradient. In directions orthogonal to the gradient (contour lines in two dimensions), the elevation change is zero.

Example 1. We regard the function $f(x, y) := ax + by$ with the constants $a, b \in \mathbb{R}$. The graph of f is a slanted plane. In a slice in x-direction, passing through y_o, we have $f(x, y_o) = ax + by_o$, the term by_o is constant and therefore disappears when taking the derivative. Hence,

$$\frac{\partial f}{\partial x}(x, y) = a \quad \text{and} \quad \frac{\partial f}{\partial y}(x, y) = b.$$

Thus, the gradient takes the constant value (a, b), which is the slope direction of the plane. It is constant, i.e. it does not depend on x or y. This is consistent with the fact that the direction of the steepest slope on the plane is the same everywhere.

Example 2. Consider $f(x, y) := x^2 + y^2$. The graph of f is a paraboloid, i.e. a shape similar to an inverted sugar loaf with the tip at the position $(0, 0)$. In a slice in x-direction passing through y_o on the y-axis, $f(x, y_o) = x^2 + y_o^2$, the term y_o^2 is constant and will therefore vanish in the derivative. We have:

$$\frac{\partial f}{\partial x}(x, y) = 2x \quad \text{and} \quad \frac{\partial f}{\partial y}(x, y) = 2y.$$

The gradient vectors of f form a radial pattern of vectors which point outward from the origin (the tip of the sugar loaf).

3.5.2 Integrals of Functions of Several Variables

In one-dimensional functions, we explained the integral by approximating the area under a curve by a set of rectangles and improved the approximation by making the x-interval of each rectangle smaller and smaller. In two dimensions, the notion of an area under the curve is replaced by that of a volume under a surface. We consider a function defined on the square interval $[0, 1] \times [0, 1]$. We decompose this interval into a set of n^2 squares with size $\frac{1}{n} \times \frac{1}{n}$. The volume under the surface can then be approximated by a set of quadratic blocks elevated on these squares, i.e. with a lower corner at $(\frac{i}{n}, \frac{j}{n}, 0)$ and elevation of $f(\frac{i}{n}, \frac{j}{n})$ where $i, j = 1, ..., n$. In analogy to Eq. 3.6, we then obtain for definite integrals

$$\int_0^1 \int_0^1 f(x, y)dxdy := \lim_{n\to\infty} \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} f(\frac{i}{n}, \frac{j}{n}). \tag{3.29}$$

Multiple integrals are usually written in a nested order, i.e. the first integral sign specifies the limits for the last integration variable and so on.

It is clear from Eq. 3.29 that the ordering with which the volume of the individual blocks on the right side are summed up, is arbitrary. We could just as well have written $\sum_{j=1}^{n} \sum_{i=1}^{n}$ since in the end, all blocks are considered anyway. For the continuous integral, this means that the order of integration can be inverted:

$$\int_c^d \int_a^b f(x, y)dx\ dy = \int_a^b \int_c^d f(x, y)dy\ dx, \tag{3.30}$$

as long as all one-dimensional integrals exist. This result is known as Fubini's theorem.

Undefinite and improper integrals are defined for functions of multiple variables in complete analogy to the one-dimensional case. Also, multidimensional integrals can be considered as averages over the integration domain.

References

Glaser G. 2004. *Der mathematische Werkzeugkasten. Anwendungen in Natur und Technik.* München: Elsevier Spektrum. Kapitel 5, 7

Kemnitz A. (2006) *Mathematik zum Studienbeginn* 7. Auflage, Vieweg und Teubner
Spivak M. (2006) *Calculus* 3. Edition, Cambridge University Press.

Chapter 4

Fitting Curves to Data

4.1 General

4.1.1 Data and Models

Quantitative measurements are often organized as measurements of a *curve*, i.e. by varying an experimental parameter (the independent variable) and measuring a functional value (the dependent variable) at each of these measurement points. Examples include the size of a growing plant depending on time (growth curve), the frequency of a subjects' correct decisions in a recognition task depending on the number of test runs (learning curve), the probability of detecting a stimulus depending on the stimulus strength (psychometric function), the response rate of a visual neuron depending on the orientation of a bar-shaped stimulus in the visual field (tuning curve), the decay of tracer activity in a certain brain volume through local cerebral blood flow (washout-curve), etc.

In these examples, two types of variables need to be distinguished

- The *independent variable* is set or chosen by the experimenter. In the examples above it is realized as the time at which a measurement is taken, the number of learning steps, the strength of the stimulus, or the orientation of the stimulus in the visual field. It does not depend on the outcome of the experiment (hence the name) and will be assumed to be uneffected of error (at least in this chapter). We will use the letter x to denote the independent variable.
- The *dependent variable* or measurand is the quantity actually observed or measured in the experiment, i.e. for each setting of the independent variable. In the above examples, the dependent variables are size, number of correct decisions, probability of detection, activity of the neuron, and the residual concentration of the tracer. We call this variable y.

We denote the data points in a set of n measurements as pairs

$$(x_i, y_i), i = 1, \ldots, n. \tag{4.1}$$

In MATLAB the x and y- values would be combined to a n-dimensional vector, `x`, `y` respectively. Note that multiple measurements at the same parameter setting x are counted as extra data points.

Mathematically, the curve which is measured through the single points is a function

$$f : x \to y. \tag{4.2}$$

In the case of ideal (i.e. not error-prone) measurements, we expect

$$y_i = f(x_i) \quad \text{for all} \quad i. \tag{4.3}$$

In MATLAB the application of function to vectors works component by component. We may therefore write

```
y = f(x)
```
(4.4)

If the function is given explicitly, e.g., $f(x) = x^2$, then the component-wise execution in MATLAB must be specified explicitly. This is done by putting a point "." in front of the power operator:

```
y = x .^ 2
```
(4.5)

The expression `x ^ 2` (squaring of a vector) is not defined in MATLAB.

4.1.2 Interpolation vs. Fitting

Since measurements can be taken only at a set of discrete locations x_i, there are always many possible curves passing through the recorded points. In addition, one is often not interested in having curves strictly passing through the points since the measurements are error-prone anyway and the 'real' value is therefore likely to deviate from the data. This is the case particularly when multiple measurements are performed for one value of the independent variable and means and standard deviations (error bars) are plotted in the diagram. Basically, there are two possibilities for the determination of a curve through data points: Interpolation and fitting[1].

Interpolation. In this case, one creates a curve passing through the measured points. Often, interpolations are *linear*, data points are connected by straight line segments. The MATLAB-routine `plot` is doing this automatically. Here, kinks usually arise between the individual line segments. Smoother interpolations can be obtained from so-called polynomial or Spline-interpolations. We will not pursue this issue further in this course.

Interpolations mostly serve graphic purposes, e.g., in order to show which points belong together in a diagram with multiple curves. Also, they allow estimates about functional values between points of measurement. Those estimates are reliable only if the sampling is dense enough to capture the true course of a curve.

Curve fitting. If the general course of the function f is known or can be assumed from theoretical considerations, better curves can be plotted. Usually, the curve is known only up to a number of 'parameters'. If, for example, one assumes for theoretical reasons that f is a straight line, then slope and y-intercept are free parameters which have to be determined from the data; if one assumes a Gaussian bell curve, then the mean, the amplitude and the width are free parameters. We denote such parameters as $p_1, p_2, ...$ and write them in the argument of the function: $f(x; p_1, p_2, ...p_k)$.

4.2 Calculating a Fit-Curve

The task of fitting a curve to data amounts to determining the parameters of a model in a way that the resulting curve fits the data as well as possible. The deviation between the curve and the measurements should be as small as possible and distributed evenly among the measurement points.

The problem of fitting curves to data can be stated by four components:

1. the data, given as two vectors `x` and `y` of the same dimension with the interpretation that the components `x(i)` and `y(i)` are the coordinates of one data point (independent and dependent variable).

[1]The German word for fitting is "Ausgleichsrechnung"; fit curves are "Ausgleichskurven". However, the English terms are more common.

Table 4.1: Examples for non-linear fitting functions

Name	Function	Range & Parameters
Maximum curves (e.g. peaked a tuning curve)		
cosine	$f(x;p_1) := 1 + p_1 \cos(x - p_2)$	$-\pi < x < \pi$ p_1 peak amplitude p_2 peak position
Gauss-Function	$f(x;p_1,p_2) := \exp\{\frac{(x-p_1)^2}{p_2^2}\}$	$-\infty < x < \infty$ $0 < f(x) < 1$ p_1 peak position p_2 peak width
Saturation curves (e.g., learning curve)		
Exponential function	$f(t;p_1,p_2) := 1 - p_1 \exp\{-p_2 t\}$	$0 < x < \infty$ $0 < f(x) < 1$ p_1, p_2 initial value, slope
Sigmoidal curves (e.g., psychometric function)		
Arctangent (Inverse of the tangent function)	$f(x;p_1,p_2) := 1 + \frac{2}{\pi} \arctan\, p_2(x - p_1)$	$-\infty < x < \infty$ $0 < f(x) < 1$ p_1 inflection point p_2 slope
Error Function; Distribution function of the normal distribution	$f(x;p_1,p_2) := \frac{1}{\sqrt{2\pi}p_2} \int_{-\infty}^{x} \exp\{-\frac{(x-p_1)^2}{2p_2^2}\}$	$-\infty < x < \infty$ $0 < f(x) < 1$ p_1 inflection point p_2 inverse slope

2. a model $f(x;p_1,...,p_k)$ with free parameters $p_1,...p_k$. For given parameters and a value of independent variables the model predicts the measured value. The choice of the type of function f follows mostly theoretical considerations; when in doubt, the simplest function or the function with the least number of free parameters should be chosen (cf. table 4.1). This will often be the regression line.

3. an error function indicating how well the data are modeled by the curve with a given parameter-set. For a given data set, the error function depends only on these free parameters of the model, since the data are treated as fixed. The error function used most is the sum of the square deviations E_{ssd}:

$$E_{ssd}(p_1,...,p_k) = \sum_{i=1}^{n} (y_i - f(x_i;p_1,...,p_k))^2 \tag{4.6}$$

The square is chosen so that over- and underestimates of data do not cancel each other out. In addition, it has useful mathematical characteristics that we will return to later.

Other error functions can be obtained by considering the probability with which the observed measurement is to be expected, under the assumption that the model is correct. This probability should be as large as possible. The resulting estimate of the model parameters is called *Maximum Likelihood* Estimate.

4. an optimization procedure that finds the parameter values for which the error function takes its best value, i.e. minimal for the sum of squared deviations and maximal for the likelihood approach. In many cases, this optimization will have to be solved numerically, but there are special cases where analytical solutions exist, in particular the important class of linear optimization problems.

The simplest (and possibly most important) example of curve fitting is the determination of a regression line which we discuss as a special case of the linear regression in the next section.

4.3 Regression Line

In linear regression, a linear relation between x and y is expected. In this case, the fit-function is

$$f(x; a, b) := ax + b. \tag{4.7}$$

One example for this would be the growth of a plant as a function of time. If one expects an exponential relationship, then one can generate a linear connection by taking the logarithm of the data.

The variables a and b are the free parameters of our model. The task is therefore to determine a and b in a way that the error, i.e. the deviation between straight line and data, is minimized.

4.3.1 Solution in Matrix-Notation

If n measurements $x_1, \ldots, x_n$ are available, we need to consider the values of f only at these positions, $y_i = f(x_i)$. Eq. 4.7 can then be written as a matrix equation:

$$\begin{pmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{pmatrix} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \begin{pmatrix} b \\ a \end{pmatrix}, \tag{4.8}$$

or shorter

$$f(\vec{\boldsymbol{x}}) = G\vec{\boldsymbol{p}}. \tag{4.9}$$

G is called the design matrix of the problem since it depends on the experimental design, in particular the settings of the independent variable used in the measurement.

The goal of the fitting process is to determine the parameter vector $\vec{\boldsymbol{p}} = (b, a)'$ with which the model matches the data values as closely as possible:

$$\vec{\boldsymbol{y}} \approx G\vec{\boldsymbol{p}} \tag{4.10}$$

If we could 'divide' this equation by the matrix G, we could solve this equation for $\vec{\boldsymbol{p}}$ and would have the result immediately. However, G is a $n \times 2$-matrix and therefore in general not even square. As a kind of substitute for the division by G we thus consider the minimization task:

$$\|\vec{\boldsymbol{y}} - G\vec{\boldsymbol{p}}\| \to \min; \tag{4.11}$$

The expression on the left corresponds (except for the square root) to the sum of squared deviations from Eq. 4.6:

$$E_{ssd}(a,b) = \sum_{i=1}^{n}(y_i - (ax_i + b))^2 = \|\vec{\boldsymbol{y}} - G\vec{\boldsymbol{p}}\|^2 \tag{4.12}$$

For the result of the optimization the square root is irrelevant. Since it is a monotonic function, minimizing the sum of squares or their square root will yield the same result.

MATLAB provides its own arithmetic operations for solving this kind of tasks, in particular the so-called backslash-operator (\) and the more generally applicable Moore-Penrose Pseudo-Inverse (`pint`). The solution of the above-mentioned minimization task is reached via the command:

```
popt = G \ y.
```
(4.13)

Reading the right side of this equation backwards, one obtains "y/G", that is "y divided by G" which can be used as mnemonic for the meaning of the backslash-operator.

Mathematically, the pseudo-inverse is obtained from Eq. 4.9 by first multiplying from the left with the transposed design matrix G':

$$G'\vec{\boldsymbol{y}} = G'G\vec{\boldsymbol{p}} \tag{4.14}$$

Since $G'G$ is always a square matrix, it will also be invertible in many cases. One can then solve the equation for $\vec{\boldsymbol{p}}$:

$$(G'G)^{-1}G'\vec{\boldsymbol{y}} = \vec{\boldsymbol{p}}. \tag{4.15}$$

Both backslash and pseudo-inverse will find the least square solution, if a unique such solution exists.

Example. We now apply this strategy on the problem of the regression line. Hence, we look at the following 'measurement':

x	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0
y	0.1	0.3	1.5	3.1	6.3	5.2	5.5	9.2	8.1	9.1

The input in MATLAB is:

```
>> x = (1:10)' ;
>> y = [0.1 0.3 1.5 3.1 6.3 5.2 5.5 9.2 8.1 9.1]' ;
>> plot(x,y,'o')
```

The last command generates a graphic of data points.

Next, we specify the design matrix G from Eq. 4.8 by the MATLAB code

```
>> G = [ones(size(x)), x] ;
```

With the help of the backslash-operator we obtain the optimal parameters as a two-dimensional vector:

```
popt = G \ y;
```

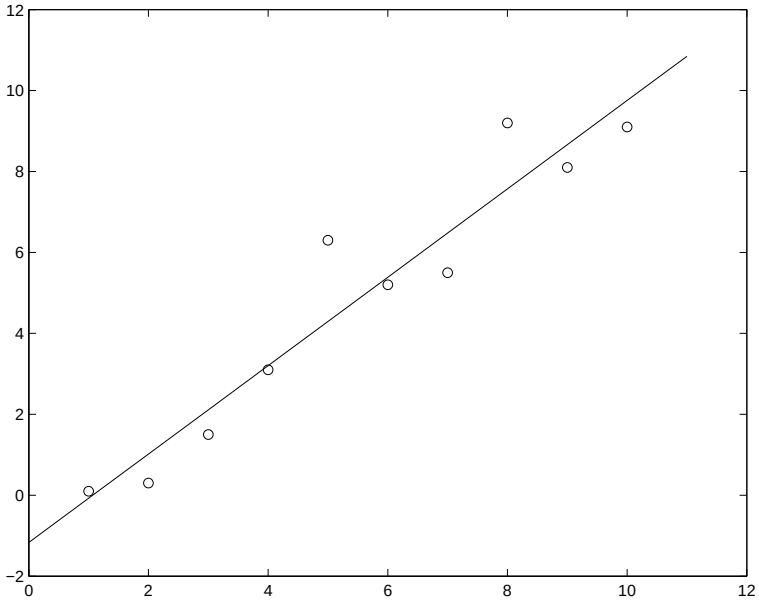


Figure 4.1: Example for the calculation of a regression line with the MATLAB-code presented in the text.

If we now want to plot the data together with the regression line, we have to calculate some points along the theoretical line. Of course, for a straight line two points suffice, but if the model is a more general curve, a denser sampling is necessary. For the sampling points of the theoretical curve, we use the terms `X` and `Y`, in contrast to the data points `x, y`.

```
X = (0:11)';
Y = [ones(size(X)), X] * popt;
plot(X,Y,'-',x,y,'o')
```

The result is shown in Fig. 4.1.

4.3.2 Elementary Computation of the Regression Line

The regression line may also be calculated using elementary mathematics. In that case, one starts with the general error function Eq. 4.6 and obtains

$$E_{ssd}(a,b) = \sum_{i=1}^{n}(y_i - (ax_i + b))^2 \tag{4.16}$$

$$= \sum_{i=1}^{n}\left(y_i^2 + a^2x_i^2 + b^2 - 2ax_iy_i - 2by_i + 2abx_i\right) \tag{4.17}$$

$$= a^2\sum_{i=1}^{n}x_i^2 + nb^2 + 2ab\sum_{i=1}^{n}x_i - 2a\sum_{i=1}^{n}x_iy_i - 2b\sum_{i=1}^{n}y_i + \sum_{i=1}^{n}y_i^2. \tag{4.18}$$

The last equation looks more complicated than the first, but for the task of minimizing E in a and b, it is indeed simpler if we keep in mind that the sums are constants. This is the case since the measurements are given and we want to determine from them the unknown parameters a and b. We now introduce the terms s_x (sum of the x-values), s_y (sum of the y-values), s_{xx} (sum of the squares of the x-Values), s_{yy} (sum of the squares of the y-values) and s_{xy} (sum of the products of the x- and y-Values); see Table 4.2.

Table 4.2: Characteristic variables of the dataset used in the calculation of the regression line

Standard-Notation	MATLAB
$s_x = \sum_{i=1}^{n} x_i$	`sx = sum(x)`
$s_y = \sum_{i=1}^{n} y_i$	`sy = sum(y)`
$s_{xx} = \sum_{i=1}^{n} x_i^2$	`sxx = x * x'` (or: `sxx = sum(x .  ∧ 2)`)
$s_{yy} = \sum_{i=1}^{n} y_i^2$	`syy = y * y'` (or: `syy = sum(y .  ∧ 2)`)
$s_{xy} = \sum_{i=1}^{n} x_i y_i$	`sxy = x * y'` (or: `sxy = sum(x .  * y)`)

With those terms we obtain from Eq. 4.18

$$E(a,b) = a^2 s_{xx} + nb^2 + 2abs_x - 2as_{xy} - 2bs_y + s_{yy}. \tag{4.19}$$

It now becomes apparent that E depends quadratically on a and b. Sketching E in three dimensions over the (a,b)-plane, a parabolic pot will arise, much as discussed in Section 3.5.1, Example 2. Here, however, it will in general not be a centered paraboloid of revolution with circular contour lines, but shifted, compressed side-ways, and rotated about the verical axis. Contour lines will be ellipses while the intersections with the coordinate planes remain to be parabolas.

The error paraboloid (Eq. 4.19) will have its minimum not at the coordinate origin, but at a different position marking the optimal values for a and b. In order to find the minimum of E, we consider the partial derivatives of E with respect to a and b (see Figure 3.7) and set them to zero. From this, the following constraints arise:

$$\frac{\partial}{\partial a} E(a,b) \stackrel{!}{=} 0 \quad \Leftrightarrow \quad as_{xx} + bs_x = s_{xy} \tag{4.20}$$

$$\frac{\partial}{\partial b} E(a,b) \stackrel{!}{=} 0 \quad \Leftrightarrow \quad as_x + nb = s_y. \tag{4.21}$$

On the right side, we now have two linear equations for two unknowns. From these, we finally obtain the solutions:

$$a = \frac{ns_{xy} - s_x s_y}{ns_{xx} - (s_x)^2} \quad \text{and} \quad b = \frac{s_y s_{xx} - s_x s_{xy}}{ns_{xx} - (s_x)^2}. \tag{4.22}$$

a and b are called slope and y-intercept of the regression line for the dataset (x_i, y_i).

4.3.3 Comparison of the Elementary and the Matrix-Calculation

For a deeper understanding of the matrix solution (pseudo-inverse) given in Section 4.3.1, we compare the two approaches in this section. We start by considering the components of the

pseudo-inverse Eq. 4.15:

$$G'G = \begin{pmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_n \end{pmatrix} \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} = \begin{pmatrix} n & s_x \\ s_x & s_{xx} \end{pmatrix}. \tag{4.23}$$

The square matrix $G'G$ thus contains the same sum terms occurring also in the elementary calculation. Note how these sums are calculated implicit as a result of the matric multiplication. We now consider the inversion step,

$$(G'G)^{-1} = \frac{1}{s_{xx} - (s_x)^2} \begin{pmatrix} s_{xx} & -s_x \\ -s_x & n \end{pmatrix}, \tag{4.24}$$

which involves a division by the determinant of $G'G$. The same terms occur in the denominators of Eq. 4.22. Finally, we observe

$$G'\vec{\boldsymbol{y}} = \begin{pmatrix} s_y \\ s_{xy} \end{pmatrix} \tag{4.25}$$

By expanding the products, we obtain

$$\vec{\boldsymbol{p}}_{opt} = \frac{1}{ns_{xx} - (s_x)^2} \begin{pmatrix} s_y s_{xx} - s_x s_{xy} \\ -s_x s_y + n s_{xy} \end{pmatrix}, \tag{4.26}$$

in accordance with Eq. 4.22.

4.4 General Linear Models

The regression line considered in the last section is a fitting curve determined by the method of minimal square deviation for a given set of data. In contrast to the general fitting problem, solutions for the regression line could be found analytically. Generally, one can show that analytical computations are always possible when the model is linear in its parameters, i.e. when the fit-function can be represented as:

$$f(x; p_1, ..., p_k) = \sum_{l=1}^{k} p_l g_l(x). \tag{4.27}$$

In this approach, the model is a weighted sum (linear combination) of arbitrary functions g_l, which themselves need not be linear. Indeed, only the functional values $g_l(x_i)$ at the nodes $(x_i)_{i=1,\dots,n}$ are of interest anyway. One could therefore confine the continuous Eq. 4.27 to these x-values and apply a matrix with random coefficients g_{li} instead of the functions g_l. Such problems are called *linear* optimization problems, because the functions g_l are combined linearly. MATLAB takes advantage of the existence of analytical solutions and treats general linear problems different from other, non-linear ones.

The regression equation will read:

$$\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix} = \begin{pmatrix} g_1(x_1) & g_2(x_1) & \dots & g_k(x_1) \\ g_1(x_2) & g_2(x_2) & \dots & g_k(x_2) \\ \vdots & \vdots & & \vdots \\ g_1(x_n) & g_2(x_n) & \dots & g_k(x_n) \end{pmatrix} \begin{pmatrix} p_1 \\ \vdots \\ p_k \end{pmatrix} \tag{4.28}$$

or shorter:

$$\vec{\boldsymbol{f}} = G\vec{\boldsymbol{p}}. \tag{4.29}$$

This latter equation is the same that occurred in simple linear regression and may be solved in the same way. Keep in mind, however, that we now may have more than two parameters, resulting in a $n \times k$ design matrix with $k > 2$.

Example 1. Consider the quadratic model function

$$f(x; p_0, p_1, p_2) := p_0 + p_1 x + p_2 x^2. \tag{4.30}$$

This is clearly a non-linear function of x, but its dependence on the parameters is linear. We set $g_0(x) = 1$, $g_1(x) = x$, $g_2(x) = x^2$, and obtain

$$\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_1 & x_1^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{pmatrix} \begin{pmatrix} p_o \\ p_1 \\ p_2 \end{pmatrix} \tag{4.31}$$

Example 2. Consider a sinusoidal modulation as it might occur in daily activity data in behavioral measurements. We assume the model

$$f(x; p_0, p_1, p_2) := p_0 + p_1 \sin \omega x + p_2 \cos \omega x, \tag{4.32}$$

where $\omega = 2\pi/\text{day}$ is a daily frequency. As in Example 1, the function is non-linear in x, but linear in the parameters p_0 through p_2. Note that if we made ω a free parameter, a non-linear regression problem would arise.

For the regression equation, we obtain

$$\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & \sin \omega x_1 & \cos \omega x_n \\ \vdots & \vdots & \vdots \\ 1 & \sin \omega x_n & \cos \omega x_n \end{pmatrix} \begin{pmatrix} p_o \\ p_1 \\ p_2 \end{pmatrix} \tag{4.33}$$

In the solution, the parameter p_o describes the average activity, $\sqrt{p_1^2 + p_2^2}$ is the amplitude of the circadian activity modulation, and $\arctan(p_2/p_1)$ is the phase of the activity onset with respect to $x = 0$, measured in radians.

General linear regression is a powerful piece of theory with wide-ranging applications. It is also underlying the so-called "general linear models" in which the residual error $(\vec{\boldsymbol{y}} - f(\vec{\boldsymbol{x}}))$ is written explicitly as

$$\vec{\boldsymbol{f}} = G\vec{\boldsymbol{p}} + \vec{\boldsymbol{\epsilon}}. \tag{4.34}$$

The design-matrix may then take arbitrary components, not necessarily reflecting the values of a function which is to be fitted to the data. In the analysis of imaging-data, e.g., in the functional magnetic resonance imaging, the vector $\vec{f}$ contains the measurements of the BOLD-signal of a voxel for the different experimental conditions and test subjects. The design-matrix describes the experimental design, meaning the conditions under which the individual measurements have come about. From the regression coefficients $\vec{p}$ the statistical significances for every 'contrast' can then be determined (c. Friston et al. 1995).

4.5 Nonlinear Regression

In this case, the minimization of the error function (e.g., Eq. 4.6) has to be solved numerically; MATLAB provides routines for this. The most important algorithms are gradient descent (in the case of explicit determinable partial derivations) and the simplex-procedure.

In the application of these algorithms one has to note that they solve the optimization problem 'iteratively', i.e. by stepwise approach to the optimal point in parameter space. Since this approach is based on local information from the error function, these algorithms may be trapped in so-called local minima, i.e. points in parameter space where the error is below all error values in a neighborhood, but not necessarily the lowest of the entire parameter space. In the case of an optimization with two parameters one can imagine the error function $E(a, b)$ as a landscape over the a, b-plane. When starting the iterative algorithm, the starting point or initial condition (a_o, b_o) has to be specified from which the algorithm searches its path 'downhill'. The algorithm stops when there is no such direction anymore. Here, two problems can arise:

1. The starting point is coincidentally located in an area of the error landscape that is more or less level. Then there is no direction downhill and the algorithm fails. Of course, this does not exclude the possibility that steeper slopes exist elsewhere on the landscape, but the algorithm cannot reach these.

2. The error landscape features several 'valleys' or depressions (local minima). The choice of the starting point then determines which valley will be found. The lowest point of such a valley is not necessarily the lowest point of the whole range.

In both cases it can help to try out different starting points. If the algorithm delivers various solutions for various starting points, then the solution with the least error is to be preferred; a general procedure for the determination of the global minimum does not exist. Incidentally, it should be noted that these problems cannot occur in linear optimization with the SSD error function, since in this case the error landscape forms a paraboloid which possesses only a global and no local minima and no level areas other than the minimal point.

4.6 Examples

4.6.1 Screen Calibration

In the analysis of the visual system, stimuli are presented on monitor screens or video-projectors. Stimuli are generated as an array of brightness values, i.e. numbers I in the interval of $[0, 1]$. We assume that all stimuli are gray-level images and neglect the problems arising from intensity discretization, i.e. the restriction of the monitor to $2^8 = 256$ gray levels.

Physically, the brightness of a pixel is the so-called radiance measured as a power per pixel area and irradiated solid angle. Taking into account the wavelength-dependent sensitivity of the human eye, the relevant quantity is luminance L, measured in Candela/m^2. The relation between

the image intensity I specified in some program or image file and the actually incurring luminance L, however, is not linear but controlled by the so-called γ-correction which takes into account the properties of the human eye and the light-producing components of the screen (LCD, phosphors in CRT-Monitors). In order to understand γ-correction, we first have to introduce the notion of contrast.

Assume a stimulus presenting two fields with constant luminance L_1 and L_2, and let $L_1 \geq L_2$. The two-point of Michelson contrast between these two fields is then defined by the relation

$$c := \frac{L_1 - L_2}{L_1 + L_2}. \tag{4.35}$$

Contrast is a dimensionless variable ranging from 0 and 1 and does not change its value when all luminances are being amplified or scaled-down by a fixed factor. If a constant illumination is added, however, e.g., by illuminating the screen with ambient light, the contrast decreases since the ambient light cancels out in the enumerator but not in the denominator of the fraction. If $L_2 = 0$ and $L_1 > 0$, the contrast is always maximal.

The calibration curve of a monitor screen specifies which luminance L is produced when an image with the intensity of $I \in [0, 1]$ is shown. It has the form of an exponential function,

$$L(I) = a\ r^I. \tag{4.36}$$

Note that $L(0) = a = L_{min}$ is the smallest luminance that can be generated by the screen which here is assumed to be above zero. In CRT-Monitors, this reflects effects of pre-heating of the monitor tube. Of course, the function can easily be modified to allow for true zero luminance if $I = 0$.

The number r is the dynamic range (contrast ratio) of the screen defined as $r = L_{max}/L_{min}$. For LCD screens it is a dimensionless number around 4000. We have $L(0) = a = L_{min}$ and $L(1) = a\ r = L_{max}$.

The reason for using exponential calibration curve is to depict equal differences of the image intensity I with equal contrast. To see how the exponential calibration function does this, consider the contrast of two screen areas which show the intensities I_o and $I_o + \Delta I$. The according contrast is:

$$c_{\Delta I} = \frac{L(I_o + \Delta I) - L(I_o)}{L(I_o + \Delta I) + L(I_o)} = \frac{ar^{I_o+\Delta I} - ar^{I_o}}{ar^{I_o+\Delta I} + ar^{I_o}} \tag{4.37}$$

We can now cancel the expression ar^{I_o} out of the fraction and obtain

$$c_{\Delta I} = \frac{r^{\Delta I} - 1}{r^{\Delta I} + 1}. \tag{4.38}$$

Due to the exponential calibration function, the presented contrast $c_{\Delta I}$ therefore does not depend on I_o but only on ΔI. Equal differences in intensity are thus indeed shown with the same contrast.

Due to the built-in γ-correction, it cannot be assumed in psychophysical experiments that the intensity value specified, e.g. in a MATLAB routine will be the intensity actually presented on the screen. Therefore, it is necessary to first measure $L(I)$ for a series of I-values and then fit the function

$$L(I; a, r) := ar^I \tag{4.39}$$

through these points. This power equation can be linearized by calculating a logarithm on both sides:

$$\log L = \log a + I \log r. \tag{4.40}$$

By means of regression of $\log L$ to I estimates for a and r are obtained. For a required luminance value L^*, one calculates I^* from

$$I^* = \frac{\log L^* - \log a}{\log r}. \tag{4.41}$$

4.6.2 Psychometric Function

Consider a subject watching a screen on which a stimulus may appear at one of two positions ('left' and 'right', say). In every test run the subject shall specify if the stimulus appeared left or right (so-called *two alternative forced choice* Paradigm; 2AFC). Now the stimulus strength (e.g., light intensity) is being varied in the experiment so that some trials are easy while others are impossible to solve. We denote by x the stimulus strength and assume that x is in the interval $[0, 1]$. We now measure the relative frequency of correct answers given a stimulus strength x_i and call this frequency $p(x_i) = P(\text{right answer}|x = x_i)$. We expect $p(x_i)$ to be greater the stronger the stimulus strength x is. Is $x = 0$, the subject can only guess and we expect $p(0) = \frac{1}{2}$. The probability of giving a correct answer by pure guessing is called the (*chance level*), C.

So, at the end of the experiment, a set of measurements $y_i = p(x_i)$, i.e. the relative frequency of correct answers will be available. We now look for a function $\Psi(x)$ fitting the data; it is called a psychometric function. Such a function should grow monotonically from C to 1 and show a saturation for large x. The x-value for which it assumes the mean value between C and 1 is called threshold of perception. The mathematical description of a psychometric function will usually have two free parameters specifying the threshold and the slope.

For the sake of simplicity we now consider sigmoidal functions attaining values between 0 and 1. If $f(x)$ is such a function, then it can be scaled to the interval $[C, 1]$ by

$$\Psi(x) := C + (1 - C)f(x). \tag{4.42}$$

For the modeling of psychometric functions, various mathematical formula are used including, the the aforementioned Arcustangent-Function as well as the error function (see Table 4.1). Both are point-symmetrical to their turning point. Another fit-function that is often used for psychometric functions is the logistic function

$$\Psi(x; \mu, \vartheta) := C + \frac{1 - C}{1 + \exp\{-\frac{x-\mu}{\vartheta}\}}. \tag{4.43}$$

Here the parameter μ shifts the function along the x-axis. The slope is determined by the parameter ϑ. A function with non-symmetrical shape that has been shown to give good results is the Weibull-Function

$$\Psi(x; \alpha, \beta) := 1 - (1 - C)\exp\{-\left(\frac{x}{\alpha}\right)^{\beta}\} \tag{4.44}$$

After having settled for one the these functions, the error term according to Eq. 4.6 has to be considered. The minimization problem is non-linear which should not make any difficulties when using MATLAB. From the parameters determined in the optimization process, the stimulus parameter x can be calculated for which the threshold value $C + (1 - C)/2$ is obtained. This value is the threshold.

4.6.3 Psychometric Functions and Maximum-Likelihood Estimations

In the case of the psychometric function the theoretical value $\Psi(x)$ is a probability, namely the probability of giving the right answer at the stimulus level x. For a fixed value of x, the subject will answer correct or false with a fixed probability and presumably independently across the different trials ("Bernoulli-experiment"). The answers should thus follow a binomial distribution with $p = \Psi(x)$. As a result, the expected error will depend on the value $\Psi(x)$; it is small for small and large values of Ψ and large for medium values. A fitting procedure that assigns equal weight to all errors occurring for different settings of x may therefore run into problems. Rather, errors at each x-level are weighted by the local expected error.

One algorithm that solves this problem is the maximum-likelihood approach. Rather than considering the total square deviation between curve and data, this approach is based on the probability with which the recorded data were to be expected, if the given parameter choice were true. To calculate this probability, we first consider the distribution of the individual measurements, i.e. the binomial distribution mentioned above. The probability to obtain H correct decisions out of n trials with the same stimulus level is:

$$P(H) = \left(\begin{array}{c} n \\ H \end{array} \right) p^H (1-p)^{n-H} = \frac{n!}{H!(n-H)!} p^H (1-p)^{n-H} \tag{4.45}$$

Next, we consider an experiment with stimulus levels $(x_1, x_2, ..., x_n)$ and measured frequencies of correct decisions $(H_1, H_2, ..., H_n)$. Given the theoretical psychometric function $\Psi_{\mu,\vartheta}$, the probability of obtaining these measurements is given by the product

$$L(\mu, \vartheta) := \prod_{i=1}^{n} \left(\begin{array}{c} n \\ H_i \end{array} \right) \Psi_{\mu,\vartheta}(x_i)^{H_i} (1 - \Psi_{\mu,\vartheta}(x_i))^{n-H_i}. \tag{4.46}$$

L is called a "likelihood" (rather than a probability) since it is considered a function of μ and ϑ and the sum of all L-values over these parameters is not 1. $L(\mu, \vartheta)$ is used as an (inverse) error term in the optimization process described in Section 4.5. In practical applications, it is normally replaced by its logarithm,

$$LL(\mu, \vartheta) := \log L(\mu, \vartheta) \tag{4.47}$$

$$= \sum_{i=1}^{n} \log \left(\begin{array}{c} n \\ H_i \end{array} \right) + \sum_{1=1}^{n} H_i \log \Psi_{\mu,\vartheta}(s_i) + (n - H_i) \log(1 - \Psi_{\mu,\vartheta}(s_i)) \tag{4.48}$$

turning the product into a sum. The terms collected in the first sum do not depend on the parameters and can therefore be neglected in the optimization process. Routines for maximum likelihood estimation of psychometric functions are available in the `psignifit` toolbox for MATLAB and python.

The maximum-likelihood process requires the error distributions of each measuring point to be known explicitly. If the errors are normally distributed at every point with the same standard deviation, then maximum-likelihood and least square algorithms give identical results.

Further Reading

Christensen R (2002) *Plane Answers to Complex Questions. The Theory of Linear Models.* Chapter 6: Regression Analysis. Springer Verlag

Macmillan NA, Creelman CD (1991) *Detection Theory: A User's Guide.* Cambridge University Press. (page 186-190: section on "'psychometric functions"')

Friston KJ, Holmes AP, Worsley J-P, Frith CD, Frackowiak RSJ (1995) Statistical parametric maps in functional imaging: a general linear approach. *Human Brain Mapping* 2:189- 210

Wichmann F.A., Hill, N.J. (2001) The psychometric function: I. fitting, sampling, and goodness of fit. *Perception and Psychophysics* 63:1293 – 1313

Chapter 5

Statistical Tests

5.1 Statistical Test Theory

The testing of hypotheses by experimental data is the most important application of probability theory in the empirical sciences. We will start this section by briefly reviewing the basic ideas.

As an example of a simple test problem, consider a coin toss with the possible outcomes 'heads' or 'tails'. The coin is called 'fair', if the probability of the head being up is 1/2. In order to test fairness, one tosses the coin n times and obtains a *random sample* $(x_1, ...x_n)$ of results. The x_i can thereby adopt the values 0 (for head) and 1 (for tails). One will accept the coin as 'fair' if 1 occurs in 'about' half of the trials. If there are notably more or less outcomes 1, the coin will be rejected as unfair.

A statistical test substantiates and quantifies this intuition. It consists of three components:

1. the hypotheses to be tested which are generally formulated as a so-called null hypothesis (H_o) and its logical contraposition (the alternative, H_1). The null hypothesis must be formulated in a way that allows to calculate the probability of the outcome of the test statistic under the condition that the null hypothesis holds true.

2. a test statistic, i.e., a random variable, to be calculated from the random sample (e.g., mean, sample variance, t-value, etc.);

3. a limit or criterion for the test statistic such that the null hypothesis is rejected if this criterion is exceeded. The criterion will depend on the probability distribution of the test statistic under the assumption that the null hypothesis holds true.

5.1.1 Hypotheses

In our example we want to check whether the probability p with which the coin toss leads to the result 'heads' equals to 0.5. We state the hypotheses H_o and H_1:

H_o: $p = 0.5$ (null hypothesis)

H_1: $p \neq 0.5$ (alternative)

The result of the test is a decision for one of the two hypotheses, i.e., acceptance or rejection of the null hypothesis H_o. Intuitively, we will reject H_o if the mean of x_i, $\bar{x}$, deviates substantially from 0.5. However, if $\bar{x}$ is close to 0.5, the null hypothesis is still not proven, since we can never rule out

Table 5.1: Logical and statistical inference

Induction	observation ⇒ proposition
Falsification	absence of observation ⇒ falsehood of proposition
Statistical Test	P(observation|proposition) small ⇒ reject proposition

that there may be other hypotheses predicting the same result. For a coin coming out 50 times head and 50 times tails in 100 tosses, the hypothesis $p = 0.5001$, say, is just as acceptable as the obvious hypothesis $p = 0.5$. In fact, this is the statistical version of a fundamental problem of epistemology; data cannot prove theories but only disprove them (so-called falsification principle). Therefore, if one intends to prove (or support) a given hypothesis, the statistical testing procedure requires to formulate the opposite as a null hypothesis and then attempt to reject this null hypothesis. This statistical falsification of the contraposition is as close as one can get to the intended "proof" of the original hypothesis.

The logic of statistical testing thus relies on the notion of a logical *contraposition*. Assume that a (theoretical) assumption A strictly implies that a certain observation B follows necessarily as in 'if it has rained, the street will be wet'. We write $A \Rightarrow B$. The converse conclusion would be $B \Rightarrow A$, i.e., 'if the street is wet then it has rained'. However, this conclusion is not necessarily correct because there might be other reasons why the street could be wet, such as a water-pipe burst. If, however, the predicted observation does not come true and it was essentially required by the theory, then the theory must be false ($\neg B \Rightarrow \neg A$; 'if the street is not wet then it has not rained'). This latter inference is the contraposition.

The test rejects the null hypothesis if the experimental result is improbable under the assumption that the null hypothesis was accurate. Therefore, the null hypothesis has to be stated in a way that actually allows to calculate probabilities of the outcome. Here, one can distinguish between two basic types of null hypotheses (or tests):

- 'Two-sided tests have null hypotheses in the form of $H_o : p = p_o$, meaning they specify the parameter to exactly one value. The null hypothesis is rejected if p is smaller than $p_o - c$ or larger than $p_o + c$ for a suitable criterion c. Note that both condition can be written jointly as $|p - p_o| \geq c$
- 'One-sided' tests have null hypotheses of the form $H_o : p \leq p_o$, or $H_o : p \geq p_o$, so they only reject if the deviation is to one side.

Beyond this distinction, there are also other types of hypotheses, e.g., those which do not only test numbers but concern entire distributions. An example is the u-test discussed below (Section 5.3).

5.1.2 α- and β-errors (Errors of first and second kind)

Depending on whether the null hypothesis is correct or not and depending on whether the test rejects the null hypothesis or not, one can distinguish four situations, two of which are correct decisions while the other ones are errors:

		ground truth	
		H_o	H_1
test decision	H_o	correct	β-error
	H_1	α-error	correct

The two errors are of a very different kind. The α-error is the rejection of the null hypothesis when it is actually true. This is the more serious error of the two error types, since it results in an erroneous claim. The β-error is the acceptance of the null hypothesis, although it is wrong. In this case, no strong claim is made (the null hypothesis was the default assumption anyway), but one misses the chance of disproving a false hypothesis. This asymmetry becomes particularly clear when considering hypotheses about side effects of medications. If one tests the null hypothesis 'no side effects', then the α-error is to launch a drug with side effects, thereby putting the health of the users at risk. In contrast, the β-error amounts to withholding a non-hazardous drug and reducing the company's sales. In the context of a scientific publication the α-error is in the release of false results while with the β-error accurate results are held back. In statistical testing, it is always assumed that the α-error is the more serious one, meaning it should be eliminated with first priority. In practice, this is more a statement about the wording of the hypotheses: one always has to pick H_o in such a way that the more severe error appears as the α-error.

For each application of a test, the probability of an α value is reported as the "significance" of the test (or the data tested). Generally, many testing schemes with the same probability of an α-error exist for a given problem. Among these, the test with the smallest probability of a β-error is generally chosen. The probability of a β-error is known as the "power"[1] of a test.

5.1.3 Test Statistic, Criterion, Significance in the Binomial Test

The objective of a statistic test is to refute the null hypothesis and to keep the probability of an α-error low. This is done by first calculating a so-called test statistic[2] from the random sample. In the example of the coin toss this is the mean

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i, \tag{5.1}$$

where n is the sample size. The statistic must reflect the relevant information as good as possible and at the same time lend itself to the calculation of the error probability as shown below. Examples for suboptimal choices of a statistic for our problem would be the outcome of just one trial, x_o, because it ignores information, or the logarithm of the x_i, because it unnecessarily complicates the probability computations.

We now compute the probability of an α-error. The α-error occurs when the null hypothesis is correct and the mean $\bar{x}$ nonetheless deviates strongly from the expected value 0.5. Let us assume we would reject the null hypothesis 'fair coin' if $\bar{x}$ exceeds 0.7 or falls below 0.3. Since we selected the sample size $n = 10$, the possible outcomes of $\bar{x}$ are the numbers $0.0, 0.1, 0.2, \ldots, 1.0$ of which the values $\bar{x} = 0;\ 0{,}1;\ 0.2;\ 0.8;\ 0.9$ and $\bar{x} = 1$ have a sufficient deviation to reject H_o. How big, then, is the probability of committing an α-error, i.e., to reject although H_o is correct? We label this likelihood with the letter p; it is called the *significance* of the test.

An α-error can occur only when H_o applies. In our example the distribution of the test statistic is easily specified for this case, it is a binomial distribution with the expectation value $r = 0.5$ and $n = 10$:

$$\mathrm{P}\left(\bar{x} = \frac{k}{n}\right) \quad = \quad \left(\begin{array}{c} n \\ k \end{array} \right) r^k (1-r)^{n-k} \tag{5.2}$$

[1] in German: Güte
[2] in German: Prüfgröße, Statistik

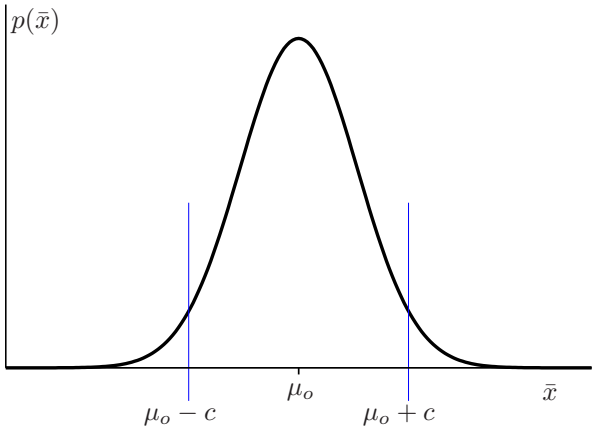


Figure 5.1: Distribution densitiy of a continuous test statistic $\bar{x}$ under the null hypothesis $\mu_o = 0.5$. The test accepts H_o if $\bar{x}$ falls between the two blue markings, i.e. if $|\bar{x} - \mu_o| < c$ holds. The areas below the two 'tails' of the distributions add up to the probability of an error of the 1. kind, α, the so-called significance of the test. The example shows the bilateral test situation. With only one blue marker for the same criterion one gets a one-sided test with significance level $\alpha/2$.

$$= \left(\begin{array}{c} n \\ k \end{array} \right) (\frac{1}{2})^n = \frac{1}{2^n} \frac{n!}{k!(n-k)!} \tag{5.3}$$

From this we obtain:

$\bar{x}$	0.0	0.1	0.2	0.3	0.4	0.5
$\mathrm{P}(\bar{x})$	0.0010	0.0098	0.0439	0.1172	0.2051	0.2461

$\bar{x}$	0.6	0.7	0.8	0.9	1,0
$\mathrm{P}(\bar{x})$	0.2051	0.1172	0.0439	0.0098	0.0010

The probability of $\bar{x}$ becoming smaller than 0.3 can then be calculated by summing the probabilities of the cases $\bar{x} = 0.0,\ 0.1$, and 0.2:

$$\mathrm{P}(\bar{x} < 0.3) = 0.0010 + 0.0098 + 0.0439 = 0.0547. \tag{5.4}$$

Similarly, we get

$$\mathrm{P}(\bar{x} > 0.7) = 0.0439 + 0.0098 + 0.0010 = 0.0547. \tag{5.5}$$

In total, we obtain for the two-sided test problem

$$p = \mathrm{P}(\bar{x} < 0.3|H_o) + \mathrm{P}(\bar{x} > 0.7|H_o) = 0.1094. \tag{5.6}$$

Our test therefore has the significance level 0.1094. The probability that it rejects although H_o applies amounts to 10.94 %.

In experimental work, $p = 0.1094$ is not considered a significant result. We would therefore have to increase our criterion, i.e. $\|\bar{x} - 0.5\| \geq 0.4$ in which case we obtain $p = 0.0216$, a result generally considered as significant. Alternatively, we could increase the sample size n.

Note that in this example, p can only take discrete values, since x and $\bar{x}$ are discrete random variables. In continuous situations, arbitrary values of p can be specified (see Figure 5.1).

If a test with a given p value rejects the H_o, any test with a lower p-value will also reject. In practical applications, one therefore considers the highest $p - value$ that still allows the rejection of the H_o. It equals the probability of the α-error which may therefore be identified with the test significance. These significance values are usually tabulated as function of the test statistic. The computation of those tables, i.e., the determination of the likelihood distribution of the test statistic under the assumption that the null hypothesis is correct, is a problem solved in mathematical statistics.

5.1.4 Effect Size

Strong effects may reach significance already with small sample sizes while weaker effects may be brought to significance by increasing the sample size. Significance itself is therefore not a suitable measurement for the strength or size of an effect. In recent years, it has been realized that effect sizes are an important characteristic of an experimental result, in addition to and partially independent of the significance. In the two-sample t-test discussed below, effect size is related to the difference of the sample means, divided by the square root of the average variance. A general definition of effect size, however, does not exist, rather, the problem has to be dealt with in specific ways for each test problem.

One strategy to avoid significance of tiny effects is to restrict the sample size already when first designing the experiment. An effect that does not reach significance with 20 participants, say, is not a strong effect, even though it might reach significance when 20 more participants are included in the study.

5.2 The t-Test

If one collects two samples in two populations for one measured variable, then one gets two mean values $\bar{x}_1$ and $\bar{x}_2$ with the associated standard deviations. Hence, the question is if the two mean values deviate from one another significantly. This will be the case, e.g., when the difference of the mean values is large but their scattering small. If in reverse the scattering is notably greater than the difference of the mean values, then it is possible that the theoretical expectation values (i.e., the mean values of the population) are the same.

As example we consider an experiment where test subjects are supposed to solve a navigation task in an unknown, virtual environment (i.e., an environment presented as interactive computer graphics) (Restat et al. 2004). The observable (dependent variable) is the number of navigation errors, i.e. the number of path decisions which lead away from the goal. The two 'populations' are two groups of subjects who were trained and tested in two different conditions of the environment. In condition 1 the surroundings were plane, in condition 2 they were inclined in the shape of a uniform slope.

For better illustration of the test procedures we are going to use fictitious data. Fig. 5.2 shows the histograms for the variables x_1 and x_2 as well as the associated mean values and error bars. The hypotheses are:

H_o: $Ex_1 - Ex_2 = 0$

H_1: $Ex_1 - Ex_2 \neq 0$

Ex is the expectation value of x, i.e., the 'mean of the population' approached for infinite sample sizes. The hypotheses formulate a two-sided test.

The test statistic of the t-test is essentially the difference of the sample means divided by the standard deviation of this difference. We assume first that the variables x_1 and x_2 are independent of each other in the sense of probability theory. In the example this condition would be met when the navigation experiment is carried out in the two conditions with different groups of test subjects (between-subject design). In this case the variance of the difference equals the sum of the variances:

$$\mathrm{var}(\bar{x}_1 - \bar{x}_2) = \mathrm{var}\ \bar{x}_1 + \mathrm{var}\ \bar{x}_2 = \frac{\mathrm{var}\ x_1}{n_1} + \frac{\mathrm{var}\ x_2}{n_2} \tag{5.7}$$

For simplicity's sake we assume that the random sample sizes n_1 and n_2 are equal. For a test statistic we get:

$$t := \sqrt{n}\ \frac{|\bar{x}_1 - \bar{x}_2|}{\sqrt{s_1^2 + s_2^2}} \tag{5.8}$$

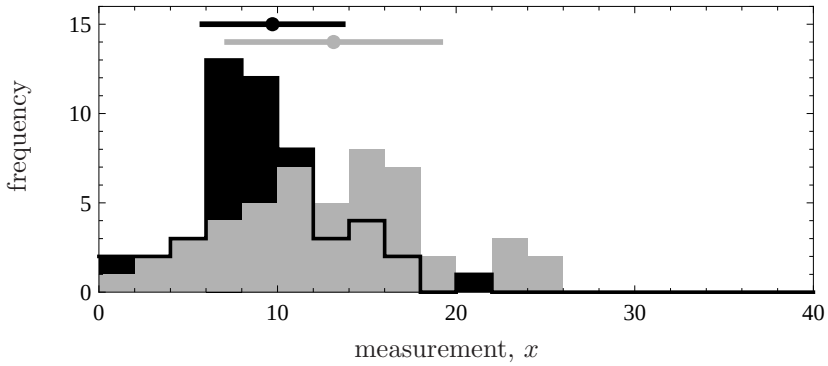


Figure 5.2: Sample data for the t-test. Random samples are being drawn respectively from 50 specimen from two independent populations. The figure shows the histograms of such random samples together with their mean values and standard deviations. The t-test is significant at the 0.1%-level.

where s_1 is the sample variance of x_1,

$$s_1^2 := \frac{1}{n-1}\left(\sum_{i=1}^{n} x_{1i}^2 - \frac{1}{n}\left(\sum_{i=1}^{n} x_{1i}\right)^2\right) = \frac{1}{n-1}\left(\sum_{i=1}^{n} x_{1i}^2 - n\bar{x}_1^2\right), \tag{5.9}$$

and s_2^2 the one of x_2 accordingly.

It should be kept in mind that the test statistic is itself a random variable. If one repeats the experiment, then the measurement values x_{1i}, x_{2i}, their mean values and standard deviations as well as the test statistic t will take different values. The distribution of the test statistic t is known if it is additionally assumed that the data have normal distribution. It is the so-called Student t-Distribution with $2n-2$ degrees of freedom. The number of the degrees of freedom is a parameter of the distribution which we do not need to interpret further.[3]

In the example of Fig. 5.2 the test statistic takes the value $t = 3.386$ with $n = 50$. For the two-sided test with 98 degrees of freedom one reads from the published tables for the t-test: $p = 0.001$. The probability to obtain this large t value if the populations are actually equal (null hypothesis) is therefore just 0.1%. We can thus reject the null hypothesis at the 0.1% level.

In case of the measurements not being statistically independent, one speaks of a t-test with repeated measures[4]. In this case, we proceed just as before, but have to use a different table to obtain the significance of a given t-value.

The t-test can be used also for testing the deviation of a mean value from a theoretical value (one-sample-t-test). In this case the test statistic is given by

$$t := \sqrt{n}\frac{|\bar{x} - \mu_o|}{s} \tag{5.10}$$

where μ_o is the theoretical mean and s the sample variance as before.

[3]Typically, the number of the degrees of freedom is the number of the single measurements minus the number of variables estimated from these measurements and used in the computation (e.g. sample mean, random sample variance). Why this correction is necessary can be illustrated in the sample variance where a division by $n-1$ rather than by n is involved (see Eq. 5.9). The calculation of the sample variance relies on the sample mean which is itself estimated from the same data and tends to absorb part of the variance. The formula given in Eq. 5.9 can be shown to be un unbiased estimator of the population variance.

[4]in German: t-Test mit Messwiederholung oder mit verbundenen Stichproben

The one-sample-t-test generalizes our example of testing the fairness of a coin to the case of continuous variables or large samples. It is also used in connection with the regression analysis in order to determine if the residues of the regression (difference between data and fit) are significant.

The t-test presumes that the measurements are distributed normally. Tests that do not need this assumption are called 'non-parametric'. There are non-parametric tests for comparing two distributions or their mean values for the task, e.g., the sign test and the u-test discussed below. If the assumption of the normal distribution is correct, the parametric tests are usually preferable because for a fixed significance level α they are less compromised by β-errors, i.e., they exert a hight 'power'.

5.3 A Rank-Test

A test for the equality of the mean values of two random samples that does not assume normal distribution of the data is the Mann-Wilcoxon-Whitney-u-test. It does assume that the distributions of the samples differ only by a shift which implies that the variances and all higher central moments are equal. The null hypothesis states that also the means are equal while the alternative assumes different means.

Consider two samples with the sample sizes $n = m = 4$.

$$\begin{aligned}(x_i)_{i=1,\ldots,n} &= (6, 5, 15, 11)\\ (y_i)_{i=1,\ldots,m} &= (17, 12, 16, 10).\end{aligned}$$

We now pool both samples, arrange the elements according to size and determine the 'rank number', i.e., the position in the common set. At the same time, we keep in mind from which sample the values originated.

Value	5	6	10	11	12	15	16	17
Origin	X	X	Y	X	Y	X	Y	Y
Rank	1	2	3	4	5	6	7	8

If the samples are similar, then the X- and Y-entries in the second line will be homogeneously mixed. To obtain a test statistics, we determine the ''rank sums' separately for the values of the two samples:

$$\begin{aligned}R_X = \quad & 1 + 2 + \quad 4 + \quad 6 \quad\quad\quad = 13\\ R_Y = \quad & \quad\quad 3 + \quad 5 + \quad 7 + 8 \quad = 23\end{aligned}$$

From these, the u-statistics are calculated as

$$u_X := nm + \frac{m(m+1)}{2} - R_X \tag{5.11}$$

$$u_Y := nm + \frac{n(n+1)}{2} - R_Y. \tag{5.12}$$

In the example we get $u_X = 13$ and $u_Y = 3$. As a test, observe that the relation $u_X + u_Y = nm$ must always hold. From those two u-values one chooses the smaller one and reads the significance from the table. In the example we can reject the null-hypothesis (equality of the mean values) on the 10%-level.

References

Bortz, J. (2005). *Statistik für Sozialwissenschaftler.* Springer Verlag, Heidelberg, 6. Aufl..

Fisz, M. (1980). *Wahrscheinlichkeitsrechnung und mathematische Statistik.* VEB Deutscher Verlag der Wissenschaften, Berlin.

Hussy, W., Jain A. (2002). *Experimentelle Hypothesenprüfung in der Psychologie.* Hogrefe, Göttingen etc.

Larsen, R.J., Marx, M.L. (2011). *An Introduction to Mathematical Statistics and Its Applications.* 5th Edition, Pearson.

Sachs, L. (2003). *Angewandte Statistik.* Springer Verlag, Berlin, Heidelberg, New York, 11. Aufl.

Restat, J., Steck, S. D., Mochnatzki, H. F., and Mallot, H. A. (2004). Geographical slant facilitates navigation and orientation in virtual environments. *Perception*, 33:667 – 687.

Chapter 6

Analysis of Variance

6.1 Experimental Design: Factors and Factor Levels

The t- and u-tests described in the previous chapter allow to compare pairs of samples occurring in experiments with two experimental conditions (treatments), one of which may be a control. Often, however, multiple conditions are involved in an experiment, for example a number of doses of one drug or the combined doses of two drugs being delivered to the subject in each condition. Here it is important to distinguish between "factors" and "factor levels" which can be thought of as the dimensions along which the conditions are varied and the specific values taken by each factor in the various conditions.

In the example of the dose of a drug delivered to a subject, the factor levels are numbers, i.e. follow an interval or even ratio scale (cf. Section 1.2). In this case, the relation between dose and effect may be considered a continuous function and the data can be analyzed with regression approaches (see Chapter 4). In the analysis of variance, interval scaling of factor levels is not required, however. If, for example, different drugs are tested each with just one dose level, these drugs may be treated as levels of a factor "medication" which has nominal scale. Clearly, curve fitting does not make any sense in this situation. Still, it should be kept in mind that the factors correspond to the independent variables of an experiment and the factor levels are the values (or nominal cases) chosen or set by the experimenter.

Experimental conditions may vary different factors simultaneously. This would be the case, for example, if two drugs are administered jointly, each in one of a number of dose levels. Here, the doses would be the factor levels and the experimental question would probably be about the effect of each drug independent of the other, as well as about possible drug interactions. Clearly, the experiment requires to test experimental conditions using combinations of the levels from both factors. If all possible combinations are indeed tested, the experiment is said to have a "full factorial design". In this case, the number of conditions equals the product of the number of levels of each factor. Since the number of conditions can get quite large, designs involving only part of the combination have been developed.

6.2 Multiple Tests

In an experiment involving k conditions, $k(k-1)/2$ pair comparisons between the samples obtained in each condition can be carried out with the t- or u-tests. Besides the computational effort of this procedure, the risk for one of the individual pair comparisons to reach significance coincidentally increases with a growing number of tests. In other words, a statement that is based on six individual

Table 6.1: Data table for a one-factorial analysis of variance

factor	e.g., 'drug dose'			
levels	1	2	3	
single measurements	x_{11} x_{12} $\vdots$ x_{1n_1}	x_{21} x_{22} $\vdots$ x_{2n_2}	x_{31} x_{32} $\vdots$ x_{3n_3}	
sums	$\sum_{j=1}^{n_1} x_{1j} =: x_{1\cdot}$	$\sum_{j=1}^{n_2} x_{2j} =: x_{2\cdot}$	$\sum_{j=1}^{n_3} x_{3j} =: x_{3\cdot}$	
group means	$\bar{x}_{1\cdot} := x_{1\cdot}/n_1$	$\bar{x}_{2\cdot} := x_{2\cdot}/n_2$	$\bar{x}_{3\cdot} := x_{3\cdot}/n_3$	
sample size				$n := \sum_{i=1}^{k} n_i$
grand sum				$x_{\cdot\cdot} := \sum_{i=1}^{k} \sum_{j=1}^{n_i} x_{ij}$
grand mean				$\bar{x}_{\cdot\cdot} := x_{\cdot\cdot}/n$

tests (as would occur in a design with 4×3 factor levels), each with significance level α has an overall probability of false rejection of $1-(1-\alpha)^6 \approx 6\alpha$ (at least if independence is assumed). E.g., if the individual pair comparison is significant with $\alpha = 0.01$, the overall significance level would be reduced to 0.059. One can try to compensate this through increased significance requirements for the individual tests, i.e. by setting the initial significance levels of the individual pair comparisons to $\alpha/6$. This approach is known as the Bonferoni-correction; it is associated with a loss in test power.

6.3 One-Factorial Analysis of Variance

6.3.1 Hypotheses

The 'analysis of variance' (*ANOVA*) avoids the problem of multiple pair comparisons. In the one-factorial case, we have a dependent variable x which in our examples represents drug action, learning success, etc. With x_1 to x_k we label the measurements performed at the factor levels $1, ..., k$ sampling the independent variable. Let the number of measurements per factor level be n_i with $n = \sum_{i=1}^{k} n_i$ being the total sum of measurements. We denote the individual measurements with a second index, x_{ij}, where j now runs from 1 to n_i. In an example, x_{ij} could be the drug action found for the j-th subject in the i-th dose level. An overview of the terminology is given in Table 6.1. Note that in the sums and means the index over which summation was performed is replaced by a dot.

The hypotheses one wants to test in this case are:

H_o: $Ex_1 = Ex_2 = ... = Ex_k$

H_1: There exist (at least) two factor levels l, m, such that $Ex_l \neq Ex_m$.

An alternative way to formulate the ANOVA hypotheses uses a linear model of the form

$$x_{ij} = \mu + \alpha_i + \epsilon_{ij}, \tag{6.1}$$

Table 6.2: Two measurements with equal total variance but different partitionings.

	x_1	x_2	x_3
	1	2	3
	1	2	3
	1	2	3
	1	2	3
$\bar{x}_{i\cdot} =$	1	2	3

	y_1	y_2	y_3
	1	3	3
	3	3	1
	2	1	2
	2	1	2
$\bar{y}_{i\cdot} =$	2	2	2

where μ is the expected value of x over all conditions, approximated by $\bar{x}_{\cdot\cdot}$ in Table 6.1. The α_i are the deviations from μ of the expectation values of the different factor levels, approximated by the $\bar{x}_{i\cdot} - \bar{x}_{\cdot\cdot}$, and the ϵ_{ij} describe the residual error, which is assumed to be distributed normally in the parametric ANOVA. The null hypothesis is then equivalent to the assumption that all α_i disappear simultaneously. Eq. 6.1 can also be understood as regression equation, connecting ANOVA to the regression analysis and general linear models from Section 4.4.

6.3.2 Test Statistic: Partitioning the Sum of Squares

In order to derive a test statistic for the above hypotheses, we consider the overall variance of the total sample, i.e. after lumping all measurements from all conditions into one sample. If H_o holds, the total variance is equal to the sum of the group variances, at least as long as the groups can be treated as statistically independent (this is not always true, see comment at the end of the section). If the group means are different, however, this difference will generate a between-group variance contributing to the the total variance. The idea of ANOVA, therefore, is to ask whether the variance between the groups is greater than the variance within the groups. Under the null hypothesis, the variance between groups should be close to zero while all existing variance would be generated by the error term ϵ_{ij} from Eq. 6.1.

As an example, consider the ficticious data given in Table 6.2: For the the variable x (left table) the variance within each group is zero, while the group means vary. For the variable y (right table) the group means are equal such that the group variance is zero. Note that the individual values $\{x_{ij}\}$ and $\{y_{ij}\}$ occuring in each table are identical, and so are the total variances for both variables.

Instead of the variances we now turn to the sums of the squared differences. Note that the sum of squared differences equals the sample variance up to a normalizing factor. We start with the lumped sample:

$$Q_{total} = \sum_{i=1}^{k}\sum_{j=1}^{n_i}(x_{ij} - \bar{x}_{\cdot\cdot})^2 = \sum_{i=1}^{k}\sum_{j=1}^{n_i}\left(x_{ij} - \frac{1}{n}\sum_{i=1}^{k}\sum_{j=1}^{n_i}x_{ij}\right)^2 \tag{6.2}$$

Calculating Q_{total} for the variables x and y from Table 6.2 yields the $Q_{total} = 8$ for both variables alike, since the order of the terms is immaterial in the sum.

Within each group we get the square differences:

$$Q_i = \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_{i\cdot})^2 = \sum_{j=1}^{n_i} \left(x_{ij} - \frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij} \right)^2 . \tag{6.3}$$

The sum of this Q_i is a measure for the total variation within the groups:

$$Q_{within} = \sum_{i=1}^{k} Q_i = \sum_{i=1}^{k} \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_{i\cdot})^2 = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left(x_{ij} - \frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij} \right)^2 . \tag{6.4}$$

For the variable x in Table 6.2 all Q_i and Q_{within} take the value 0; for the variable y, we have $Q_1 = Q_3 = 2$, $Q_2 = 4$ and $Q_{within} = 8$.

Finally, we consider the square deviations which arise when we replace every value with its group mean and then calculate the total deviation:

$$Q_{between} := \sum_{i=1}^{k} n_i (\bar{x}_{i\cdot} - \bar{x}_{\cdot\cdot})^2 = \sum_{i=1}^{k} n_i \left(\frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij} - \frac{1}{n} \sum_{i=1}^{k} \sum_{j=1}^{n_i} x_{ij} \right)^2 \tag{6.5}$$

The factor n_i results from the fact that every entry in the group i has been replaced by the group mean $\bar{x}_{i\cdot}$; this value therefore appears n_i times. For the variable x in Table 6.2 $Q_{between}$ takes the value 8, for y we find $Q_{between} = 0$. Generally, one can show that the relation

$$Q_{total} = Q_{within} + Q_{between} \tag{6.6}$$

is always true for independent random variables (cf. e.g. Fisz 1980, p. 611).

Equation 6.6 is known as the partitioning of the sum of squares. For the test statistic, we go back to the sample variances and calculate the ratio between the sample variances in the groups and between the groups. For this, we have to include a factor involving the total sample size n and the number of groups (factor levels) k. Similar to the situation for the t-test (Section 5.2) the numbers $k-1$ and $n-k$ are called the degrees of freedom of the F-statistic. They reflect the fact that in the estimation of within and between variances, means are involved which are also estimated from the same data. $Q_{between}/(k-1)$ and $Q_{within}/(n-k)$ are the unbiased estimators for the variances $V_{between}$ and V_{within}, respectively.

We obtain the test-statistic (F-statistic)

$$F = \frac{\text{Variance between the groups}}{\text{Variance within the groups}} = \frac{\frac{1}{k-1}\, Q_{between}}{\frac{1}{n-k}\, Q_{within}} \tag{6.7}$$

We expand $Q_{between}$ and Q_{within} from the above equations and apply the so-called Steiner formula:

$$\frac{1}{n} \sum_{i=1}^{n} \left(a_i - \frac{1}{n} \sum_{i=1}^{n} a_i \right)^2 = \frac{1}{n} \sum_{i=1}^{n} a_i^2 - \left(\frac{1}{n} \sum_{i=1}^{n} a_i \right)^2 , \tag{6.8}$$

which can be proven elementally by the calculation of the binomial term.

Finally, we obtain the test statistic for the one-factorial (one-way) ANOVA as

$$F = \frac{n-k}{k-1} \frac{\sum_{i=1}^{k} \frac{x_{i\cdot}^2}{n_i} - \frac{x_{\cdot\cdot}^2}{n}}{\sum_{i=1}^{k} \sum_{j=1}^{n_i} x_{ij}^2 - \sum_{i=1}^{k} \frac{x_{i\cdot}^2}{n_i}} \tag{6.9}$$

$$= \frac{n-k}{k-1} \frac{\sum_{i=1}^{k} \frac{1}{n_i} \left(\sum_{j=1}^{n_i} x_{ij} \right)^2 - \frac{1}{n} \left(\sum_{i=1}^{k} \sum_{j=1}^{n_i} x_{ij} \right)^2}{\sum_{i=1}^{k} \sum_{j=1}^{n_i} x_{ij}^2 - \sum_{i=1}^{k} \frac{1}{n_i} \left(\sum_{j=1}^{n_i} x_{ij} \right)^2} \tag{6.10}$$

The distribution of the random variable F is the so-called Fisher-F-Distribution with degrees of freedom $k-1$ and $n-k$. This distribution is tabulated. For a given value of the test statistic the significance is being determined automatically in MatLab or read from according tables (e.g. in Bortz 2005).

As in the t-test we have presumed that the populations x_i are statistically independent. For this, every experimental condition has to be carried out by different subjects; this is called 'between subject design'. If for one given experiment the performance of all conditions by each subject is preferred, a 'within subject design' is obtained. In that case, statistical independence does not obtain. The overall procedure for carrying out the ANOVA is the same in this situation, but a different table for the F-distribution has to be used, known as the table for ('repeated measures ANOVA') as described, e.g., in the textbooks of Bortz (1999) or Sachs & Hedderich (2009).

A non-parametrical variety of the analysis of variance is the Friedman-Test, cf. e.g. Larsen & Marx (1986).

6.4 Two-Factorial Analysis of Variance

In many cases the various experimental conditions will be organized by more than one factor. When simultaneously treating with two different medications, one may ask if (i) drug 1 has an effect, or (ii) drug 2 has an effect, or if (iii) both drugs are effective only in combination, but not on their own. A corresponding experimental design has got two factors whose factor levels match the dosage of the two drugs. In the cases (i) and (ii) mentioned above one speaks of *Main Effects*, in the case (iii) of an *Interaction*.

The linear model for the two-factorial case is:

$$x_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk}. \tag{6.11}$$

x_{ijk} is the k-th single measurement in the experimental condition where the factor 1 has level i and factor 2 has level j. If there is a main effect in factor 1, then $\alpha_i \neq 0$ for at least on level i. With a main effect in factor 2, we have $\beta_j \neq 0$ for some j. The term $(\alpha\beta)_{ij}$ is not meant as product, but as a coefficient of the interaction; an interaction exists if $(\alpha\beta)_{ij}$ is different from zero for at least one combination of i, j. The remaining error ϵ carries three indices, two for the factor levels and one for the individual measurement. As in the one-factorial ANOVA one assumes that the ϵ_{ijk} and with that the x_{ijk} are normally distributed. As for the one-factorial case, a repeated measures ANOVA is one where the x_{ijk} are statistically dependent.

Fig. 6.1 schematically presents the various effects and interactions which can occur in a two-factorial model. For factor 1 three levels are assumed which are arranged equidistantly on the ordinate. If one changes the ordering of the factor levels on the axis, then the straight lines would become curved ones. This does not influence the statistical analysis with ANOVA. Ultimately, ANOVA only tests if the mean values of the respective groups deviate from one another or not.

Note that the visualization of the data in Figure 6.1 is asymmetrical in the sense that factor 1 is represented along the horizontal axis whereas factor 2 is depiced by the two lines in each plot. Of course we could have chosen to plot the effects just the other way round, with two ticks on the

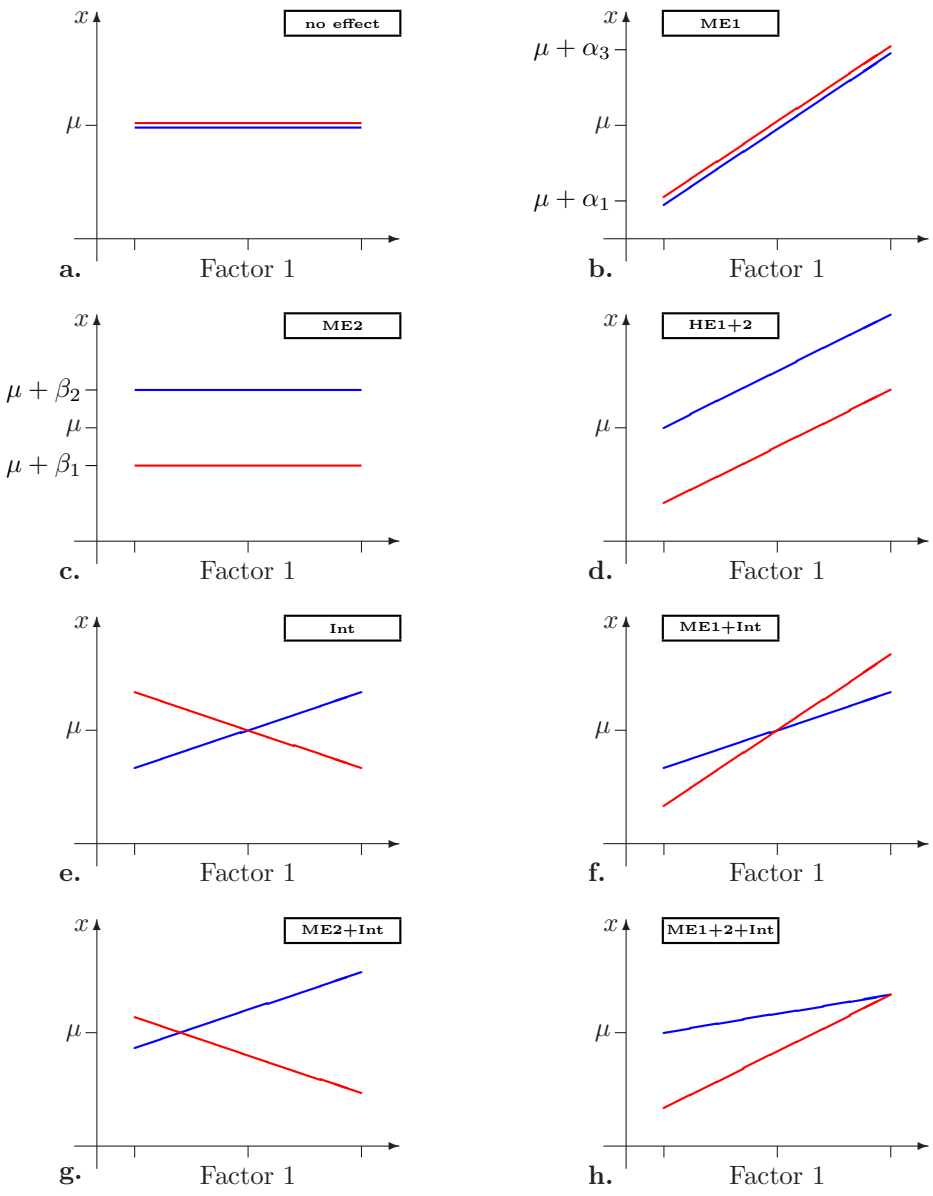


Figure 6.1: Overview of possible effects in a two-factorial ANOVA with three levels for factor 1 (horizontal axis) and two levels for factor 2 (blue and red lines). **a.** No effect; in Eq. 6.11, $\alpha_i = \beta_j = (\alpha\beta)_{ij} = 0$. **b.** Main effect 1 without further effects. **c.** Main effect 2 without further effects. **d.** Combination of both main effects. **e.** Interaction without main effects. **f.** Main effect 1 together with an interaction. **g.** Main effect 2 together with an interaction. **h.** Main effects 1 and 2 together with an interaction.

horizontal axis for the levels of factor 2 and three lines in the plot for the levels of factor 1. More naturally, the whole thing can be plotted as a 3D surface or staircase with coordinate axes for both factors. A main effect for factor 1 would then amount to a slope in one coordinate direction, a main effect of factor 2 for a slope in the other coordinate direction, and an interaction to a saddle-shaped surface.

6.5 References

Bortz, J. (2005). *Statistik für Sozialwissenschaftler*. Springer Verlag, Heidelberg, 6. Aufl..

Fisz, M. (1980). *Wahrscheinlichkeitsrechnung und mathematische Statistik*. VEB Deutscher Verlag der Wissenschaften, Berlin.

Larsen, R.J., Marx, M.L. (2011). *An Introduction to Mathematical Statistics and Its Applications*. 5th Edition, Pearson.

Sachs, L., Hedderich J. (2009). *Angewandte Statistik*. Springer Verlag, Berlin, Heidelberg, New York, 13. Aufl.

Chapter 7

Bivariate Statistics

In the two-factorial ANOVA studied in the previous chapter, we have considered measurements with two independent variables. Here we study the reverse case, i.e. multiple dependent variables, which occurs if more than one variable is simultaneously measured in each trial of an experiment or each specimen of a sample. We start with the bivariate case, i.e. two dependent variables. The multivariate case will be considered in the next chapter.

In bivariate measurement, two distinct variables x and y are measured simultaneously, i.e. for one value of the independent variable(s). Examples include:

- *Feature Vectors:* In this case two (or more) features of a specimen are measured, for example the length of the hind limb and the circumference of the head in a mouse, the frequencies of two alleles in a population, time and maximum amplitude of an action potential, or the intensity of a vocalization in two different frequency bands.
- *State Variables:* In a neural net, the activity of two (or more) cells at a time is measured. Likewise, in a blood sample, one could measure simultaneously the concentrations of oxygen and glucose.
- *Spatial Position*: In the plane, position measures are always two-dimensional, expressed for example as x- and y- coordinates. Position measurements occur, e.g., in eye movement studies (saccades to flashed targets), spatial cognition (accuracy of place recognition), or eye-hand coordination (hitting a target with the finger), etc.

Multivariate measurements are not the exception but the rule. An extreme case is the recording of images where one can interpret the intensity value of each of $n \times m$ pixels as a variable of its own, i.e. as a component of an nm-dimensional data vector. Likewise, time signals such as action potentials can be sampled at n time-steps and treated as n-dimensional data vectors. The mathematical concepts in the bivariate and multivariate case are surprisingly similar. We will come back to multivariate problems in the next chapter.

7.1 Correlation and Covariance

The data points of a bivariate measurement can be written as two-dimensional vectors (x_i, y_i) where i indicates the number of the measurement and includes information about the setting of the independent variable(s). We assume that n such measurements exist.

An important characteristic for the relations of variables to each other is the covariance, which is based on the deviations of the variables from their respective means. If, for a given i, both

variables tend to deviate from their means in the same direction, i.e. if large values for x tend to be associated with large values of y, and small values of x with small values of y, then the covariance is positive. If the reverse is true, (large x associated with small y and small x associated with large y), covariance will be negative. We write:

$$\mathrm{cov}(x,y) := \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y}); \tag{7.1}$$

where

$$\bar{x} := \frac{1}{n}\sum_{i=1}^{n}x_i \quad \text{and} \quad \bar{y} := \frac{1}{n}\sum_{i=1}^{n}y_i \tag{7.2}$$

denote the mean values of x and y.

Equation 7.1 defines the sample covariance which is a random variable depending on the data sample drawn from some population. For the covariance of the entire population, the term $\frac{1}{n-1}\sum_{i=1}^{n}$ has to be replaced by the expected value E. The sample covariance is an unbiased estimator of the population covariance for the case that the expected values of x and y are estimated by the sample means (see also the comment after Eq. 5.9 on page 51).

The covariance of a variable with itself is called variance; it cannot become negative.

$$\mathrm{var}(x) := \mathrm{cov}(x,x) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2 = \tag{7.3}$$

By normalizing the covariance to the variances of the involved variables, we obtain a quantity which is limited to values between -1 and $+1$; it is called correlation:

$$\mathrm{cor}(x,y) := \frac{\mathrm{cov}(x,y)}{\sqrt{\mathrm{var}(x)\mathrm{var}(y)}}. \tag{7.4}$$

Consider a sample of n bivariate measurements. We subtract the means and arrange the data as a $n \times 2$ matrix D,

$$D = \begin{pmatrix} x_1-\bar{x} & \dots & x_n-\bar{x} \\ y_1-\bar{y} & \dots & y_n-\bar{y} \end{pmatrix}, \tag{7.5}$$

we can calculate the variances and covariances in the form of a covariance matrix C as

$$C = \frac{1}{n-1}D^\top D = \begin{pmatrix} \mathrm{var}(x) & \mathrm{cov}(x,y) \\ \mathrm{cov}(x,y) & \mathrm{var}(y) \end{pmatrix} \tag{7.6}$$

(see also Section 2.3.4). Covariance is commutative, $\mathrm{cov}(x,y) = \mathrm{cov}(y,x)$, and the covariance metric is symmetric, $C^\top = C$.

The two-dimensional data values (x_i, y_i) can be visualized as a scatter plot with coordinate axes x and y. The center, or more precisely the center of gravity of this scatter plot is the mean vector taken over all measurements; it has the components $(\bar{x}, \bar{y})$. The total area covered by the cloud of dots is related to the variances and the covariances of the variables. In particular, $\mathrm{var}(x)$ indicates the one-dimensional spread in x-direction and $\mathrm{var}(y)$ the one-dimensional spread in y-direction. The area of the rectangle spanned by the two variances, i.e. the product $\mathrm{var}(x)\mathrm{var}(y)$, can be thought of as the total (two-dimensional) spread of the data set if covariance is zero. If not, $\mathrm{cov}(x,y)$ corresponds to the deviation of the two-dimensional spread from the rectangle formed by the two variances. If the covariance is large, then the two variables are linearly connected, and the cloud of dots will be elongated along a straight line which lies diagonally in the rectangle spanned by the variances.

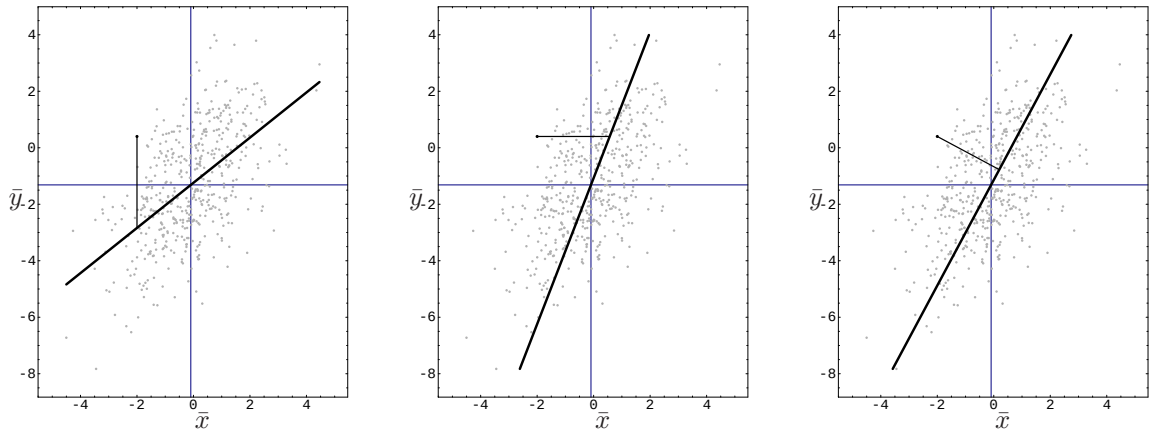


Figure 7.1: Two-dimensional cloud of dots. The data are drawn randomly from a two-dimensional normal distribution. **a.** Regression of y to x (cf. section 4.3) minimizes the sum of squared y-distances of the points from the regression line. **b.** Regression from x to y minimizes the sum of squared x-distances of the points from the regression line. **c.** The first principal component minimizes the sum of squared distances measured orthogonally to the straight line.

7.2 Regression Lines and Principal Components

In order to estimate a linear relation between x and y from a sample of bivariate date, three possibilities exist (Fig. 7.1):

1. Regression from y to x: This case was discussed in Section 4.3. One obtains a straight line that minimizes the distances of the points in the y-direction (Fig. 7.1a).

2. Regression from x to y: When exchanging the roles of x and y in the calculation of the regression line, one gets again a regression which, however, will minimize the error in x-direction. The two regression lines are generally not the same; they differ from each other the more, the smaller the correlation coefficient $\mathrm{cor}(x, y)$ becomes (Fig. 7.1b).

3. Finally, there is the possibility of measuring the deviation between data points and a straight fitting line perpendicular to this line. Note that in this case the direction of measurement depends on the fit. This process is symmetric in the sense that the exchange of x and y will now produce the same line. It is called the first principal component of the cloud of dots. If the shape of the cloud of dots can be approximated by an ellipse, the first principal component equals the major semi-axis of this ellipse (Fig. 7.1c).

If, in the experimental design used, x is the independent variable and y the (sole) dependent variable, regression of y on x is the right thing to do. The independent variable is then also called "regressor" and the dependent variable "regressand". In the bivariate case, however, both x and y are dependent variables, and it is not always clear which direction to choose for a regression analysis. In the bivariate case, therefore, principle components are generally more meaningful.

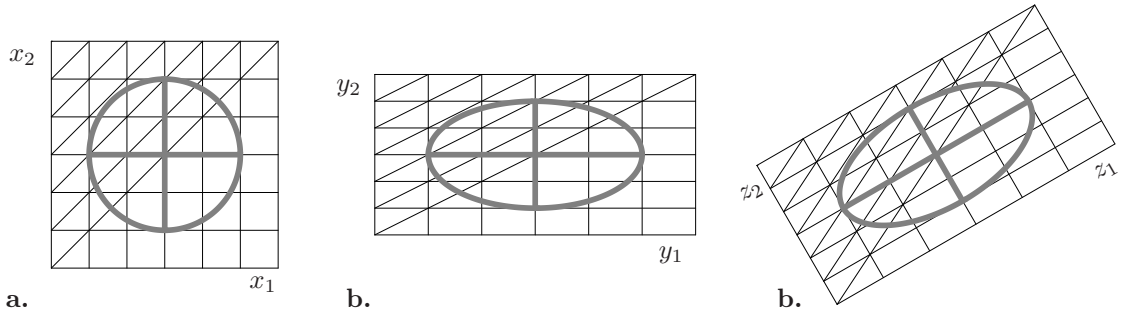


Figure 7.2: Formation of an ellipse from a circle (**a.**) through stretching or compression along the coordinate axes and (**b.**) subsequent rotation (**c.**). The large and small axis of the ellipsis are marked.

7.3 Detour: Ellipses

Point clouds as depicted in Fig. 7.1 can be described by elliptic areas covering a certain percentage of the data points. In this section we discuss briefly the analytical geometry of such ellipses in matrix-notation. It will help to better understand the two-dimensional normal distribution but may be skipped at first reading.

In general, ellipses can be thought of as transformations of circles by a sequence of two of the similarity transformations introduced in Section 2.3.1. First, the circle is stretched or compressed along the coordinate axes, thereby creating an ellipse whose major and minor axis are aligned with the coordinate axes y_1, y_2 (Fig. 7.2b). If the radius of the original circle was 1, the lengths of the major and minor semi-axes will equal the stretch or compression factors of the transformation. In the second step the ellipse is rotated without changing the lengths of the semi-axes (Fig. 7.2c).

A circular arc about the origin with radius 1 includes all points (vectors) of the plane which satisfy the condition

$$x_1^2 + x_2^2 = \vec{\boldsymbol{x}}'\vec{\boldsymbol{x}} = 1. \tag{7.7}$$

As stretching/compression operator, we consider a matrix A with

$$A = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}. \tag{7.8}$$

It is easy to see that the vectors $\vec{\boldsymbol{e_1}} = (1,0)'$ and $\vec{\boldsymbol{e_2}} = (0,1)'$ are eigenvectors of this matrix, and that the eigenvalues are λ_1 and λ_2. The coordinate transform describing the transition from Fig.7.2a to Fig. 7.2b is, then:

$$\vec{\boldsymbol{x}} \mapsto \vec{\boldsymbol{y}} := A\vec{\boldsymbol{x}}. \tag{7.9}$$

We now solve this equation for $\vec{\boldsymbol{x}}$ and substitute the result in the equation of the circle, Eq. 7.7.

$$(A^{-1}\vec{\boldsymbol{y}})'(A^{-1}\vec{\boldsymbol{y}}) = 1 \tag{7.10}$$

$$\vec{\boldsymbol{y}}'(A^{-1})'A^{-1}\vec{\boldsymbol{y}} = 1. \tag{7.11}$$

Here, the relation $(AB)' = B'A'$ was used which is a general rule for the transposition of matrix products.

Equation 7.11 describes an ellipse with coordinate-aligned semi-axes from Fig. 7.2b. We denote the matrix in the center of Eq. 7.11 by M and obtain:

$$M := (A^{-1})' A^{-1} = \begin{pmatrix} \frac{1}{\lambda_1^2} & 0 \\ 0 & \frac{1}{\lambda_2^2} \end{pmatrix}. \tag{7.12}$$

The according equations of the ellipse in matrix- and conventional notation are

$$\vec{\boldsymbol{y}}' M \vec{\boldsymbol{y}} = 1 \quad \Longleftrightarrow \quad \frac{y_1^2}{\lambda_1^2} + \frac{y_2^2}{\lambda_2^2} = 1. \tag{7.13}$$

From this, one reads that the lengths of the half axes of the ellipse $\vec{\boldsymbol{y}}' M \vec{\boldsymbol{y}} = 1$ are the square roots of the eigenvalues of M. This result holds for any diagonal 2×2-matrix.

We now describe the rotation by a matrix

$$D = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix} \tag{7.14}$$

and the transformation

$$\vec{\boldsymbol{y}} \mapsto \vec{\boldsymbol{z}} := D\vec{\boldsymbol{y}} = DA\vec{\boldsymbol{x}}. \tag{7.15}$$

Before carrying out the transformation of the ellipse equation $\vec{\boldsymbol{y}}' M \vec{\boldsymbol{y}} = 1$, we note that the inverse of a rotation matrix with the angle of rotation ϕ is equal to the rotation matrix with angle $-\phi$. Since $\cos(-\phi) = \cos\phi$ and $\sin(-\phi) = -\sin\phi$, we have:

$$D^{-1} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} = D'. \tag{7.16}$$

The characteristic $D^{-1} = D'$ is typical for rotations and reflections. Matrixes which fulfill this characteristic are called *unitary*.

We now insert Eq. 7.15 into the conditional equation of the ellipse and obtain:

$$(D^{-1}\vec{\boldsymbol{z}})' M (D^{-1}\vec{\boldsymbol{z}}) = 1 \tag{7.17}$$

$$\vec{\boldsymbol{z}}' D M D' \vec{\boldsymbol{z}} = 1. \tag{7.18}$$

This is the general conditional equation of an ellipse in matrix notation. The matrix DMD' is symmetrical; its eigenvectors are the directions of the major and minor semi-axes of the ellipse. The square roots of the eigenvalues are the lengths of the semi-axes.

7.4 The Two-Dimensional Normal Distribution

7.4.1 Uncorrelated data

We now return to two-dimensional random samples of the form $(x_1, y_1), (x_2, y_2), \ldots (x_n, y_n)$. In many cases, such samples will follow a two-dimensional normal distribution. It is characterized by the well known bell-shaped density function which can be thought of as the limit for large samples of a two-dimensional histogram over the x, y values in the sample. In the one-dimensional case, the density function reads:

$$p(x) = \frac{1}{\sqrt{2\pi \mathrm{var}(x)}} \exp\left\{ -\frac{1}{2} \frac{(x - \bar{x})^2}{\mathrm{var}(x)} \right\} \tag{7.19}$$

The square root of the variance is also called standard deviation, $\sigma := \sqrt{\mathrm{var}(x)}$.

If y is also normally distributed and x and y are uncorrelated, then one gets the joint distribution density (in the sample: the joint histogram) simply by the multiplication of the one-dimensional (or 'marginal') densities[1]

$$p(x,y) = \frac{1}{2\pi\sqrt{\mathrm{var}(x)\mathrm{var}(y)}} \exp\left\{-\frac{1}{2}\left(\frac{(x-\bar{x})^2}{\mathrm{var}(x)} + \frac{(y-\bar{y})^2}{\mathrm{var}(y)}\right)\right\} \tag{7.20}$$

Equation 7.20 describes a two-dimensional bell surface above the (x,y)-plane. The bell shape itself is of secondary significance for the further discussion. We rather turn to the contour lines of the bell surface for the following discussion. One obtains such contour lines by setting $p(x,y)$ to a constant p_o (the elevation of the contour). We choose $p_o := e^{-1/2}/(2\pi\sqrt{\mathrm{var}(x)\mathrm{var}(y)})$ and obtain the conditional equation of a contour line:

$$\frac{(x-\bar{x})^2}{\mathrm{var}(x)} + \frac{(y-\bar{y})^2}{\mathrm{var}(y)} = 1 \tag{7.21}$$

If $\mathrm{var}(x) = \mathrm{var}(y) =: \sigma^2$, this equation describes a circle with the radius σ about the center of gravity (i.e., mean) of the point cloud, $(\bar{x},\bar{y})$. In fact, all other contours obtained for different elevations p_o will be circles about the mean with various radii.

If the variances are unequal, e.g. $\mathrm{var}(x) > \mathrm{var}(y)$, then we get an ellipse with coordinate-aligned semi-axes, that is an ellipsis whose large semi-axis runs horizontally (in x-direction) and the small semi-axis vertically (in y-direction). If $\mathrm{var}(y) > \mathrm{var}(x)$, then the larger semi-axis is in y-direction. In both cases, the semi-axes of the ellipsis coincide with the axes of the coordinate system; oblique ellipses cannot exist with uncorrelated data. Contour lines of the two-dimensional normal distribution are known as error ellipses.

To conclude this section, we write down Eq. 7.21 in matrix notation, which will help us define the multi-dimensional standard distribution in the next sections.

$$(x-\bar{x}, y-\bar{y}) \begin{pmatrix} \frac{1}{\mathrm{var}(x)} & 0 \\ 0 & \frac{1}{\mathrm{var}(y)} \end{pmatrix} \begin{pmatrix} x-\bar{x} \\ y-\bar{y} \end{pmatrix} = 1 \tag{7.22}$$

7.4.2 Correlated Data

In the case of correlated data the ellipsis enclosing the cloud of dots will be rotated away from the coordinate axes. The equations of such ellipses do not only include terms in x^2 and y^2, but also a mixed term in xy. In the matrix notation (Eq. 7.22) one obtains such terms simply by replacing the zeros in the antidiagonal by suitable factors. As mentioned above, the covariance will play a role there.

Without demonstrating the calculation in detail we just state the result. The error ellipsis of a general two-dimensional random variable has the form:

$$(x-\bar{x}, y-\bar{y}) \begin{pmatrix} \delta\mathrm{var}(y) & -\delta\mathrm{cov}(x,y) \\ -\delta\mathrm{cov}(x,y) & \delta\mathrm{var}(x) \end{pmatrix} \begin{pmatrix} x-\bar{x} \\ y-\bar{y} \end{pmatrix} = 1. \tag{7.23}$$

[1] Note that this property, $p(x,y) = q(x)r(y)$ is actually the definition of statistical *independence* of two distributions with densities q and r. In general, statistically independent variables are always uncorrelated (i.e., have correlation zero). The reverse is true only for normally distributed variables; uncorrelated, normally distributed variables are also independent.

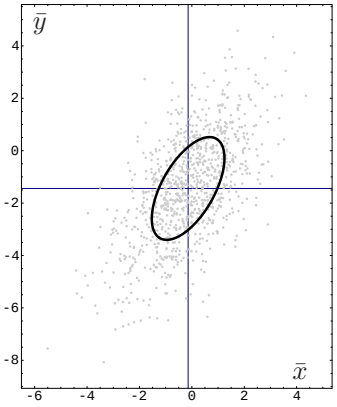


Figure 7.3: Error ellipse for the sample from Fig. 7.1. The major semi-axis of the ellipse lies in the direction of the first principle component of the data set. For normally distributed data, 39,35% of the dots fall within the ellipsis.

with

$$\delta := \frac{1}{\mathrm{var}(x)\mathrm{var}(y) - \mathrm{cov}^2(x,y)} \tag{7.24}$$

The matrix in Eq. 7.23 thus equals the matrix DMD' in the general ellipsis equation, Eq. 7.18. Equation 7.23 becomes equivalent to Eq. 7.22 if the covariance is zero.

The matrix in Eq. 7.23 is indeed the inverse of the covariance matrix C of the data set, introduced in Section 7.1:

$$C := \begin{pmatrix} \mathrm{var}(x) & \mathrm{cov}(x,y) \\ \mathrm{cov}(x,y) & \mathrm{var}(y) \end{pmatrix}. \tag{7.25}$$

Hence the general form of the two-dimensional standard distribution is:

$$p(x,y) = \frac{1}{2\pi\sqrt{\det C}} \exp\left\{-\frac{1}{2}(x-\bar{x}, y-\bar{y})C^{-1}(x-\bar{x}, y-\bar{y})'\right\} \tag{7.26}$$

where "det" denotes the determinant of a matrix,

$$\det C = \mathrm{var}(x)\mathrm{var}(y) - \mathrm{cov}^2(x,y); \tag{7.27}$$

it is a measure for total variance of the distribution.

Equation 7.26 shows the two-dimensional standard distribution in the closest possible analogy to the one-dimensional case. In comparison to the one-dimensional case the division by the variance (in the exponent) is being replaced by the multiplication with the inverse of the covariance matrix, the standard deviation in the pre-factor is being replaced by the square root of the determinant of the covariance matrix. In the one-dimensional case, the covariance-matrix is 1×1 with the only coefficient $\mathrm{var}(x)$. In this case, the two-dimensional case becomes completely equivalent to the one-dimensional equation except for the normalization factor which is $1/(2\pi)$ in the two-dimensional case and $1/\sqrt{2\pi}$ in one dimension. In the general, n-dimensional case the normalization factor is $1/\sqrt{2\pi}^n$.

7.5 Plotting Error Ellipses

The covariance matrix to a given sample can be calculated quite simply according to the equations 7.1 - 7.3 and 7.25. The error ellipse then has the equation:

$$(x-\bar{x}, y-\bar{y})C^{-1}(x-\bar{x}, y-\bar{y})' = 1. \tag{7.28}$$

Eq. 7.28 is a conditional equation, i.e. it specifies a property that any point on the arc must satisfy. Unfortunately, it does not lend itself easily for plotting. In order to plot this ellipsis, one first needs to determine the axes. They are obtained as eigenvectors of the matrix C^{-1}. We call them $\vec{e}_1$ and $\vec{e}_2$. We now show that the eigenvectors of C are also eigenvectors of C^{-1}. Let $\vec{e}$ be an eigenvector of C with eigenvalue $\lambda \neq 0$. We have:

$$C\vec{e} = \lambda\vec{e}. \tag{7.29}$$

Multiplying from the left with C^{-1} yields

$$C^{-1}C\vec{e} = \lambda C^{-1}\vec{e} \tag{7.30}$$

$$\frac{1}{\lambda}\vec{e} = C^{-1}\vec{e}. \tag{7.31}$$

I.e., $\vec{e}$ is also an eigenvector of C^{-1}, however with the eigenvalue $1/\lambda$. Since one generally orders the eigenvectors according to the size of their respective eigenvalues, the first eigenvector of a 2×2 matrix is the second eigenvector of the inverse of this matrix and vice versa.

For plotting our error ellipse we need the eigenvalues and eigenvectors of C^{-1} which, however, can just as well be obtained from C, i.e. without the inversion. It is sufficient to calculate the eigenvectors and eigenvalues of C; they are called $\lambda_1, \vec{\boldsymbol{e_1}}$ and $\lambda_2, \vec{\boldsymbol{e_2}}$.

We now look for a parametric description of the ellipsis which we can enter in a plot-routine. For this, we think of a rotating arrow (much as the hand of a clock) circulating about the center of the cloud of dots. The length of this clock-hand shall be specified for every angle ϕ in a way that when completing the turn, the tip of the clock-hand describes an elliptic trajectory. We obtain the length in the direction of the first eigenvector from the following argument:

Let r be the sought length, the point on the ellipsis is therefore $r\vec{e}_1$. From the conditional equation we read:

$$1 = (r\vec{e}_1)'C^{-1}(r\vec{e}_1) \tag{7.32}$$

$$= \frac{r^2}{\lambda_1}\|\vec{e}_1\|^2 \tag{7.33}$$

and further $r = \sqrt{\lambda_1}$. Analogously one determines the length in the direction of the second eigenvector. Finally, we combine these values in the parametric equation of the error ellipse,

$$r(\phi) = (\bar{x}, \bar{y})' + \sqrt{\lambda_1}\cos\phi\ \vec{e}_1 + \sqrt{\lambda_2}\sin\phi\ \vec{e}_2. \tag{7.34}$$

We draw this curve for $\phi \in (0, 2\pi)$ on top of the scatter plot and obtain with it the desired error ellipsis.

For a two-dimensional standard distribution the error ellipsis contains 39,35 % of the measured values. This value is calculated from the integration of the distribution density over the area enclosed by the error ellipse. We can instead consider the standard normal distribution where the covariance matrix is the unit matrix and integrate over a circular disc with the radius 1:

$$\int_{-\pi}^{\pi}\int_0^1 \frac{1}{2\pi}\exp\{-\frac{r^2}{2}\}r dr d\phi = \tag{7.35}$$

$$= \int_0^1 r\exp\{-\frac{r^2}{2}\}dr = \left[-\exp\{-\frac{r^2}{2}\}\right]_0^1 = 1 - \frac{1}{\sqrt{e}} = 0.3935. \tag{7.36}$$

We call this ellipse the standard error ellipse. An ellipse with doubled semi-axes (two times standard deviation) contains 86,47 % of the measured values as can be shown analogously.

Lastly, we look at the area of the error ellipse from Equation 7.34. If a and b are the half axes of an ellipsis, then their surface is $A = \pi ab$. With the the matrix notation used in Eq. 7.28 we have $A = \pi\sqrt{\det C}$ (cf. Eq. 7.27). If λ_1 and λ_2 are calculated explicitly, the equivalence of the two definitions becomes apparent. The area of the standard error ellipse equals π times the square root of the determinant of the covariance matrix. The determinant can therefore be considered a measure of the total, two-dimensional spread of the cloud of data points or the total variance of the vector (x, y).

7.6 Reference

Rencher, A. C. (2002). *Methods of Multivariate Analysis. 2. Edition.* Wiley Interscience (John Wiley and Sons). Kapitel 4.

Chapter 8

Principal Component Analysis

Principal Component Analysis (PCA) is a method for analyzing multi-variate data in which a number of factors (causes, sources, etc.) superimpose. In general, one has more dependent variables than factors and wants to filter out the underlying factors. We start with a few examples. For a comprehensive description of the mathematical background see Mardia et al. (1979) and Rencher (2002).

8.1 Examples

8.1.1 Average Grades

Consider a sample of n students of whom grades are available in m topics of study. Per student, we thus have a m-dimensional data vector $\vec{\boldsymbol{x}}_i = (x_{i1}, x_{i2}, ..., x_{im})'$. Now, an overall score per student is to be determined. In the simplest case, this could be the mean of the grades, i.e.,

$$y_i = \frac{1}{m}\sum_{j=1}^{m} x_{ij} = \left(\frac{1}{m}, \ldots, \frac{1}{m}\right) \begin{pmatrix} x_{i1} \\ \vdots \\ x_{im} \end{pmatrix}. \tag{8.1}$$

If a student has a special linguistic aptitude, say, the average grade would probably be better with the addition of another foreign language to the curriculum. To compensate for this effect, one could switch to weighted means where all languages taken together would count just as much as the sum of the scientific subjects. Generally, such a weighted mean with coefficients (weights) $w_1, ..., w_m$ can be given as

$$y_i = \sum_{j=1}^{m} w_j x_{ij} = (w_1, \ldots, w_m) \begin{pmatrix} x_{i1} \\ \vdots \\ x_{im} \end{pmatrix}. \tag{8.2}$$

In the case of average grades, the weights are non-negative and add to 1. In the applications considered below, however, we will allow negative weights and require that the sum of squared weights be 1. For a weight vector $\vec{\boldsymbol{v}}$ not satisfying this condition, we can always obtain a normalized one by dividing every weight by $\|\vec{\boldsymbol{v}}\| = \sqrt{\sum_j v_j^2}$, $\vec{\boldsymbol{w}} = \vec{\boldsymbol{v}}/\|\vec{\boldsymbol{v}}\|$.

Besides the obvious weight set used in Eq. 8.1, many others are possible. For the selection, two criteria can be used:

1. Which weight set $\vec{\boldsymbol{w}}$ describes the variation in the data set best? This will be important for judging the overall performance of a student as "good" or "poor" from just one averaged grade.

2. Are there further distinctions that can be made besides the overall performance? A possible example might be "linguistically apt" vs. "scientifically apt"? How many such factors (or axes) are needed?

8.1.2 Genetic Variation

In n populations of one species, allele frequencies of m alleles are determined. One would like to know then if the observed variation is 'homogenous', or if taxonomically relevant clusters (subspecies etc.) showing "co-variations" of particular allele patterns can be demonstrated.

For every population, one gets a m-dimensional vector of allele frequencies which can be thought of as point in a m-dimensional feature space[1] If two subspecies can now be distinguished, the feature vectors of the according populations should be in two separate clusters in this feature space. The direction along which the clusters are separated is again a vector in feature space, let it be called $\vec{\boldsymbol{w}}$, i.e. it has one component for every allele examined. Since only the direction is of interest here, we define $\vec{\boldsymbol{w}}$ as unit vector, $\sum w_j^2 = \vec{\boldsymbol{w}}'\vec{\boldsymbol{w}} = 1$, as above. If the vectors of allele frequency are projected onto the line through the origin defined by $\vec{\boldsymbol{w}}$, one gets two accumulation points that might correspond to two subspecies. Mathematically, this projection is realized by the dot product (see. Chapter 1):

$$y_i := \vec{\boldsymbol{w}}'\vec{\boldsymbol{x}}_i = \sum_{j=1}^{m} w_j x_{ij} \tag{8.3}$$

The obvious (though not always successful) strategy for choosing $\vec{\boldsymbol{w}}$ is to maximize the spread of the projections[2], i.e. the variance of y_i. The problem, then, is to find the direction of projection maximizing this variance.

Cavalli-Sforza et al. (1994) show with this method that the genetic variation within the species *Homo sapiens* is of no taxonomic value.

8.1.3 Spike-Sorting

In extracellular electrophysiology, an electrode is introduced into the tissue and electrical activities are recorded with result mostly from action potentials of the neurons in the vicinity of the electrode. Recordings from one electrode may result from multiple cells whose signals will differ due to the distance of each neuron from the electrode tip and the relative position of the electrode with respect to each neurons dendrite, soma, and axon. Assume we have a recording with n spike events of different shape and amplitude. One would then like to know from how many different neurons the potentials originate and classify the potentials as originating from one cell or another.

We assume that no temporal overlap of two or more action potentials occur. The first processing step then is to define the exact starting points of the action potential and sample the subsequent

[1] As mentioned before, one cannot imagine a m-dimensional space for $n > 3$. The distinction of m dimensions means here that the data vector can at least in principle vary in m independent ways, in particular for every observed allele.

[2] Indeed, clusters my be sharply separated in one dimension but widespread in another one, to the point that the largest variation is not the one allowing cluster separation. In this case, PCA may fail. An alternative approach used for this problem is *independent component analysis*, ICA.

interval of about 2 milliseconds into m steps. Each of the n measured potentials then yields a m-dimensional data vector $(x_{i1}, \dots x_{ij}) = \vec{\boldsymbol{x}}_i$. The sought variables are the the number of neurons recorded from and an estimate of the time course of each neuron's potential, again sampled in m steps. In PCA-based spike sorting, one first defines optimally discriminating features for the given set of spike events. Each feature is defined by a m-dimensional vector $\vec{\boldsymbol{w}}_l$; the dot product of this vector and the data vector $\vec{\boldsymbol{x}}_i$ is the feature value.

$$y_{lj} := \vec{\boldsymbol{w}}_l' \vec{\boldsymbol{x}}_j = \sum_{j=1}^{m} w_{lj} x_{ij} \tag{8.4}$$

As in the previous examples one is again interested in projections through which the y come to lie apart as far as possible (cf. Lewicki 1998).

8.1.4 Allometry

Here we repeat a simplified version of an example given by Mardia (1979; page 241f). In a sample of 54 apple trees two variables are measured:

x^*: trunk diameter
y^*: total height of the tree

Since the two variables and their variances differ by orders of magnitude, it makes sense to normalize them by their standard deviations. We set up the zero-mean variables

$$x := \frac{x^* - \bar{x}}{\sigma_{x^*}} \quad \text{and} \quad y := \frac{y^* - \bar{y}}{\sigma_{y^*}}. \tag{8.5}$$

Since we have already divided by the standard deviations, the covariances equal the correlations. We calculate the correlation matrix

$$C_R := \begin{pmatrix} \mathrm{var}(x) & \mathrm{cor}(x,y) \\ \mathrm{cor}(x,y) & \mathrm{var}(y) \end{pmatrix} = \begin{pmatrix} 1,0 & 0,5 \\ 0,5 & 1,0 \end{pmatrix}, \tag{8.6}$$

where the variances take the value 1 by definition and the correlation of 0.5 is assumed to be the experimental result.

Using this example, we can for once calculate the eigensystem of the matrix C_R with elementary methods. Of course, MATLABprovides a function for this.

Let λ be an eigenvalue and $\vec{\boldsymbol{e}}$ an eigenvector, then

$$C_R \vec{\boldsymbol{e}} - \lambda \vec{\boldsymbol{e}} = 0 \quad \text{or} \quad \begin{pmatrix} 1,0-\lambda & 0,5 \\ 0,5 & 1,0-\lambda \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = 0.$$

From the right side version of this equation, it follows that the determinant of the matrix has to disappear, $(1-\lambda)^2 - 1/4 = 0$. One gets the results $\lambda_{1,2} = 1 \pm 0,5$. For determining the eigenvectors one inserts one of the two results for λ and solves the emerging equations system with two unknowns. The end result is:

$$\lambda_1 = \frac{3}{2}; \qquad \vec{\boldsymbol{e}}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{8.7}$$

$$\lambda_2 = \frac{1}{2}; \qquad \vec{\boldsymbol{e}}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \tag{8.8}$$

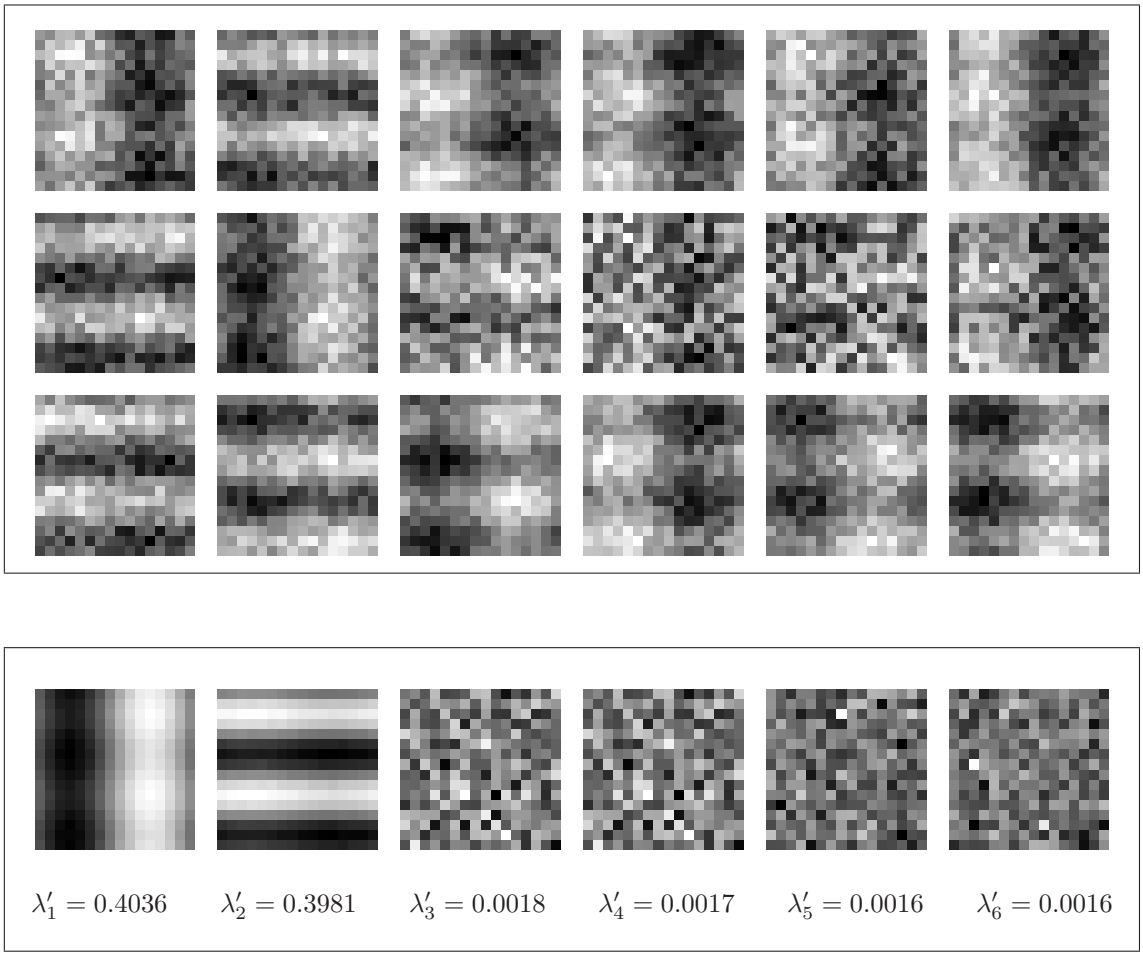


Figure 8.1: Above: Examples from a sample of images with $16 \times 16 = 256$ pixels respectively. Below: The first 6 principal components of the data set. $\lambda'_i := \lambda_i / \sum_{i=1}^{256} \lambda_i$.

The interpretation of this result is that the trees can be described by two independent quantities, in particular an overall 'size' $x + y$ reflecting both trunk diameter and growth height, containing $\lambda_1/(\lambda_1 + \lambda_2) = 75$ % of the total variance, and a 'form factor' $x - y$ reflecting 25 % of the total variance. The new variables are uncorrelated, i.e. the descriptions size (small / large) and form (lean / heady) can now be defined independent of each other. This computation (coordinate-transform towards the eigensystem of the correlation- or covariance- matrix) is also called de-correlation.

8.1.5 Images as Multivariate Data

Black and white images are arrays of pixels whose intensity values can be seen as components of a high-dimensional vector. We will discuss image processing more thoroughly in a later chapter, but consider a a simple example here.

A frequently occuring problem in image processing is to treat a sample of images as a superposition of prototype images, each with a random weight, and some independent noise process. If proper prototypes are found, image storage and transmission can be reduced to transmitting and

storing the weight factors which will result in a substantial reduction of the required storage or transmission capacities. An example is given in Fig. 8.1. Here, two images are essentially sufficient to explain the variation in the set of data; the remaining variation is white noise and can be suppressed easily if the principal components are known. With other tasks like e.g., classification this noise suppression is advantageous as well.

The 'prototypes' always indicate the difference from the average image of the sample calculated by averaging each pixel value over all images in the sample. Accordingly, the prototypes can take on negative values since a given image can deviate from the mean both upward and downward. In Fig. 8.1 the mean value image is not shown. The 'prototypes' in the bottom row are displayed so that a medium gray corresponds to the value 0.

With image data one has to note that the images, just as in the other examples, should be regarded as vectors, not matrices. In MATLAB one therefore transforms them into a simple list by hanging the single image lines behind one another. Afterwards, one needs an operation that turns those vectors into images again in order to be able to present the results.

With this in mind, we consider images as data vectors with dimension m; in the example of Fig. 8.1 is $m = 16 \times 16 = 256$. We denote the sought prototypes with $\vec{\boldsymbol{e}}_l$, $l = 1, ..., k$ and require these images to be *orthogonal*, i.e.,

$$\vec{\boldsymbol{e}}_i' \vec{\boldsymbol{e}}_j = 0 \quad \text{for all} \;\; i \neq j. \tag{8.9}$$

The reconstruction of the sample images from the prototype images is achieved by a weighted superposition of the prototypes,

$$\vec{\boldsymbol{x}}_i \approx \vec{\boldsymbol{\mu}} + \sum_{l=1}^{k} a_{il} \vec{\boldsymbol{e}}_l \tag{8.10}$$

or more precisely

$$\sum_{i=1}^{n} \left\| \vec{\boldsymbol{x}}_i - (\vec{\boldsymbol{\mu}} + \sum_{l=1}^{k} a_{il} \vec{\boldsymbol{e}}_l) \right\| \rightarrow \text{minimal} \tag{8.11}$$

for suitable coefficients a_{il}. With $\vec{\boldsymbol{\mu}} := (\sum_{i=1}^{n} \vec{\boldsymbol{x}}_i)/n$ we denote the average image. The corresponding prototypes and coefficients can be found with the help of the principal axis transformation.

In image processing one applies the principal component transformation to sets of smaller images obtained by decomposing the original images into sets of image patches. The principal components are then image features by which images are analyzed (or so-called Karhunen-Loéve-Transformation). In an example from the perception, the principal component transformation is applied to entire images, especially images of faces in order to define 'feature dimensions' like e.g., male/female; young/old; European/Asian, etc. (Valentin et al. 1997, Calder & Young 2005). Decomposition into and reconstruction from a set of "basis images" also underlies the well-known JPEG encoding. In this case the base images (prototypes) are stripe pattern (sinusoidals) of various width, as also occur in Fourier analysis.

8.1.6 Summary

From the above examples three basic applications of the principal component analysis can be seen. These are not mutually exclusive but defer to slightly different issues.

- Identification of 'factors' or relevant variables. This intention is underlying the examples 'grading' and 'allometry'. The factors are the principal components themselves.
- Classification. In the examples from genetics and spike-classification the values ("scores") associated with the factors were used for the classification.

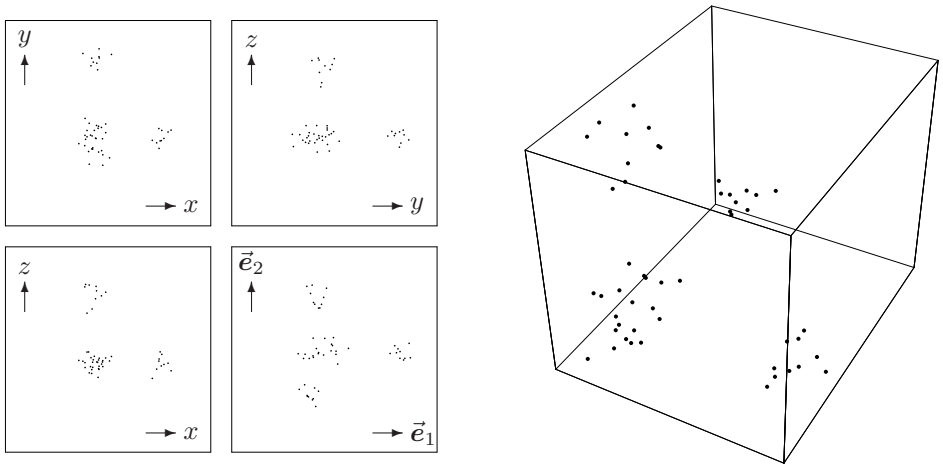


Figure 8.2: Projection of three-dimensional data onto two-dimensional planes. Left: The projections on the coordinate planes ((x, y), (x, z) and (y, z)) show only three point clusters respectively. The projection onto an optimal separating plane (spanned by $\vec{e}_1$ and $\vec{e}_2$), however, shows four point clusters. Right: Actual arrangement of the point clusters in the three- dimensional space. The optimal separating plane lies slanted in this space.

- Data reduction. In many cases one can assume that the higher principal components reflect noise, rather than the sought signal (image processing, spike-sorting). In this case, the projection onto the subspace of the first k principal components produces an optimal dimension reduction of the data, meaning one with minimal loss of information.

8.2 Feature Spaces, Projections and Reconstructions

The above examples have in common that there are high-dimensional data vectors which cannot be easily visualized in a diagram. A crucial concept, therefore, is that of a projection to some axis, which leads to simple, one-dimensional data. The central idea of PCA is to find the optimal axes for such projections and use them for data analysis.

Let $\vec{\boldsymbol{x}}_i = (x_{i1}, x_{i2}, ..., x_{im})'$ be a vector from the m-dimensional data set and $\vec{\boldsymbol{w}} = (w_1, w_2, ..., w_m)'$ a random vector in this space of which we want to assume again that it has the length (norm) 1:

$$\|\vec{\boldsymbol{w}}\| = \sum_{j=1}^{m} w_j^2 = 1. \tag{8.12}$$

The projection of $\vec{\boldsymbol{x}}_i$ to $\vec{\boldsymbol{w}}$ is the point on the straight line running through the origin in direction $\vec{\boldsymbol{w}}$, which is closest to the point $\vec{\boldsymbol{x}}_i$. If one drops a perpendicular from $\vec{\boldsymbol{x}}_i$ onto the straight line, then it meets at exactly that point.

Note that projections projection in many dimensions work exactly as in the two-dimensional case. This is because the point $\vec{\boldsymbol{x}}_i$ and the line define a plane which can be sketched out on a sheet of paper. With this, the problem is reduced to two dimensions.

The projection of $\vec{\boldsymbol{x}}_i$ onto $\vec{\boldsymbol{w}}$ is obtained with the help of the dot product, cf. Eq. 1.7:

$$P : \vec{\boldsymbol{x}}_{\boldsymbol{i}} \mapsto a_i \vec{\boldsymbol{w}} = (\vec{\boldsymbol{x}}_i' \vec{\boldsymbol{w}}) \vec{\boldsymbol{w}} = \left(\sum_{j=1}^{m} x_{ij} w_j \right) \vec{\boldsymbol{w}}. \tag{8.13}$$

Here, P denotes the projection operation. If e.g., $\vec{\boldsymbol{w}}$ is a coordinate vector, say $\vec{\boldsymbol{w}} = (1, 0, 0, ...0)'$, then $P(\vec{\boldsymbol{x}}_i) = x_{1i}$. For general projection axis, we can therfore think of the projection as some sort of coordinate of the point with respect to the general axis $\vec{\boldsymbol{w}}$. We call this coordinate a_i.

If all n data vectors are projected onto the same axis $\vec{\boldsymbol{w}}$, a set of n numbers $(a_1, ..., a_n)$ will result. These projection will reproduce the more information about the original sample, the larger their spread, or variance is. Maximization of the variance of projection is therefore a general aim in PCA.

Projections are closely related to coordinate transformation. To see this, we consider an example in the two-dimensional space. We denote the coordinate axes with the letters

$$\vec{\boldsymbol{u}}_1 := \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \vec{\boldsymbol{u}}_2 := \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{8.14}$$

Obviously, we have $\vec{\boldsymbol{u}}_1'\vec{\boldsymbol{u}}_1 = 1$, $\vec{\boldsymbol{u}}_2'\vec{\boldsymbol{u}}_2 = 1$ and $\vec{\boldsymbol{u}}_1'\vec{\boldsymbol{u}}_2 = 0$. The basis system $\vec{\boldsymbol{u}}_1$, $\vec{\boldsymbol{u_2}}$ is therefore said to be *orthonormal*. The components of a vector $\vec{\boldsymbol{x}} = (x_1, x_2)'$ in this coordinate frame can be obtained by projection of the vector onto the coordinate axes by means of the dot product:

$$x_1 := \vec{\boldsymbol{x}}'\vec{\boldsymbol{u}}_1 \quad \text{and} \quad x_2 := \vec{\boldsymbol{x}}'\vec{\boldsymbol{u}}_2. \tag{8.15}$$

Conversely, one can reconstruct $\vec{\boldsymbol{x}}$ as sum of the basis vectors weighted with the coordinates:

$$\vec{\boldsymbol{x}} = x_1\vec{\boldsymbol{u}}_1 + x_2\vec{\boldsymbol{u}}_2. \tag{8.16}$$

Both characteristics of a basis hold also when skipping to random, orthonormal basis vectors $\vec{\boldsymbol{e}}_1$, $\vec{\boldsymbol{e}}_2$. In this case the projection values

$$y_1 = \vec{\boldsymbol{x}}'\vec{\boldsymbol{e}}_1 \quad \text{and} \quad y_2 := \vec{\boldsymbol{x}}'\vec{\boldsymbol{e}}_2 \tag{8.17}$$

can again be used to 'reconstruct' the vector $\vec{\boldsymbol{x}}$:

$$\vec{\boldsymbol{x}} = y_1\vec{\boldsymbol{e}}_1 + y_2\vec{\boldsymbol{e}}_2. \tag{8.18}$$

This relation between projection and reconstruction is a consequence of the ortho-normality of the basis and is true for any number of dimensions. In the examples, use was made mostly of the projection property. One application of a the reconstruction property was given in the case of image processing.

8.3 Performing PCA

We have discussed the essential mathematics already in the previous chapter and do not need to repeat this here. The principal component analysis is a coordinate transformation in the data space which brings the coordinate axes are brought into alignment with the principal axes of the error ellipsoid. The numbering of the axes is such that the axis with the highest data variance has the number 1, the second highest the number 2 and so on.

Here we summarize the steps needed for carrying out the principal component analysis.

8.3.1 Data Matrix

We assume a number n of m-dimensional data vectors

$$\vec{\boldsymbol{x}}_1 \quad = \quad (x_{11}, x_{12}, \ldots x_{1m})$$

$$\begin{aligned} \vec{\boldsymbol{x}}_2 &= (x_{21}, x_{22}, \ldots x_{2m}) \\ &\vdots \\ \vec{\boldsymbol{x}}_n &= (x_{n1}, x_{n2}, \ldots x_{nm}) \end{aligned}$$

Next, we calculate the average data vector

$$\vec{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} \vec{\boldsymbol{x}}_i = \frac{1}{n}\left(\sum_{i=1}^{n} x_{i1},\ \sum_{i=1}^{n} x_{i2},\ \ldots,\ \sum_{i=1}^{n} x_{im}\right) \tag{8.19}$$

We now subtract the mean values from the x_{ij} and obtain so-called zero-mean data. We continue using the letter x but assume that $\sum_i x_{ij}$ now disappears for all j. In the example with the grades this means that instead of a student's actual grades only his or her individual deviation from the mean over all students is considered.

The zero-mean data are arranged in a data matrix

$$D = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1m} \\ x_{21} & x_{22} & \ldots & x_{2m} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{nm} \end{pmatrix} \tag{8.20}$$

8.3.2 Covariance Matrix

The covariance matrix of the data set is obtained from

$$C = \frac{1}{n} D'D \tag{8.21}$$

(cf. section 2.3.4).

At this point one should check that C is indeed a $m \times m$-matrix and not perhaps a $n \times n$-matrix. This error occurs if D and D' have been confused.

8.3.3 Eigensystem

Next, one determines the eigensystem of c. Recall the definition of an eigenvector $\vec{\boldsymbol{e}}$ of a $m \times m$-matrix M as a m-dimensional vector satisfying:

$$M\vec{\boldsymbol{e}} = \lambda\vec{\boldsymbol{e}} \quad \text{where} \quad \lambda \in \mathbb{R}. \tag{8.22}$$

The number λ is the eigenvalue belonging to $\vec{\boldsymbol{e}}$. Eigenvectors are always given as unit vectors and MATLAB also applies this rule. A symmetrical $m \times m$-matrix like the covariance matrix can have up to m different eigenvalues; the corresponding eigenvectors are pairwise orthogonal to each other. If two eigenvalues λ_i, λ_j are equal, then every vector $a\vec{\boldsymbol{e}}_i + b\vec{\boldsymbol{e}}_j$ is an eigenvector; the resulting plane is sometimes called a (two-dimensional) eigenspace.

MATLAB will return the eigensystem as a matrix of eigenvectors and a vector of eigenvalues,

$$E = [\vec{\boldsymbol{e}}_1, \ldots, \vec{\boldsymbol{e}}_m] \quad \text{and} \quad \Lambda = (\lambda_1, \ldots, \lambda_m)' \tag{8.23}$$

8.3.4 Principal Components

The eigenvectors $\vec{\boldsymbol{e}}_j, j = 1, ..., m$ are the sought principal components. Their vector components e_{ji} are sometimes called "loadings".

The principal components are m-dimensional unit vectors. For normally distributed data the principal axes are the axes of the error ellipsoid. They are ortho-normal, i.e., $\vec{\boldsymbol{e}}_i'\vec{\boldsymbol{e}}_j = 0$ for all $i \neq j$ and $\vec{\boldsymbol{e}}_i'\vec{\boldsymbol{e}}_i = 1$ for all i. Since there are as many principal axes as dimensions in the data space (namely m), the principal axes define an ortho-normal coordinate transform, i.e. a combined rotation and/or mirroring in the feature space. In the new coordinate system, the data dimensions will be uncorrelated..

The principal components have the same number of dimensions as the data vectors and can therefore be treated as prototypical data vectors. Principal components of the image data set from Fig. 8.1 are therefore again images, principal axes of a data set of spike courses are temporal signals and so forth. This is especially interesting for the constructive interpretation of the principal axes where data vectors are approximated by superposition of the principal axes.

8.3.5 Scores

Projection of the data-vector $\vec{\boldsymbol{x}}_i$ onto the principal axis $\vec{\boldsymbol{e}}_j$ are called "scores". They are defined by

$$a_{ij} = (\vec{\boldsymbol{x}}_i \cdot \vec{\boldsymbol{e}}_j) \tag{8.24}$$

and are numbers which can be thought of as coordinates of the data vector $\vec{\boldsymbol{e}}_i$ in the new coordinate system generated by the principal components.

For pairs of principal components $\vec{\boldsymbol{e}}_k, \vec{\boldsymbol{e}}_l$, data can be represented in a "score-score-plot" containing a dot at (a_{ik}, a_{il}) for each data vector $\vec{\boldsymbol{x}}_i$ (cf. Fig. 8.2) Score-score plots can be used for data clustering.

8.3.6 Eigenvalues

The eigenvalues λ_i of the covariance matrix equal the variances of the corresponding scores:

$$\text{var } a_j = \lambda_j \tag{8.25}$$

The sum of the eigenvalues is the total variance of the data set. It can also be calculated as the "trace", i.e. the sum of the diagonal elements, of the covariance matrix,

$$V_{tot} = \text{tr}C = \sum_{j=1}^{m} \lambda_j. \tag{8.26}$$

Analysis f the eigenvalues allows to decide how many of the principle components should be used in a given analysis.

One way to do this is to keep just as many dimensions that the sum of the eigenvalues amounts to some prescribed percentage of the total variance. A somewhat more systematic approach uses a bar diagram of the λ-values in a decreasing series and looks for a kink in the enveloping line, i.e., a λ_k such that all λ_i with $i > k$ roughly equal λ_k. This means that the residues $\vec{\boldsymbol{x_i}} - \sum_{j=1}^{k} a_{ij}\vec{\boldsymbol{e}}_j$ compose an approximately spherical point cloud.

8.3.7 Reconstruction and Noise Reduction

Often, the first principle components will reflect the actual signal measured in a experiment while the higher components can be considered noise. This can be used to reduce noise by omitting these

higher components. A pattern $\vec{\boldsymbol{x}}_i$ would thus be replaced by the approximation

$$\vec{\boldsymbol{\xi}}_i = \sum_{j=1}^{k} a_{ij} \vec{\boldsymbol{e}}_j \tag{8.27}$$

for some $k < m$. (Clearly, if $k = m$, we have $\vec{\boldsymbol{\xi}}_i = \vec{\boldsymbol{x}}_i$.) If $k = 2$, this can be interpreted in the following way: The first two principal components span a plane in the feature space approximating the location of the data points as closely as possible. The vector $\vec{\boldsymbol{\xi}}_i$ is the point in this plane closest to the actual data vector $\vec{\boldsymbol{x}}_i$. If we sketch the plane separately, we obtain a score-score-plot for the first two principal components.

8.4 Literature

Calder A.J., Young, A. W. (2005) Understanding the recognition of facial identity and facial expression. *Nature Reviews Neuroscience*, 6:641-651

Cavalli-Sforza, L. L., Menozzi, P., and Piazza, A. (1994). *The history and geography of human genes*. Princeton University Press, Princeton, NJ.

Lewicki, M. S. (1998). A review of methods for spike sorting: the detection and classification of neural action potentials. *Network: Computation in Neural Systems*, 9:R53 – R78.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate Analysis*. Academic Press, London.

Rencher, A. C. (2002). *Methods of Multivariate Analysis*. John Wiley, New York. 2. Aufl. 2002. Kapitel 12.

Valentin, D., Abdi, H., Edelman, B., and O'Toole, A. J. (1997). Principal component and neural network analysis of face images: What can be generalized in gender classification? *Journal of Mathematical Psychology*, 41:398 – 413.

Chapter 9

Fourier-Analysis

Fourier analysis is a powerful tool of mathematical analysis that arises ion many different contexts, some of which will be addressed in later chapters on linear systems and time series analysis. Here we summarize the essential facts about Fourier transforms without going into the mathematical details. Following the general scope of this course, the emphasis is on getting some intuitions for possible applications, rather than on the deep mathematical issues occuring in this field.

9.1 Periodic Functions

9.1.1 Time-Dependent Potentials

Periodic functions occur in many contexts and are crucial in the analysis of time-dependent variables. As examples, we consider time-dependent potentials as are generated in a bicycle dynamo ("magneto") or alternating current generators. The period is given by a rotation of the generator in this case. Periodic functions satisfy

$$U(t+T) = U(t) \quad \text{for all } t. \tag{9.1}$$

Here, T is called period or wave-length. In the case of a bicycle dynamo, it is the time needed for one revolution of the rotor. Clearly, we have $U(t+nT) = U(t)$ for any $n \in \mathbb{N}$.

Sinusoidal Potentials

The most important function for describing time-dependent voltages and currents is the sine function,

$$U(t) = U_o \sin(\omega t + \varphi). \tag{9.2}$$

Here, U_o is the amplitude, φ the phase and ω the so-called angular frequency[1]. Since the period of the sine is 'naturally' 2π or $360°$, the voltage oscillates exactly once per time unit if $\omega = 2\pi$. The unit of the frequency is the Hertz; $1\text{Hz} = 1\text{sec}^{-1}$ is one oscillation per second. Frequency can be

[1] In German: Kreisfrequenz

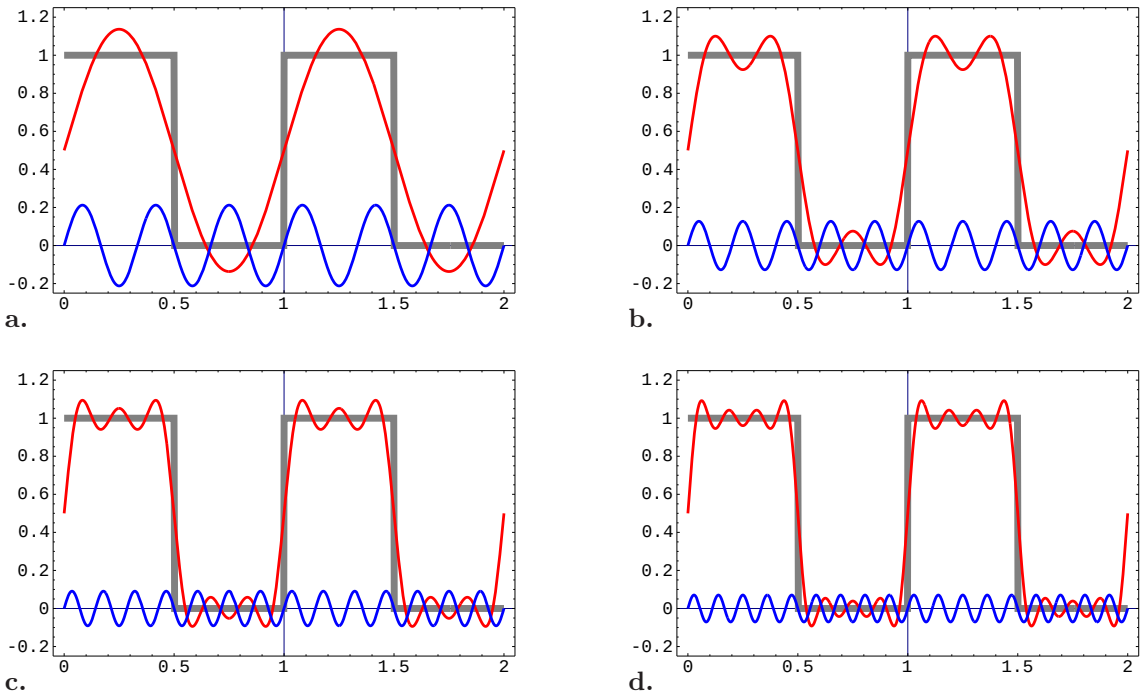


Figure 9.1: Approximation of the 'box function' (Equation 9.6 with $\omega = 1$) by a Fourier-sine-series (Equation 9.8). For explanations see text.

expressed in various ways whose relation can be summarized as follows frequency:

$$\begin{array}{lllll} f & = & \frac{1}{T} & = & \frac{\omega}{2\pi} \quad \text{frequency (Hz} = 1/\text{s)} \\ T & = & \frac{1}{f} & = & \frac{2\pi}{\omega} \quad \text{cycle duration, period (s)} \\ \omega & = & \frac{2\pi}{T} & = & 2\pi f \quad \text{angular frequency (rad/s)} \end{array} \tag{9.3}$$

Square-Wave Function

The square-wave potential in the interval $[0, T]$ is defined by the so-called boxcar function

$$U(t) = \begin{cases} U_o & \text{for } 0 \le t < \tau \\ 0 & \text{for } \tau \le t < T \end{cases} . \tag{9.4}$$

Outside of the interval the function continues periodically. The relation of the time during which the potential is switched on (τ) and total time T, is called 'duty cycle'. The square-wave voltage has a direct current (DC)[2] component of $U_o\tau/T$ which can be eliminated through subtraction. The result is called a bimodal square-wave voltage.

[2]In German: Gleichstrom

General periodic Functions

General periodic functions can be are expressed as sums of sine- and cosine-waves. The fact that this is generally possible is the central result of the theory of the Fourier-series which we will discuss in more detail later in this chapter. Such sums are also called 'trigonometric polynomials':

$$p_n(x) := \frac{a_o}{2} + \sum_{k=1}^{n} a_k \cos k\omega x + \sum_{k=1}^{n} b_k \sin k\omega x. \tag{9.5}$$

Here, $\omega/2\pi$ is the 'fundamental frequency'; all other frequencies of the form $k\omega/2\pi$ are multiples of this fundamental frequency and, especially in acoustical applications, are called 'harmonics'. In acoustics, the pitch of a note is given by the fundamental frequency while the distribution of the overtones determines the timbre. If playing the standard pitch a^1 on the piano, on the flute, or singing it with the vowel A or O, then there is always the same fundamental frequency of 440 Hz, but with a different composition of the overtones. This composition is given by the coefficients a_k and b_k, or their combined power $\sqrt{a_k^2 + b_k^2}$; together, they form the "sound spectrum".

The term 'spectrum' was originally introduced in optics where the individual spectral components of a light beam, that is electromagnetic waves of certain frequencies, can be separated by means of a prism. Mixed (i.e., non-spectral) lights may contain any frequency, therefore they are not limited to integer multiples of a fundamental frequency. In Fourier analysis, the term spectrum is generally used for the frequency-dependent set of the coefficients of the sine- and cosine-components.

9.2 Fourier-Series

Trigonometric polynomials as defined in Eq. 9.5 are a powerful tool in many fields of scientific computing. This is due to the fact that periodic functions can generally be approximated by a trigonometric polynomial to arbitrary accuracy if we allow the number of coefficients n to become sufficiently large. In this section, we briefly discuss the theory for periodic functions (Fourier series). Extensions to non-periodic functions and more dimensions will be briefly discussed below. For reference see Tolstov (1962), Bracewell (1986), Butz (2010).

9.2.1 Examples

Consider once more the boxcar function with duty cycle 1/2,

$$b(x) := \begin{cases} 1 & \text{if } \operatorname{mod}(x, T) < T/2 \\ 0 & \text{else} \end{cases} \tag{9.6}$$

(see Figure 9.1). It can be approximated by a series of sine waves given by the equation

$$g_n(x) = \frac{1}{2} + \frac{2}{\pi}\sin \nu x + \frac{2}{3\pi}\sin 3\nu x + \ldots + \frac{2}{(2n-1)\pi}\sin(2n-1)\nu x \tag{9.7}$$

$$= \frac{1}{2} + \frac{2}{\pi}\sum_{k=1}^{n} \frac{\sin(2k-1)\nu x}{2k-1} \tag{9.8}$$

Here, we have used the basic angular frequency

$$\nu := \frac{2\pi}{T}. \tag{9.9}$$

Clearly, g_n is a "trigonometric polynomials" as in Eq. 9.5; it is also the Fourier series expansion of the boxcar function, Equation 9.6. Fig. 9.1 shows the boxcar function (gray line) together with the approximation $g_i(x)$ for $i = 1$ through 4. In the lower part of each panel, the next term of the sum is shown as a blue line which is then accumulated to the current approximation in the following panel. For each x satisfying $\mathrm{mod}(x, T) \neq 0.5$ and $\mathrm{mod}(x, T) \neq 1$, i.e. for each x where $b(x)$ is continuous, the series converges towards the true functional value:

$$\lim_{n\to\infty} g_n(x) = b(x). \tag{9.10}$$

Figure 9.2 shows a slightly more general case where the sinusoids needed to reconstruct the signal have different phases. The periodic function is now a random noise function on the interval $[0, 1]$, which is periodically repeated along the real axis. The relevance of the phases of the sinusoidal components can be seen by checking the value at $x = 1$ of the correction term (lower sinusoid in each panel); while this is zero for all frequencies in Fig. 9.1, the value now changes from panel to panel.

9.2.2 Finding the coefficients

So far, we have seen that trigonometric polynomials can approximate continuous functions. As already mentioned above, we may write the general form of such polynomials as

$$p_n(x) := \frac{a_o}{2} + \sum_{k=1}^{n} a_k \cos k\nu x + \sum_{k=1}^{n} b_k \sin k\nu x. \tag{9.11}$$

Here, $\nu/2\pi$ is again the fundamental frequency of the signal, i.e. p_n repeats itself with a wave-length of $T = 2\pi/\nu$.

How can we find the coefficients a_k, b_k? Under the assumption, that a suitable set of coefficients exists, we can find the coefficients by exploiting the so-called orthogonality relations of sinusoids which hold for all $k, l > 0$:

$$\int_0^{2\pi} \sin kx \sin lx\, dx = \begin{cases} \pi & \text{if } k = l \\ 0 & \text{if } k \neq l \end{cases} \tag{9.12}$$

$$\int_0^{2\pi} \cos kx \cos lx\, dx = \begin{cases} \pi & \text{if } k = l \\ 0 & \text{if } k \neq l \end{cases} \tag{9.13}$$

$$\int_0^{2\pi} \sin kx \cos lx\, dx = 0 \tag{9.14}$$

With these relations, we obtain:

$$a_k = \frac{2}{T}\int_0^T g(x) \cos k\nu x\, dx; \quad k \in \{0, 1, 2, \ldots\} \tag{9.15}$$

$$b_k = \frac{2}{T}\int_0^T g(x) \sin k\nu x\, dx; \quad k \in \{1, 2, 3, \ldots\} \tag{9.16}$$

We call a_k the Fourier-sine coefficient for the frequency $k\nu$ and b_k the Fourier-cosine coefficient for the frequency $k\nu$.

For example, if $g(x) = \sin m\nu x$, i.e., if the original function is a sine, we have $a_k = 0$ for all k, $b_m = 1$, and $b_k = 0$ for all $k \neq m$.

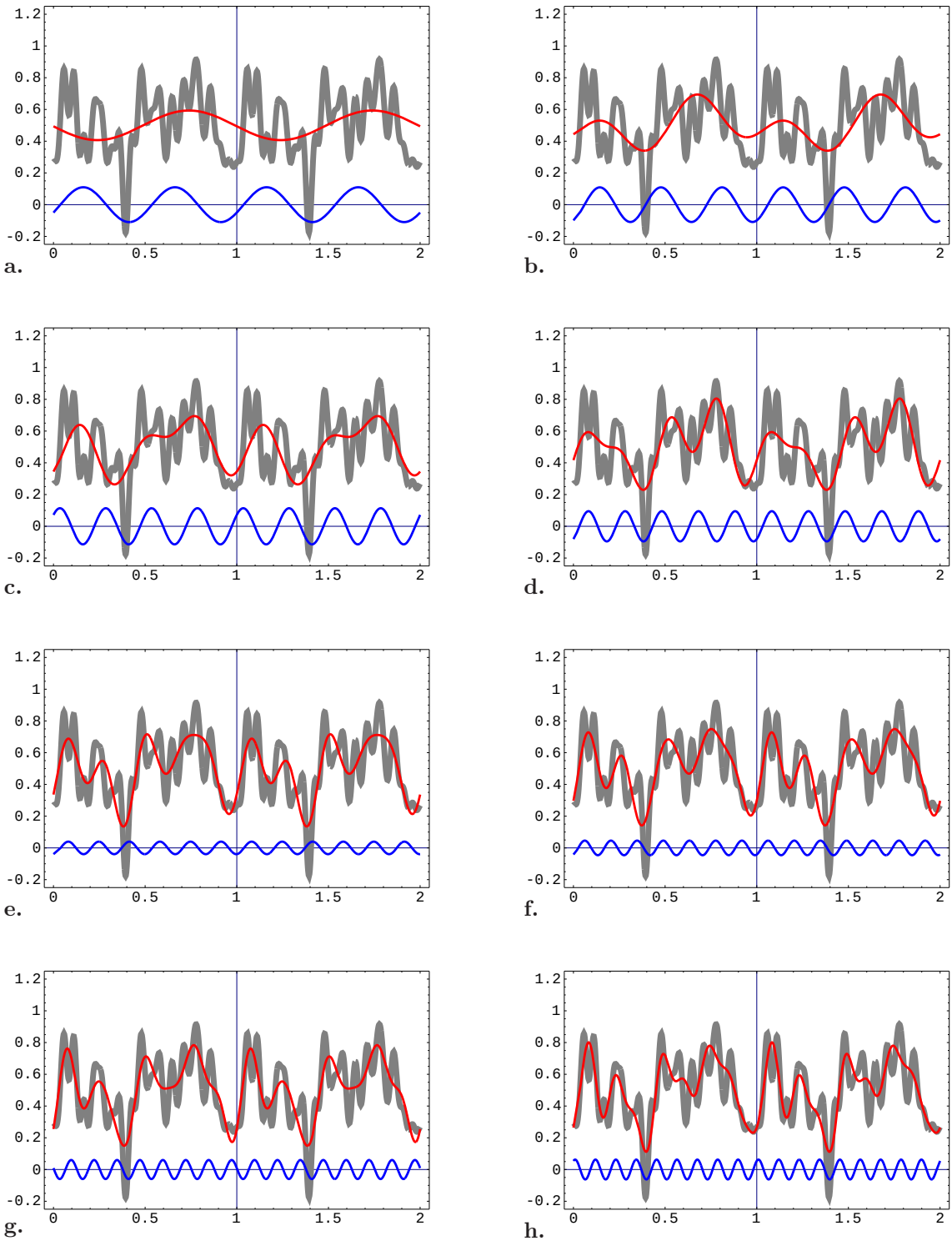


Figure 9.2: Approximation of an arbitrary periodic function by a Fourier-series. For explanations see text.

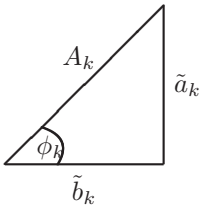


Figure 9.3: Geometrical interpretation of the relation of the quantities a_k, b_k, A_k and ϕ_k as described in Eq. 9.18
.

9.2.3 Complex notation

For each frequency, the sine and cosine components in Eq. 9.11 add up to a general sinusoidal of the form

$$f_k(x) = a_k \cos k\nu x + b_k \sin k\nu x =: A_k \cos(k\nu x - \phi_k) \tag{9.17}$$

Joint amplitude and phase can be obtained from the addition theorems of trigonometry. Simple calculation yields:

$$\begin{aligned} A_k &= \sqrt{a_k^2 + b_k^2} \\ \phi_k &= \arctan\frac{a_k}{b_k}. \end{aligned} \tag{9.18}$$

The notations can be simplified by the use of complex numbers. Complex numbers arise in algebra from solving equations like $x^2 + 1 = 0$. The solution of this equation, i.e. the square root of -1 is called the imaginary unit $i = \sqrt{-1}$. Complex numbers are linear combinations of a real and an imaginary part,

$$z = a + ib \tag{9.19}$$

where $a = Re(z)$ and $b = Im(z)$ are real numbers.

In the summing operation, complex numbers behave like two-dimensional vectors,

$$z_1 + z_2 = (a_1 + ib_1) \ + \ (a_2 + ib_2) \ = \ \underbrace{a_1 + a_2}_{Re(z_1+z_2)} + i \underbrace{(b_1 + b_2)}_{Im(z_1+z_2)}. \tag{9.20}$$

Multiplication, however, turns out to be somewhat more intricate:

$$\begin{aligned} z_1 z_2 &= (a_1 + ib_1)(a_2 + ib_2) = a_1 a_2 + ia_1 b_2 + ia_2 b_1 + (i)^2 b_1 b_2 \\ &= \underbrace{a_1 a_2 - b_1 b_2}_{Re(z_1 z_2)} + i \underbrace{(a_1 b_2 + a_2 b_1)}_{Im(z_1 z_2)}. \end{aligned} \tag{9.21}$$

To each complex number $z = a + ib$ is associated a so-called complex conjugate $z^* = a - ib$. From the multiplication rule above, it follows that $zz^* = a^2 + b^2$, a real number. The square root of zz^* is called the absolute value or modulus of z. In the 2D vector analogy, it is the squared length of the vector.

The usefulness of complex numbers in Fourier theory rests on Euler's formula extending the exponential function to complex numbers:

$$\exp\{i\varphi\} = \cos\varphi + i\sin\varphi \tag{9.22}$$

$$\exp\{z\} = \exp\{Re(z)\}(\cos Im(z) + i\sin Im(z)) \tag{9.23}$$

Euler's formula can be inverted to

$$\cos\varphi = \frac{1}{2}\left(e^{i\varphi} + e^{-i\varphi}\right) \tag{9.24}$$

$$\sin\varphi = \frac{1}{2i}\left(e^{i\varphi} - e^{-i\varphi}\right). \tag{9.25}$$

The crucial step is now to switch from sinusoidal functions to complex exponentials by the above formulae. For each individual frequency, we obtain:

$$\begin{aligned} f_k(x) &= a_k \cos k\nu x + b_k \sin k\nu x = A_k \cos(k\nu x - \phi_k) && (9.26)\\ &= c_{-k} \exp\{-ik\nu x\} + c_k \exp\{ik\nu x\}. && (9.27) \end{aligned}$$

Finally, Eq. 9.11 reduces to

$$p_n(x) := \sum_{k=-n}^{n} c_k \exp\{ik\nu x\} \tag{9.28}$$

with

$$\begin{aligned} c_o &= \frac{a_o}{2} \\ c_k &= \frac{1}{2}(a_k - ib_k) \\ c_{-k} &= \frac{1}{2}(a_k + ib_k). \end{aligned}$$

We will use the complex notation in the sequel.

9.3 Non-Periodic Functions: the Fourier Transform

The generalization to non-periodic functions is mathematically difficult, but intuitively quite easy, if we consider functions of increasing period length T. For a given T, for example $T = 2\pi$, we have coefficients at the multiples of the wave length $\nu = 2\pi/T = 1$,

$$\omega = k\nu = \frac{2k\pi}{T} \in \{1, 2, 3, 4, 5, \ldots\}. \tag{9.29}$$

If the period is twice as long, $T = 4\pi$, we obtain $\nu = 1/2$ and

$$\omega = k\nu = \frac{2k\pi}{T} \in \{\frac{1}{2}, 1, \frac{3}{2}, 2, \frac{5}{2}, \ldots\}. \tag{9.30}$$

In the end, if the period is infinite (i.e. if the function is no more periodic at all), we get a "coefficient" for every value of ω, i.e. a function of frequency. Switching back to the complex notation, we thus obtain the Fourier transform:

$$\tilde{g}(\omega) := \int_{-\infty}^{\infty} g(x) \exp\{i\omega x\} dx. \tag{9.31}$$

By the same token, the trigonometric series becomes:

$$g(x) := \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{g}(\omega) \exp\{-i\omega x\} dx \tag{9.32}$$

Eq. 9.31 is called the Fourier forward transformation and Eq. 9.32 the Fourier backward transformation. Applying both in a sequence reconstructs the original function as long as this was continuous and its square was integrable.

The function $\tilde{g}(\omega)$ in Eq. 9.31 is a complex function of the real argument ω. By Euler's formula (Eq. 9.22) the complex number $\tilde{g}(\omega_o) = \tilde{g}_c(\omega_o) + i\tilde{g}_s(\omega_o)$ for each ω_o gives the power and phase shift

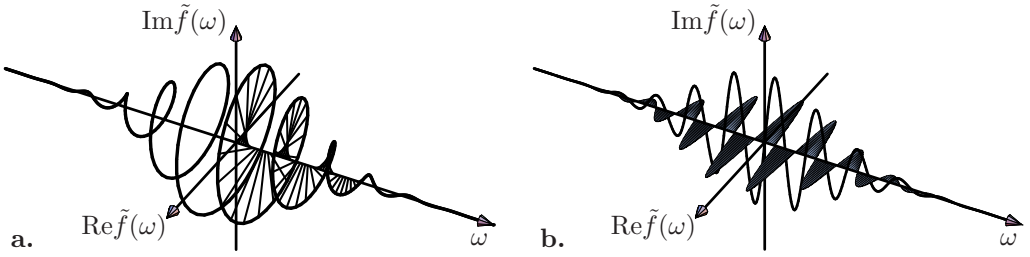


Figure 9.4: Complex Fourier transform of an eccentric Gaussian, $\exp\{-(x-x_o)^2\}$. **a.** 3D plot showing the complex functional values of each frequency ω as "vectors", or pointers in the complex plane. **b.** Real and imaginary parts of the same function shown separately. The lengths of the pointers in Fig. a correspond to the power of the signal (Fourier transform of autocorrelation). The angle of the pointer in the complex plane is the Fourier phase.

of the component with spatial frequency ω_o. If only the spatial frequencies present in a pattern are to be considered, one often uses the so-called power spectrum of g, i.e. the square of the absolute value (modulus) of $\tilde{\omega}$. A famous theorem in Fourier theory states that the power spectrum equals the Fourier transform of the autocorrelation function given by:

$$\Phi_{gg}(y) := \int g(x)g(x+y)dx, \tag{9.33}$$

in formal notation:

$$\tilde{\Phi}_{gg}(\omega) = |\tilde{g}(\omega)|^2 = \tilde{g}\tilde{g}^*. \tag{9.34}$$

We will discuss the autocorrelation function in more detail in the chapter on time series.

9.4 Fourier-Transforms in Two and More Dimensions

The Fourier transform generalizes to functions of two or more variables, such as images or spatio-temporal intensity distributions. The sinusoidal must in this case be replaced by a plane wave, e.g.

$$\sin(\omega_x x + \omega_y y) = \sin(\vec{\boldsymbol{\omega}} \cdot \vec{\boldsymbol{x}}). \tag{9.35}$$

Intuitively, these plane waves look like corrugated surfaces or wash-boards whose contour lines form a set of parallel straight lines. The orientation of these contour lines is orthogonal to the vector (ω_x, ω_y), the separation of wave peaks (wave length) is $2\pi/\sqrt{\omega_x^2 + \omega_y^2}$.

The Fourier transform then becomes a complex function of two or more real frequency variables:

$$\tilde{g}(\omega_x, \omega_y) := \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\exp\{i(\omega_x x + \omega_y y)\}dxdy. \tag{9.36}$$

Each point in the frequency plane (ω_x, ω_y) corresponds to one plane wave.

9.5 Summary

1. Every (sufficiently) continuous function can be unambiguously and reversibly represented by its Fourier transform:

$$\begin{aligned} \text{forward:} \qquad & \tilde{g}(\omega) := \int g(x) \exp\{i\omega x\} dx, \\ \text{backward:} \qquad & g(x) := \frac{1}{2\pi} \int \tilde{g}(\omega) \exp\{-i\omega x\} dx. \end{aligned} \tag{9.37}$$

   The real and imaginary part of $\tilde{g}$ are also called the Fourier cosine and Fourier sine transforms. Intuitively, Equation 9.37 says that every continuous function can be represented as the sum of sine and cosine functions.

2. (Shift theorem.) Let $g(x)$ be a function with Fourier transform $\tilde{g}(\omega)$ and $s \in \mathbb{R}$ a number specifying a shift of g. The shifted version of g, $g_s(x) := g(x + s)$ has the Fourier transform

$$\tilde{g}_s(\omega) = \exp\{-i\omega s\}\tilde{g}(\omega) \tag{9.38}$$

9.6 References

Bracewell, R. N. (1986). *The Fourier transform and its applications*. McGraw-Hill, New York, 2. edition.

Butz, T. (2010). *Fourier Transformations for Pedestrians*. Springer, Berlin, Heidelberg.

Tolstov, G. P. (1962). *Fourier Series*. Prentice-Hall, Inc., Englewood Cliffs, NJ.

Chapter 10

Linear Systems

10.1 Linear systems

In systems theory, a *system* is defined as a mapping assigning to each input function an output function. The term mapping was introduced in Chapter 3.1 as a generalization of the notion of a function, where the range and the domain are not numbers but elements of more general sets. In the case of a system, range and domain are sets of functions, the input of the system being in the domain set and the output being in the range set. In functional analysis, such systems are also known as "operators". They are closely related to mappings from an input vector to an output vector, if samplings of the respective functions are considered. Indeed, we will also consider discrete systems, where the input and output are given as sequences of numbers, x_i, y_i.

Both input and output functions may depend on time, space, or both. A typical example of a system in this sense is a receptor cell, for example for light, which receives a time-dependent input (light intensity) and a time-dependent output (transmitter release). If the output depends only on the immediate input, the system can be described by a simple input-output relation, i.e. a transfer function (in German 'Kennlinie'), since the input at previous times will have no influence on the current output. Such systems are called static, since the time dependence of the input plays no role. If, however, outputs depend on inputs during an entire time window preceeding the current instant, more complex descriptions are required.

10.1.1 Linearity

An important subclass of systems is characterized by linear superposition. Consider a system with the following input-output relations

$$x_1(t) \quad \mapsto \quad y_1(t) \tag{10.1}$$
$$x_2(t) \quad \mapsto \quad y_2(t). \tag{10.2}$$

A system is called linear if the following two conditions hold for all input functions f_1, f_2 and all real numbers λ:

$$(x_1(t) + x_2(t)) \quad \mapsto \quad (y_1(t) + y_2(t)) \tag{10.3}$$
$$\lambda x_1(t) \quad \mapsto \quad \lambda x_1(t). \tag{10.4}$$

Note that the second condition follows from the first one if λ is a positive integer.

The above conditions are also known as the superposition principle. Consider as a spatiotemporal system the surface of a lake, as the input to the system a pebble been thrown into the lake

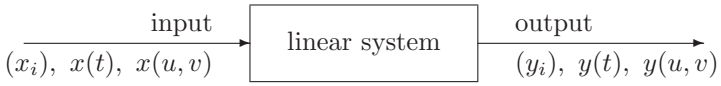


Figure 10.1: Black box representation of a "system". A system is defined here by its input-output relations, i.e. as a mapping from the set of possible input functions to an output function. In mathematics (functional analysis), such mappings are known as "operators". The different terms for inputs and outputs refer to discrete systems, time-dependent systems, and spatial systems, respectively.

and as the output the ripple pattern formed on the surface, conceptualized as the elevation or depression of the water surface as a function of position and time. In this case, linear superposition means that the ripple pattern formed by throwing two pebbles is the sum of the ripples generated by each pebble alone.

Strict linearity is a fairly strong constraint. For example, if $\lambda = -1$, it says that inverted inputs are answered by inverted outputs. In systems that cannot output negative values (e.g., in neurons no negative spike-rates exist), this implies that they cannot be linear in a strict sense. Likewise, if λ takes large positive values, linear systems would have to amplify their input without any distortion. Again, this is not possible in systems with saturation, i.e., limited outputs. A neuron that responds to a given stimulus with 10 spikes per second cannot produce 1000 spikes per second if the stimulus is amplified 100-fold. Again, this is not strictly linear. Technical systems can be linear in good approximation, but ideal linearity will never be obtained.

The importance of linear systems stems from two facts. First, a closed mathematical theory exists for a large subclass of linear systems (stationary or translation-invariant, see below). We will consider this theory in the sequel. Second, linearity is the first-order approximation of general systems if small inputs or small deviations from an average input are considered. Even if non-linear theory is applied, linear theory will always be used as a first approximation.

10.1.2 Stationarity and Translation-Invariance

A system $f(t) \mapsto h(t)$ is called stationary if for all $t_o \in \mathbb{R}$ the following condition holds:

$$x(t - t_o) \ \mapsto \ y(t - t_o). \tag{10.5}$$

This is to say, the system response depends only on the time interval passed between the arrival of an input and the output, not on the absolute time at which the system is considered. A stationary system needs not be static since output may depend on a whole temporal interval of inputs. However, this dependence is the same for all absolute times. In the above pebbles-and ripples example, the system is stationary as long as the same pebble throw will always yield the same ripple pattern. If, however, the pattern changes e.g., due to alternations in water level, stationary would be violated. Further examples for a non-stationarity are learning and growth, in which case the absolute time (i.e., the state of learning or growth) will matter.

In spatial systems, the analogous property is translation-invariance (also called space-invariance or shift-invariance). A system $x(u, v) \mapsto y(u, v)$ is called space-invariant if for all real numbers $u_o, v_o \in \mathbb{R}$ the following condition holds

$$x(u - u_o, v - v_o) \ \mapsto \ y(u - u_o, v - v_o). \tag{10.6}$$

As an example, consider a slide projector as a system with an input image (the slide) and an output image (the projection on the screen). Both images can be described as intensity functions

over spatial coordinates. Translation invariance means in this case that the blurring disk generated on the projection screen should be the same everywhere in the image. This can be violated as a consequence of imperfect lenses, for example if the blurring disk is enlarged or distorted at the image margins.

10.1.3 Examples

Sliding window

Consider a discrete system with input and output sequences $(x_i)_{i\in\mathbb{N}}$ and $(y_i) - i \in \mathbb{N}$. We can specify the output by the following equation:

$$y_1 := x_1 \tag{10.7}$$

$$y_i := \frac{1}{2}(x_{i-1} + x_i) \text{ for } i > 1 \tag{10.8}$$

At each instant, the output is calculated as the average of the current and the previous input, i.e., over a backward, sliding, window of width 2. Clearly, this idea can be extended to larger windows, or to windows with weights. A weighted window is characterized by a set of weights $(w_j)_{j=0,\ldots,J}$ where each w_j specifies the weight of an input that occurred j timesteps ago, i.e.,

$$y_i = \sum_{j=0}^{J} w_j x_{i-j} \quad \text{for } i \geq J. \tag{10.9}$$

If $i < j$, the sum will only be taken from 0 to i. For the sliding window of width 2, we obtain $w_0 = w_1 = \frac{1}{2}$.

The system described above is linear due to the linear equation for y_i. It is also stationary since the window or weight function does not change as it slides along. If all weights are positive, the system is said to have low-pass properties. That is to say, it forms some sort of sliding average of the input. If in a second step the output is substracted from a copy of the input, only the short-term modulations survive; the system is then called a high-pass.

Iterative low-pass filter

A special low-pass filter with certain mathematically useful properties is given by the iterative equation

$$y_i = kx_i + (1-k)y_{i-1}. \tag{10.10}$$

with 0‘k‘1. The equation describes a relaxation process; at each timestep, the current output is attenuated by a factor $k-1$ and augmented by the weighted current input. The weights are chosen such that a constant input will yield a constant output.

Eq. 10.10 can be expanded into a non-iterative equation by substituting $y_{i-1} = kx_{i-1} + (1-k)y_{i-2}$:

$$y_i = kx_i + k(1-k)x_{i-1} + (1-k)^2 y_{i-2} \tag{10.11}$$

$$= k\sum_{j=0}^{i}(1-k)^j x_{i-j}. \tag{10.12}$$

Here we have assumed that $y_o = 0$.

The expanded equation has the same structure as Eq. 10.9, with the weights $w_j = k(1-k)^j$. This describes an exponential decay, or relaxation. If we write $(1-k)^j = \exp\{j\log(1-k)\}$, we see that the weight function will be decayed to $1/e$ of its initial value if $j = -1/\log(1-k)$. This value is called the time constant of the decay. In our discrete example, the decay proceeds in steps such that the level $1/e$ may not be exactly reached.

The system described in Equation 10.12 is called a low-pass of order one with time-constant $1/\log(1-k)$.

10.2 Convolution

10.2.1 δ-impulse and impulse response

Eq. 10.9 is already the general equation of a linear, spationary system in the discrete, one-dimensional case. Before we discuss the continuous case, we consider a special input sequence, which we will call an "impulse":

$$d_i := (1,0,0,...) = \begin{cases} 1 & \text{for } i = 0 \\ 0 & \text{for } i > 0 \end{cases} \tag{10.13}$$

If we use this as an input in Eq. 10.9, we obtain

$$y_i = \sum_{j=0}^{i} w_j d_{i-j} = w_i. \tag{10.14}$$

That is to say, if stimulated with a short impulse stimulus, the response of the system will be the weight function, which is therefore also called the *impulse response*. In the spatial, 2D case, the impulse is replaced by a point stimulus in the plane and the system's response is called its *point-spread-function*.

In the continuous case, the definition of the impulse is slightly more complicated. The classical δ-impulse is generally defined as the limit of a sequence of functions such as

$$f_i(t) := \begin{cases} i & \text{if } -\frac{1}{i} < t \leq 0 \\ 0 & \text{else} \end{cases} \tag{10.15}$$

$$\delta(t) = \lim_{i\to\infty} f_i(t). \tag{10.16}$$

Mathematically a slightly incorrect, one can say that $\delta(t) = 0$ for all $t \neq 0$ and $\int_{-\infty}^{\infty} \delta(t)dt = 1$. One obtains this result also for other function sequences, i.e., Gaussian functions whose width approaches zero while at the same time the amplitude approaches infinity.

δ is called the Dirac-impulse or the δ-function. Mathematically, it is not really a function since the function value at the location $t = 0$ ("∞") is not a number. One obtains a formally satisfying definition of δ from the theory of the so-called distributions in Functional Analysis.

10.2.2 Convolution Integral

The relevance of the δ-impulse results from the fact that each function $f(t)$ can be formally divided into a series of chronologically consecutive pulses where the pulse at time t' has the amplitude $f(t')$ and therefore the form $f(t')\delta(t-t')$. This is fully analogous to the partitioning of a function into

rectangles that we had carried out for the introduction of the (Riemannian) integral in Fig. 3.2.2. If one reassembles the function from the pulses, one gets the (slightly tautological) equation

$$x(t) = \int_{-\infty}^{\infty} x(t')\delta(t-t')dt'. \tag{10.17}$$

The benefit of this equation follows from the linearity of the system. If one assumes to know the answer of the system to a δ-impulse and that this impulse response does not change depending on the moment of applying of the δ-impulse, then one can use this equation to predict the output to arbitrary input functions. We denote the impulse response by $g(t)$.

$$\delta(t) \quad \longrightarrow \text{Filter} \longrightarrow \quad g(t) \tag{10.18}$$

$$x(t) = \int_{-\infty}^{\infty} x(t')\delta(t-t')dt' \quad \longrightarrow \text{Filter} \longrightarrow \quad y(t) = \int_{-\infty}^{\infty} x(t')g(t-t')dt' \tag{10.19}$$

The right side of Eq. 10.19 is the convolution integral. We say that x is convolved with g. Note that the integration variable t' occurs in the two function under the integral with opposite sign which is charateristic for equations of the convolution type. A similar type of equation where the integration variable has the same sign in both arguments is called the correlation integral.

The integration boundaries are choosen to include the entire set of real numbers. Of course, if we think of time-dependent systems, the impulse response can start only after the impulse is delivered. We can therefore assume that $g(t) = 0$ for $t < 0$. This property is sometimes called "causality". For the integral, it means that we could have taken the boundaries from $-\infty$ to t.

The convolution operation is commutative, that is to say that the roles of input and impulse response can be exchanged. With the explicit boundaries, we may write

$$\int_{-\infty}^{t} x(t')g(t-t')dt' = \int_{0}^{\infty} x(t-t')g(t')dt'. \tag{10.20}$$

In the left integral, t' is an instant in the past, i.e., before the current instant t, while $t-t'$ is the time elapsed since t'. Inputs delivered at t' are weighted depending on the time passed since t', and summed. In the right integral, we think about inputs that occured an interval t' back in time and sum over all these intervals. The two intuitions are equivalent and reflect the commutativity of convolution.

10.2.3 Convolution in Two Dimensions

In two-dimensional systems, the same logic applies. If $g(u,v)$ is the point-spread-function of a system, the equations read

$$y(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(u',v')g(u-u',v-v')du'dv' \tag{10.21}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(u-u',v-v')g(u',v')du'dv'. \tag{10.22}$$

In space, "causality"" is not an issue: the system may spread input in all directions in space, but only forward in time. Indeed, in many cases point spread functions will be circularly symmetric, or "isotropic".

10.3 Convolution and Fourier transforms

10.3.1 Sinusoidal inputs

A central result of linear system theory states that sinusoidal inputs applied to a linear, shift-invariant system yield sinuoidals of the same frequency that may, however, differ in amplitude and phase. The prove of this fact is straight forward and relies on the well known addition theorem for trigonometric functions: $\sin(\alpha - \beta) = \sin\alpha \sin\beta - \cos\alpha \sin\beta$. Consider a system with impulse response $g(t)$. If we convolve this with a sine, we obtain

$$\int g(t')sin(\omega(t-t'))dt' \tag{10.23}$$

$$= \int g(t')\sin(\omega t)\cos(\omega t')dt' + \int g(t')\cos(\omega t)\sin(\omega t')dt' \tag{10.24}$$

$$= \sin(\omega t)\int g(t')\cos(\omega t')dt' + \cos(\omega t)\int g(t')\sin(\omega t')dt' \tag{10.25}$$

The output is thus the sum of a sine and cosine term with the same frequency. This imposes on the output an amplification (or attenuation) and a phase shift while the sinusoidal shape and the frequency remain unchanged. We introduce the notations

$$\tilde{g}_c := \int g(t')\cos(\omega t')dt' \tag{10.26}$$

$$\tilde{g}_s := \int g(t')\sin(\omega t')dt' \tag{10.27}$$

and observe that $\tilde{g}_c$ and $\tilde{g}_s$ are equivalent to the Fourier sine- and cosine-coefficients defined in Eqs. 9.15 and 9.16, up to some multiplicative constant.

From $\tilde{g}_c$ and $\tilde{g}_s$, the amplification and phase shift can be calculated as

$$A(\omega) = \sqrt{\tilde{g}_c^2 + \tilde{g}_s^2} \tag{10.28}$$

$$\phi(\omega) = \arctan\frac{\tilde{g}_c}{\tilde{g}_s}. \tag{10.29}$$

Clearly, the amplitude and phase factors can also be combined in one complex number, as discussed in section 9.2.3:

$$\tilde{g}(\omega) = A(\omega)e^{i\phi(\omega)} \tag{10.30}$$

10.3.2 Filters: The Modulation Transfer function

In Eqs. 10.28 and 10.29, we have explicitly written the frequency of the input signal as a variable. Indeed, in the identification of linear, shift-invariant systems, one approach is to apply sinusoidal inputs of various frequencies and measure the output amplitude and phase as a function of input frequency. The result is called the modulation transfer function (MTF)[1]. The amplitude part[2] and the phase part[3] of the MTF are then plotted in a so-called Bode[4]-diagram, showing amplitude in double logarithmic scale and phase in single logarithmic scale (see Fig. 10.2).

[1] in German: Modulationsübertragungsfunktion, MÜF
[2] in German: Amplitudenfrequenzgang
[3] in German: Phasenfrequenzgang
[4] Hendrik Wade Bode, 1905-1982, American engineer

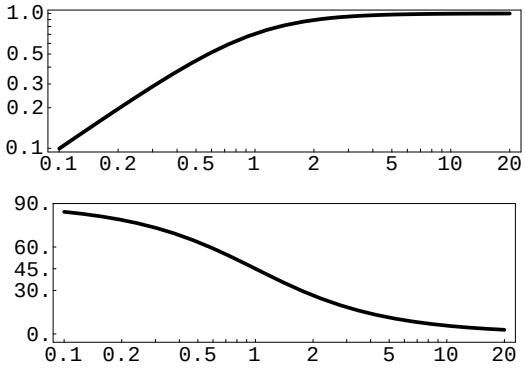


Figure 10.2: Bode diagram (modulation transfer function) of a high pass. Top: Amplitude, (Modulationsübertragungsfunktion), below: phase.

Consider now a general input signal $x(t)$. As explained in Section 9 (Eq. 9.28), it can be expressed as a sum of complex sinusoidals

$$x(t) = \sum_{k=-\infty}^{\infty} c_k \; e^{ik\omega)} \tag{10.31}$$

From the shift-invariance property of the system (Eq. 10.6, it follows that the response to pure sine and cosine inputs will differ only in a phase offset of 90°. From the linearity, it follows that the output elicited by a sum of sine and cosine inputs is the sum of the outputs elicited by each component individually. With the modulation transfer function, we know these individual outputs; they are generated by applying the frequency-specific amplitude and phase factors to each component. In the complex notation, amplitude factor and phase shift together amount to a multiplication. We can therefore calculate the overall output by

$$y(t) = \sum_{k=-\infty}^{\infty} \tilde{g}_k(\omega) \; c_k \; e^{ik\omega)}. \tag{10.32}$$

Eg. 10.32 describes a filtering process in the sense that frequency components for which the MTF is different from zero will be passed whereas frequency components for which the MTF is zero, will be filtered out. A filter passing high but not low frequency is called a highpass filter. By the same logic, lowpass and bandpass filters can be defined. A notch filter is a filter passing everything but a specific band of frequencies. If $y_L(t)$ denotes the lowpass filtered version of some signal $x(t)$, a highpass filtered version may be obtained as $y_H(t) = x(t) - y_L(t)$.

10.3.3 The Convolution Theorem

We have now collected the main results to formulate the central result of linear systems theory, i.e. the Convolution Theorem. In short, it states that the Fourier transform of the convolution of two functions equals the pointwise product of the Fourier transforms of the same two functions.

Figure 10.3 summarizes the situation so far. Systems can be identified either by their impulse response $g(t)$ or by their modulation transfer function $\tilde{g}(\omega)$. In the case of the impulse response, the system response is calculated by convolution of the input with the impulse response. If the MTF is used, outputs to arbitrary inputs are calculated by taking the Fourier transform of the input, multiplying it pointwise (i.e., for each frequency) with the according value of the MTF and reconstructing the output by Fourier backwards transformation of the pointwise product.

The only remaining point for the formulation of the convolution theorem concerns the relation between the impulse response and the MTF. The answer to this question is already contained in

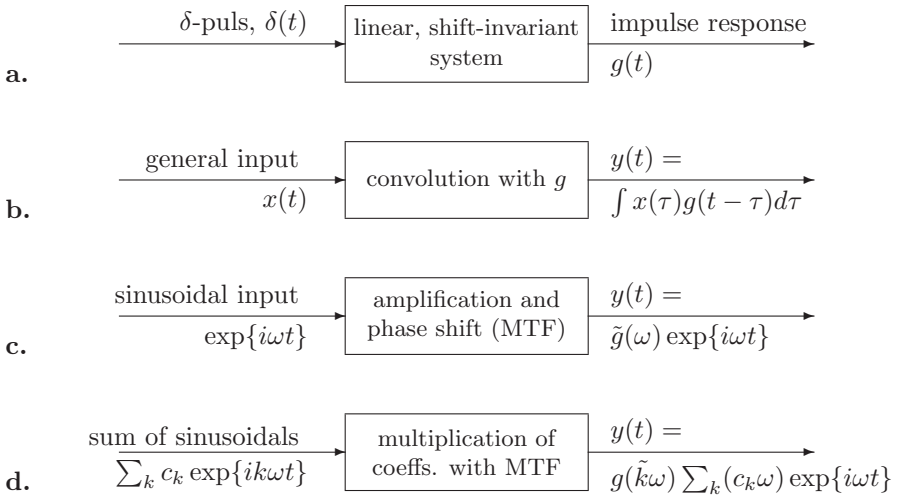


Figure 10.3: System identification. **a**: Stimulating the system with a δ-impulse yields the impulse response $g(t)$. **b.** From the impulse response, responses to general input functions can be calculated by convolution. **c.** Stimulation with a sinusoidal yields an amplified and phase shifted sinusoidal of the same frequency. The complex amplitude and phase factor equals the modulation transfer function $\tilde{g}(\omega)$ evaluated at the frequency of the sinusoidal. **d.** A sum of sinusoidals as the input yields a sum of sinusoidals as output, where each coefficient is multiplied by the respective value of the MTF.

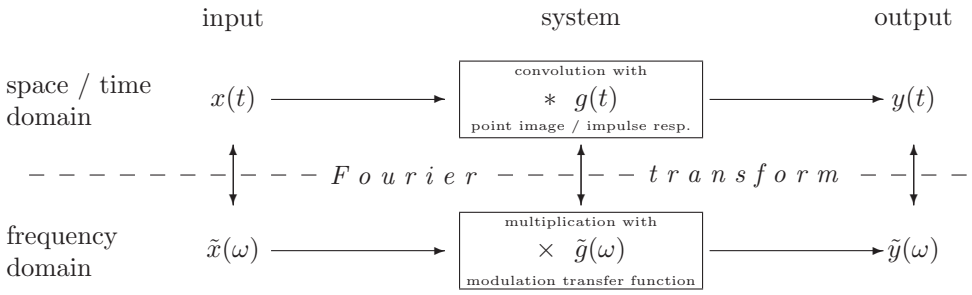


Figure 10.4: Summary of the relation of Fourier transform and linear systems theory.

Eq. 10.26 and 10.27: the MTF is the Fourier transform of the impulse response. Of course, this is the reason why we used the notation $\tilde{g}(\omega)$ from the start.

In conclusion, the theorem then says:

$$(f * g)\tilde{} = \tilde{f}\tilde{g}. \tag{10.33}$$

In words, the Fourier transform of a convolution equals the pointwise product of the Fourier transformations. This theorem thus simplifies system analysis by reducing convolution to a multiplication. In system analysis, one therefore tends to think "in the frequency domain", i.e, in terms of the Fourier transforms of the signals involved. The situation is summarized also in Figure 10.4.

10.4 References

Diniz, P.S.R., da Silva, E.A.B., Netto, S. L. (2010) *Digital Signal Processing. System Analysis and Design* Cambridge University Presss, 2. edition.

Ziemer, R. E., Tranter W. H., Fannin, D. R. Bracewell, R. N. (2009). *Signals & Systems. Continuous and Discrete*. Pearson. 4. edition.

Chapter 11

Time Series

11.1 Basic Properties

A series of measurements taken over a continuous temporal variable, either at regular or at arbitrary intervals, is called a time series. We will use both the continuous formulation $x(t)$ for some time interval (t_o, t_1) and a sequence notation $x_i, i = 1, 2, ...$

The idea of time series analysis is to find regularities in time series and, mostly in stock marked applications, to predict future development of these series. Here, we are not so much interested in predictions, but in identifying underlying rules. Both problems are, however, related since regularities can be used to make predictions. We start by constructing some typical series.

11.1.1 White noise

Assume a time series is sampled at discrete times yielding the values

$$x_1, x_2, x_3, ... \tag{11.1}$$

We assume that each value is drawn from a fixed random distribution. For example, we could toss a coin at each time and set x_t to 1 or 0 depending on the outcome of the coin toss. In this case, it is clear that the values at time t and time $t+1$ will be uncorrelated, as will be all combinations x_t and $x_{t+\tau}$. Generally, a time series satisfying this condition, i.e., a time series where neighboring points are uncorrelated, is called a white noise process.

Uncorrelatedness can be obtained with the described binary distribution (the coin toss) as well as with other random distributions from which the individual x_i-values are drawn. The most important noise process is Gaussian white noise where the value at each instant in time is drawn from a normal distribution. The motivation for the term "white" noise will become clear below.

If we add up the individual x_i values iteratively, we obtain a new time series y_i defined by

$$\begin{aligned} y_0 &= 0 \\ y_i &= y_{i-1} + x_i = \sum_{j=1}^{i} x_j \end{aligned} \tag{11.2}$$

This time series is called a (one-dimensional) random walk. Two-dimensional random walks can be defined analogously and can be used to model chance level in experiments measuring animal trajectories.

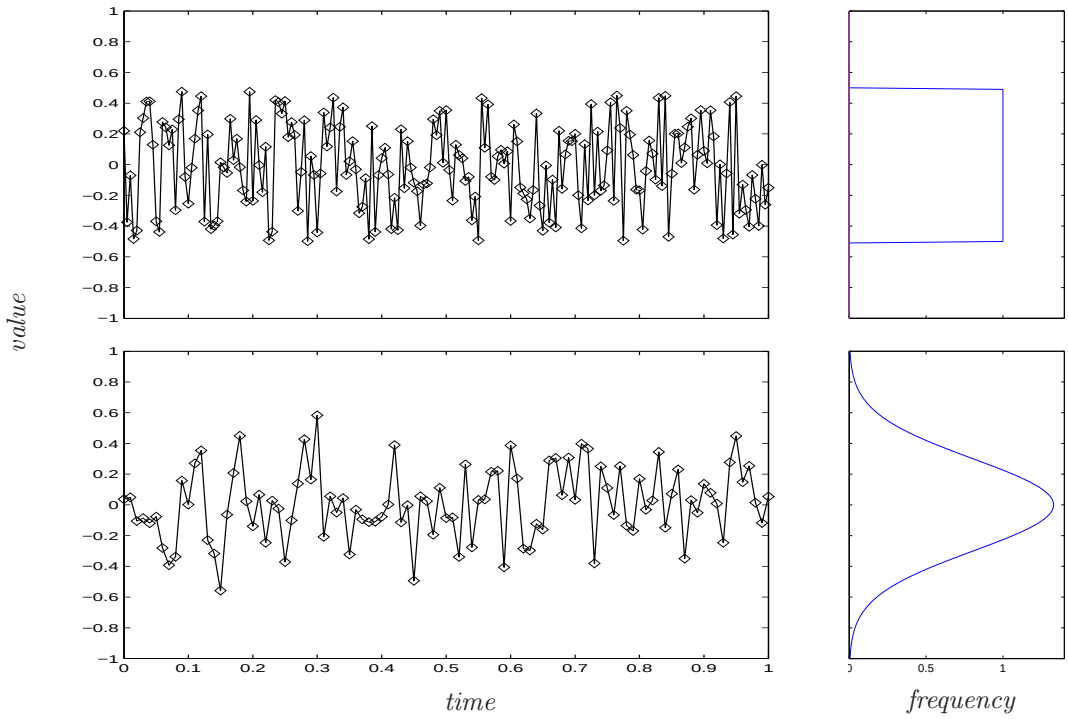


Figure 11.1: Discrete white noise processes (left) and the underlying distributions (right). Top: uniform distribution in the interval $(-0.5, 0.5)$. Bottom: normal distribution ("Gaussian white noise") with standard deviation 0.3.

11.1.2 Variance

The variance of the random distribution, from which the individual samples are drawn, is also the variance of the random process. Throughout this chapter, we will assume that the average value of the time series is zero. We consider the values of x obtained at different instances of time as samples and obtain the variance

$$\gamma_x := \lim_{T\to\infty} \frac{1}{T} \int_0^T x^2(t)dt \tag{11.3}$$

or, in discrete notation

$$\gamma_x := \lim_{N\to\infty} \frac{1}{N} \sum_i^N x_i^2. \tag{11.4}$$

Usually, the limit is not actually evaluated but replaced by some sufficiently large value of T or N, respectively. We can ignore the particular choice of T altogether as long as we are only interested in the position of peaks in the auto- and cross-correlation functions described below. In the following text, we will not be strict with the time interval as long as no confusions can arise.

Alternatively to Eq. 11.3, where averages are taken over time, we could also have averaged over the distribution from which the individual elements of the time series are drawn. In our examples, this will yield the same result, but in general, deviations are possible. If the average of one representative of the process taken over time is equal to averages taken over multiple realizations of the same process are equal, the process is called ergodic.

11.1.3 Moving averages

Consider a new time series y_i generated from the white noise process by the following rule:

$$y_i = \frac{1}{2}(x_i + x_{i-1}) \tag{11.5}$$

Clearly, this is not a white noise process, since samples $y_i = (x_i + x_{i-1})/2$ and $y_{i+1} = (x_{i+1} + x_i)$ share the component x_i. The series y_i is a low-pass filtered version of the original series x_i. Eq. 11.5 can be considered a discrete convolution of the series $(x_i)_i$ with the kernel $(\frac{1}{2}, \frac{1}{2})$. Of course, other smoothing schemes can be described by choosing appropriate convolution kernels. Low-pass filtered versions of a white noise process are sometimes called pink noise.

11.1.4 Point processes

If the random process can only take the values zero or one, an alternative way of representing it is by reporting the time steps, where a one occured. For example, the series $1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1$ would thus be represented as $1, 4, 5, 9, 10, 12$. This type of random process is called a point process since it gives the point in time, where something happened, without specifying the event itself. An example from physics is radioactive decay. In biology, point processes arise in the description of spike series or behavioral events.

A related scheme is counting the number of subsequent zeros until the next value of one occurs ("run-length encoding"). For example, the series $1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1$ would thus be represented as $(1, 3, 1, 4, 1, 2)$. Clearly, this is most appropriate for series where ones are rare as compared to the time rate of sampling.

11.1.5 Markov chains

In Section 2.3.3, we have already seen an example for a Markov chain, i.e. a random process switching between discrete states (the pub visited by the student) with fixed probabilities of transition. Markov chains are a general tool for the analysis of behavior and, in particular, in speech processing. Here, the discrete states are the syllables produced by the subject and syllable recognition is guided by known or estimated transition probabilities between syllables.

As compared to the example from Section 2.3.3, a general Markov chain is defined by the rule that the probability distribution of the state x_i at time i depends on the previous outcome x_{i-1} and maybe earlier outcomes x_{i-k}. However, the range of previous states affecting the current state is limited by some threshold k_{max}. Earlier outcomes do not affect the current probabilities. General Markov chains may thus involve higher order transition matrices and continuous outcomes.

An example for a Markov chain with continuous outcome and $k_{max} = 1$ is the random walk described in Section 11.1.1.

11.2 Periodicity: The Autocorrelation function

An important property of a time series concerns its internal dependencies: how strong does a value at time t depend on values taken at time $t - 1$, $t - 2$, or generally at time $t - \tau$. As often in statistics, we replace dependency with correlation and study the autocorrelation function

$$\Phi_{xx}(\tau) := \int x(t)x(t - \tau)dt, \tag{11.6}$$

or, in discrete notation,

$$\Phi_{xx}(\tau) := \sum_i x_i x_{i-\tau}. \tag{11.7}$$

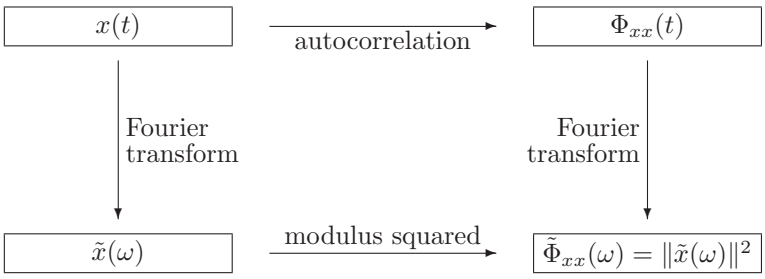


Figure 11.2: Autocorrelation and power spectrum, illustrating the Wiener-Khinchin theorem.

The integral is taken over the available data set. Since we are usually interested only in peaks of the autocorrelation function, normalizing factors are not important.

Properties

1. $\Phi_{xx}(\tau)$ takes its maximal value at shift $\tau = 0$.

$$\Phi_{xx}(0) \geq \Phi_{xx}(\tau) \quad \text{for all } \tau. \tag{11.8}$$

   This is due to the fact that the maximal similarity between $x(t)$ and $x(t-\tau)$ is of course obtained for $\tau = 0$. If a normalization is needed, it is often convenient to consider $\Phi_{xx}(\tau)/\Phi_{xx}(0)$. Indeed, when ignoring the normalization factor, $\Phi_{xx}(0)$ is simply the variance γ_X of the signal as defined in Eq. 11.3.

2. The auto-correlation function is symmetric,

$$\Phi_{xx}(-\tau) = \Phi_{xx}(\tau). \tag{11.9}$$

3. If $x(t)$ is periodic with length T, the autocorrelation function will also be periodic with length T, i.e. it will have a peak at $t = T$.

4. Let $y(t)$ be a shifted version of $x(t)$, i.e. $y(t) = x(t+a)$. Then, the autocorrelation functions of x and y are the same. This implies for example, that the autocorrelations of the sine and the cosine function are the same. In both cases, the first peak outside the coordinate origin will appear at $T = 2\pi/\omega$, indicating the frequency of the sinusoidal. The phase (sine vs. cosine) is lost.

5. The Fourier transform of the autocorrelation function equals the absolute value (modulus) of the complex Fourier transform of x:

$$\tilde{\Phi}_{xx}(\omega) = \|\tilde{x}(\omega)\|^2 \tag{11.10}$$

   (Wiener[1]- Khinchin[2] theorem). This function is also known as the power spectrum (dt: Leistungsdichtespektrum) of x. For an illustration of this theorem, see Fig. 11.2.

6. If x is a white noise process, $x(t)$ will be uncorrelated with $x(t-\tau)$ for all $\tau \neq 0$, i.e. $\Phi_{xx}(0) = 1$ and $\Phi_{xx}(t) = 0$ for all $t \neq 0$. The Fourier transform of this so-called impulse function, i.e. the power spectrum of the white noise process, is a constant, indicating that all temporal frequencies are included in the process. This is why the name "white" noise was chosen in the first place. Similarly, low-pass filtering of a white noise process reduces the higher frequencies, leading to a higher relative content of low frequencies. In analogy to the visual spectrum where low frequencies appear red, such noise processes are sometimes called pink.

[1] Norbert Wiener, 1894-1964.
[2] Alexandr Yakovlevich Khinchin, 1894-1959.

Application

The main application of both the autocorrelation function and the power spectrum is to find periodicities in time series. As compared to simple Fourier transform, it has the advantage of neglecting phase differences, i.e. periodicities in the sine and cosine components both contribute to the same peak.

11.3 Influences between two time series: Cross-correlation

If two simultaneous time series are considered, $x(t)$ and $y(t)$, say, one may be interested to know whether the $x(t)$ influences $y(t)$ or the other way round. This can be studied by the cross-correlation function:

$$\Phi_{xy}(\tau) := \frac{1}{\sqrt{\gamma_x \gamma_y}} \int x(t)y(t-\tau)dt. \tag{11.11}$$

Again, integration is taken over the biggest available interval.

Properties

1. The autocorrelation defined above can be considered the cross-correlation of a function with itself. The cross-correlation function does not generally peak at zero.

2. The symmetry relation generalizes to:

$$\Phi_{xy}(-\tau) = \Phi_{yx}(\tau). \tag{11.12}$$

   In general, therefore, the cross-correlation function is not symmetric.

3. If $y(t)$ follows $x(t)$ with some delay a, i.e., $y(t) = x(t-a)$, the cross-correlation function of x and y will have a peak at $\tau = -a$.

$$\Phi_{xy}(\tau) := \int x(t)y(t-\tau)dt = \int x(t)x(t-\tau-a)dt = \Phi_{xx}(\tau+a) \tag{11.13}$$

$$\Phi_{xy}(-a) = \Phi_{xx}(0). \tag{11.14}$$

   This is the most important property of the cross-correlation function. If x drives y with some delay, this delay can be detected as an off-zero peak in Φ_{xy}.

4. There exists also a relation to the Fourier transforms of x and y, expressed in complex number notation:

$$\tilde{\Phi}_{xy}(\omega) = \tilde{x}(\omega)\tilde{y}(\omega)^* \tag{11.15}$$

   Here, the asterisk denotes the complex conjugate, i.e. $(a+ib)^* = (a-ib)$. Since $(a+ib)(a-ib) = a^2+b^2$, Eq. 11.15 is consistent with Eq. 11.10.

   $\tilde{\Phi}_{xy}$ is sometimes called the cross-spectrum.

EEG coherence

Cross-correlations are frequently applied in the analysis of multiple spike trains. If a peak is found in the cross-correlation of spike trains from two different cells, it is often assumed that the leading cell is driving the trailing cell. However, correlations may also arise from common input without direct connectivity between the cells.

The cross-spectrum is used to define synchrony between EEG signals from multiple cortical areas. A typical definition of coherence in EEG is

$$\mathrm{Coh}_{xy}(\omega) := \frac{\tilde{\Phi}_{xy}(w)^2}{\tilde{\Phi}_{xx}(w)\tilde{\Phi}_{yy}(w)}. \tag{11.16}$$

This function can be integrated within each of the typical EEG frequency bands (e.g., $\alpha = 8-13$ Hz, $\beta = 13 - 30$ Hz).

11.4 The Poisson process

As another application of time series analysis, we consider data giving the times of occurrence of an event. Examples for such point or signal processes include the measurement of the occurrence of a behavior (e.g., the interruption of a light beam by an animal passing by), the reported "flipping" of the perception of a bistable figure such as the Necker cube, or neural spike trains.

Many such data can be described by Poisson[3] processes, which can be thought of as a generalization of binomial distribution for rare events. In the case of the Poisson process (rather than the Poisson distribution), the argument is as follows.

Consider a short time interval Δt. We require that the probability of an event occuring in that time interval is

$$p_1(\Delta t) = \lambda \Delta t \quad \text{for } \lambda > 0. \tag{11.17}$$

The subscript 1 in p_1 indicates that one and only one event is to take place in the interval Δt. Eq. 11.17 states that the probability for an event occuring in a given interval is proportional to the length of the interval. This will not hold for long intervals, since in this case more than one event is likely to occur and p_1 will therefore decrease.

A second requirement states that events do not occur in pairs or triplets etc., but independent of each other. Therefore, the probability that more than one event occurs during a short interval is zero,

$$p_o(\Delta t) + p_1(\Delta t) = 1. \tag{11.18}$$

These two requirements define the Poisson process. It is possible to prove that the distribution of the variable X_t, i.e., the number of events occuring from time zero to time t, has the distribution

$$P(X_t = i) = \frac{(\lambda t)^i}{i!} \exp(-\lambda t). \tag{11.19}$$

Here, $i!$ is the factorial, i.e. $i! = i \times (i-1) \times (i-2) \times ... \times 2 \times 1$, $0! = 1! = 1$. Eq. 11.19 is the standard Poisson distribution with parameter λt. Mean and variance of this distribution are equal and take the value of the parameter, i.e. λt. The derivation of the distribution function from the requirements can be found in Fisz (1980), Section 8.3.

Eq. 11.19 describes the number of events occuring in a time interval from 0 to t. In measurements, we are often not so much interested in this number than in the interval, or pauses between events. Let τ_k be the time of occuring of the kth event. We can then measure an interval preceeding the kth event. Its length is $U_k = \tau_k - \tau_{k-1}$. Note that while X_t is a discrete random variable, taking values $1, 2, 3, ...$, U_k is a continuous variable with positive real values. We note without proof that the probability density function of U is given by

$$p.d.f.(u) = \lambda \exp\{-\lambda u\} \tag{11.20}$$

(cf. Fisz 1980, problem 8.13.3). The probability density function can be approximated in measurements by histograms of the frequency of various interval durations.

[3] Siméon Denis Poisson, 1781-1840

11.5 References and suggested reading

Fisz M. (1980) Wahrscheinlichkeitsrechung und mathematische Statistik, 10. Auflage. Berlin: VEB Deutscher Verlag der Wissenschaften.

Shumway RH, Stoffler DS. (2000) Time Series Analysis and Its Applications. New York: Springer.

Chapter 12

Image Processing

12.1 Images

Images are two-dimensional distributions of color- or gray-scale values. On the computer such images are always sampled and discretized:

Sampling: Image space is split up into a set of **"pixels"** ("picture elements") which in MATLAB are numbered beginning with $(1, 1)$ from the upper left corner. The pixel with the number (i, j) is located in line i and column j. We denote the width of the image (number of columns) with I, the height (number of lines) with J.

Discretization: The continuous color- or gray-scale values of natural images are presented in the computer as discrete numbers. MATLAB uses three types of numbers:

- `uint8` (unsigned integer, 8 bit). These are the color- or gray-scales from 0 to 255. Three color values with 8 bit respectively amount to 24 bit or $2^{24} = 16.777.216$ different colors.
- `uint16` (unsigned integer, 16 bit). These are the color- or gray-scales from 0 to 65535.
- `double` (double precision floating point). In this case, the color- or gray-scales are depicted as floating point numbers in the interval $[0, 1]$.

Pixels are assigned color values in one of two ways:

- 'True color image' (RGB): Here, three color values are stored for every pixel. This can be done with any of the above mentioned resolution. If one uses `uint8`, the image is a $I \times J \times 3$-matrix of 8-bit-numbers.
- "'Mapped images"': If less color gradations are needed, one can define these in a palette or colormap whose colors are numbered consecutively from 1 to n. For every pixel one indicates this number ("index") as integer-number. For presenting the image, the color has to be looked up in the colormap. If using a palette of 16 color- and intensity values, then the image consists of a $I \times J$-matrix of index values as well as a colormap of 16×3 color values. Those color values can be of the type `uint8`, `uint16`, or `double` again.

  Gray-value images are treated in MATLAB as a special case of indexed images where the colormap only contains entries with three equal values for R, G, and B, respectively. The command `colormap(gray)` produces such a color chart.

The MATLAB-command `pixval` shows in a given image the (i, j)- coordinates as well as the value of the pixel over which the mouse pointer is currently positioned.

For the computer, images are therefore nothing else than matrices of color- and intensity values which in itself have no meaning at all. If wishing to mark off and measure regions in the image, find objects or detect movements, then one has to start from these matrices. The according operations and algorithms are the scope of image processing.

Through the usage of cameras for measuring problems of all kinds image processing becomes more and more important in data analysis. In Chapter 8, we have already considered an example where images were treated as statistical data vectors. Other examples are the tracking of movements (of animals, limbs, eyes) in behavioral experiments, the various tomographic or imaging procedures in medicine, or the quantitative evaluation of anatomical and histological images. In addition to this, methods of image processing are also deployed in general when two-dimensional distributions of data are calculated, so for example also in chromatography and acoustics (sonograms).

12.2 Filtering

Images can be treated as matrices or vectors (chaining row by row in one long sequence), and the usual rules of arithmetics apply. Adding up two images intuitively corresponds to the superposition of two images projected from two beamers, say, where the intensity values of the pixels falling over each other are added. Another important operation is spatial shift where, for a one-pixel shift to the left, each intensity value is replaced by that of the neighboring pixel to the left.

From shifting, multiplication and addition the most important image processing operation, namely the filtering or correlation with a mask can be generated. As a simple example we consider an image given by a gray-level matrix G. In order to make the image "smoother", e.g. making it more blurred in order to suppress pixel noise, one can create a new image H by averaging at every pixel (i, j) over the next neighbors, i.e.:

$$H(i,j) := \frac{1}{5}(G(i,j) + G(i-1,j) + G(i+1,j) + G(i,j-1) + G(i,j+1)). \tag{12.1}$$

An enhancement of the contrasts between different pixels can be achieved by subtracting the mean gray-level of a neighborhood from the individual gray-scale values, i.e.:

$$H(i,j) := G(i,j) - \frac{1}{4}(G(i-1,j) + G(i+1,j) + G(i,j-1) + G(i,j+1)). \tag{12.2}$$

This equation will not work for pixels that lie on the margin of the image; for example, $i - 1$ might evaluate to 0, but $G(0, j)$ is not in the image. In such cases, one simply sets pixels that lie outside of the image to zero ($G(0, j) := 0$) or continues the image matrix at the margin constantly ($G(0, j) := G(1, j)$). MATLAB offers both possibilities as options. In any case, the size of the result image equals the size of the input image.

In Equations 12.1 and 12.2, the influence of a pixel $G(i, j)$ on a pixel $H(k, l)$ of the new image depends only on the difference of the coordinates $(k - i, l - j) = (m, n)$, i.e. at every place in the image the same procedure is being performed. Such operations are called "space-invariant" or "translation-invariant". The most universal case of this kind of operations is obtained when specifying a "weight" $C(m, n)$ for every coordinate difference with which the observed pixel acts on another. We call the matrix C a *mask* and write:

$$\begin{aligned} H &= G \otimes C, \\ H(k,l) &= \sum_{m=-M}^{M} \sum_{n=-N}^{N} C(m,n)\, G(k+m, l+n). \end{aligned} \tag{12.3}$$

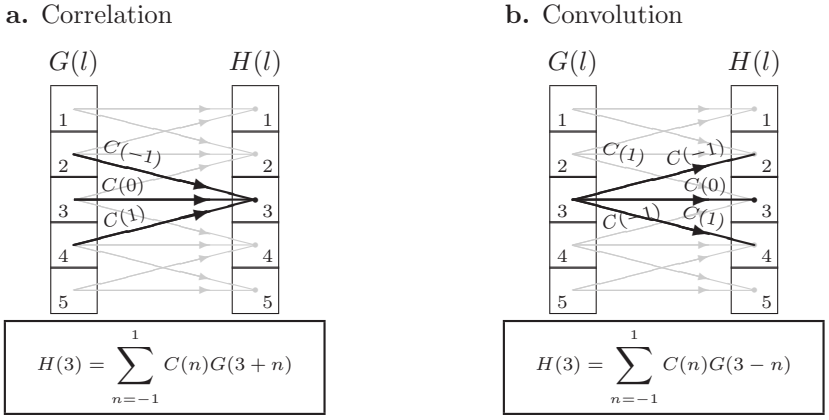


Figure 12.1: Correlation and Convolution. Both operations differ from each other only in the numbering of the pixels of the mask. In correlation (**a.**) this happens from the perspective of the result image H. If one imagines looking "back" from H onto G, then the negatively indicated pixels are on the right, the positively indicated pixels on the left. The mask describes the convergence (the "gathering") of signals from the input image. For determining the signal strength at one fixed pixel of the input image the mask does not have to be moved. In convolution (**b.**) the pixels of the mask are numbered from the perspective of the input image. If one imagines looking "forward" from G to H, then the negatively indicated pixels are on the left, the positively indicated on the right. The mask describes the divergence (the "spread") of signals from the input image. For determining the signal strength of the result image the mask has to be shifted over the input image.

Here, we have assumed that the components of the matrix C are not numbered from 0, as usual, but from $-M$ to M or $-N$ to N. The number of rows and columns is therefore always odd. For the two above mentioned examples (Eq. 12.1 and 12.2) there are

$$C = \begin{array}{|r|r|r|}\hline 0 & 1/5 & 0 \\ \hline 1/5 & 1/5 & 1/5 \\ \hline 0 & 1/5 & 0 \\ \hline \end{array} \quad \text{and} \quad C = \begin{array}{|r|r|r|}\hline 0 & -1/4 & 0 \\ \hline -1/4 & 1 & -1/4 \\ \hline 0 & -1/4 & 0 \\ \hline \end{array}. \tag{12.4}$$

The operation from Eq. 12.3 is called *correlation* or *filtering*. In MATLAB it is performed by the function `imfilter(G,C)`. At times, one also uses the 'convolution' which is mathematically by and large equivalent to the correlation. The intuition, however, is different: in correlation the mask entry $C(m,n)$ indicates how strongly a pixel of the input image, shifted by (m,n), influences the current pixel of the result image. In convolution one proceeds from the input image; the mask entry $C(m,n)$ indicates how strongly the current pixel of the input image influences one pixel of the result image that is shifted by (m,n) (cf. Fig. 12.1). The equation for the convolution operation is:

$$\begin{aligned} H &= G * C, \\ H(k,l) &= \sum_{m=-M}^{M} \sum_{n=-N}^{N} C(m,n)\, G(k-m, l-n). \end{aligned} \tag{12.5}$$

Convolution is transformed into a correlation if the mask is rotated by 180°, or if a point reflection is made at its central pixel. For point-symmetric masks (i.e., masks with the characteristic of $C(-m,-n) = C(m,n)$) correlation and convolution are the same.

Convolution and correlation are operations which map one image to another one. This mapping is *linear*, ie. it satisfies:

$$(F+G)*C = F*C+G*C;\ \ (\lambda F)*C=\lambda(F*C), \tag{12.6}$$
$$(F+G)\otimes C = F\otimes C+G\otimes C;\ \ (\lambda F)\otimes C=\lambda(F\otimes C). \tag{12.7}$$

There, $\lambda \in \mathbb{R}$ is a real number; addition of images and the multiplication with λ are performed pixel-wise. Therefore, if one wants to average, say, images (i.e., adding the gray-scale values pixel-wise and dividing by the total number of images), then it is irrelevant if one uses a filter operation before or after the averaging.

Specially for convolution two more relations hold:

Associativity: If G is an image and C, D are two masks, then let:

$$(G*C)*D = G*(C*D). \tag{12.8}$$

Instead of convolving an image successively with two masks, one can therefore convolve first the masks together and then use the combined mask for the image. The associated formula for the correlation is $(G\otimes C)\otimes D = G\otimes(C*D)$.

Commutativity: If one convolves an image with two masks C, D successively, then the order of those masks does not matter:

$$G*C*D = G*D*C. \tag{12.9}$$

The mask, like the image, is a matrix, even though usually much smaller than the image. Except for the boundary effects at places where the mask extends beyond the image, the commutativity therefore applied: $G*C=C*G$.

In contrast, correlation is not commutative. If one exchanges the order of image and mask, one obtains a point-symmetrically reflected result.

Convolution may yield negative image values or image values exceeding 255. By default, 8-bit images in MATLAB will be "truncated", e.g. pixel values in the interval from 0 to 255 will be rounded to integer numbers and values above 255 or below 0 are set to 255 or 0, respectively. In order to avoid the resulting errors, images should be converted to floating point numbers ("double"). For masks with purely positive entries the range of values from 0 to 255 will not be left as long as the sum of all entries is 1. In this case, though, the mask is not integer anymore; if it is applied on an 8-bit image, rounding errors will occur. This problem can also be solved by using floating point numbers from the beginning. Since the monitor provides only a fixed gray-value scale from black to white, one has to present the 0 with a medium gray at negative image values. Negative values appear darker then, positive ones brighter.

For color images the definition of convolution turns out to be more difficult. In the simplest case, one just uses the same mask separately for each of the three color channels and puts the three results together again into a RGB-image. In general, for every position of the mask one has 3×3 weights which reproduce the influences of all three color channels of the input image on each of the three color channels of the output image.

Example 1

Consider a one-dimensional "image" with pixels numbered $(-5,...,-2,-1,0,1,2,...,5)$ and intensities

$$G_o(i) = \begin{cases} 1 & \text{for} \quad i=0 \\ 0 & \text{for} \quad i\neq 0 \end{cases}. \tag{12.10}$$

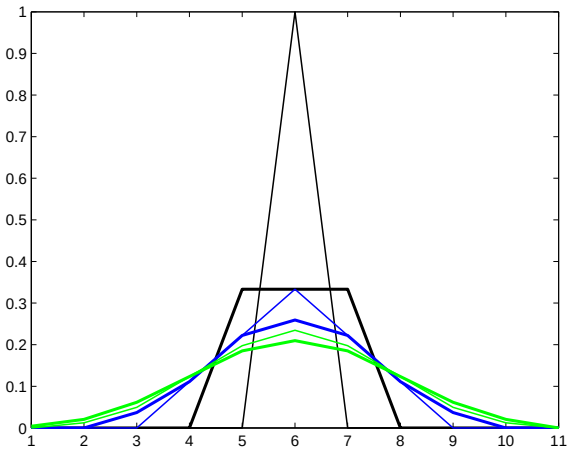


Figure 12.2: Iterated convolution of a point image with a box-shaped mask. The first function ($g_0(x)$, light black line) shows the original image, the second function($g_1(x)$, heavy black line) reproduces the mask. The further iterations are: $g_2(x)$, light blue line; $g_3(x)$, heavy blue line; $g_4(x)$, light green line, and $g_5(x)$, heavy green line.

As a mask, consider the 3-pixel mask $m = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. We will now denote by

$$G_k = G_{k-1} * m \tag{12.11}$$

the k-fold convolution of G_o with the mask m. The result is shown in the following table:

k	$g_k(-5)$					$g_k(0)$					$g_k(5)$	
0	0	0	0	0	0	1	0	0	0	0	0	
1	0	0	0	0	1	1	1	0	0	0	0	/3
2	0	0	0	1	2	3	2	1	0	0	0	/9
3	0	0	1	3	6	7	6	3	1	0	0	/27
4	0	1	4	10	16	19	16	10	4	1	0	/81
5	1	5	15	30	45	51	45	30	15	5	1	/243

As can be seen from Figure 12.2, the functions $g_k(x)$ approach a bell-shaped curve for large k. Indeed, the central limit theorem guarantees that this limit function is the Gaussian.

Example 2

The discrete, two-dimensional convolution operation introduced in Eq. 12.5 is a basic operation occuring in many fields of mathematics and quite different contexts. In its continuous version, it is the basis of linear systems theory discussed in Chapter 10. Here, we briefly discuss a further application of discrete convolution in probability theory.

When rolling a pair of dice, the probability of obtaining a particular number on each individual die is equally distributed, i.e.

$$\mathrm{P}(x_i = j) \;=\; \frac{1}{6} \quad \text{for} \quad i = 1, 2 \quad \text{and} \quad j = 1, ..., 6. \tag{12.12}$$

Let Y denote the sum of the points from the two dice. Clearly, Y can vary between 2, if both dice show "1", and 12, if both dice show "6". The probability of these results equals $\left(\frac{1}{6}\right)^2 = \frac{1}{36}$.

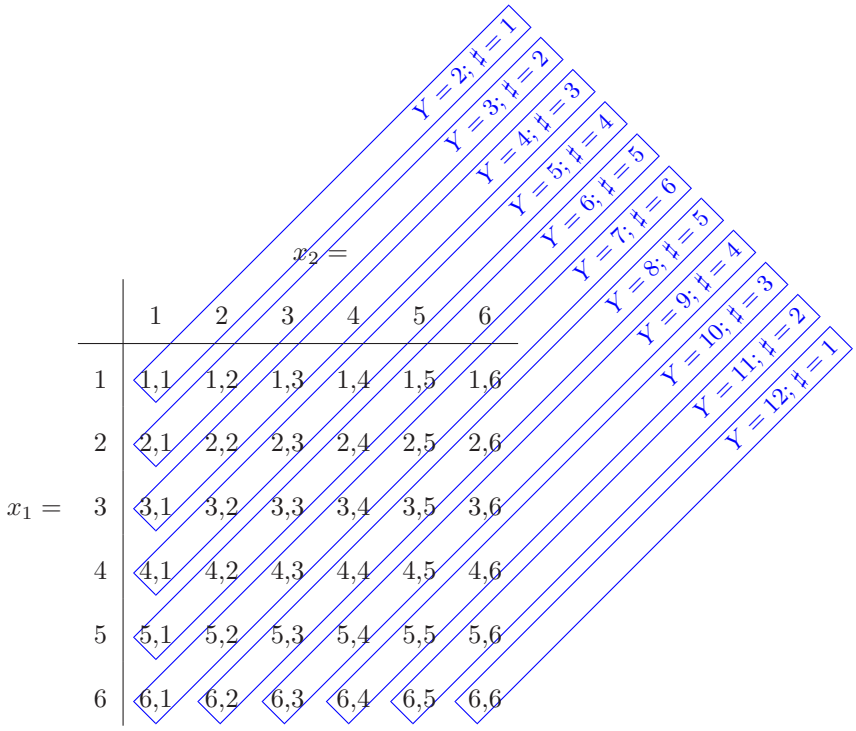


Figure 12.3: Convolution and the probability distribution of sums. The black table shows the possible outcomes of a pair of dice rolled together. The blue boxes group all outcomes yielding the same sum. Clearly, the sum 2 will occur less frequently than the sum 7, say, since six possible outcomes yield 7 while only one possible outcome yields 2.

Intermediate sums such as 5 can be obtained by a number of pairs, i.e., $(1,4), (2,3), (3,2), (4,1)$, and therefore have higher probabilities, in our example $\frac{4}{36}$.

In general, all possible outcomes yielding the sum i can be written as $(j, i-j)$, or more explicitly

$$\mathrm{P}(Y = i) = \sum_{j=\max(i-6,1)}^{\min(6,i-1)} \mathrm{P}(X_1 = j)\ \mathrm{P}(X_2 = i - j). \tag{12.13}$$

As depicted in Fig. 12.3, $\mathrm{P}(Y = i)$ takes a triangular shape, just as the convolution of the box function with itself (light blue line in Fig. 12.2). In general, the distribution density function of a sum of two random variables can be shown to equal the convolution of the distribution density functions of the two individual distribution functions.

12.3 Simple Edge Detection

Edges are places in the image where the gray-scale value changes abruptly or at least "swiftly". Edges are characterized by the following properties:

- **Orientation:** Locally, edges are always line segments; the image changes quickly perpendicular to the edge, but hardly or not at all in the direction of the edge. Small line segments that indicate place and the local direction of the edge are called "edge segments", their angle to the horizontal "orientation". It lies between $-90°$ and $90°$.

- **Polarity**: Two edges of the same orientation are said to have the same polarity if the brighter part of the image is on the same side of both of the segments. Orientation and polarity can be summarized to one angle between $-180°$ and $180°$.
- **Step- and Contour Edges**: Step edges are borders between regions of varying image intensity. They can be emphasized by antisymmetric masks of the form $(-0.5, 0, 0.5)$ or $(0.5, 0, -0.5)$, respectively. Contour edges consist of two close step edges of the same orientation and opposite polarity. They can be emphasized by symmetrical masks of the form $(-0.5, 1, -0.5)$ and $(0.5, -1, 0.5)$, respectively.
- **Resolution (scale)**: Edges appear on varying resolution levels which have to be processed with masks of different size.

In the output image of an edge detector, a pixel adopts the value 0 when no edge was detected, and the value 1 when one edge was detected. A basic and often applied edge detector is the Sobel-operator; it uses two oriented edge detectors for vertical and horizontal edges,

$$V = \begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \quad \text{and} \quad H = \begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array}.$$

The two resulting images are squared pixel by pixel then added up:

$$G_{out} = (G_{in} \otimes V)^2 + (G_{in} \otimes H)^2.$$

As edge elements we consider such pixels where G_{out} exceeds a given threshold value. The Sobel-operator evaluates the gradient or steepest uphill direction in the gray-scale value landscape; intuitively, it provides the square of the length of the gradient (vector of the partial derivations) in the gray-scale value range. It is non-linear.

12.4 Correlation and Matched Filters

We have already pointed out that the mask of a filter operation can be understood as small image. This small image, or template, is moved over the analyzed image. The result of the pixel-wise multiplication of image and mask with subsequent summation will be the bigger, the more similar image and mask locally are. If one wants to search, e.g. the letter "A" in a scanned text in a perpendicular orientation and in a font size of 9 pt, one can define a mask with the shape

$$A := \begin{array}{|c|c|c|c|c|c|c|c|c|}\hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ \hline 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ \hline 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ \hline \end{array} \tag{12.14}$$

and correlate the image with it. At those places where an "A" is printed white on black background, this mask will provide maximal outcome.

In order to observe this, we consider the number of the pixels of the mask A filled with the value 1. In the example of Eq. 12.14 those are $w_M = 40$. The remaining $b_M = 41$ pixels of the mask are black (value 0). If one puts this mask on a purely white image, meaning an area whose pixels all have the value 1, then the outcome of the filtering operation will be $w_M = 40$. Since the mask contains no negative weights, this is the maximal possible value. If one puts this mask now on an image with white and black pixels, then the result of the filtering H will adopt the maximal value $H = w_M$ as long as all black pixels of the image are aligned with black pixels of the mask. The "darkest" image, e.g. the image with the most black pixels that still supplies $H = w_M$, has got black pixels wherever the mask has got black pixels too, so it shows an "A" as well. We call the number of white pixels in the image w_I and consider the "matching strength" $m := H/w_I$. It assumes its maximal value when the image corresponds with the mask, since $H = w_M$ is then maximal and w_I will be small. If we now slide the mask over the image, then we can detect an "A" at those places where $m = H/w_B$ or more generally

$$m(k,l) = \frac{\sum_m \sum_n A(m,n)\ G(k+m, l+n)}{\sqrt{\sum_m \sum_n G^2(k+m, l+n)}} \tag{12.15}$$

has a maximum. If we denote by E a mask of the same size as A whose entries are all 1 and by G^2 the pixel-wise squaring operation, we can also write:

$$m = \frac{G \otimes A}{\sqrt{G^2 \otimes E}}. \tag{12.16}$$

There, the division is defined pointwise. If m exceeds a threshold, then this points to the presence of the letter A in the image.

The process also works for continuous gray-values. Mathematically, it is a consequence of the so-called Cauchy-Schwarz inequality.

12.5 Motion

We consider two images of a sequence, G_1 and G_2. Changes in the image can be detected by simply subtracting the images from each other. Such changes do not always result from motions, though, but may also be caused by changes in lighting, etc. Even when image differences are produced by motions, one cannot in general read from the difference image in which direction the motion took place. A simple motion detector which provides this information can be constructed as follows:

First, we consider a motion to the right by 3 pixels and down by 1 pixel. In this case, one will expect that in the moved area of the image the relation

$$G_2(i,j) = G_1(i-3, j-1) \tag{12.17}$$

will approximately hold. In order to verify this, we need a shift operation first. Image shifts can be realized as filtering. For the shift by the vector $(3,1)$ we need the following 7×3 mask:

$$S_{3,1} = \begin{array}{c} \\ \text{-1} \\ \text{0} \\ \text{1} \end{array} \begin{array}{c} \begin{array}{ccccccc} \text{-3} & \text{-2} & \text{-1} & \text{0} & \text{1} & \text{2} & \text{3} \end{array} \\ \begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array} \end{array} \tag{12.18}$$

One easily verifies the relation $G_2 = G_1 \otimes S_{3,1}$.

As evidence for the shift of the entire image by 3 pixels to the right and 1 pixel down we now regard the square difference between G_2 and $G_1 \otimes S_{3,1}$:

$$SSD_{3,1} := \|G_2 - G_1 \otimes S_{3,1}\|^2. \tag{12.19}$$

With the double bars $\|\cdot\|$ we denote the "norm" of an image (here of the difference image) where the image is now interpreted as the vector of all its pixels, $\|G\| := \sqrt{\sum_{i,j} G(i,j)^2}$. The abbreviation SSD stands for *sum of squared differences*. $SSD_{3,1}$ is therefore a number which becomes zero when G_2 and the shifted version of G_1 are exactly congruent, and which becomes the larger the more the images differ from each other.

In order to measure the motion, we have to carry out this operation for all image shifts (v_1, v_2) in a realistic range and then choose the shift (i.e. motion) that produces the smallest quadratic difference.

Motion detection is a well developed field in image processing where many variants exist. Here, just the basic idea of the so-called correlation procedure was outlined; more elaborate algorithms can be found in the referenced literature.

12.6 References and Further Reading

B. Jähne. *Digitale Bildverarbeitung und Bildgewinnung.* Springer Verlag, Berlin, Heidelberg, New York, 7. neubearbeitete Auflage, 2012.

B. D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th International Joint Conference on Artificial Intelligence* (IJCAI'81), pp 674-679, 1981.

H. A. Mallot. *Computational Vision. Information Processing in Perception and Visual Behavior.* The MIT Press, Cambridge, MA, 2000.

J. R. Parker. *Algorithms for Image Processing and Computer Vision.* John Wiley & Sons, 2nd Edition 2010.

A. Rosenfeld und A. C. Kak. *Digital Picture Processing, Vols. 1 und 2.* Academic Press, Orlando, Fla. and London, 2. Auflage, 1982.

E. Trucco und A. Verri. *Introductory techniques for 3–D computer vision.* Prentice Hall, 1998.