Advanced Mathematics for Engineers

Wolfgang Ertel

translated by Elias Drotleff and Richard Cubek

October 1, 2012

Hochschule
Ravensburg-Weingarten
University of Applied Sciences

Preface

Since 2008 this mathematics lecture is offered for the master courses computer science, mechatronics and electrical engineering. After a repetition of basic linear algebra, computer algebra and calculus, we will treat numerical calculus, statistics and function approximation, which are the most important mathematics basic topics for engineers.
We also provide an introduction to Computer Algebra. Mathematica, Matlab and Octave are powerful tools for the Exercises. Event though we favour the open source tool Octave, the student is free to choose either one of the three.
We are looking forward to work with interesting semesters with many motivated and eager students who want to climb up the steep, high and fascinating mountain of engineering mathematics together with us. I assure you that we will do our best to guide you through the sometimes wild, rough and challenging nature of mathematics. I also assure you that all your efforts and your endurance in working on the exercises during nights and weekends will pay off as good marks and most importantly as a lot of fun.
Even though we repeat some undergraduate linear algebra and calculus, the failure rate in the exams is very high, in particular among the foreign students. As a consequence, we strongly recommend all our students to repeat undergraduate linear algebra such as operation on matrices like solution of linear systems, singularity of matrices, inversion, eigenvalue problems, row-, column- and nullspaces. You also should bring decent knowledge of one-dimensional and multidimensional calculus, e.g. differentiation and integration in one and many variables, convergence of sequences and series and finding extrema with constraints of multivariate functions. And basic statistics is also required. To summarize: **If you are not able to solve problems (not only know the terms) in these fields, you have very little chances to successfully finish this course.**

History of this Course

The first version of this script was created in the winter semester 95/96. I had included in this lecture only Numerics, although I wanted to cover initially Discrete Mathematics too, which is very important for computer scientists. If you want to cover both in a lecture of three semester week hours, it can happen only superficially. Therefore I decided to focus like my colleagues on Numerics. Only then it is possible to impart profound knowledge.
From Numerical Calculus besides the basics, systems of linear equations, various interpolation methods, function approximation, and the solution of nonlinear equations will be presented. An excursion into applied research follows, where e.g. in the field of benchmarking of Microprocessors, mathematics (functional equations) is influencing directly the practice of computer scientists.
In summer 1998 a chapter about Statistics was added, because of the weak coverage at our University till then. In the winter semester 1999/2000, the layout and structure were improved, as well some mistakes have been removed.
In the context of changes in the summer semester 2002 in the curriculum of Applied Computer science, statistics was shifted, because of the general relevance for all students, into the lecture Mathematics 2. Instead of Statistics, contents should be included, which are specifically relevant for computer scientists. The generation and verification of random numbers is an important topic, which is finally also covered.
Since summer 2008, this lecture is only offered to Master (Computer Science) students. Therefore the chapter about random numbers was extended. Maybe other contents will be included in the lecture. For some topics original literature will be handed out, then student

have to prepare the material by themselves.
To the winter semester 2010/11 the lecture has now been completely revised, restructured and some important sections added such as radial basis functions, Gaussian processes and statistics and probability. These changes become necessary with the step from Diploma to Master. I want to thank Markus Schneider and Haitham Bou Ammar who helped me improve the lecture.
To the winter semester 2010/11 the precourse will be integrated in the lecture in order to give the students more time to work on the exercises. Thus, the volume of lecture grows from 6 SWS to 8 SWS and we will now split it into two lectures of 4 SWS each.
In the winter semester 2012/13 we go back to a one semester schedule with 6 hours per week for computer science and mechatronics students. Electrical engineering students will only go for four hours, covering chapters one to six.

Wolfgang Ertel

Contents

Chapter 1

Linear Algebra

1.1 Video Lectures

We use the excellent video lectures from G. Strang, the author of [1], available from: http://ocw.mit.edu/courses/mathematics/18-06-linear-algebra-spring-2010. In particular we show the following lectures:

Lec #	Topics
1	The geometry of linear equations (lecture 01)
2	Transposes, Permutations, Spaces R^n (lecture 05)
3	Column Space and Nullspace (lecture 06)
4	Solving $Ax = 0$: Pivot Variables, Special Solutions (lecture 07)
5	Independence, Basis, and Dimension (lecture 09)
6	The Four Fundamental Subspaces (lecture 10)
7	Orthogonal Vectors and Subspaces (lecture 14)
8	Properties of Determinants (lecture 18)
9	Determinant Formulas and Cofactors (lecture 19)
10	Cramer's rule, inverse matrix, and volume (lecture 20)
11	Eigenvalues and Eigenvectors (lecture 21)
12	Symmetric Matrices and Positive Definiteness (lecture 25)
13	Linear Transformations and Their Matrices (lecture 30)

1.2 Exercises

Exercise 1.1 Solve the nonsingular triangular system

$$u + v + w = b_1 \tag{1.1}$$
$$v + w = b_2 \tag{1.2}$$
$$w = b_3 \tag{1.3}$$

Show that your solution gives a combination of the columns that equals the column on the right.

Exercise 1.2 Explain why the system

$$u + v + w = 2 \tag{1.4}$$
$$u + 2v + 3w = 1 \tag{1.5}$$
$$v + 2w = 0 \tag{1.6}$$

is singular, by finding a combination of the three equations that adds up to $0 = 1$. What value should replace the last zero on the right side, to allow the equations to have solutions, and what is one of the solutions?

Inverses and Transposes

Exercise 1.3 Which properties of a matrix A are preserved by its inverse (assuming A^{-1} exists)?

(1) A is triangular

(2) A is symmetric

(3) A is tridiagonal

(4) all entries are whole numbers

(5) all entries are fractions (including whole numbers like $\frac{3}{1}$)

Exercise 1.4

a) How many entries can be chosen independently, in a symmetric matrix of order n?

b) How many entries can be chosen independently, in a skew-symmetric matrix of order n?

Permutations and Elimination

Exercise 1.5

a) Find a square 3×3 matrix P, that multiplied from left to any $3 \times m$ matrix A exchanges rows 1 and 2.

b) Find a square $n \times n$ matrix P, that multiplied from left to any $n \times m$ matrix A exchanges rows i and j.

Exercise 1.6 A permutation is a bijective mapping from a finite set onto itself. Applied to vectors of length n, a permutation arbitrarily changes the order of the vector components. The word "ANGSTBUDE" is a permutation of "BUNDESTAG". An example of a permutation on vectors of length 5 can be described by

$$(3, 2, 1, 5, 4).$$

This means component 3 moves to position 1, component 2 stays where it was, component 1 moves to position 3, component 5 moves to position 4 and component 4 moves to position 5.

a) Give a 5×5 matrix P that implements this permutation.

b) How can we come from a permutation matrix to its inverse?

Exercise 1.7

a) Find a 3×3 matrix E, that multiplied from left to any $3 \times m$ matrix A adds 5 times row 2 to row 1.

b) Describe a $n \times n$ matrix E, that multiplied from left to any $n \times m$ matrix A adds k times row i to row j.

c) Based on the above answers, prove that the elimination process of a matrix can be realized by successive multiplication with matrices from left.

Column Spaces and NullSpaces

Exercise 1.8 Which of the following subsets of R^3 are actually subspaces?

a) The plane of vectors with first component $b_1 = 0$.

b) The plane of vectors b with $b_1 = 1$.

c) The vectors b with $b_1 b_2 = 0$ (this is the union of two subspaces, the plane $b_1 = 0$ and the plane $b_2 = 0$).

d) The solitary vector $b = (0, 0, 0)$.

e) All combinations of two given vectors $x = (1, 1, 0)$ and $y = (2, 0, 1)$.

f) The vectors (b_1, b_2, b_3) that satisfy $b_3 - b_2 + 3b_1 = 0$.

Exercise 1.9 Let P be the plane in 3-space with equation $x + 2y + z = 6$. What is the equation of the plane P_0 through the origin parallel to P? Are P and P_0 subspaces of R^3?

Exercise 1.10 Which descriptions are correct? The solutions x of

$$Ax = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{1.7}$$

form a plane, line, point, subspace, nullspace of A, column space of A.

$Ax = 0$ and Pivot Variables

Exercise 1.11 For the matrix

$$A = \begin{bmatrix} 0 & 1 & 4 & 0 \\ 0 & 2 & 8 & 0 \end{bmatrix} \tag{1.8}$$

determine the echelon form U, the basic variables, the free variables, and the general solution to $Ax = 0$. Then apply elimination to $Ax = b$, with components b_1 and b_2 on the right side; find the conditions for $Ax = b$ to be consistent (that is, to have a solution) and find the general solution in the same form as Equation (3). What is the rank of A?

Exercise 1.12 Write the general solution to

$$\begin{bmatrix} 1 & 2 & 2 \\ 2 & 4 & 5 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \end{bmatrix} \tag{1.9}$$

as the sum of a particular solution to $Ax = b$ and the general solution to $Ax = 0$, as in (3).

Exercise 1.13 Find the value of c which makes it possible to solve

$$u + v + 2w = 2 \tag{1.10}$$
$$2u + 3v - w = 5 \tag{1.11}$$
$$3u + 4v + w = c \tag{1.12}$$

Solving $Ax = b$

Exercise 1.14 Is it true that if v_1, v_2, v_3 are linearly independent, that also the vectors $w_1 = v_1 + v_2$, $w_2 = v_1 + v_3$, $w_3 = v_2 + v_3$ are linearly independent? (Hint: Assume some combination $c_1w_1 + c_2w_2 + c_3w_3 = 0$, and find which c_i are possible.)

Exercise 1.15 Find a counterexample to the following statement: If v_1, v_2, v_3, v_4 is a basis for the vector space R^4, and if W is a subspace, then some subset of the v's is a basis for W.

Exercise 1.16 Suppose V is known to have dimension k. Prove that

a) any k independent vectors in V form a basis;

b) any k vectors that span V form a basis.

In other words, if the number of vectors is known to be right, either of the two properties of a basis implies the other.

Exercise 1.17 Prove that if V and W are three-dimensional subspaces of R^5, then V and W must have a nonzero vector in common. Hint: Start with bases of the two subspaces, making six vectors in all.

The Four Fundamental Subspaces

Exercise 1.18 Find the dimension and construct a basis for the four subspaces associated with each of the matrices

$$A = \begin{bmatrix} 0 & 1 & 4 & 0 \\ 0 & 2 & 8 & 0 \end{bmatrix} \quad \textit{and} \quad U = \begin{bmatrix} 0 & 1 & 4 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{1.13}$$

Exercise 1.19 If the product of two matrices is the zero matrix, $AB = 0$, show that the column space of B is contained in the nullspace of A. (Also the row space of A is the left nullspace of B, since each row of A multiplies B to give a zero row.)

Exercise 1.20 Explain why $Ax = b$ is solvable if and only if rank A = rank A', where A' is formed from A by adding b as an extra column. Hint: The rank is the dimension of the column space; when does adding an extra column leave the dimension unchanged?

Exercise 1.21 Suppose A is an m by n matrix of rank r. Under what conditions on those numbers does

a) A have a two-sided inverse: $AA^{-1} = A^{-1}A = I$?

b) $Ax = b$ have infinitely many solutions for every b?

Exercise 1.22 If $Ax = 0$ has a nonzero solution, show that $A^Ty = f$ fails to be solvable for some right sides f. Construct an example of A and f.

Orthogonality

Exercise 1.23 In R^3 find all vectors that are orthogonal to (1, 1, 1) and (1, -1, 0). Produce from these vectors a mutually orthogonal system of unit vectors (an orthogonal system) in

R^3.

Exercise 1.24 Show that $x - y$ is orthogonal to $x + y$ if and only if $\|x\| = \|y\|$.

Exercise 1.25 Let P be the plane (not a subspace) in 3-space with equation $x+2y-z = 6$. Find the equation of a plane P' parallel to P but going through the origin. Find also a vector perpendicular to those planes. What matrix has the plane P' as its nullspace, and what matrix hast P' as its row space?

Projections

Exercise 1.26 Suppose A is the 4×4 identity matrix with its last column removed. A is 4×3. Project $b = (1, 2, 3, 4)$ onto the column space of A. What shape is the projection matrix P and what is P?

Determinants

Exercise 1.27 How are $\det(2A)$, $\det(-A)$, and $\det(A^2)$ related to $\det A$, when A is n by n?

Exercise 1.28 Find the determinants of:

a) a rank one matrix

$$A = \begin{bmatrix} 1 \\ 4 \\ 2 \end{bmatrix} \begin{bmatrix} 2 & -1 & 2 \end{bmatrix} \tag{1.14}$$

b) the upper triangular matrix

$$U = \begin{bmatrix} 4 & 4 & 8 & 8 \\ 0 & 1 & 2 & 2 \\ 0 & 0 & 2 & 6 \\ 0 & 0 & 0 & 2 \end{bmatrix} \tag{1.15}$$

c) the lower triangular matrix U^T;

d) the inverse matrix U^{-1};

e) the "reverse-triangular" matrix that results from row exchanges,

$$M = \begin{bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 2 & 6 \\ 0 & 1 & 2 & 2 \\ 4 & 4 & 8 & 8 \end{bmatrix} \tag{1.16}$$

Exercise 1.29 If every row of A adds to zero prove that $\det A = 0$. If every row adds to 1 prove that $\det(A - I) = 0$. Show by example that this does not imply $\det A = 1$.

Properties of Determinants

Exercise 1.30 Suppose A_n is the n by n tridiagonal matrix with 1's everywhere on the three diagonals:

$$A_1 = \begin{bmatrix}1\end{bmatrix}, \quad A_2 = \begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}, \quad A_3 = \begin{bmatrix}1 & 1 & 0\\ 1 & 1 & 1\\ 0 & 1 & 1\end{bmatrix}, ... \tag{1.17}$$

Let D_n be the determinant of A_n; we want to find it.

a) Expand in cofactors along the first row of A_n to show that $D_n = D_{n-1} - D_{n-2}$.

b) Starting from $D_1 = 1$ and $D_2 = 0$ find $D_3, D_4, ..., D_8$. By noticing how these numbers cycle around (with what period?) find D_{1000}.

Exercise 1.31 Explain why a 5 by 5 matrix with a 3 by 3 zero submatrix is sure to be a singular (regardless of the 16 nonzeros marked by x's):

$$\text{the determinant of } A = \begin{bmatrix}x & x & x & x & x\\ x & x & x & x & x\\ 0 & 0 & 0 & x & x\\ 0 & 0 & 0 & x & x\\ 0 & 0 & 0 & x & x\end{bmatrix} \text{ is zero.} \tag{1.18}$$

Exercise 1.32 If A is m by n and B is n by m, show that

$$det = \begin{bmatrix}0 & A\\ -B & I\end{bmatrix} = \det AB. \quad \left(\text{Hint: Postmultiply by } \begin{bmatrix}I & 0\\ B & I\end{bmatrix}.\right) \tag{1.19}$$

Do an example with $m < n$ and an example with $m > n$. Why does the second example have det $AB = 0$?

Cramers' rule

Exercise 1.33 The determinant is a linear function of the column 1. It is zero if two columns are equal. When $b = Ax = x_1a_1 + x_2a_2 + x_3a_3$ goes into the first column of A, then the determinant of this matrix B_1 is

$$|b \quad a_2 \quad a_3| = |x_1a_1 + x_2a_2 + x_3a_3 \quad a_2 \quad a_3| = x_1|a_1 \quad a_2 \quad a_3| = x_1 \det A$$

a) What formula for x_1 comes from left side = right side?

b) What steps lead to the middle equation?

Eigenvalues and Eigenvectors

Exercise 1.34 Suppose that λ is an eigenvalue of A, and x is its eigenvector: $Ax = \lambda x$.

a) Show that this same x is an eigenvector of $B = A - 7I$, and find the eigenvalue.

b) Assuming $\lambda \neq 0$, show that x is also an eigenvector of A^{-1} and find the eigenvalue.

Exercise 1.35 Show that the determinant equals the product of the eigenvalues by imagining that the characteristic polynomial is factored into

$$\det(A - \lambda I) = (\lambda_1 - \lambda)(\lambda_2 - \lambda)\cdots(\lambda_n - \lambda) \tag{1.20}$$

and making a clever choice of λ.

Exercise 1.36 Show that the trace equals the sum of the eigenvalues, in two steps. First, find the coefficient of $(-\lambda)^{n-1}$ on the right side of (15). Next, look for all the terms in

$$\det(A - \lambda I) = \det \begin{bmatrix} a_{11} - \lambda & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} - \lambda & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} - \lambda \end{bmatrix} \tag{1.21}$$

which involve $(-\lambda)^{n-1}$. Explain why they all come from the product down the main diagonal, and find the coefficient of $(-\lambda)^{n-1}$ on the left side of (15). Compare.

Diagonalization of Matrices

Exercise 1.37 Factor the following matrices into $S\Lambda S^{-1}$:

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad \textit{and} \quad A = \begin{bmatrix} 2 & 1 \\ 0 & 0 \end{bmatrix}. \tag{1.22}$$

Exercise 1.38 Suppose $A = uv^T$ is a column times a row (a rank-one matrix).

a) By multiplying A times u show that u is an eigenvector. What is λ?

b) What are the other eigenvalues (and why)?

c) Compute $\text{trace}(A) = v^T u$ in two ways, from the sum on the diagonal and the sum of λ's.

Exercise 1.39 If A is diagonalizable, show that the determinant of $A = S\Lambda S^{-1}$ is the product of the eigenvalues.

Symmetric and Positive Semi-Definite Matrices

Exercise 1.40 If $A = Q\Lambda Q^T$ is symmetric positive definite, then $R = Q\sqrt{\Lambda}Q^T$ is its symmetric positive definite square root. Why does R have real eigenvalues? Compute R and verify $R^2 = A$ for

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad \textit{and} \quad A = \begin{bmatrix} 10 & -6 \\ -6 & 10 \end{bmatrix}. \tag{1.23}$$

Exercise 1.41 If A is symmetric positive definite and C is nonsingular, prove that $B = C^T AC$ is also symmetric positive definite.

Exercise 1.42 If A is positive definite and a_{11} is increased, prove from cofactors that the determinant is increased. Show by example that this can fail if A is indefinite.

Linear Transformation

Exercise 1.43 Suppose a linear mapping T transforms (1, 1) to (2, 2) and (2, 0) to (0, 0). Find $T(v)$:

$$(a) \quad v = (2, 2) \qquad (b) \quad v = (3, 1) \qquad (c) \quad v = (-1, 1) \qquad (d) \quad v = (a, b)$$

Exercise 1.44 Suppose T is reflection across the 45^o line, and S is reflection across the y axis. If $v = (2, 1)$ then $T(v) = (1, 2)$. Find $S(T(v))$ and $T(S(v))$. This shows that generally $ST \neq TS$.

Exercise 1.45 Suppose we have two bases $v_1, ..., v_n$ and $w_1, ..., w_n$ for R^n. If a vector has coefficients b_i in one basis and c_i in the other basis, what is the change of basis matrix in $b = Mc$? Start from

$$b_1 v_1 + ... + b_n v_n = Vb = c_1 w_1 + ... + c_n w_n = Wc. \tag{1.24}$$

Your answer represents $T(v) = v$ with input basis of v's and output basis of w's. Because of different bases, the matrix is not I.

Chapter 2

Computer Algebra

Definition 2.1 Computer Algebra = Symbol Processing + Numerics + Graphics

Definition 2.2 Symbol Processing is calculating with symbols (variables, constants, function symbols), as in Mathematics lectures.

Advantages of Symbol Processing:

- often considerably less computational effort compared to numerics.
- symbolic results (for further calculations), proofs in the strict manner possible.

Disadvantages of Symbol Processing:

- often there is no symbolic (closed form) solution, then Numerics will be applied, e.g.:
  - Calculation of Antiderivatives
  - Solving Nonlinear Equations like: $(e^x = \ sinx)$

Example 2.1

1. symbolic:

$$\lim_{x\to\infty}\left(\frac{\ln x}{x+1}\right)' =? \quad (asymptotic \quad behavior)$$

$$\left(\frac{\ln x}{x+1}\right)' = \frac{\frac{1}{x}(x+1)-\ln x}{(x+1)^2} = \frac{1}{(x+1)x} - \frac{\ln x}{(x+1)^2}$$

$$\underline{x\to\infty}: \left(\frac{\ln x}{x+1}\right)' \to \frac{1}{x^2} - \frac{\ln x}{x^2} \to \frac{\ln x}{x^2} \to 0$$

2. numeric:

$$\lim_{x\to\infty} f'(x) =?$$

Example 2.2 Numerical solution of $x^2 = 5$

$$x^2 = 5, \quad x = \frac{5}{x}, \quad 2x = x + \frac{5}{x}$$

$$x = \frac{1}{2}\left(x + \frac{5}{x}\right)$$

iteration:

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{5}{x_n}\right)$$

n	x_n
0	2 ← Startwert
1	2.25
2	2.236111
3	2.23606798
4	2.23606798

$$\Rightarrow \sqrt{5} = 2.23606798 \pm 10^{-8}$$

(approximate solution)

2.1 Symbol Processing on the Computer

Example 2.3 Symbolic Computing with natural numbers:
Calculation rules, i.e. Axioms necessary. $\Rightarrow$ Peano Axioms e.g.:

$$\forall x, y, z : x + y = y + x \tag{2.1}$$
$$x + 0 = x \tag{2.2}$$
$$(x + y) + z = x + (y + z) \tag{2.3}$$

Out of these rules, e.g. $0 + x = x$ can be deduced:

$$0 + x \underbrace{=}_{(2.1)} x + 0 \underbrace{=}_{(2.2)} x$$

Implementation of symbol processing on the computer by "Term Rewriting".

Example 2.4 (Real Numbers) Chain Rule for Differentiation:

$$[f(g(x))]' \Rightarrow f'(g(x))g'(x)$$

$$\sin(\ln x + 2)' = \cos(\ln x + 2)\frac{1}{x}$$

Computer: (Pattern matching)

$$\sin(Plus(\ln x, 2))' = \cos(Plus(\ln x, 2))Plus'(\ln x, 2)$$

$$\sin(Plus(\ln x, 2))' = \cos(Plus(\ln x, 2))Plus(\ln' x, 2')$$

$$\sin(Plus(\ln x, 2))' = \cos(Plus(\ln x, 2))Plus\left(\frac{1}{x}, 0\right)$$

$$\sin(Plus(\ln x, 2))' = \cos(Plus(\ln x, 2))\frac{1}{x}$$

$$\sin(Plus(\ln x, 2))' = \frac{\cos(\ln x + 2)}{x}$$

Effective systems:

- Mathematica (S. Wolfram & Co.)
- Maple (ETH Zurich + Univ. Waterloo, Kanada)

2.2 Short Introduction to Mathematica

Resources:

- Library: Mathematica Handbook (Wolfram)
- Mathematica Documentation Online: http://reference.wolfram.com
- http://www.hs-weingarten.de/~ertel/vorlesungen/mae/links.html

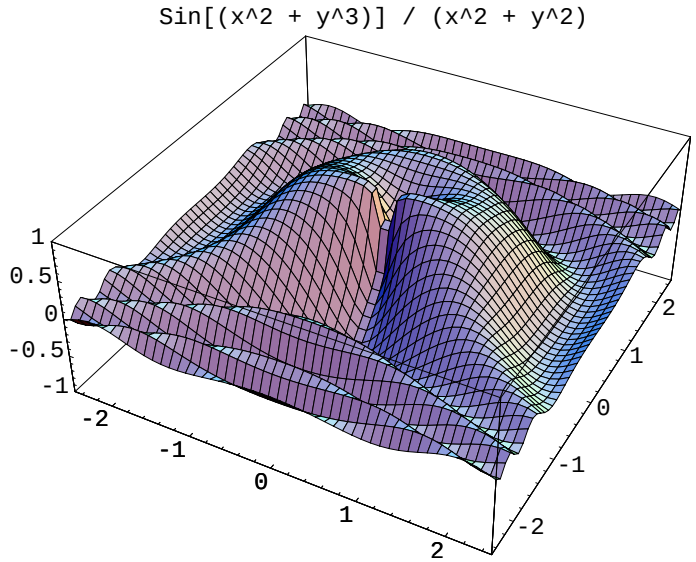
2.2.0.1 Some examples as jump start

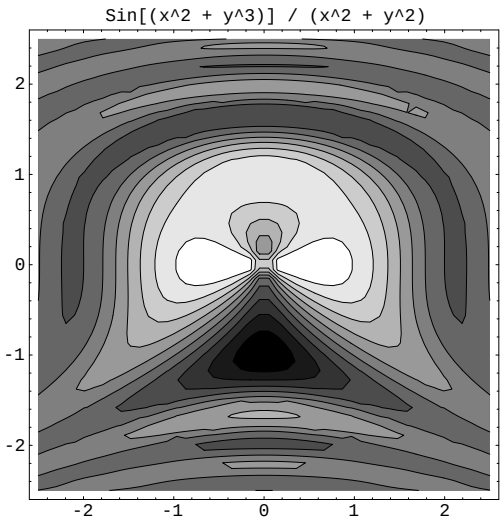
```
In[1]:= 3 + 2^3

Out[1]= 11

In[2]:= Sqrt[10]

Out[2]= Sqrt[10]

In[3]:= N[Sqrt[10]]

Out[3]= 3.16228

In[4]:= N[Sqrt[10],60]

Out[4]= 3.162277660168379331998893544432718533719555139325216826857 5

In[5]:= Integrate[x^2 Sin[x]^2, x]

           3                                    2
        4 x  - 6 x Cos[2 x] + 3 Sin[2 x] - 6 x  Sin[2 x]
Out[5]= ------------------------------------------------
                              24

In[7]:= D[%, x]

            2       2
        12 x  - 12 x  Cos[2 x]
Out[7]= ----------------------
                  24
```

```
In[8]:= Simplify[%]

         2      2
Out[8]= x  Sin[x]

In[9]:= Series[Exp[x], {x,0,6}]

                2    3    4    5     6
               x    x    x    x     x         7
Out[9]= 1 + x + -- + -- + -- + --- + --- + O[x]
               2    6    24   120   720

In[10]:= Expand[(x + 2)^3 + ((x - 5)^2 (x + y)^2)^3]

                       2    3          6          7         8         9
Out[10]= 8 + 12 x + 6 x  + x  + 15625 x  - 18750 x  + 9375 x  - 2500 x  +

          10       11    12          5             6            7
>    375 x   - 30 x   + x   + 93750 x  y - 112500 x  y + 56250 x  y -

           8           9          10        11             4  2
>    15000 x  y + 2250 x  y - 180 x   y + 6 x   y + 234375 x  y  -

            5  2          6  2          7  2         8  2        9  2
>    281250 x  y  + 140625 x  y  - 37500 x  y  + 5625 x  y  - 450 x  y  +

         10  2          3  3          4  3          5  3         6  3
>    15 x   y  + 312500 x  y  - 375000 x  y  + 187500 x  y  - 50000 x  y  +

          7  3        8  3       9  3          2  4          3  4
>    7500 x  y  - 600 x  y  + 20 x  y  + 234375 x  y  - 281250 x  y  +

            4  4         5  4        6  4        7  4       8  4
>    140625 x  y  - 37500 x  y  + 5625 x  y  - 450 x  y  + 15 x  y  +

             5          2  5         3  5         4  5        5  5
>    93750 x y  - 112500 x  y  + 56250 x  y  - 15000 x  y  + 2250 x  y  -

          6  5      7  5         6           6        2  6        3  6
>    180 x  y  + 6 x  y  + 15625 y  - 18750 x y  + 9375 x  y  - 2500 x  y  +

          4  6       5  6    6  6
>    375 x  y  - 30 x  y  + x  y

In[11]:= Factor[%]

                      2       3    4               2       3        2
Out[11]= (2 + x + 25 x  - 10 x  + x  + 50 x y - 20 x  y + 2 x  y + 25 y  -

             2    2  2                2      3       4       5       6
>      10 x y  + x  y ) (4 + 4 x - 49 x  - 5 x  + 633 x  - 501 x  + 150 x  -

           7    8                2          3          4         5
>      20 x  + x  - 100 x y - 10 x  y + 2516 x  y - 2002 x  y + 600 x  y -

           6       7        2       2        2  2        3  2
>      80 x  y + 4 x  y - 50 y  - 5 x y  + 3758 x  y  - 3001 x  y  +

            4  2       5  2     6  2          3        2  3       3  3
```

```
>      900 x  y  - 120 x  y  + 6 x  y  + 2500 x y  - 2000 x  y  + 600 x  y  -

          4  3      5  3       4          4       2  4       3  4    4  4
>      80 x  y  + 4 x  y  + 625 y  - 500 x y  + 150 x  y  - 20 x  y  + x  y )
```

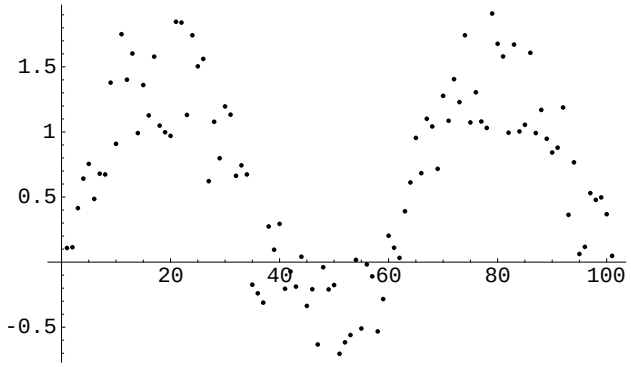
```
In[12]:= InputForm[%7]

Out[12]//InputForm= (12*x^2 - 12*x^2*Cos[2*x])/24

In[20]:= Plot[Sin[1/x], {x,0.01,Pi}]

Out[20]= -Graphics-

In[42]:= Plot3D[x^2 + y^2, {x,-1,1}, {y,0,1}]

Out[42]= -SurfaceGraphics-

In[43]:= f[x_,y_] := Sin[(x^2 + y^3)] / (x^2 + y^2)

In[44]:= f[2,3]

         Sin[31]
Out[44]= -------
           13

In[45]:= ContourPlot[x^2 + y^2, {x,-1,1}, {y,-1,1}]

Out[45]= -SurfaceGraphics-

In[46]:= Plot3D[f[x,y], {x,-Pi,Pi}, {y,-Pi,Pi}, PlotPoints -> 30,
         PlotLabel -> "Sin[(x^2 + y^3)] / (x^2 + y^2)", PlotRange -> {-1,1}]

Out[46]= -SurfaceGraphics-
```






```
In[47]:= ContourPlot[f[x,y], {x,-2,2}, {y,-2,2}, PlotPoints -> 30,
         ContourSmoothing -> True, ContourShading -> False,
         PlotLabel -> "Sin[(x^2 + y^3)] / (x^2 + y^2)"]

Out[47]= -ContourGraphics-
```

```
In[52]:= Table[x^2, {x, 1, 10}]

Out[52]= {1, 4, 9, 16, 25, 36, 49, 64, 81, 100}

In[53]:= Table[{n, n^2}, {n, 2, 20}]

Out[53]= {{2, 4}, {3, 9}, {4, 16}, {5, 25}, {6, 36}, {7, 49}, {8, 64},
>    {9, 81}, {10, 100}, {11, 121}, {12, 144}, {13, 169}, {14, 196},
>    {15, 225}, {16, 256}, {17, 289}, {18, 324}, {19, 361}, {20, 400}}

In[54]:= Transpose[%]

Out[54]= {{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
>     20}, {4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 256,
>     289, 324, 361, 400}}

In[60]:= ListPlot[Table[Random[]+Sin[x/10], {x,0,100}]]

Out[60]= -Graphics-
```




```
In[61]:= x = Table[i, {i,1,6}]

Out[61]= {1, 2, 3, 4, 5, 6}

In[62]:= A = Table[i*j, {i,1,5}, {j,1,6}]

Out[62]= {{1, 2, 3, 4, 5, 6}, {2, 4, 6, 8, 10, 12}, {3, 6, 9, 12, 15, 18},
>    {4, 8, 12, 16, 20, 24}, {5, 10, 15, 20, 25, 30}}

In[63]:= A.x

Out[63]= {91, 182, 273, 364, 455}

In[64]:= x.x

Out[64]= 91

In[71]:= B = A.Transpose[A]

Out[71]= {{91, 182, 273, 364, 455}, {182, 364, 546, 728, 910},
>    {273, 546, 819, 1092, 1365}, {364, 728, 1092, 1456, 1820},
>    {455, 910, 1365, 1820, 2275}}

In[72]:= B - IdentityMatrix[5]

Out[72]= {{90, 182, 273, 364, 455}, {182, 363, 546, 728, 910},
```

```
>    {273, 546, 818, 1092, 1365}, {364, 728, 1092, 1455, 1820},
>    {455, 910, 1365, 1820, 2274}}
```

`%`	last command
`%n`	nth last command
`?f`	help for function f
`??f`	more help for f
`f[x_,y_] := x^2 * Cos[y]`	define function $f(x, y)$
`a = 5`	assign a constant to variable a
`f = x^2 * Cos[y]`	assign an expression to variable f
(f is only a placeholder for the expression, not a function!)	
`D[f[x,y],x]`	derivative of f with respect to x
`Integrate[f[x,y],y]`	antiderivative of f with respect to x
`Simplify[expr]`	simplifies an expression
`Expand[expr]`	expand an expression
`Solve[f[x]==g[x]]`	solves an equation
`^C`	cancel
`InputForm[Expr]`	converts into mathematica input form
`TeXForm[Expr]`	converts into the LaTeXform
`FortranForm[Expr]`	converts into the Fortran form
`CForm[Expr]`	converts into the C form
`ReadList["daten.dat", {Number, Number}]`	reads 2-column table from file
`Table[f[n], {n, n_min, n_max}]`	generates a list $f(n_{min}), \ldots, f(n_{max})$
`Plot[f[x],{x,x_min,x_max}]`	generates a plot of f
`ListPlot[Liste]`	plots a list
`Plot3D[f[x,y],{x,x_min,x_max},{y,y_min,y_max}]`	generates a three-dim. plot of f
`ContourPlot[f[x,y],{x,x_min,x_max},{y,y_min,y_max}]`	generates a contour plot of f
`Display["Dateiname",%,"EPS"]`	write to the file in PostScript format

Table 2.2: Mathematica – some inportant commands

Example 2.5 (Calculation of Square Roots)

```
(*********** square root iterative **************)
sqrt[a_,genauigk_] := Module[{x, xn, delta, n},
        For[{delta=9999999; n = 1; x=a}, delta > 10^(-accuracy), n++,
          xn = x;
          x = 1/2(x + a/x);
          delta = Abs[x - xn];
          Print["n = ", n, "  x = ", N[x,2*accuracy], "  delta = ", N[delta]];
        ];
        N[x,genauigk]
]
sqrt::usage = "sqrt[a,n] computes the square root of a to n digits."

Table[sqrt[i,10], {i,1,20}]
```

```
(********** square root recursive **************)
x[n_,a_] := 1/2 (x[n-1,a] + a/x[n-1,a])
x[1,a_] := a
```

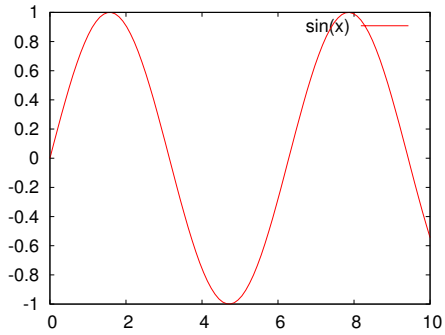
2.3 Gnuplot, a professional Plotting Software

Gnuplot is a powerful plotting programm with a command line interface and a batch interface. Online documentation can be found on www.gnuplot.info.

On the command line we can input

```
plot [0:10] sin(x)
```

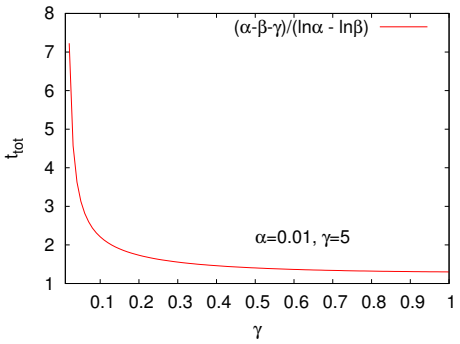
to obtain the graph



Almost arbitrary customization of plots is possible via the batch interface. A simple batch file may contain the lines

```
set terminal postscript eps color enhanced 26
set label "{/Symbol a}=0.01, {/Symbol g}=5" at 0.5,2.2
set output "bucket3.eps"
plot [b=0.01:1] a=0.01, c= 5, (a-b-c)/(log(a) - log(b)) \
title "({/Symbol a}-{/Symbol b}-{/Symbol g})/(ln{/Symbol a} - ln{/Symbol b})"
```

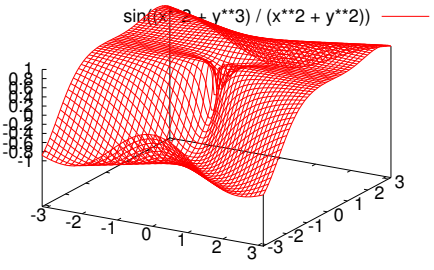
producing a EPS file with the graph



3-dimensional plotting is also possible, e.g. with the commands

```
set isosamples 50
splot [-pi:pi][-pi:pi] sin((x**2 + y**3) / (x**2
+ y**2))
```

which produces the graph



2.4 Short Introduction to MATLAB

Effective systems:

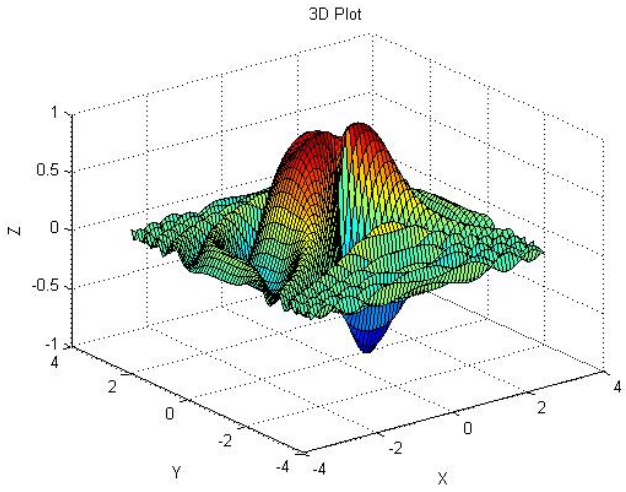
- MATLAB & SIMULINK (MathWorks)

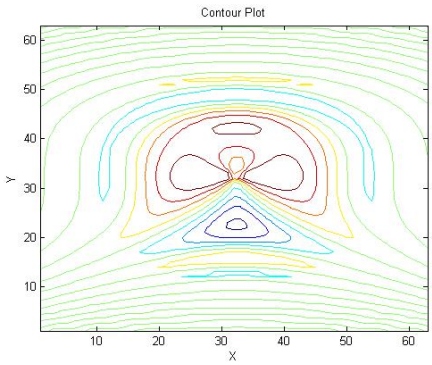
2.4.0.2 Some examples as jump start

```
Out(1)=3+2^3

ans =    11

Out(2)=sqrt(10)

ans =    3.1623

Out(3)=vpa(sqrt(10),60)

 =       3.16227766016837933199889354443271853371955513932521682685750

syms x
syms y
y=x^2sin(x)^2
                                  2       2
                                 x  sin(x)
z=int(y,x)

   2                                              2                                    3
  x  (- 1/2 cos(x) sin(x) + 1/2 x) - 1/2 x cos(x)  + 1/4 cos(x) sin(x) + 1/4 x - 1/3 x

Der=diff(z,x)
                                       2            2             2
  2 x (- 1/2 cos(x) sin(x) + 1/2 x) + x  (1/2 sin(x)  - 1/2 cos(x)  + 1/2)

                     2                                  2          2
         - 1/4 cos(x)  + x cos(x) sin(x) - 1/4 sin(x)  + 1/4 - x
Simple=simplify(Der)
                                  2       2
                                 x  sin(x)
Series=Taylor(exp(x),6,x,0)
                               2        3         4          5
                1 + x + 1/2 x  + 1/6 x  + 1/24 x  + 1/120 x

(x+2)^2+((x+5)^2(x+y)^2)^3
                              2          6         6
                        (x + 2)  + (x - 5)  (x + y)
Exp_Pol=expand(Pol)

            2          6          5             4  2           3  3
  4 + 4 x + x  + 15625 x  + 93750 x  y + 234375 x  y  + 312500 x  y

                    2  4            5      11         10  2       9  3
>         + 234375 x  y  + 93750 x y  + 6 x   y + 15 x   y  + 20 x  y

                 8  4      7  5    6  6        10          9  2        8  3
>         + 15 x  y  + 6 x  y  + x  y  - 180 x   y - 450 x  y  - 600 x  y

                  7  4        6  5          6    12       11        10         9
>         - 450 x  y  - 180 x  y  + 15625 y  + x   - 30 x   + 375 x   - 2500 x
```

```
                  8          7       5  6       9  8        2   7         3
>         + 9375 x  - 18750 x  - 30 x  y+ 2250 x  y + 5625 x  y  + 7500 x  y

                  6  4         5  5        4  6          8           7  2
>         + 5625 x  y  + 2250 x  y  + 375 x  y  - 15000 x  y - 37500 x  y

                   6  3         5  4         4  5         3  6          7
>         - 50000 x  y  - 37500 x  y  - 15000 x  y  - 2500 x  y  + 56250 x  y

                   6  2          5  3          4  4         3  5
>         + 140625 x  y  + 187500 x  y  + 140625 x  y  + 56250 x  y

                  2  6          6            5  2          4  3
>         + 9375 x  y  - 112500 x  y - 281250 x  y  - 375000 x  y

                   3  4          2  5           6
>         - 281250 x  y  - 112500 x  y  - 18750 x y
```

```
t=0:0.01:pi
plot(sin(1./t))
--Plot Mode---

[X,Y]=meshgrid(-1:0.01:1,-1:0.01:1)
Z=sin(X.^2+Y.^3)/(X.^2+Y.^2)
surf(X,Y,Z)
```






```
x=1:1:10
y(1:10)=x.^2

y =

[   1,   4,   9,  16,  25,  36,  49,  64,  81, 100]

A_1=[1 2 4; 5 6 100; -10.1 23 56]

A_1 =
    1.0000    2.0000    4.0000
    5.0000    6.0000  100.0000
  -10.1000   23.0000   56.0000

A_2=rand(3,4)
```

```
A_2 =
    0.2859    0.7157    0.4706    0.7490
    0.5437    0.8390    0.5607    0.5039
    0.9848    0.4333    0.2691    0.6468

A_2'=
    0.3077    0.1387    0.4756
    0.3625    0.7881    0.7803
    0.6685    0.1335    0.0216
    0.5598    0.3008    0.9394

A_1.*A_2=
    3.1780    5.9925    5.0491    3.0975
   43.5900   94.5714   92.6770   29.3559
   26.3095   57.1630   58.7436   17.5258

[U L]=lu(A_1)

U =
   -0.0990    0.2460    1.0000
   -0.4950    1.0000         0
    1.0000         0         0

L =
  -10.1000   23.0000   56.0000
         0   17.3861  127.7228
         0         0  -21.8770

[Q R]=qr(A_1)

Q =
   -0.0884   -0.2230    0.9708
   -0.4419   -0.8647   -0.2388
    0.8927   -0.4501   -0.0221

R =
  -11.3142   17.7035    5.4445
         0  -15.9871 -112.5668
         0         0  -21.2384

b=[1;2;3]
x=A_1\b

b =
     1
     2
     3

x =
    0.3842
    0.3481
   -0.0201

A_3=[1 2 3; -1 0 5; 8 9 23]

A_3 =
     1     2     3
    -1     0     5
```

```
     8     9    23

Inverse=inv(A_3)

Inverse =
   -0.8333   -0.3519    0.1852
    1.1667   -0.0185   -0.1481
   -0.1667    0.1296    0.0370
```

Example 2.6 (Calculation of Square Roots)

```
(********** root[2] iterative **************)
function [b]=calculate_Sqrt(a,accuracy)
clc;
x=a;
delta=inf;
while delta>=10^-(accuracy)
        Res(n)=x;
        xn=x;
        x=0.5*(x+a/x);
       delta=abs(x-xn);
end
b=Res;
```

2.5 Short Introduction to GNU Octave

From the Octave homepage: GNU Octave is a high-level interpreted language, primarily intended for numerical computations. It provides capabilities for the numerical solution of linear and nonlinear problems, and for performing other numerical experiments. It also provides extensive graphics capabilities for data visualization and manipulation. Octave is normally used through its interactive command line interface, but it can also be used to write non-interactive programs. The Octave language is quite similar to Matlab so that most programs are easily portable.

Downloads, Docs, FAQ, etc.:

http://www.gnu.org/software/octave/

Nice Introduction/Overview:

http://math.jacobs-university.de/oliver/teaching/iub/resources/octave/octave-intro/octave-intro.html

Plotting in Octave:

http://www.gnu.org/software/octave/doc/interpreter/Plotting.html

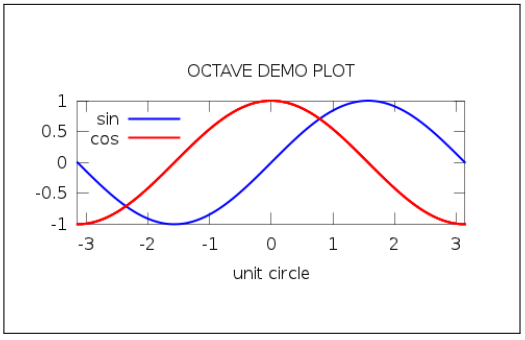
```
// -> comments


BASICS
======

octave:47> 1 + 1
ans =  2
octave:48> x = 2 * 3
x =  6

// suppress output

octave:49> x = 2 * 3;
octave:50>

// help

octave:53> help sin
'sin' is a built-in function
 -- Mapping Function:  sin (X)
     Compute the sine for each element of X in radians.
...


VECTORS AND MATRICES
====================

// define 2x2 matrix

octave:1> A = [1 2; 3 4]
A =
   1   2
   3   4

// define 3x3 matrix

octave:3> A = [1 2 3; 4 5 6; 7 8 9]
A =
   1   2   3
   4   5   6
   7   8   9

// access single elements

octave:4> x = A(2,1)
x =  4

octave:17> A(3,3) = 17
A =
    1    2    3
    4    5    6
    7    8   17

// extract submatrices

octave:8> A
A =
   1   2   3
```

```
   4   5   6
   7   8  17

octave:9> B = A(1:2,2:3)
B =
   2   3
   5   6

octave:36> b=A(1:3,2)
b =

   2
   5
   8

// transpose

octave:25> A'
ans =
    1    4    7
    2    5    8
    3    6   17

// determinant

octave:26> det(A)
ans = -24.000

// solve Ax = b

// inverse

octave:22> inv(A)
ans =
  -1.54167   0.41667   0.12500
   1.08333   0.16667  -0.25000
   0.12500  -0.25000   0.12500

// define vector b

octave:27> b = [3 7 12]'
b =
    3
    7
   12

// solution x

octave:29> x = inv(A) * b
x =
  -0.20833
   1.41667
   0.12500

octave:30> A * x
ans =
    3.0000
    7.0000
   12.0000
```

```
// try A\b

// illegal operation

octave:31> x * b
error: operator *: nonconformant arguments (op1 is 3x1, op2 is 3x1)

// therefore allowed

octave:31> x' * b
ans =  10.792

octave:32> x * b'
ans =
   -0.62500   -1.45833   -2.50000
    4.25000    9.91667   17.00000
    0.37500    0.87500    1.50000

// elementwise operations

octave:11> a = [1 2 3]
a =
   1   2   3

octave:10> b = [4 5 6]
b =
   4   5   6

octave:12> a*b
error: operator *: nonconformant arguments (op1 is 1x3, op2 is 1x3)
octave:12> a.*b
ans =
    4   10   18

octave:23> A = [1 2;3 4]
A =
   1   2
   3   4

octave:24> A^2
ans =
    7   10
   15   22

octave:25> A.^2
ans =
    1    4
    9   16

// create special vectors/matrices

octave:52> x = [0:1:5]
x =
   0   1   2   3   4   5

octave:53> A = zeros(2)
A =
   0   0
```

```
   0   0

octave:54> A = zeros(2,3)
A =
   0   0   0
   0   0   0

octave:55> A = ones(2,3)
A =
   1   1   1
   1   1   1

octave:56> A = eye(4)
A =
Diagonal Matrix
   1   0   0   0
   0   1   0   0
   0   0   1   0
   0   0   0   1

octave:57> B = A * 5
B =
Diagonal Matrix
   5   0   0   0
   0   5   0   0
   0   0   5   0
   0   0   0   5

// vector/matrix size

octave:43> size(A)
ans =
   3   3

octave:44> size(b)
ans =
   3   1

octave:45> size(b)(1)
ans =  3


PLOTTING (2D)
============

octave:35> x = [-2*pi:0.1:2*pi];
octave:36> y = sin(x);
octave:37> plot(x,y)
octave:38> z = cos(x);
octave:39> plot(x,z)

// two curves in one plot

octave:40> plot(x,y)
octave:41> hold on
octave:42> plot(x,z)

// reset plots
```

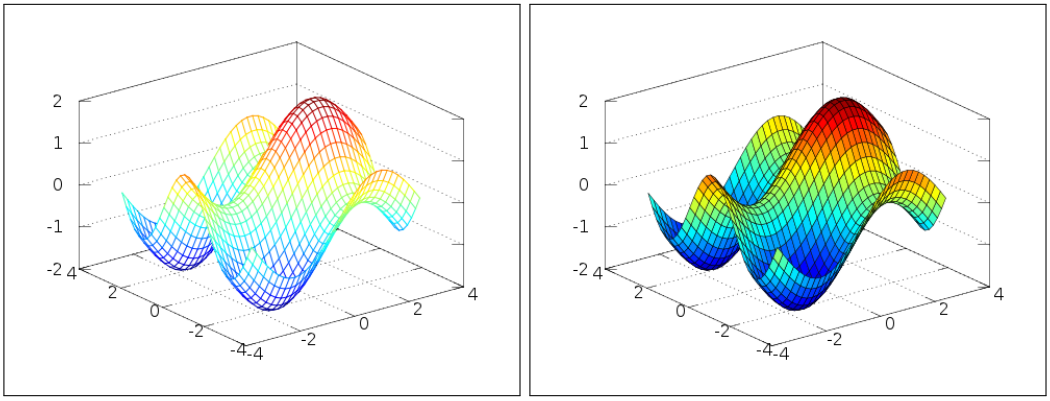
```
octave:50> close all

// plot different styles

octave:76> plot(x,z,'r')
octave:77> plot(x,z,'rx')
octave:78> plot(x,z,'go')

octave:89> close all

// manipulate plot

octave:90> hold on
octave:91> x = [-pi:0.01:pi];

// another linewidth

octave:92> plot(x,sin(x),'linewidth',2)
octave:93> plot(x,cos(x),'r','linewidth',2)

// define axes range and aspect ratio

octave:94> axis([-pi,pi,-1,1], 'equal')
-> try 'square' or 'normal' instead of 'equal' (help axis)

// legend

octave:95> legend('sin','cos')

// set parameters (gca = get current axis)

octave:99> set(gca,'keypos', 2) // legend position (1-4)
octave:103> set(gca,'xgrid','on') // show grid in x
octave:104> set(gca,'ygrid','on') // show grid in y

// title/labels

octave:102> title('OCTAVE DEMO PLOT')
octave:100> xlabel('unit circle')
octave:101> ylabel('trigon. functions')

// store as png

octave:105> print -dpng 'demo_plot.png'
```



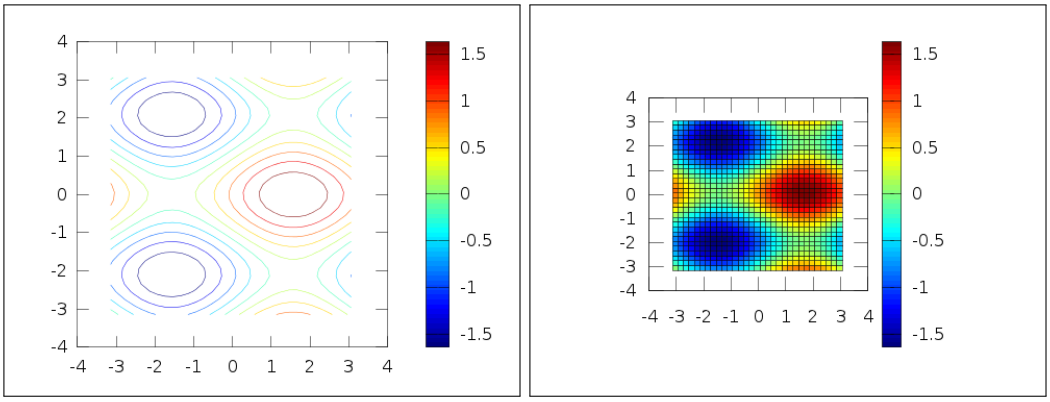
DEFINE FUNCTIONS

```
================

sigmoid.m:
---
function S = sigmoid(X)
mn = size(X);
S = zeros(mn);
for i = 1:mn(1)
for j = 1:mn(2)
S(i,j) = 1 / (1 + e ^ -X(i,j));
end
end
end
---

easier:
---
function S = sigmoid(X)
S = 1 ./ (1 .+ e .^ (-X));
end
---

octave:1> sig + [TAB]
sigmoid    sigmoid.m
octave:1> sigmoid(10)
ans =  0.99995
octave:2> sigmoid([1 10])
error: for x^A, A must be square // (if not yet implemented elementwise)
error: called from:
error:   /home/richard/faculty/adv_math/octave/sigmoid.m at line 3, column 4
...
octave:2> sigmoid([1 10])
ans =
   0.73106   0.99995

octave:3> x = [-10:0.01:10];
octave:5> plot(x,sigmoid(x),'linewidth',3);


PLOTTING (3D)
============

// meshgrid

octave:54> [X,Y] = meshgrid([1:3],[1:3])
X =
   1   2   3
   1   2   3
   1   2   3

Y =
   1   1   1
   2   2   2
   3   3   3

// meshgrid with higher resolution (suppress output)

octave:15> [X,Y] = meshgrid([-4:0.2:4],[-4:0.2:4]);
```

```
// function over x and y, remember that cos and sin
// operate on each element, result is matrix again

octave:20> Z = cos(X) + sin(1.5*Y);

// plot

octave:21> mesh(X,Y,Z)
octave:22> surf(X,Y,Z)
```

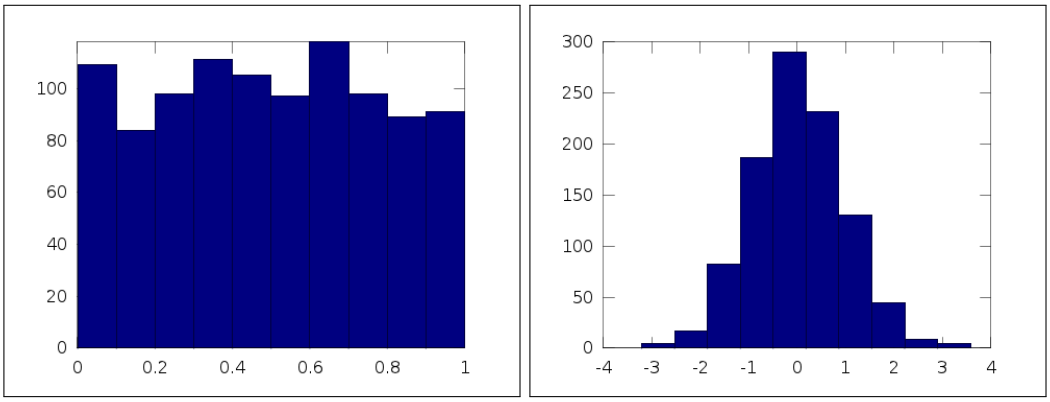


```
octave:44> contour(X,Y,Z)
octave:45> colorbar
octave:46> pcolor(X,Y,Z)
```



```
RANDOM NUMBERS / HISTOGRAMS
===========================

// equally distributed random numbers

octave:4> x=rand(1,5)
x =
   0.71696   0.95553   0.17808   0.82110   0.25843

octave:5> x=rand(1,1000);
octave:6> hist(x);

// normally distributed random numbers

octave:5> x=randn(1,1000);
octave:6> hist(x);
```

```
// try

octave:5> x=randn(1,10000);
octave:6> hist(x, 25);
```



2.6 Exercises

Mathematica

Exercise 2.1 Program the factorial function with Mathematica.

a) Write an iterative program that calculates the formula $n! = n \cdot (n-1) \cdot \ldots \cdot 1$.

b) Write a recursive program that calculates the formula

$$n! = \begin{cases} n \cdot (n-1)! & \text{if } n > 1 \\ 1 & \text{if } n = 1 \end{cases}$$

analogously to the root example in the script.

Exercise 2.2

a) Write a Mathematica program that multiplies two arbitrary matrices. Don't forget to check the dimensions of the two matrices before multiplying. The formula is

$$C_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}.$$

Try to use the functions `Table`, `Sum` and `Length` only.

b) Write a Mathematica program that computes the transpose of a matrix using the `Table` function.

c) Write a Mathematica Program that computes the inverse of a matrix using the function `Linear Solve`.

MATLAB

Exercise 2.3

a) For a finite geometic series we have the formula $\Sigma_{i=0}^{n} q^i = \frac{1-q^{n+1}}{1-q}$. Write a MATLAB function that takes q and n as inputs and returns the sum.

b) For an infinite geometic series we have the formula $\Sigma_{i=0}^{\infty} q^i = \frac{1}{1-q}$ if the series converges. Write a MATLAB function that takes q as input and returns the sum. Your function should produce an error if the series diverges.

Exercise 2.4

a) Create a 5×10 random Matrix A.

b) Compute the mean of each column and assign the results to elements of a vector called avg.

c) Compute the standard deviation of each column and assign the results to the elements of a vector called s.

Exercise 2.5 Given the row vectors $x = [4,\ 1,\ 6,\ 10,\ -4,\ 12,\ 0.1]$ and $y = [-1,\ 4,\ 3,\ 10,\ -9,\ 15,\ -2.1]$ compute the following arrays,

a) $a_{ij} = x_i y_j$

b) $b_{ij} = \frac{x_i}{y_j}$

c) $c_i = x_i y_i$, then add up the elements of c using two different programming approaches.

d) $d_{ij} = \frac{x_i}{2+x_i+y_j}$

e) Arrange the elements of x and y in ascending order and calculate e_{ij} being the reciprocal of the less x_i and y_j.

f) Reverse the order of elements in x and y in one command.

Exercise 2.6 Write a MATLAB function that calculates recursively the square root of a number.

Analysis Repetition

Exercise 2.7 In a bucket with capacity v there is a poisonous liquid with volume αv. The bucket has to be cleaned by repeatedly diluting the liquid with a fixed amount $(\beta - \alpha)v$ $(0 < \beta < 1 - \alpha)$ of water and then emptying the bucket. After emptying, the bucket always keeps αv of its liquid. Cleaning stops when the concentration c_n of the poison after n iterations is reduced from 1 to $c_n < \epsilon > 0$.

a) Assume $\alpha = 0.01$, $\beta = 1$ and $\epsilon = 10^{-9}$. Compute the number of cleaning-iterations.

b) Compute the total volume of water required for cleaning.

c) Can the total volume be reduced by reducing β? If so, determine the optimal β.

d) Give a formula for the time required for cleaning the bucket.

e) How can the time for cleaning the bucket be minimized?

Chapter 3

Calculus – Selected Topics

3.1 Sequences and Convergence

Definition 3.1 A function $\mathbb{N} \to \mathbb{R}, n \mapsto a_n$ is called sequence.
Notation: $(a_n)_{n\in\mathbb{N}}$ or $(a_1, a_2, a_3, ...)$

Example 3.1
$(1, 2, 3, 4, ...) = (n)_{n\in\mathbb{N}}$
$(1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, ...) = (\frac{1}{n})_{n\in\mathbb{N}}$
$(1, 2, 4, 8, 16, ...) = (2^{n-1})_{n\in\mathbb{N}}$

Consider the following sequences:

1. 1,2,3,5,7,11,13,17,19,23,...

2. 1,3,6,10,15,21,28,36,45,55,66,..

3. 1,1,2,3,5,8,13,21,34,55,89,...

4. 8,9,1,-8,-10,-3,6,9,4,-6,-10

5. 1,2,3,4,6,7,9,10,11,13,14,15,16,17,18,19,21,22,23,24,26,27,29,30,31,32,33,34,35,36, 37,..

6. 1,3,5,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,31,33, 35, 37,38,39,41,43,..

Find the next 5 elements of each sequence. If you do not get ahead or want to solve other riddles additionaly, have a look at *http://www.oeis.org*.

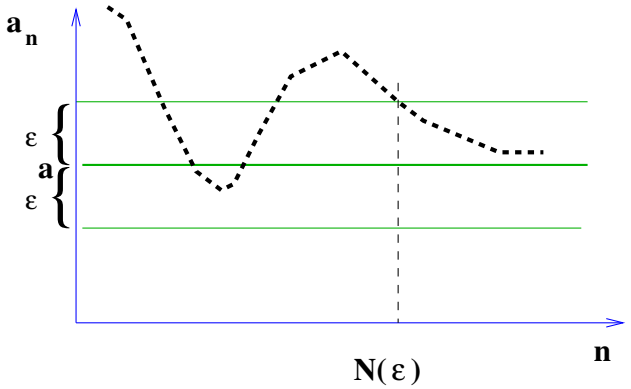
Definition 3.2 $(a_n)_{n\in\mathbb{N}}$ is called **bounded**, if there is $A, B \in \mathbb{R}$ with $\forall n \quad A \leq a_n \leq B$

$(a_n)_{n\in\mathbb{N}}$ is called **monotonically increasing/decreasing**, iff $\forall n \quad a_{n+1} \geq a_n \quad (a_{n+1} \leq a_n)$

Definition 3.3 A sequence of real numbers $(a_n)_{n\in\mathbb{N}}$ **converges to** $a \in \mathbb{R}$, iff:

$$\forall \varepsilon > 0 \quad \exists N(\varepsilon) \in \mathbb{N}, \quad so \quad that \quad |a_n - a| < \varepsilon \quad \forall n \geq N(\varepsilon)$$

Notation: $\lim\limits_{n\to\infty} a_n = a$



Definition 3.4 A sequence is called **divergent** if it is not **convergent**.

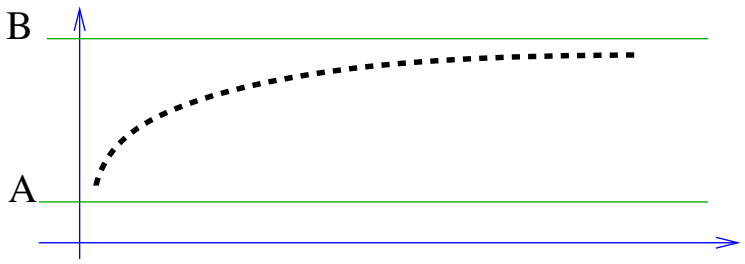
Example 3.2

1.) $(1, \frac{1}{2}, \frac{1}{3}, ...)$ converges to 0 (zero sequence)
2.) $(1, 1, 1, ...)$ converges to 1
3.) $(1, -1, 1, -1, ...)$ is divergent
4.) $(1, 2, 3, ...)$ is divergent

Theorem 3.1 Every convergent sequence is **bounded**.

Proof: for $\varepsilon = 1 : N(1)$, first $N(1)$ terms bounded, the rest bounded through $a \pm N(1)$.
Note: Not every bounded sequence does converge! (see exercise 3), but:

Theorem 3.2 Every bounded monotonic sequence is convergent



3.1.1 Sequences and Limits

Let $(a_n), (b_n)$ two convergent sequences with: $\lim\limits_{n\to\infty} a_n = a$, $\lim\limits_{n\to\infty} b_n = b$, then it holds:

$$\begin{aligned}
\lim_{n\to\infty}(a_n \pm b_n) &= \lim_{n\to\infty} a_n \pm \lim_{n\to\infty} b_n \\
&= a \pm b \\
\lim_{n\to\infty}(c \cdot a_n) &= c \cdot \lim_{n\to\infty} a_n \\
&= c \cdot a \\
\lim_{n\to\infty}(a_n \cdot b_n) &= a \cdot b \\
\lim_{n\to\infty}\left(\frac{a_n}{b_n}\right) &= \frac{a}{b} \quad if \quad b_n, b \neq 0
\end{aligned}$$

Example 3.3 Show that the sequence $a_n = \left(1 + \frac{1}{n}\right)^n, n \in \mathbb{N}$ converges:

n	1	2	3	4	10	100	1000	10000
a_n	2	2.25	2.37	2.44	2.59	2.705	2.717	2.7181

The numbers (only) suggest that the sequence converges.

1. Boundedness: $\forall n \quad a_n > 0$ and

$$\begin{aligned}
a_n &= \left(1 + \frac{1}{n}\right)^n \\
&= 1 + n \cdot \frac{1}{n} + \frac{n(n-1)}{2} \cdot \frac{1}{n^2} + \frac{n(n-1)(n-2)}{2 \cdot 3} \cdot \frac{1}{n^3} + \ldots + \frac{1}{n^n} \\
&= 1 + 1 + \frac{1}{2}\left(1 - \frac{1}{n}\right) + \frac{1}{2 \cdot 3}\left(1 - \frac{1}{n}\right)\left(1 - \frac{2}{n}\right) + \ldots + \frac{1}{n!}\left(1 - \frac{1}{n}\right)\left(1 - \frac{2}{n}\right) \cdot \ldots \\
&\quad \ldots \cdot \left(1 - \frac{n-1}{n}\right) \\
&< 1 + 1 + \frac{1}{2} + \frac{1}{2 \cdot 3} + \ldots + \frac{1}{n!} \\
&< 1 + 1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \ldots + \frac{1}{2^n} \\
&< 1 + 1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \ldots \\
&= 1 + \frac{1}{1 - \frac{1}{2}} \\
&= 3
\end{aligned}$$

2. Monotony: Replacing n by $n + 1$ in (1.) gives $a_n < a_{n+1}$, since in line 3 most summands in a_{n+1} are bigger!

The limit of this sequence is the *Euler number*:

$$e := \lim_{n\to\infty}\left(1 + \frac{1}{n}\right)^n = 2.718281828\ldots$$

3.2 Series

Definition 3.5 Let $(a_n)_{n\in\mathbb{N}}$ be a sequence of real numbers. The sequence

$$s_n := \sum_{k=0}^{n} a_k \quad , n \in \mathbb{N}$$

of the partial sums is called (infinite) series and is defined by $\sum_{k=0}^{\infty} a_k$.
If $(s_n)_{n\in\mathbb{N}}$ converges, we define

$$\sum_{k=0}^{\infty} a_k := \lim_{n\to\infty} \sum_{k=0}^{n} a_k.$$

Example 3.4

n	0	1	2	3	4	5	6	7	8	9	10	...
Sequence a_n	0	1	2	3	4	5	6	7	8	9	10	...
Series $S_n = \sum_{k=0}^{n} a_k$	0	1	3	6	10	15	21	28	36	45	55	...

n	0	1	2	3	4	5	6	7	8	9	10
Sequence a_n	1	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{64}$	$\frac{1}{128}$	$\frac{1}{256}$	$\frac{1}{518}$	$\frac{1}{1024}$
Series S_n	1	$\frac{3}{2}$	$\frac{7}{4}$	$\frac{15}{8}$	$\frac{31}{16}$	$\frac{63}{32}$	$\frac{127}{64}$	$\frac{255}{128}$	$\frac{511}{256}$	$\frac{1023}{512}$	$\frac{2047}{1024}$
(decimal)	1	1.5	1.75	1.875	1.938	1.969	1.984	1.992	1.996	1.998	1.999

3.2.1 Convergence criteria for series

Theorem 3.3 (Cauchy) The series $\sum_{n=0}^{\infty} a_n$ converges iff

$$\forall \varepsilon > 0 \quad \exists N \in \mathbb{N} \quad \left| \sum_{k=m}^{n} a_k \right| < \varepsilon$$

for all $n \geq m \geq N$

Proof: Let $s_p := \sum_{k=0}^{p} a_k$. Then $s_n - s_{m-1} = \sum_{k=m}^{n} a_k$. Therefore $(s_n)_{n\in\mathbb{N}}$ Cauchy sequence $\Leftrightarrow$ (s_n) is convergent.

Theorem 3.4 A series with $a_k > 0 \quad for \quad k \geq 1$ converges iff the sequence of partial sums is bounded.

Proof: as exercise

Theorem 3.5 (Comparison test)
Let $\sum_{n=0}^{\infty} c_n$ a convergent series with $\forall n \quad c_n \geq 0$ and $(a_n)_{n\in\mathbb{N}}$ a sequence with $|a_n| \leq c_n \quad \forall n \in \mathbb{N}$. Then $\sum_{n=0}^{\infty} a_n$ converges.

Theorem 3.6 (Ratio test)
Let $\sum_{n=0}^{\infty} a_n$ a series with $a_n \neq 0$ for all $n \geq n_0$. A real number q with $0 < q < 1$ exists, that $\left|\frac{a_{n+1}}{a_n}\right| \leq q$ for all $n \geq n_0$. Then the series $\sum_{n=0}^{\infty} a_n$ converges.
If, from an index n_0, $\left|\frac{a_{n+1}}{a_n}\right| \geq 1$, then the series is divergent.

Proof idea (f. 1. Part): Show that $\sum_{n=0}^{\infty} |a_0|q^n$ is a majorant.
Example 3.5

$$\sum_{n=0}^{\infty} \frac{n^2}{2^n} \quad converges.$$

Proof:

$$\left|\frac{a_{n+1}}{a_n}\right| = \frac{(n+1)^2 2^n}{2^{n+1}n^2} = \frac{1}{2}(1+\frac{1}{n})^2 \underset{\substack{\uparrow \\ \text{for } n\geq 3}}{\leq} \frac{1}{2}(1+\frac{1}{3})^2 = \frac{8}{9} < 1.$$

3.2.2 Power series

Theorem 3.7 and defintion For each $x \in \mathbb{R}$ the power series

$$\exp(x) := \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

is convergent.

Proof: The ratio test gives

$$\left|\frac{a_{n+1}}{a_n}\right| = \left|\frac{x^{n+1}n!}{(n+1)!x^n}\right| = \frac{|x|}{n+1} \leq \frac{1}{2} \quad for \quad n \geq 2|x| - 1$$

Definition 3.6 Euler's number $e := \exp(1) = \sum_{n=0}^{\infty} \frac{1}{n!}$
The function $\exp : \mathbb{R} \to \mathbb{R}^+ \quad x \mapsto \exp(x)$ is called exponential function.

Theorem 3.8 (Remainder)

$$\exp(x) = \sum_{n=0}^{N} \frac{x^n}{n!} + R_N(x) \qquad N-th \quad approximation$$

with $|R_N(x)| \leq 2\frac{|x|^{N+1}}{(N+1)!}$ for $|x| \leq 1 + \frac{N}{2}$ or $N \geq 2(|x|-1)$

3.2.2.1 Practical computation of $\exp(x)$:

$$\begin{aligned}
\sum_{n=0}^{N} \frac{x^n}{n!} &= 1 + x + \frac{x^2}{2} + \ldots + \frac{x^{N-1}}{(N-1)!} + \frac{x^N}{N!} \\
&= 1 + x(1 + \frac{x}{2}(1 + \ldots + \frac{x}{N-2}(1 + \frac{x}{N-1}(1 + \frac{x}{N}))\ldots)) \\
e &= 1 + 1 + \frac{1}{2}(\ldots + \frac{1}{N-2}(1 + \frac{1}{N-1}(1 + \frac{1}{N}))\ldots) + R_N \qquad with \quad R_N \leq \frac{2}{(N+1)!}
\end{aligned}$$

For $N = 15$**:** $|R_{15}| \leq \frac{2}{16!} < 10^{-13}$
$e = 2.718281828459 \pm 2 \cdot 10^{-12}$ (rounding error 5 times 10^{-13}!)

Theorem 3.9 The functional equation of the exponential function
$\forall x, y \in \mathbb{R}$ it holds: $\exp(x+y) = \exp(x) \cdot \exp(y)$.

Proof: The proof of this theorem is via the series representation (definition 3.6). It is not easy, because it requires another theorem about the product of series (not covered here).
Conclusions:
a) $\forall x \in \mathbb{R} \quad \exp(-x) = (\exp(x))^{-1} = \frac{1}{exp(x)}$
b) $\forall x \in \mathbb{R} \quad \exp(x) > 0$
c) $\forall n \in \mathbb{Z} \quad \exp(n) = e^n$
Notation: Also for real numbers $x \in \mathbb{R}$: $\quad e^x := \exp(x)$
Proof:
a) $\exp(x) \cdot \exp(-x) = \exp(x-x) = \exp(0) = 1 \Rightarrow \exp(-x) = \frac{1}{\exp(x)} \quad x \neq 0$
b) 1.Case $x \geq 0$: $\exp(x) = 1 + x + \frac{x^2}{2} + \ldots \geq 1 > 0$
2.Case $x < 0$: $-x < 0 \Rightarrow \exp(-x) > 0 \Rightarrow \exp(x) = \frac{1}{\exp(-x)} > 0.$
c) Induction $\exp(1) = e \quad \exp(n) = \exp(n-1+1) = \exp(n-1) \cdot e = e^{n-1} \cdot e$
Note: for large $x := n + h \quad n \in \mathbb{N} \qquad \exp(x) = \exp(n+h) = e^n \cdot \exp(h)$
(for large x faster then series expansion)

3.3 Continuity

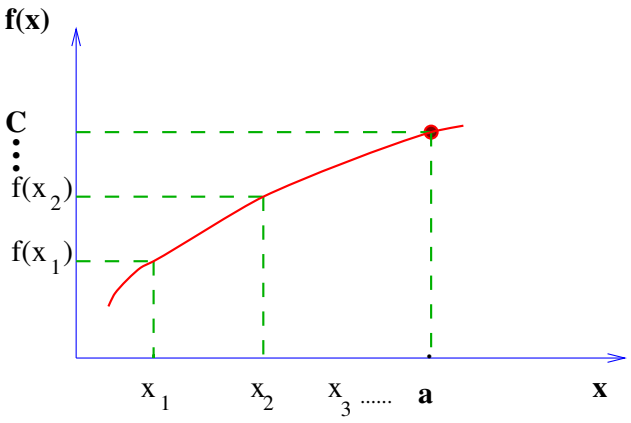
Functions are characterized among others in terms of "'smoothness"'. The weakest form of smoothness is the continuity.

Definition 3.7 Let $D \subset \mathbb{R}$, $f : D \to \mathbb{R}$ a function and $a \in \mathbb{R}$. We write

$$\lim_{x \to a} f(x) = C,$$

if **for each sequence** $(x_n)_{n \in \mathbb{N}}$, $(x_n) \in D$ with $\lim_{n \to \infty} x_n = a$ holds:

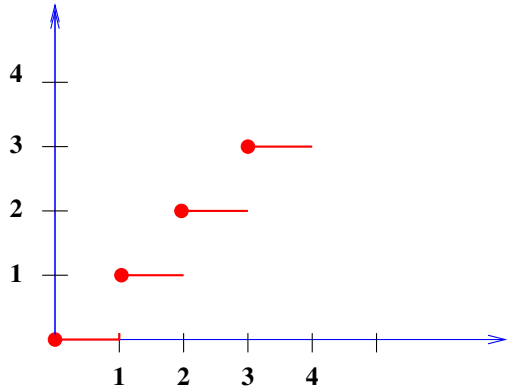
$$\lim_{n \to \infty} f(x_n) = C.$$



Definition 3.8 For $x \in \mathbb{R}$ the expression $\lfloor x \rfloor$ denotes the unique integer number n with $n \leq x < n + 1$.

Example 3.6 1. $\lim_{x \to 0} \exp(x) = 1$

2. $\lim_{x \to 1} \lfloor x \rfloor$ does not exist!
left-side limit $\neq$ right-side limit



3. Let $f : \mathbb{R} \to \mathbb{R}$ polynomial of the form $f(x) = x^k + a_1 x^{k-1} + \ldots + a_{k-1} x + a_k, \quad k \geq 1$.
Then it holds: $\lim_{x \to \infty} f(x) = \infty$

$$\text{and} \quad \lim_{x \to -\infty} f(x) = \begin{cases} \infty & , \ if \quad k \quad even \\ -\infty & , \ if \quad k \quad odd \end{cases}$$

Proof: for $x \neq 0$

$$f(x) = x^k (1 + \underbrace{\frac{a_1}{x} + \frac{a_2}{x^2} + \ldots + \frac{a_k}{x^k}}_{=: g(x)})$$

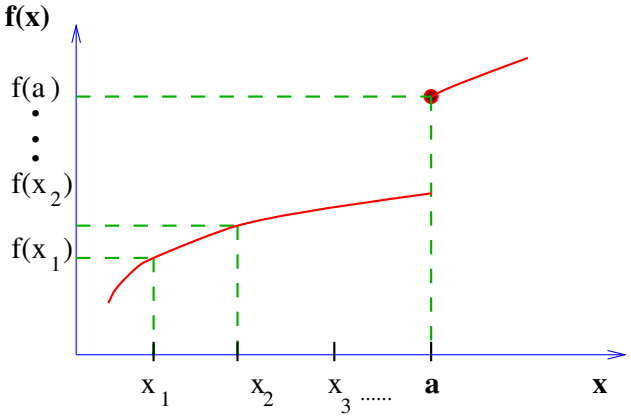
since $\lim_{x\to\infty} g(x) = 0$, it follows $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} x^k = \infty$.

Application: The asymptotic behavior for $x \to \infty$ of polynomials is always determinated by the highest power in x.

Definition 3.9 (Continuity)
Let $f : D \to \mathbb{R}$ a function and $a \in D$. The function f is called **continuous** at point a, if

$$\lim_{x\to a} f(x) = f(a).$$

f is called continuous in D, if f is continuous at every point of D.



For the depicted function it holds $\lim_{x\to\infty} f(x) \neq a$. f is discontinuous at the point a.

Example 3.7 1.) $f : x \mapsto c$ (constant function) is continuous on whole $\mathbb{R}$.
2.) The exponential function is continuous on whole $\mathbb{R}$.
3.) The identity function $f : x \mapsto x$ is continuous on whole $\mathbb{R}$.

Theorem 3.10 Let $f, g : D \to \mathbb{R}$ functions, that are at $a \in D$ continuous and let $r \in \mathbb{R}$. Then the functions $f + g$, rf, $f \cdot g$ at point a are continuous, too. If $g(a) \neq 0$, then $\frac{f}{g}$ is continuous at a.

Proof: Let (x_n) a sequence with $(x_n) \in D$ and $\lim_{n\to\infty} x_n = a$.

$$\left.\begin{array}{rrcl} \textit{to show}: & \lim_{n\to\infty}(f+g)(x_n) & = & (f+g)(a) \\ & \lim_{n\to\infty}(rf)(x_n) & = & (rf)(a) \\ & \lim_{n\to\infty}(f\cdot g)(x_n) & = & (f\cdot g)(a) \\ & \lim_{n\to\infty}(\frac{f}{g})(x_n) & = & (\frac{f}{g})(a) \end{array}\right\} \textit{holds because of rules for sequences.}$$

Definition 3.10 Let A, B, C subsets of $\mathbb{R}$ with the functions $f : A \to B$ and $g : B \to C$. Then $g \circ f : A \to C$, $\quad x \mapsto g(f(x))$ is called the composition of f and g.

Example 3.8

$$\begin{array}{rrcl} 1.) & f \circ g(x) & = & f(g(x)) \\ 2.) & \sqrt{\ } \circ \sin(x) & = & \sqrt{\sin(x)} \\ 3.) & \sin \circ \sqrt{\ }(x) & = & \sin(\sqrt{x}) \end{array}$$

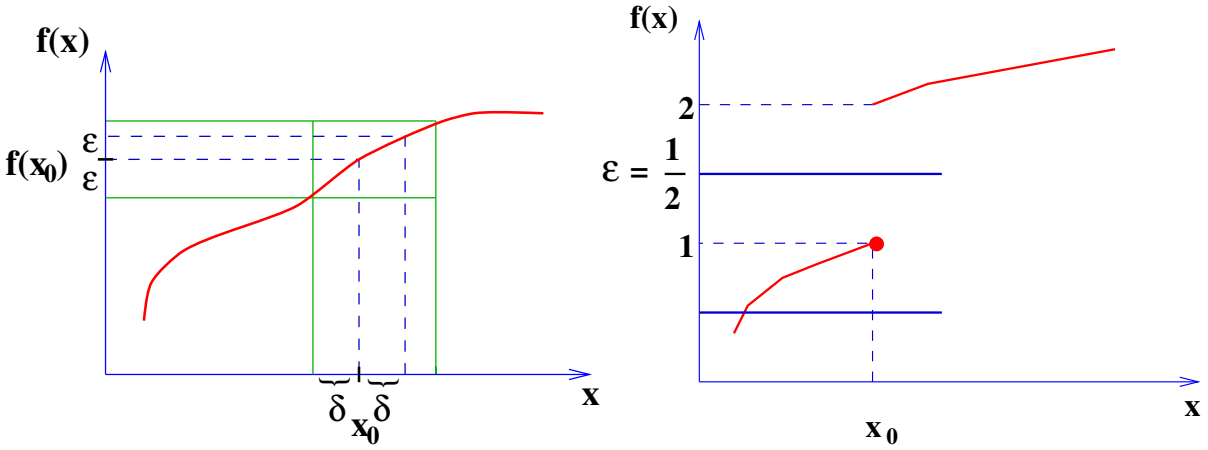
Theorem 3.11 Let $f : A \to B$ continuous at $a \in A$ and $g : A \to C$ continuous at $y = f(a)$. Then the composition $g \circ f$ is continuous in a, too.

Proof: to show: $\lim_{n\to\infty} x_n = a \quad \Rightarrow \quad \lim_{n\to\infty} f(x_n) = f(a) \quad \Rightarrow \quad \lim_{n\to\infty} g(f(x_n)) = g(f(a))$.
($\Rightarrow$: continuity of f; $\Rightarrow$: continuity of g)

Example 3.9 $\dfrac{x}{x^2 + a}$ is continuous on whole $\mathbb{R}$, because $f(x) = x^2, g(x) = f(x) + a$ and $h(x) = \dfrac{x}{g(x)}$ are continuous.

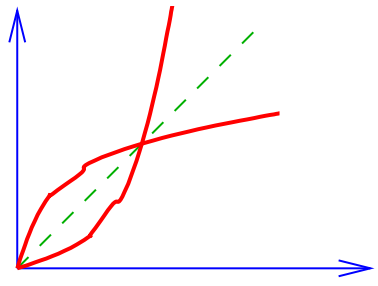
Theorem 3.12 (ε δ Definition of Continuity)
A function $f : D \to \mathbb{R}$ is continuous at $x_0 \in D$ iff:

$$\forall \varepsilon > 0 \quad \exists \delta > 0 \quad \forall x \in D \quad (|x - x_0| < \delta \Rightarrow |f(x) - f(x_0)| < \varepsilon)$$

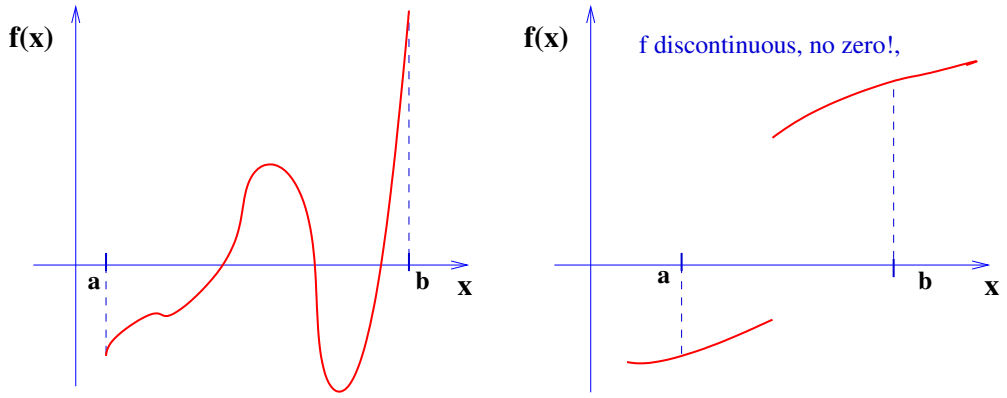


Theorem 3.13 Let $f : [a, b] \to \mathbb{R}$ continuous and strictly increasing (or decreasing) and $A := f(a), B := f(b)$. Then the inverse function $f^{-1} : [A, B] \to \mathbb{R}$ (*bzw.* $[B, A] \to \mathbb{R}$) is continuous and strictly increasing (or decreasing), too.

Example 3.10 (Roots)
Let $k \in \mathbb{N}, k \geq 2$. The function $f : \mathbb{R}^+ \to \mathbb{R}^+, \quad x \mapsto x^k$ is continuous and strictly increasing. The inverse function $f^{-1} : \mathbb{R}^+ \to \mathbb{R}^+, x \mapsto \sqrt[k]{x}$ is continuous and strictly increasing.



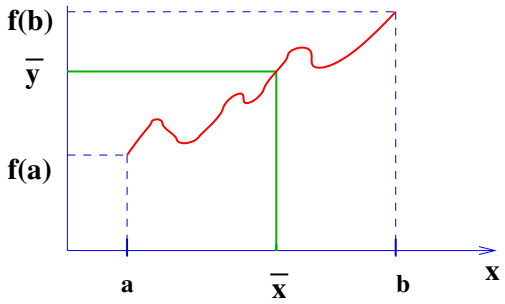
Theorem 3.14 (Intermediate Value)
Let $f : [a, b] \to \mathbb{R}$ continuous with $f(a) < 0$ and $f(b) > 0$. Then there exists a $p \in [a, b]$ with $f(p) = 0$.



Note: if $f(a) > 0, f(b) < 0$ take $-f$ instead of f and apply the intermediate value theorem.

Example 3.11 $D = \mathbb{Q}:\quad x \mapsto x^2 - 2 = f(x)\quad f(1) = -1, f(2) = 2$ there is a $p \in D$ with $f(p) = 0$.

Corollar 3.3.1 Is $f : [a, b] \to \mathbb{R}$ continuous and $\bar{y}$ is any number between $f(a)$ and $f(b)$, then there is at least one $\bar{x} \in [a, b]$ with $f(\bar{x}) = \bar{y}$.



Note: Now it is clear that every continuous function on $[a, b]$ assumes every value in the interval $[f(a), f(b)]$.

3.3.1 Discontinuity

Definition 3.11 We write $\lim\limits_{x \searrow a} f(x) = c \quad (\lim\limits_{x \nearrow a} f(x) = c)$, if for every sequence (x_n) with $x_n > a \quad (x_n < a)$ and $\lim\limits_{x \to \infty} x_n = a$ holds: $\lim\limits_{n \to \infty} f(x_n) = c$.
$\lim\limits_{x \searrow a} f(x) \quad (\lim\limits_{x \nearrow a} f(x))$ is called right-side (left-side) limit of f at $x = a$.

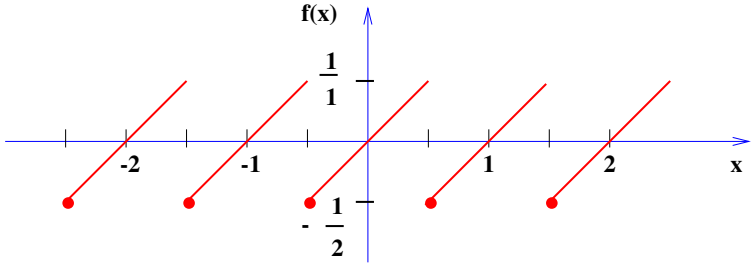
Theorem 3.15 A function is continuous at point a, if the right-side and left-side limit are equal.

Lemma 3.1 A function is discontinuous at the point a, if limit $\lim\limits_{x \to a} f(x)$ does not exist.

Conclusion: A function is discontinuous at the point a, if there are two sequences $(x_n), (z_n)$ with $\lim x_n = \lim z_n = a$ and $\lim f(x_n) \neq \lim f(z_n)$.

Example 3.12 1. **Step:** $\lim\limits_{x \nearrow a} f(x) = c_1 \neq c_2 = \lim\limits_{x \searrow a} f(x)$

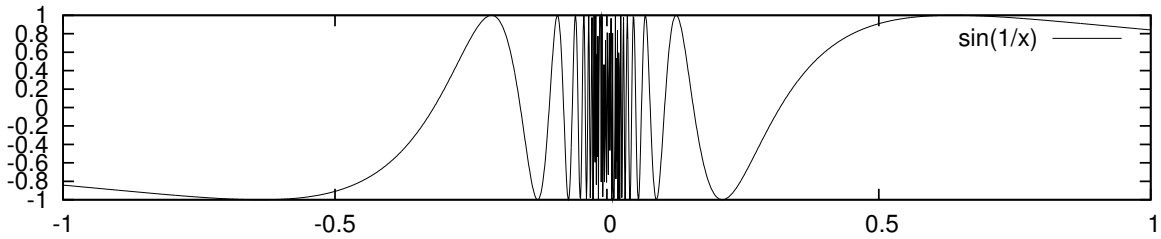
$f(x) = x - n \quad for \quad n - \frac{1}{2} \leq x < n + \frac{1}{2} \quad n \in \mathbb{Z}$



2. **Pole:** $\lim\limits_{x \to x_0} f(x) = \infty$
   or $\lim\limits_{x \to x_0} f(x) = -\infty$

   Example: $f(x) = \dfrac{1}{x^2}$

3. **Oscillation:**
   The function $f(x) = \sin \dfrac{1}{x}, \quad x \neq 0$ is discontinuous at $x = 0$



Proof: let $x_n = \dfrac{1}{\frac{\pi}{2} + n \cdot 2\pi} \quad n \in \mathbb{N}$

$\Rightarrow \sin \dfrac{1}{x_n} = 1 \Rightarrow \lim\limits_{n \to \infty} x_n = 0, \quad \lim\limits_{n \to \infty} \sin \dfrac{1}{x_n} = 1$

but: let $z_n = \dfrac{1}{n \cdot \pi}, \quad n \in \mathbb{N}$

$\Rightarrow \lim\limits_{n \to \infty} z_n = 0, \quad \lim\limits_{n \to \infty} \sin \dfrac{1}{z_n} = 0$

$\rightarrow$ Limit is not unique, therefore $\sin \frac{1}{x}$ is discontinuous.

Note: Is a function f continuous $\forall x \in [a, b]$, then it holds for any convergent sequence (x_n) :

$$\lim_{n \to \infty} f(x_n) = f(\lim_{n \to \infty} x_n).$$

Proof: as exercise
Conclusion: Continuity of f at $x_0 = \lim\limits_{n \to \infty} x_n$ means that f and $\lim\limits_{n \to \infty}$ can be exchanged.

3.4 Taylor–Series

The Taylor series is a representation of a function as an infinite sum of powers of x.
Goals:

1. Simple representation of functions as polynomials, i.e.:
$$f(x) \approx a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \cdots + a_n x^n$$
2. Approximation of functions in the neighborhood of a point x_0.

Ansatz:

$$P(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)^2 + a_3(x - x_0)^3 + \cdots + a_n(x - x_0)^n$$

coefficients $a_0, \cdots, a_n$ are sought such that

$$f(x) = P(x) + R_n(x)$$

with a remainder term $R_n(x)$ and $\lim_{n\to\infty} R_n(x) = 0$, ideally for all x.
We require for some point x_0 that
$f(x_0) = P(x_0), f'(x_0) = P'(x_0), \cdots, f^{(n)}(x_0) = P^{(n)}(x_0)$

Computation of Coefficients:

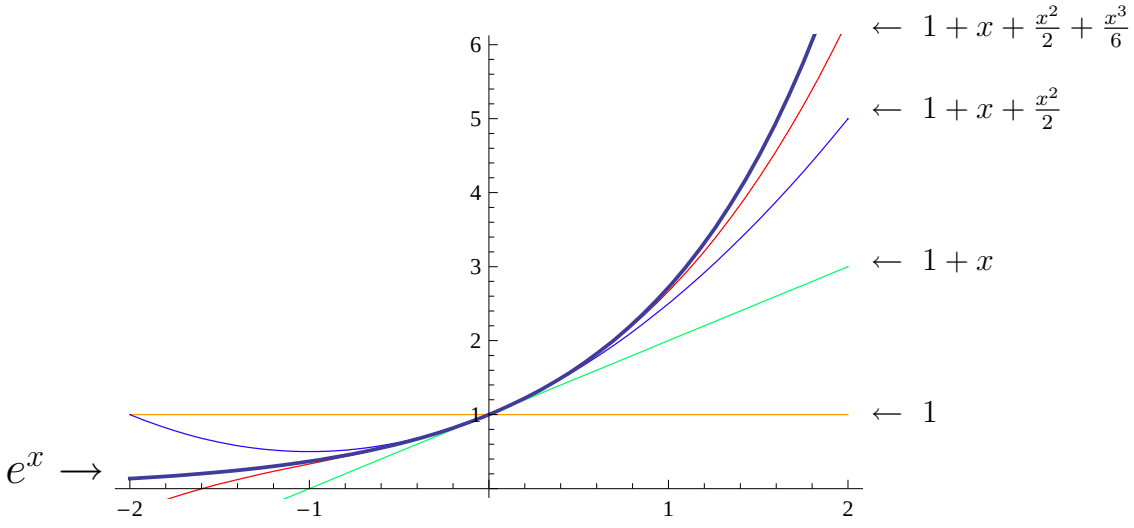
$$P(x_0) = a_0, \quad P'(x_0) = a_1, \quad P''(x_0) = 2a_2, \quad \cdots, \quad P^{(k)}(x_0) = k!a_k, \quad \cdots$$
$$\Rightarrow\ f^{(k)}(x_0) = k!a_k \ \Rightarrow\ a_k = \frac{f^{(k)}(x_0)}{k!}$$

Result:

$$f(x) = \underbrace{f(x_0) + \frac{f'(x_0)}{1!}(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n}_{P(x)} + R_n(x)$$

Example 3.13 Expansion of $f(x) = e^x$ in the point $x_0 = 0$:

$$f(x_0) = f(0) = 1, \quad f'(0) = 1, \quad f''(0) = 1, \quad \cdots, \quad f^{(n)} = 1$$
$$\Rightarrow\ e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots + \frac{x^n}{n!} + R_n(x)$$



Theorem 3.16 (Taylor Formula) Let $I \subset \mathbb{R}$ be an interval and $f : I \to \mathbb{R}$ a $(n+1)$–**times continuously differentiable** function. Then for $x \in I, x_0 \in I$ we have

$$f(x) = f(x_0) + \frac{f'(x_0)}{1!}(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n + R_n(x)$$

with

$$R_n(x) = \frac{1}{n!}\int_{x_0}^{x}(x - t)^n f^{(n+1)}(t)dt$$

Theorem 3.17 (Lagrangian form of the remainder term) Let $f : I \to \mathbb{R}$ $(n+1)$–times continuously differentiable and $x_0, x \in I$. Then there is a z between x_0 and x such that

$$R_n(x) = \frac{f^{(n+1)}(z)}{(n+1)!}(x - x_0)^{n+1}.$$

Example 3.14 $f(x) = e^x$ Theorems 3.16 and 3.17 yield

$$e^x = \sum_{k=0}^{n}\frac{x^k}{k!} + \underbrace{\frac{e^z}{(n+1)!}x^{n+1}}_{=R_n(x)} \quad for \quad |z| < |x|$$

Convergence:

$$|R_n(x)| \le \frac{e^{|x|}\,|x|^{n+1}}{(n+1)!} =: b_n$$

$$\left|\frac{b_{n+1}}{b_n}\right| = \frac{|x|}{n+2} \to 0 \quad for \quad n \to \infty$$

the ratio test implies convergence of $\sum_{n=0}^{\infty} b_n$.

$$\Rightarrow \lim_{n\to\infty} b_n = 0 \quad \Rightarrow \lim_{n\to\infty} R_n(x) = 0 \quad \textbf{for all } x \in \mathbb{R}$$

Thus the Taylor series for e^x converges to $f(x)$ for all $x \in \mathbb{R}$!

Example 3.15 Evaluation of the integral

$$\int_0^1 \sqrt{1 + x^3}\,dx.$$

As the function $f(x) = \sqrt{1 + x^3}$ has no simple antiderivative (primitive function), it can not be symbolically integrated. We compute an approximation for the integral by integrating the third order Taylor polynomial

$$\sqrt{1 + x^3} = (1 + x^3)^{1/2} \approx 1 + \frac{x^3}{2}$$

and substituting this into the integral

$$\int_0^1 \sqrt{1 + x^3}\,dx \approx \int_0^1 1 + \frac{x^3}{2}\,dx = \left[x + \frac{x^4}{8}\right]_0^1 = \frac{9}{8} = 1.125$$

The exact value of the integral is about 1.11145, i.e. our approximation error is about 1%.

Definition 3.12 The series $T_f(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!}(x-x_0)^k$ is called Taylor series of f with expansion point (point of approximation) x_0.

Note:

1. For $x = x_0$ every Taylor series converges.
2. But for $x \neq x_0$ not all Taylor series converge!
3. A Taylor series converges for exactly these $x \in I$ to $f(x)$ for which the remainder term from theorem 3.16 (3.17) converges to zero.
4. Even if the Taylor series of f converges, it does not necessarily converge to f. ($\rightarrow$ example in the exercises.)

Example 3.16 *(Logarithm series)* For $0 < x \leq 2$:

$$\ln(x) = (x-1) - \frac{(x-1)^2}{2} + \frac{(x-1)^3}{3} - \frac{(x-1)^4}{4} \pm \cdots$$

Proof:

$$\ln'(x) = \frac{1}{x}, \quad \ln''(x) = -\frac{1}{x^2}, \quad \ln'''(x) = \frac{2}{x^3}, \quad \ln^{(4)}(x) = -\frac{6}{x^4}, \quad \ln^{(n)}(x) = (-1)^{n-1}\frac{(n-1)!}{x^n}$$

Induction:

$$\ln^{(n+1)}(x) = \left(\ln(x)^{(n)}\right)' = \left((-1)^{(n-1)}\frac{(n-1)!}{x^n}\right)' = (-1)^n\frac{n!}{x^{n+1}}$$

Expansion at $x_0 = 1$

$$T_{ln,1}(x) = \sum_{k=0}^{\infty} \frac{\ln^{(k)}(1)}{k!}(x-1)^k = (x-1) - \frac{(x-1)^2}{2} + \frac{(x-1)^3}{3} - \frac{(x-1)^4}{4} \pm \cdots$$

This series converges only for $0 < x \leq 2$ (without proof).

Definition 3.13 If a Taylor series converges for all x in an interval I, we call I the **convergence area.**
Is $I = [x_0 - r, x_0 + r]$ or $I = (x_0 - r, x_0 + r)$, r is the **convergence radius** of the Taylor series.

Example 3.17 *Relativistic mass increase:*
Einstein: total energy: $E = mc^2$ kinetic energy: $E_{kin} = (m - m_0)c^2$

$$m(v) = \frac{m_0}{\sqrt{1 - \left(\frac{v}{c}\right)^2}}$$

to be shown: for $v \ll c$ we have $E_{kin} \approx \frac{1}{2} m_0 v^2$

$$E_{kin} = (m - m_0)c^2 = \left(\frac{1}{\sqrt{1 - \left(\frac{v}{c}\right)^2}} - 1 \right) m_0 c^2$$

$$\frac{1}{\sqrt{1-x}} = (1-x)^{-\frac{1}{2}} = 1 + \frac{1}{2}x + \frac{\left(-\frac{1}{2}\right)\left(-\frac{3}{2}\right)}{2!}x^2 + \cdots$$
$$= 1 + \frac{1}{2}x + \frac{3}{8}x^2 + \cdots$$

for $x \ll 1$:

$$\frac{1}{\sqrt{1-x}} \approx 1 + \frac{1}{2}x$$

$$\Rightarrow E_{kin} \approx \left(1 + \frac{1}{2}\frac{v^2}{c^2} - 1\right) m_0 c^2 = \frac{1}{2} m_0 v^2 + \frac{3}{8} m_0 \frac{v^4}{c^2} + \ldots$$

3.5 Differential Calculus in many Variables

$$f : \mathbb{R}^n \to \mathbb{R}$$
$$(x_1, x_2, \cdots, x_n) \mapsto y = f(x_1, x_2, \cdots, x_n)$$

or

$$\boldsymbol{x} \mapsto y = f(\boldsymbol{x})$$

3.5.1 The Vector Space $\mathbb{R}^n$

In order to "compare" vectors, we use a norm:

Definition 3.14 Any mapping $\| \quad \| : \mathbb{R}^n \to \mathbb{R}, \boldsymbol{x} \mapsto \|\boldsymbol{x}\|$ is called **Norm** if and only if

1. $\|\boldsymbol{x}\| = 0$ iff $x = 0$
2. $\|\lambda \boldsymbol{x}\| = |\lambda| \, \|\boldsymbol{x}\| \quad \forall \lambda \in \mathbb{R}, \boldsymbol{x} \in \mathbb{R}^n$
3. $\|\boldsymbol{x} + \boldsymbol{y}\| \leq \|\boldsymbol{x}\| + \|\boldsymbol{y}\| \quad \forall x, y \in \mathbb{R}^n$ triangle inequation

the particular norm we will use here is the

Definition 3.15 *(Euklidian Norm)*
The function $| \quad | : \mathbb{R}^n \to \mathbb{R}^+ \cup \{0\}, \boldsymbol{x} \mapsto \sqrt{x_1^2 + \cdots + x_n^2}$ is the **Euklidian Norm** of the vector $\boldsymbol{x}$.

Lemma: Die Euklidian norm is a norm.

Theorem 3.18 For $\boldsymbol{x} \in \mathbb{R}^n$ we have $\boldsymbol{x}^2 = \boldsymbol{x}\boldsymbol{x} = |\boldsymbol{x}|^2$

Proof as exercise.

Note: The scalar product in $\mathbb{R}^n$ induces the Euklidian norm.

3.5.2 Sequences and Series in $\mathbb{R}^n$

analogous to Sequences and Series in $\mathbb{R}$!

Definition 3.16 A mapping $N \to \mathbb{R}^n, n \mapsto \boldsymbol{a}_n$ is called sequence.
Notation: $(\boldsymbol{a}_n)_{n\in\mathbb{N}}$

Example 3.18

$$\begin{pmatrix}1\\1\\1\end{pmatrix}, \begin{pmatrix}2\\\frac{1}{2}\\\frac{1}{2}\end{pmatrix}, \begin{pmatrix}3\\\frac{1}{3}\\\frac{1}{4}\end{pmatrix}, \begin{pmatrix}4\\\frac{1}{4}\\\frac{1}{8}\end{pmatrix}, \begin{pmatrix}5\\\frac{1}{5}\\\frac{1}{16}\end{pmatrix}, \cdots = \begin{pmatrix}\begin{pmatrix}n\\\frac{1}{n}\\\frac{1}{2^{n-1}}\end{pmatrix}\end{pmatrix}_{n\in\mathbb{N}}$$

Definition 3.17 A sequence $(\boldsymbol{a}_n)_{n\in\mathbb{N}}$ of vectors $\boldsymbol{a}_n \in \mathbb{R}^n$ **converges** to $\boldsymbol{a} \in \mathbb{R}^n$, if

$$\forall \varepsilon > 0 \quad \exists N(\varepsilon) \in N \,|\boldsymbol{a}_n - \boldsymbol{a}| < \varepsilon \quad \forall \quad n \geq N(\varepsilon)$$

Notation: $\lim\limits_{n\to\infty} \boldsymbol{a}_n = \boldsymbol{a}$

Theorem 3.19 A (vector) sequence $(\boldsymbol{a}_n)_{n\in\mathbb{N}}$ converges to $\boldsymbol{a}$ if and only if all its coordinate sequences converge to the respective coordinates of $\boldsymbol{a}$. *(Proof as exercise.)*
Notation:

$$\boldsymbol{a}_k = \begin{pmatrix}a_1^k\\\vdots\\a_n^k\end{pmatrix} \qquad (\boldsymbol{a}_k)_{k\in\mathbb{N}} \quad \boldsymbol{a}_k \in \mathbb{R}^n$$

Note: Theorem 3.19 enables us to lift most properties of sequences of real numbers to sequences of vectors.

3.5.3 Functions from $\mathbb{R}^n$ to $\mathbb{R}^m$

$m = 1$: Functions f from $D \subset \mathbb{R}^n$ to $B \subset \mathbb{R}$ have the form

$$f : D \to B \quad , \quad \boldsymbol{x} \mapsto f(\boldsymbol{x})$$

$$\begin{pmatrix}x_1\\\vdots\\x_n\end{pmatrix} \mapsto f(x_1, \cdots, x_n)$$

Example 3.19

$f(x_1, x_2) = \sin(x_1 + \ln x_2)$
$m \neq 1$: Functions f from $D \subset \mathbb{R}^n$ to $B \subset \mathbb{R}^m$ have the form

$$f : D \to B \quad , \quad \boldsymbol{x} \mapsto f(\boldsymbol{x})$$

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \mapsto \begin{pmatrix} f_1(x_1, \cdots, x_n) \\ \vdots \\ f_m(x_1, \cdots, x_n) \end{pmatrix}$$

Example 3.20

1.
$$f : \mathbb{R}^3 \to \mathbb{R}^2$$
$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \mapsto \begin{pmatrix} \sqrt{x_1 x_2 x_3} \\ \cos x_1 + \sin x_2 \end{pmatrix}$$

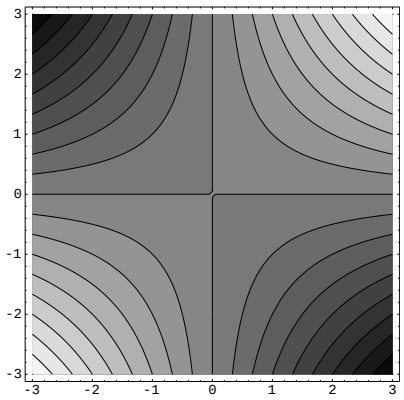
2. Weather parameters: temperature, air pressure and humidity at any point on the earth
$$f : [0°, 360°] \times [-90°, 90°] \to [-270, \infty] \times [0, \infty] \times [0, 100\%]$$
$$\begin{pmatrix} \Theta \\ \Phi \end{pmatrix} \mapsto \begin{pmatrix} temperature(\Theta, \Phi) \\ airpressure(\Theta, \Phi) \\ humidity(\Theta, \Phi) \end{pmatrix}$$

Note: The components $f_1(\boldsymbol{x}), \cdots, f_m(\boldsymbol{x})$ can be viewed (analysed) independently. Thus, in the following we can restrict ourselves to $f : \mathbb{R}^n \to \mathbb{R}$.

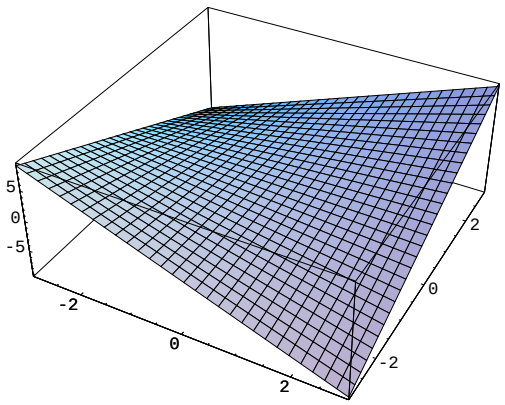
3.5.3.1 Contour Plots

Definition 3.18 Let $D \subset \mathbb{R}^2, B \subset \mathbb{R}, c \in B, f : D \to B$. The set $\{(x_1, x_2) | f(x_1, x_2) = c\}$ is called contour of f to the niveau c.

Example 3.21 $f(x_1, x_2) = x_1 x_2$
$$x_1 x_2 = c$$
for
$$x_1 \neq 0 : x_2 = \frac{c}{x_1}$$
(hyperbolas)
$$c = 0 \quad \Leftrightarrow \quad x_1 = 0 \vee x_2 = 0$$




```
ContourPlot[x y, {x,-3,3}, {y,-3,3},
  Contours -> {0,1,2,3,4,5,6,7,8,9,-1,
               -2,-3,-4,-5,-6,-7,-8,-9},
  PlotPoints -> 60]
```




```
Plot3D[x y, {x,-3,3}, {y,-3,3},
       PlotPoints -> 30]
```

3.5.4 Continuity in $\mathbb{R}^n$

analogous to continuity of functions in one variable:

Definition 3.19 Let $\boldsymbol{f} : D \to \mathbb{R}^m$ a function and $\boldsymbol{a} \in \mathbb{R}^n$. If there is a sequence $(\boldsymbol{a}_n)$ (maybe more than one sequence) with $\lim\limits_{n\to\infty} \boldsymbol{a}_n = \boldsymbol{a}$, we write

$$\lim_{\boldsymbol{x}\to\boldsymbol{a}} \boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{c},$$

if for any sequence $(\boldsymbol{x}_n), \boldsymbol{x}_n \in D$ with $\lim\limits_{n\to\infty} \boldsymbol{x}_n = \boldsymbol{a}$:

$$\lim_{n\to\infty} \boldsymbol{f}(\boldsymbol{x}_n) = \boldsymbol{c}$$

Definition 3.20 *(Continuity)*
Let $\boldsymbol{f} : D \to \mathbb{R}^m$ a function and $\boldsymbol{a} \in D$. The function $\boldsymbol{f}$ is **continuous** in $\boldsymbol{a}$, if $\lim\limits_{\boldsymbol{x}\to\boldsymbol{a}} \boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{a})$. $\boldsymbol{f}$ is continuous in D, if $\boldsymbol{f}$ is continuous in all points in D.

Note: These definitions are analogous to the one-dimensional case.

Theorem 3.20 If $\boldsymbol{f} : D \to \mathbb{R}^m, \boldsymbol{g} : D \to \mathbb{R}^m, h : D \to \mathbb{R}$ are continuous in $\boldsymbol{x}_0 \in D$, then $\boldsymbol{f} + \boldsymbol{g}, \boldsymbol{f} - \boldsymbol{g}, \boldsymbol{f}\,\boldsymbol{g}$ and $\frac{\boldsymbol{f}}{h}$ (if $h(\boldsymbol{x}_0)) \neq 0$) are continuous in $\boldsymbol{x}_0$.

3.5.5 Differentiation of Functions in $\mathbb{R}^n$

3.5.5.1 Partial Derivatives

Example 3.22

$$f : \mathbb{R}^2 \to \mathbb{R}$$
$$f(x_1, x_2) = 2x_1^2 x_2^3$$

keep $x_2 = const.$, and compute the 1–dim. derivative of f w.r.t. x_1:

$$\frac{\partial f}{\partial x_1}(x_1, x_2) = f_{x_1}(x_1, x_2) = 4x_1 x_2^3$$

analogous with $x_1 = const.$

$$\frac{\partial f}{\partial x_2} = 6x_1^2 x_2^2$$

second derivatives:

$$\left.\begin{array}{l} \frac{\partial}{\partial x_2}\frac{\partial}{\partial x_1}(x_1, x_2) = 12x_1x_2^2 \\ \frac{\partial}{\partial x_1}\frac{\partial}{\partial x_2}(x_1, x_2) = 12x_1x_2^2 \end{array}\right\} \Rightarrow \frac{\partial}{\partial x_1}\frac{\partial f}{\partial x_2} = \frac{\partial}{\partial x_2}\frac{\partial f}{\partial x_1}$$

Example 3.23

$$\Phi(u,v,w) = uv + \cos w$$
$$\Phi_u(u,v,w) = v$$
$$\Phi_v(u,v,w) = u$$
$$\Phi_w(u,v,w) = -\sin w$$

Definition 3.21 If $\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} f_1(x_1,\cdots,x_n) \\ \vdots \\ f_m(x_1,\cdots,x_n) \end{pmatrix}$ is partially differentiable in $\boldsymbol{x} = \boldsymbol{x}_0$, i.e. all partial Derivatives $\frac{\partial f_i}{\partial x_k}(\boldsymbol{x}_0)(i = 1,\cdots,m, k = 1,\cdots,n)$ exist, then the matrix

$$\boldsymbol{f}'(\boldsymbol{x}_0) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(\boldsymbol{x}_0) & \frac{\partial f_1}{\partial x_2}(\boldsymbol{x}_0) & \cdots & \frac{\partial f_1}{\partial x_n}(\boldsymbol{x}_0) \\ \frac{\partial f_2}{\partial x_1}(\boldsymbol{x}_0) & \frac{\partial f_2}{\partial x_2}(\boldsymbol{x}_0) & \cdots & \frac{\partial f_2}{\partial x_n}(\boldsymbol{x}_0) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_m}{\partial x_1}(\boldsymbol{x}_0) & \frac{\partial f_m}{\partial x_2}(\boldsymbol{x}_0) & \cdots & \frac{\partial f_m}{\partial x_n}(\boldsymbol{x}_0) \end{pmatrix}$$

is called **Jacobian matrix**.

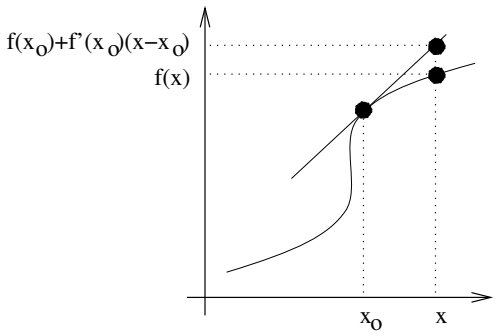
Example 3.24 Linearisation of a function: $f : \mathbb{R}^2 \to \mathbb{R}^3$ in $\boldsymbol{x}_0$

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} 2x_2 \\ \sin(x_1 + x_2) \\ \ln(x_1) + x_2 \end{pmatrix}$$

$$\boldsymbol{f}'(\boldsymbol{x}) = \begin{pmatrix} 0 & 2 \\ \cos(x_1 + x_2) & \cos(x_1 + x_2) \\ \frac{1}{x_1} & 1 \end{pmatrix}$$

1–dimensional

$$f'(x_0) = \lim_{x \mapsto x_0} \frac{f(x) - f(x_0)}{x - x_0}$$



Linearisation $\boldsymbol{g}$ of $\boldsymbol{f}$ in $\boldsymbol{x}_0 = \binom{\pi}{0}$

$$\boldsymbol{g}(x_1,x_2) = \boldsymbol{f}(\pi,0) + \boldsymbol{f}'(\pi,0)\left[\binom{x_1}{x_2} - \binom{\pi}{0}\right]$$

$$\Rightarrow \boldsymbol{g}(x_1,x_2) = \begin{pmatrix} 0 \\ 0 \\ \ln\pi \end{pmatrix} + \begin{pmatrix} 0 & 2 \\ -1 & -1 \\ \frac{1}{\pi} & 1 \end{pmatrix} \begin{pmatrix} x_1 - \pi \\ x_2 \end{pmatrix} = \begin{pmatrix} 2x_2 \\ -x_1 - x_2 + \pi \\ \frac{x_1}{\pi} + x_2 + \ln\pi - 1 \end{pmatrix}$$

Note: For $\boldsymbol{x} \to \boldsymbol{x}_0$ i.e. close to $\boldsymbol{x}_0$ the linearisation $\boldsymbol{g}$ is a good approximation to $\boldsymbol{f}$ (under which condition?).

Example 3.25 We examine the function $f : \mathbb{R}^2 \to \mathbb{R}$ with

$$f(x,y) = \begin{cases} \frac{x\,y}{\sqrt{x^2+y^2}} & \text{if } (x,y) \neq (0,0) \\ 0 & \text{if } (x,y) = (0,0) \end{cases}$$

Differentiability: f is differentiable on $\mathbb{R}^2 \backslash \{(0,0)\}$ since it is built up of differentiable functions by sum, product and division.

$$\begin{aligned} \frac{\partial f}{\partial x}(x,y) &= \frac{y}{\sqrt{x^2+y^2}} - \frac{x^2\,y}{(x^2+y^2)^{\frac{3}{2}}} \\ \frac{\partial f}{\partial x}(0,y) &= \frac{y}{y} = 1 \\ \frac{\partial f}{\partial x}(x,0) &= 0 \\ \Rightarrow \lim_{y\to 0} \frac{\partial f}{\partial x}(0,y) &\neq \lim_{x\to 0} \frac{\partial f}{\partial x}(x,0) \end{aligned}$$

$\Rightarrow$ the partial derivative $\frac{\partial f}{\partial x}$ is not continuous in $(0,0)$. $\Rightarrow f$ is in $(0,0)$ not differentiable .

Symmetries:

1. f is symmetric wrt. exchange of x and y, i.e. w.r.t. the plane $y = x$.
2. f is symmetric wrt. exchange of x and $-y$, i.e. w.r.t. the plane $y = -x$.
3. $f(-x,y) = -f(x,y)$, d.h. f is symmetric w.r.t. the y-axis.
4. $f(x,-y) = -f(x,y)$, d.h. f is symmetric w.r.t. the x-axis.

Contours:

$$\frac{x\,y}{\sqrt{x^2+y^2}} = c \quad \Leftrightarrow \quad x\,y = c\,\sqrt{x^2+y^2} \quad \Rightarrow \quad x^2\,y^2 = c^2\,(x^2+y^2)$$

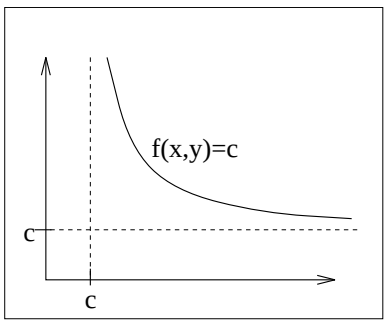
$$\Leftrightarrow \quad y^2(x^2-c^2) = c^2x^2 \quad \Rightarrow \quad y = \pm\frac{cx}{\sqrt{x^2-c^2}} \quad \Rightarrow$$

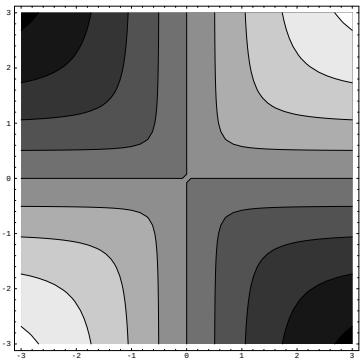
Contours:

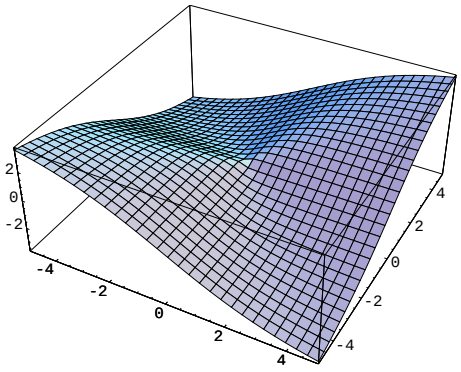
$$y = \begin{cases} \frac{cx}{\sqrt{x^2-c^2}} & \text{if } c>0, x>0 \text{ (1. Quadr.) and } c<0, x<0 \text{ (2. Quadr.)} \\ -\frac{cx}{\sqrt{x^2-c^2}} & \text{if } c>0, x<0 \text{ (3. Quadr.) and } c<0, x>0 \text{ (4. Quadr.)} \end{cases}$$

Signs in the quadrants:

-	+
+	-







Continuity: f is continuous on $\mathbb{R}^2\backslash\{(0,0)\}$, since it is built up of continuous functions by sum, product and division.

Continuity in $(0,0)$:
Let $\varepsilon > 0$ such that $|\boldsymbol{x}| = \varepsilon$, i.e. $\varepsilon = \sqrt{x^2+y^2} \Leftrightarrow y = \pm\sqrt{\varepsilon^2 - x^2}$

$$\Rightarrow f(x,y) = \pm\frac{x\sqrt{\varepsilon^2-x^2}}{\varepsilon} = \pm\frac{x\,\varepsilon\sqrt{1-x^2/\varepsilon^2}}{\varepsilon} = \pm x\sqrt{1-x^2/\varepsilon^2}$$

from $|x| \leq \varepsilon$ we get

$$|f(x,y)| \leq |x| = \varepsilon \quad \textit{and} \quad \lim_{\boldsymbol{x}\to 0} f(x,y) = 0$$

Thus f is continuous in $(0,0)$.

3.5.5.2 The Gradient

Definition 3.22 $f : D \to \mathbb{R}(D \subset \mathbb{R}^n)$

The Vector $\text{grad} f(\boldsymbol{x}) := f'(\boldsymbol{x})^T = \begin{pmatrix} \frac{\partial f}{\partial x_1}(\boldsymbol{x}) \\ \vdots \\ \frac{\partial f}{\partial x_n}(\boldsymbol{x}) \end{pmatrix}$ is called **gradient** of f.

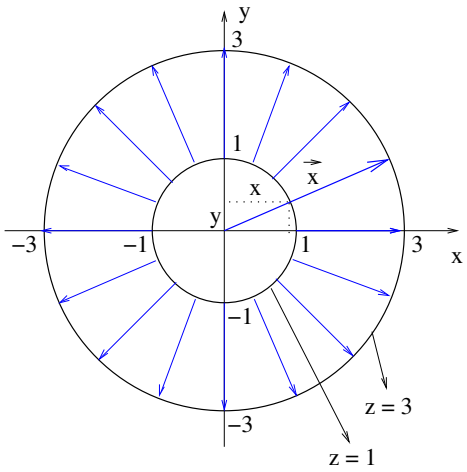
The gradient of f points in the direction of the steepest ascent of f.

Example 3.26

$$f(x,y) = x^2 + y^2$$

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

$$\Rightarrow \text{grad} f(x,y) = \begin{pmatrix} 2x \\ 2y \end{pmatrix} = 2 \begin{pmatrix} x \\ y \end{pmatrix}$$



3.5.5.3 Higher Partial Derivatives

Let $\boldsymbol{f} : D \to \mathbb{R}^m (D \subset \mathbb{R}^n)$. Thus $\frac{\partial \boldsymbol{f}}{\partial x_i}(x)$ is again a function mapping from $\overset{\circ}{D}$ to $\mathbb{R}^m$ and

$$\frac{\partial}{\partial x_k}\left(\frac{\partial \boldsymbol{f}}{\partial x_i}\right)(\boldsymbol{x}) =: \frac{\partial^2 f}{\partial x_k \partial x_i}(\boldsymbol{x}) = \boldsymbol{f}_{x_i,x_k}(\boldsymbol{x})$$

is well defined.

Theorem 3.21 Let $D \subset \mathbb{R}^n$ open and $\boldsymbol{f} : D \to \mathbb{R}^m$ two times partially differentiable. Then we have for all $\boldsymbol{x}_0 \in D$ and all $i, j = 1, \cdots, n$

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(\boldsymbol{x}_0) = \frac{\partial^2 f}{\partial x_j \partial x_i}(\boldsymbol{x}_0)$$

Consequence: If $\boldsymbol{f} : D \to \mathbb{R}^n (D \subset \mathbb{R}^n$ open) is k–times continuously partially differentiable, then

$$\frac{\partial^k \boldsymbol{f}}{\partial x_{i_k} \partial x_{i_{k-1}} \cdots \partial x_{i_1}} = \frac{\partial^k \boldsymbol{f}}{\partial x_{i_{\Pi(k)}} \ldots \partial x_{i_{\Pi(1)}}}$$

for any Permutation Π of the numbers $1, \ldots, k$.

3.5.5.4 The Total Differential

If $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}$ is differentiable, then the tangential mapping $f_t(\boldsymbol{x}) = f(\boldsymbol{x}_0) + f'(\boldsymbol{x}_0)(\boldsymbol{x} - \boldsymbol{x}_0)$ represents a good approximation to the function f in the neighborhood of $\boldsymbol{x}_0$ which can be seen in

$$f_t(\boldsymbol{x}) - f(\boldsymbol{x}_0) = f'(\boldsymbol{x}_0)(\boldsymbol{x} - \boldsymbol{x}_0).$$

With

$$df(\boldsymbol{x}) := f_t(\boldsymbol{x}) - f(\boldsymbol{x}_0) \approx f(\boldsymbol{x}) - f(\boldsymbol{x}_0)$$

and

$$d\boldsymbol{x} = \begin{pmatrix} dx_1 \\ \vdots \\ dx_n \end{pmatrix} := \boldsymbol{x} - \boldsymbol{x}_0$$

we get:

$$df(\boldsymbol{x}) = f'(\boldsymbol{x}_0)d\boldsymbol{x}$$

or

$$df(\boldsymbol{x}) = \sum_{k=1}^{n} \frac{\partial f}{\partial x_k}(\boldsymbol{x}_0)dx_k = \frac{\partial f}{\partial x_1}(\boldsymbol{x}_0)dx_1 + \cdots + \frac{\partial f}{\partial x_n}dx_n$$

Definition 3.23 The linear mapping $df = \sum_{k=1}^{n} \frac{\partial f}{\partial x_k}(\boldsymbol{x}_0)dx_k$ is called **total differential** of the function f in $\boldsymbol{x}_0$.

Note: Since in a neighborhood of $\boldsymbol{x}_0$, f_t is a good approximation of the function f, we have for all $\boldsymbol{x}$ close to $\boldsymbol{x}_0$:

$$df(\boldsymbol{x}) \approx f(\boldsymbol{x}) - f(\boldsymbol{x}_0).$$

Thus $df(\boldsymbol{x})$ gives the approximate deviation of the function value $f(\boldsymbol{x})$ from $f(\boldsymbol{x}_0)$, when $\boldsymbol{x}$ deviates from $\boldsymbol{x}_0$ a little bit.

3.5.5.5 Application: The Law of Error Propagation

Example 3.27 For a distance of $s = 10\,km$ a runner needs the time of $t = 30\,min$ yielding an average speed of $v = \frac{s}{t} = 20\frac{km}{h}$. Let the measurement error for the distance s be $\Delta s = \pm 1\,m$ and for the time we have $\Delta t = \pm 1\,sec$. Give an upper bound on the propagated error Δv for the average speed!

This can be solved as follows. To the given measurements $x_1, \cdots, x_n$, a function $f : \mathbb{R}^n \to \mathbb{R}$ has to be applied. The measurement error for $x_1, \cdots, x_n$ is given as $\pm\Delta x_1, \cdots, \pm\Delta x_n$ ($\Delta x_i > 0 \quad \forall i = 1, \cdots, n$). The law of error propagation gives as a rough upper bound for the error $\Delta f(\boldsymbol{x})$ of $f(x_1, \cdots, x_n)$ the assessment

$$\Delta f(x_1, \cdots, x_n) < \left|\frac{\partial f}{\partial x_1}(\boldsymbol{x})\right| \Delta x_1 + \ldots + \left|\frac{\partial f}{\partial x_n}(\boldsymbol{x})\right| \Delta x_n$$

Definition 3.24 We call

$$\Delta f_{max}(x_1, \cdots, x_n) := \left|\frac{\partial f}{\partial x_1}(\boldsymbol{x})\right| \Delta x_1 + \ldots + \left|\frac{\partial f}{\partial x_m}(\boldsymbol{x})\right| \Delta x_n$$

the **maximum error** of f. The ratio $\frac{\Delta f_{max}(\boldsymbol{x})}{f(\boldsymbol{x})}$ is the **relative maximum error**.

Note: Δf_{max} typically gives a too high estimate for the error of f, because this value only occurs if all measurement errors $dx_1, \cdots, dx_n$ add up with the same sign. This formula should be applied for about $n \leq 5$.

Definition 3.25 When the number of measurements n becomes large, a better estimate for the error Δf is given by the formula

$$\Delta f_{mean}(x_1, \cdots, x_n) := \sqrt{\left(\frac{\partial f}{\partial x_1}(\boldsymbol{x})\right)^2 \Delta x_1 + \ldots + \left(\frac{\partial f}{\partial x_m}(\boldsymbol{x})\right)^2 \Delta x_n}$$

for the **mean error** of f.

Example 3.28 Solution fo example 3.27. Application of the maximum error formula leads to

$$\begin{aligned}\Delta v(s,t) &= \left|\frac{\partial v}{\partial s}(s,t)\right| \Delta s + \left|\frac{\partial v}{\partial t}(s,t)\right| \Delta t = \left|\frac{1}{t}\right| \Delta s + \left|-\frac{s}{t^2}\right| \Delta t = \frac{\Delta s}{t} + \frac{s}{t^2}\Delta t \\ &= \frac{0.001}{0.5}\frac{km}{h} + \frac{10}{0.25}\frac{km}{h^2}\frac{1}{3600}h = \left(0.002 + \frac{40}{3600}\right)\frac{km}{h} = 0.013\frac{km}{h}\end{aligned}$$

This can be compactly written as the result $v = (20 \pm 0.013)\frac{km}{h}$.

Definition 3.26 Let $f : D \to \mathbb{R}$ two times continuously differentiable. The $n \times n$–Matrix

$$(\mathrm{Hess} f)(\boldsymbol{x}) := \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2}(\boldsymbol{x}) & \ldots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(\boldsymbol{x}) \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(\boldsymbol{x}) & \ldots & \frac{\partial^2 f}{\partial x_n^2}(\boldsymbol{x}) \end{pmatrix}$$

is the **Hesse–Matrix** of f in $\boldsymbol{x}$.

Note: Hess$f$ is symmetric, since

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial^2 f}{\partial x_j \partial x_i}$$

3.5.6 Extrema without Constraints

Again we appeal to your memories of one–dimensional analysis: How do you determine extrema of a function $f : \mathbb{R} \to \mathbb{R}$? This is just a special case of what we do now.

Definition 3.27 Let $D \subset \mathbb{R}^n$ and $f : D \to \mathbb{R}$ a function. A point $\boldsymbol{x} \in D$ is a **local maximum (minimum)** of f, if there is a neighborhood $U \subset D$ of $\boldsymbol{x}$ such that

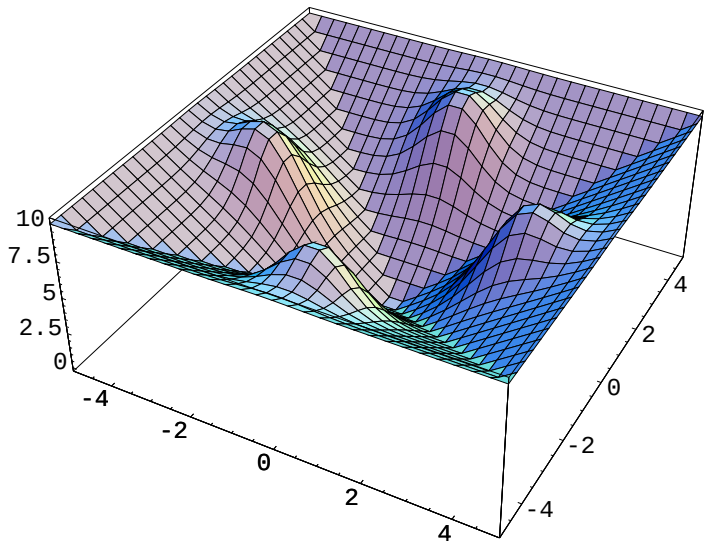
$$f(\boldsymbol{x}) \geq f(\boldsymbol{y}) \qquad (f(\boldsymbol{x}) \leq f(\boldsymbol{y})) \qquad \forall \boldsymbol{y} \in U.$$

Analogously, we have an **isolated local Maximum (Minimum)** in $\boldsymbol{x}$, if there is a neighborhood $U \subset D$ of $\boldsymbol{x}$ such that

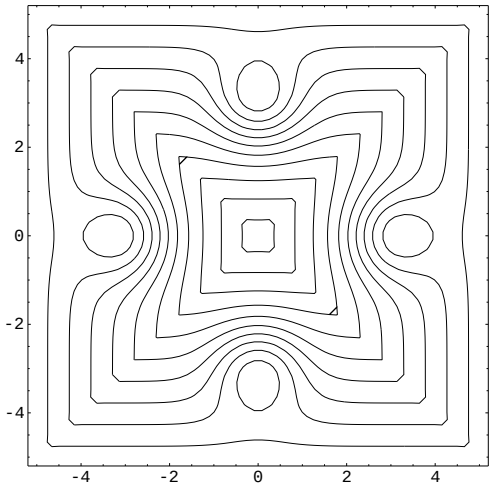
$$f(\boldsymbol{x}) > f(\boldsymbol{y}) \qquad (\text{bzw. } f(\boldsymbol{x}) < f(\boldsymbol{y})) \qquad \forall \boldsymbol{y} \in U, \quad \boldsymbol{y} \neq \boldsymbol{x}$$

All these points are called **extrema**.
If the mentioned neighborhood U of an extremum is the whole domain, i.e. $U = D$, then the extremum is **global**.

Give all local, global, isolated and non-isolated maxima and minima of the function shown in the following graphs:




```
Plot3D[f[x,y], {x,-5,5},{y,-5,5}, PlotPoints
-> 30]
```




```
ContourPlot[f[x,y],
{x,-5,5},{y,-5,5}, PlotPoints
-> 60, ContourSmoothing ->
True,ContourShading-> False]
```

Theorem 3.22 Let $D \subset \mathbb{R}^n$ be open and $f : D \to R$ partially differentiable . If f has a local extremum in $\boldsymbol{x} \in D$, then $\operatorname{grad} f(\boldsymbol{x}) = 0$.

Proof: Reduction on 1–dim. case:
For $i = 1, \cdots, n$ define $g_i(h) := f(x_1, \cdots, x_i + h, \cdots, x_n)$. If f has a local extremum in $\boldsymbol{x}$, then all g_i have a local extremum in 0. Thus we have for all i: $g_i'(0) = 0$. Since $g_i'(0) = \frac{\partial f(\boldsymbol{x})}{\partial x_i}$ we get

$$\operatorname{grad} f(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1}(\boldsymbol{x}) \\ \vdots \\ \frac{\partial f}{\partial x_n}(\boldsymbol{x}) \end{pmatrix} = 0$$

Note:

- Theorem 3.22 represents a necessary condition for local extrema.

- Why is the proposition of Theorem 3.22 false if $D \subset \mathbb{R}^n$ is no open set?

Linear algebra reminder:

Definition 3.28 Let A a symmetric $n \times n$–Matrix of real numbers.

A is **positive (negative) definite**, if all eigenvalues of A are positive (negative).
A is **positive (negative) semidefinite**, if all eigenvalues are ≥ 0 (≤ 0).
A is **indefinite**, if all eigenvalues are $\neq 0$ and there exist positive as well as negative eigenvalues.

Theorem 3.23 Criterium of Hurwitz
Let A real valued symmetric matrix. A ist positive definite, if and only if for $k = 1, \cdots, n$

$$\begin{vmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} \end{vmatrix} > 0$$

A is negative definite if and only if -A is positive definite.

Theorem 3.24 For $D \subset \mathbb{R}^n$ open and two times continuously differentiable $f : D \to \mathbb{R}$ with $\operatorname{grad} f(\boldsymbol{x}) = 0$ for $\boldsymbol{x} \in D$ the following holds:

a) $(\operatorname{Hess} f)(\boldsymbol{x})$ positive definite $\Rightarrow$ f has in $\boldsymbol{x}$ an isolated minimum

b) $(\operatorname{Hess} f)(\boldsymbol{x})$ negative definite $\Rightarrow$ f has in $\boldsymbol{x}$ an isolated maximum

c) $(\operatorname{Hess} f)(\boldsymbol{x})$ indefinite $\Rightarrow$ f has in $\boldsymbol{x}$ **no local extremum.**

Note: Theorem 3.24 is void if $(\operatorname{Hess} f)(\boldsymbol{x})$ is positive oder negative semidefinite.

Procedure for the application of theorems 3.22 and 3.23 to search local extrema of a function $f : (D \subset \mathbb{R}^n) \to \mathbb{R}$:

1. Computation of $\operatorname{grad} f$
2. Computation of the zeros $\operatorname{grad} f$
3. Computation of the Hessian matrix $\operatorname{Hess} f$
4. Evaluation of $\operatorname{Hess} f(\boldsymbol{x})$ for all zeros $\boldsymbol{x}$ of $\operatorname{grad} f$.

Example 3.29 Some simple functions $f : \mathbb{R}^2 \to \mathbb{R}$:

1. $f(x, y) = x^2 + y^2 + c$

$$\operatorname{grad} f(x, y) = \begin{pmatrix} 2x \\ 2y \end{pmatrix} \Rightarrow \operatorname{grad} f(0, 0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} = 0$$

$$\mathrm{Hess}f = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$

is positive definite on all $\mathbb{R}^2$. $\Rightarrow$ f has an isolated local minimum in 0 (paraboloid).

2. $f(x,y) = -x^2 - y^2 + c$

$$\mathrm{grad}f(0,0) = 0 \quad \mathrm{Hess}f = \begin{pmatrix} -2 & 0 \\ 0 & -2 \end{pmatrix}$$

$\Rightarrow$ isolated local maximum in 0 (paraboloid).

3. $f(x,y) = ax + by + c \quad a, b \neq 0$

$$\mathrm{grad}f = \begin{pmatrix} a \\ b \end{pmatrix} \neq 0 \quad \forall \boldsymbol{x} \in \mathbb{R}^2$$

$\Rightarrow$ no local extremum.

4. $f(x,y) = x^2 - y^2 + c$

$$\mathrm{grad}f(x,y) = \begin{pmatrix} 2x \\ -2y \end{pmatrix} \Rightarrow \mathrm{grad}f(0,0) = 0$$

$$\mathrm{Hess}f = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix}$$

$\Rightarrow$ Hess$f$ indefinite $\Rightarrow$ f has no local extremum.

5. $f(x,y) = x^2 + y^4$

$$\mathrm{grad}f = \begin{pmatrix} 2x \\ 4y^3 \end{pmatrix} \Rightarrow \mathrm{grad}f(0,0) = 0$$

$$\mathrm{Hess}f(0,0) = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$$

$\Rightarrow$ Hess$f$ positive smidefinite, but f has in 0 an isolated minimum.

6. $f(x,y) = x^2$

$$\mathrm{grad}f = \begin{pmatrix} 2x \\ 0 \end{pmatrix} \Rightarrow \mathrm{grad}f(0,y) = 0$$

$$\mathrm{Hess}f(0,0) = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$$

$\Rightarrow$ Hess$f$ positive semidefinite, but f has a (non isolated) local minimum. All points on the y–axis ($x = 0$) are local minima.

7. $f(x,y) = x^2 + y^3$

$$\mathrm{grad}f(x,y) = \begin{pmatrix} 2x \\ 3y^2 \end{pmatrix} \Rightarrow \mathrm{grad}f(0,0) = 0$$

$$\mathrm{Hess}f(0,0) = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}$$

$\Rightarrow$ Hess$f$ positive semidefinite, but f has no local extremum.

3.5.7 Extrema with Constraints

Example 3.30 Which rectangle (length x, width y) has maximal area, given the perimeter U.

Area $f(x,y) = xy$. The function $f(x,y)$ has no local maximum on $\mathbb{R}^2$!

Constraint: $U = 2(x+y)$ or $x + y = \frac{U}{2}$ substituted in $f(x,y) = xy$

$$\Rightarrow g(x) := f(x, \frac{U}{2} - x) = x(\frac{U}{2} - x) = \frac{U}{2}x - x^2$$

$$g'(x) = \frac{U}{2} - 2x = 0$$

$$\left.\begin{array}{l} x = \frac{U}{4} \\ y = \frac{U}{4} \end{array}\right\} \Rightarrow \underline{\underline{x = y}}$$

$$g''(\frac{U}{4}) = -2$$

$\Rightarrow x = y = U/4$ ist (the unique) maximum of the area for constant perimeter U!

In many cases substitution of constraints is not feasible!

Wanted: Extremum of a function $f(x_1, \cdots, x_n)$ under the p constraints

$$\begin{array}{c} h_1(x_1, \cdots, x_n) = 0 \\ \vdots \\ h_p(x_1, \cdots, x_n) = 0 \end{array}$$

Theorem 3.25 Let $f : D \to \mathbb{R}$ and $\boldsymbol{h} : D \to \mathbb{R}^p$ be continuously differentiable functions on an open set $D \subset \mathbb{R}^n, n > p$ and the matrix $\boldsymbol{h}'(\boldsymbol{x})$ has rank p for all $\boldsymbol{x} \in D$.

If $\boldsymbol{x_0} \in D$ is an **extremum of** f **under the constraint(s)** $\boldsymbol{h}(\boldsymbol{x}_0) = 0$, there exist real numbers $\lambda_1, \cdots, \lambda_p$ with

$$\frac{\partial f}{\partial x_i}(\boldsymbol{x}_0) + \sum_{k=1}^{p} \lambda_k \frac{\partial h_k}{\partial x_i}(\boldsymbol{x}_0) = 0 \quad \forall i = 1, \cdots, n$$

and

$$h_k(\boldsymbol{x}_0) = 0 \quad \forall k = 1, \cdots, p$$

Illustration:
For $p = 1$, i.e. only one given constraint, the theorem implies that for an extremum $\boldsymbol{x_0}$ of f under the constraint $h(\boldsymbol{x}_0) = 0$ we have

$$\text{grad} f(\boldsymbol{x}_0) + \lambda \text{grad} h(\boldsymbol{x}_0) = 0$$

- grad$f$ and grad$h$ are parallel in the extremum $\boldsymbol{x_0}$!
- $\Rightarrow$ Contours of f and h for $h(\boldsymbol{x}) = 0$ are parallel in $\boldsymbol{x_0}$.

- The numbers $\lambda_1, \cdots, \lambda_p$ are the **Lagrange multipliers**.

Note: We have to solve $n+p$ equations with $n+p$ unknowns. Among the solutions of this (possibly nonlinear) system the extrema have to be determined. Not all solutions need to be extrema of f under the constraint(s) $h(\boldsymbol{x}_0) = 0$ (necessary but not sufficient condition for extrema.)

Definition 3.29 Let f, h be given as in theorem 3.25. The function $L : D \to \mathbb{R}$

$$L(x_1, \cdots, x_n) = f(x_1, \cdots, x_n) + \sum_{k=1}^{p} \lambda_k h_k(x_1, \cdots, x_n)$$

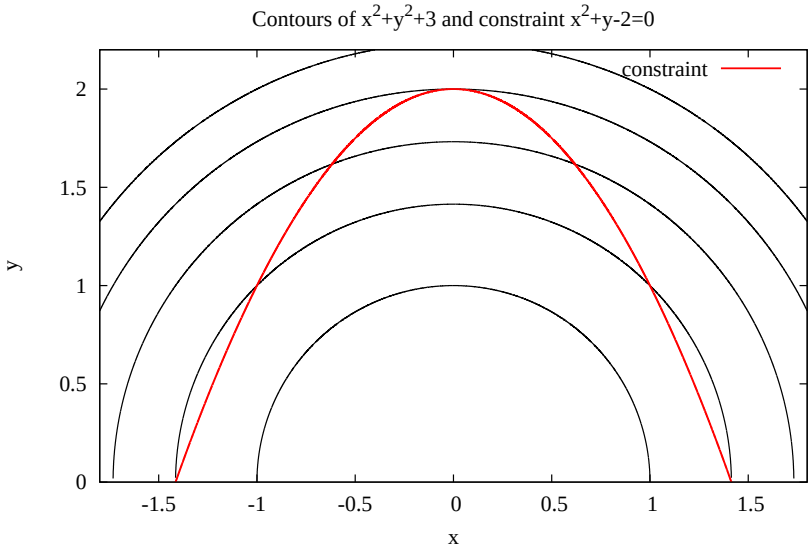
is called **Lagrange function.**

Conclusion: The equations to be solved in theorem 3.25 can be represented as:

$$\frac{\partial L}{\partial x_i}(\boldsymbol{x}) = 0 \quad (i = 1, \cdots, n)$$

$$h_k(\boldsymbol{x}) = 0 \quad (k = 1, \cdots, p)$$

Example 3.31 Extrema of $f(x,y) = x^2+y^2+3$ under the constraint $h(x,y) = x^2+y-2 = 0$



$$\begin{aligned} L(x,y) &= x^2 + y^2 + 3 + \lambda(x^2 + y - 2) \\ \frac{\partial L}{\partial x}(x,y) &= 2x + 2\lambda x \\ \frac{\partial L}{\partial y}(x,y) &= 2y + \lambda \end{aligned}$$

$$\text{grad} L(x,y) = 0 \quad , \quad h(x,y) = 0$$

$$
\begin{aligned}
2x + 2\lambda x &= 0 \qquad (1)\\
2y + \lambda &= 0 \qquad (2)\\
x^2 + y - 2 &= 0 \qquad (3)\\
\text{(2) in (1):} \quad 2x - 4xy &= 0 \qquad (4)\\
y &= 2 - x^2 \qquad \text{(3a)}\\
\text{(3a) in (4):} \quad 2x - 4x(2 - x^2) &= 0
\end{aligned}
$$

first solution: $\boldsymbol{x}_1 = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$ is a maximum.

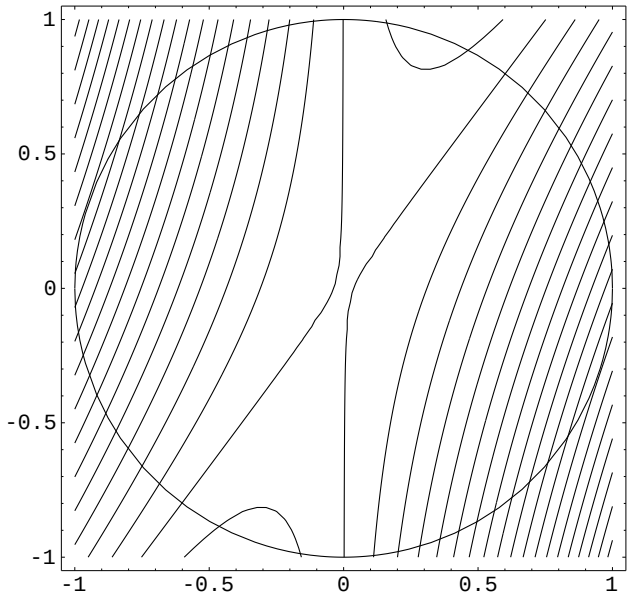
$$2 - 8 + 4x^2 = 0$$

$$4x^2 = 6$$

$$x_{2,3} = \pm\sqrt{\frac{3}{2}} \quad y_{2,3} = \frac{1}{2}$$

$\boldsymbol{x}_2 = \begin{pmatrix} \sqrt{\frac{3}{2}} \\ \frac{1}{2} \end{pmatrix}$ and $\boldsymbol{x}_3 = \begin{pmatrix} -\sqrt{\frac{3}{2}} \\ \frac{1}{2} \end{pmatrix}$ are minima.

Example 3.32 Extrema of the function $f(x,y) = 4x^2 - 3xy$ on the disc $\overline{K}_{\boldsymbol{0},1} = \{(x,y)|x^2+y^2 \le 1\}$.




```
Show[ContourPlot[4*x^2 - 3 *x*y, {x,-1,1}, {y,-1,1}, PlotPoints -> 60,
     Contours -> 20, ContourSmoothing -> True, ContourShading -> False, PlotLabel -> " "],
     Plot[{Sqrt[1-x^2],-Sqrt[1-x^2]}, {x,-1,1}], AspectRatio -> 1 ]
```

1. local extrema inside the disc $\overline{D}_{\boldsymbol{0},1}$:

$$\mathrm{grad} f(x,y) = \begin{pmatrix} 8x - 3y \\ -3x \end{pmatrix} = 0$$

$\Rightarrow \boldsymbol{x} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ is the unique zero of the gradient.

$$\mathrm{Hess} f = \begin{pmatrix} 8 & -3 \\ -3 & 0 \end{pmatrix}$$

$$|8| = 8$$

$$\begin{vmatrix} 8 & -3 \\ -3 & 0 \end{vmatrix} = 0 - 9 = -9$$

$\Rightarrow$ Hess$f$ is neither positive nor negative definite. Eigenvalues of Hess$f =: A$

$$A\boldsymbol{x} = \lambda \boldsymbol{x} \quad \Leftrightarrow \quad (A - \lambda)\boldsymbol{x} = 0$$

$$\Leftrightarrow \quad \begin{pmatrix} 8-\lambda & -3 \\ -3 & -\lambda \end{pmatrix} \boldsymbol{x} = 0 \Leftrightarrow \det \begin{pmatrix} 8-\lambda & -3 \\ -3 & -\lambda \end{pmatrix} = 0$$

$$\Leftrightarrow \quad (8-\lambda)(-\lambda) - 9 = 0 \quad \Leftrightarrow \quad \lambda^2 - 8\lambda - 9 = 0$$

$$\lambda_{1,2} = 4 \pm \sqrt{16+9}$$

$$\lambda_1 = 9$$

$$\lambda_2 = -1$$

$\Rightarrow$ Hess$f$ is indefinite
$\Rightarrow$ f has no local extremum on any open set D.
$\Rightarrow$ in particular f has on $D_{\boldsymbol{0},1}$ no extremum!

2. Local extrema on the margin, i.e. on $\partial\overline{D}_{0,1}$: local extrema von $f(x,y) = 4x^2 - 3xy$ under the constraint $x^2 + y^2 - 1 = 0$:

Lagrangefunction $L = 4x^2 - 3xy + \lambda(x^2 + y^2 - 1)$

$$\frac{\partial L}{\partial x} = 8x - 3y + 2\lambda x = (2\lambda + 8)x - 3y$$

$$\frac{\partial L}{\partial y} = -3x + 2\lambda y$$

Equations for x, y, λ:

$$\begin{array}{rrl} (1) & 8x - 3y + 2\lambda x & = 0 \\ (2) & -3x + 2\lambda y & = 0 \\ (3) & x^2 + y^2 - 1 & = 0 \\ (1)y - (2)x = (4) & 8xy - 3y^2 + 3x^2 & = 0 \end{array}$$

first solution: $(3) \Rightarrow (3a): \quad y^2 = 1 - x^2$

$$(3a) in (4) : \pm 8x\sqrt{1-x^2} - 3(1-x^2) + 3x^2 = 0$$

$$\text{Subst.: } x^2 = u : \quad \pm 8\sqrt{u}\sqrt{1-u} = 3(1-u) - 3u = 3 - 6u$$

squaring:

$$\begin{array}{rcl} 64u(1-u) & = & 9 - 36u + 36u^2 \\ -64u^2 + 64u - 36u^2 + 36u - 9 & = & 0 \\ -100u^2 + 100u - 9 & = & 0 \\ u^2 - u + \frac{9}{100} & = & 0 \\ u_{1,2} = \frac{1}{2} \pm \sqrt{\frac{1}{4} - \frac{9}{100}} & = & \frac{1}{2} \pm \sqrt{\frac{25-9}{100}} = \frac{1}{2} \pm \frac{4}{10} \\ u_1 & = & 0.1 \\ u_2 & = & 0.9 \\ x_{1,2} & = & \pm\frac{1}{\sqrt{10}} \approx \pm 0.3162 \\ x_{3,4} & = & \pm\frac{3}{\sqrt{10}} \approx \pm 0.9487 \end{array}$$

Contours:

$$f(x,y) = 4x^2 - 3xy = c$$

$$y = \frac{-c + 4x^2}{3x} = \frac{4}{3}x - \frac{c}{3x}$$

$$x_3 = \frac{3}{\sqrt{10}} \Rightarrow y_3 = \pm\sqrt{1 - x_3^2} = \pm\frac{1}{\sqrt{10}}$$

$$f\left(\frac{3}{\sqrt{10}}, \frac{1}{\sqrt{10}}\right) = 4\frac{9}{10} - 3\frac{3}{10} = \frac{27}{10}$$

$$f\left(\frac{3}{\sqrt{10}}, -\frac{1}{\sqrt{10}}\right) = 4\frac{9}{10} + 3\frac{3}{10} = \frac{45}{10}$$

$\Rightarrow f(x,y)$ has on $\bar{K}_{0,1}$ in $\boldsymbol{x}_1 = \begin{pmatrix} \frac{3}{\sqrt{10}} \\ -\frac{1}{\sqrt{10}} \end{pmatrix}$ and in $\boldsymbol{x}_2 = \begin{pmatrix} \frac{3}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} \end{pmatrix}$ isolated local maxima

$\Rightarrow f(x,y)$ has on $\bar{K}_{0,1}$ in $\boldsymbol{x}_3 = \begin{pmatrix} \frac{1}{\sqrt{10}} \\ \frac{3}{\sqrt{10}} \end{pmatrix}$ and in $\boldsymbol{x}_4 = \begin{pmatrix} -\frac{1}{\sqrt{10}} \\ -\frac{3}{\sqrt{10}} \end{pmatrix}$ isolated local minima.

3.5.7.1 The Bordered Hessian

In order to check whether a candidate point for a constrained extremum is a maximum or minimum, we need a sufficient condition, similarly to the definiteness of the Hessian in the unconstrained case. Here we need the **Bordered Hessian**

$$\overline{\text{Hess}} := \begin{pmatrix} 0 & \dots & 0 & \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_n} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & \frac{\partial h_p}{\partial x_1} & \dots & \frac{\partial h_p}{\partial x_n} \\ \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_p}{\partial x_1} & \frac{\partial^2 L}{\partial x_1^2} & \dots & \frac{\partial^2 L}{\partial x_1 \partial x_n} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial h_1}{\partial x_n} & \dots & \frac{\partial h_p}{\partial x_n} & \frac{\partial^2 L}{\partial x_n \partial x_1} & \dots & \frac{\partial^2 L}{\partial x_n^2} \end{pmatrix}$$

This matrix can be used to check on local minima and maxima by computing certain subdeterminants. Here we show this only for the two dimensional case with one constraint where the bordered Hessian has the form

$$\overline{\text{Hess}} := \begin{pmatrix} 0 & \frac{\partial h}{\partial x_1} & \frac{\partial h}{\partial x_2} \\ \frac{\partial h}{\partial x_1} & \frac{\partial^2 L}{\partial x_1^2} & \frac{\partial^2 L}{\partial x_1 \partial x_2} \\ \frac{\partial h}{\partial x_2} & \frac{\partial^2 L}{\partial x_2 \partial x_1} & \frac{\partial^2 L}{\partial x_2^2} \end{pmatrix}$$

and the sufficient criterion for local extrema is (in contrast to the unconstrained case!) the following simple determinant condition:
Under the constraint $h(x, y) = 0$ the function f has in (x, y) a

- local maximum, if $|\overline{\text{Hess}}(x, y)| > 0$
- local minimum, if $|\overline{\text{Hess}}(x, y)| < 0$.

If $|\overline{\text{Hess}}(x, y)| = 0$, we can not decide on the properties of the stationary point (x, y).
Application to example 3.31 yields

$$\text{grad} L(x, y) = \begin{pmatrix} 2x(1+\lambda) \\ 2y + \lambda \end{pmatrix}$$

$$\overline{\text{Hess}}(x, y) = \begin{pmatrix} 0 & 2x & 1 \\ 2x & 2(1+\lambda) & 0 \\ 1 & 0 & 2 \end{pmatrix}.$$

Substitution of the first solution of $\text{grad} L = 0$ which is $x = 0$, $y = 2$, $\lambda = -4$ into this matrix gives

$$|\overline{\text{Hess}}(0, 2)| = \begin{vmatrix} 0 & 0 & 1 \\ 0 & -6 & 0 \\ 1 & 0 & 2 \end{vmatrix} = 6$$

which proves that we indeed have a maximum in $(0, 2)$.

3.6 Exercises

Sequences, Series, Continuity

Exercise 3.1 Prove (e.g. with complete induction) that for $p \in \mathbb{R}$ it holds:

$$\sum_{k=0}^{n}(p+k) = \frac{(n+1)(2p+n)}{2}$$

Exercise 3.2

a) Calculate

$$\sqrt{1+\sqrt{1+\sqrt{1+\sqrt{1+\ldots}}}},$$

i.e. the limit of the sequence $(a_n)_{n\in\mathbb{N}}$ with $a_0 = 1$ and $a_{n+1} = \sqrt{1+a_n}$. Give an exact solution as well as an approximation with a precision of 10 decimal places.

b) Prove that the sequence $(a_n)_{n\in\mathbb{N}}$ converges.

Exercise 3.3 Calculate

$$1+\frac{1}{1+\frac{1}{1+\frac{1}{1+\frac{1}{1+\ldots}}}},$$

i.e. the limit of the sequence $(a_n)_{n\in\mathbb{N}}$ with $a_0 = 1$ and $a_{n+1} = 1+1/a_n$. Give an exact solution as well as an approximation with a precision of 10 decimal places.

Exercise 3.4 Calculate the number of possible draws in the German lottery, which result in having three correct numbers. In German lottery, 6 balls are drawn out of 49. The 49 balls are numbered from 1-49. A drawn ball is not put back into the pot. In each lottery ticket field, the player chooses 6 numbers out of 49. Then, what is the probability to have three correct numbers?

Exercise 3.5 Investigate the sequence $(a_n)_{n\in\mathbb{N}}$ with $a_n := 1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\frac{1}{5}+\ldots+\frac{1}{n}$ regarding convergence.

Exercise 3.6 Calculate the infinite sum $\sum_{n=0}^{\infty}\frac{1}{2^n}$.

Exercise 3.7 Prove: A series $\sum_{k=0}^{\infty} a_k$ with $\forall k:\ a_k > 0$ converges if and only if the sequence of the partial sums is limited.

Exercise 3.8 Calculate an approximation (if possible) for the following series and investigate their convergence.

a) $\sum_{n=0}^{\infty}(n+1)2^{-n}$ b) $\sum_{n=0}^{\infty}4^n(n+1)!\,n^{-n}$ c) $\sum_{n=0}^{\infty}3n[4+(1/n)]^{-n}$

Exercise 3.9 Investigate the following functions $f:\mathbb{R}\to\mathbb{R}$ regarding continuity (give an outline for each graph):

a) $f(x) = \frac{1}{1+e^{-x}}$ b) $f(x) = \begin{cases} 0 & \text{if } x = 1 \\ \frac{1}{x-1} & \text{else} \end{cases}$ c) $f(x) = \begin{cases} x+4 & \text{if } x > 0 \\ (x+4)^2 & \text{else} \end{cases}$

d) $f(x) = \begin{cases} (x-2)^2 & \text{if } x > 0 \\ (x+2)^2 & \text{else} \end{cases}$ e) $f(x) = |x|$ f) $f(x) = x - \lfloor x \rfloor$ g) $f(x) = \left| \left\lfloor x + \frac{1}{2} \right\rfloor - x \right|$

Exercise 3.10 Show that $f : \mathbb{R} \to \mathbb{R}$ with

$$f(x) = \begin{cases} 0 & \text{falls } x \text{ rational} \\ 1 & \text{falls } x \text{ irrational} \end{cases}$$

is not continuous in any point.

Taylor–Series

Exercise 3.11 Calculate the Taylor series of sine and cosine with $x_0 = 0$. Prove that the Taylor series of sine converges towards the sine function.

Exercise 3.12 Try to expand the function $f(x) = \sqrt{x}$ at $x_0 = 0$ and $x_0 = 1$ into a Taylor series. Report about possible problems.

Exercise 3.13 Let f be expandable into a Taylor series on the interval $(-r, r)$ around 0 $((r > 0)$. Prove:

a) If f is an even function $(f(x) = f(-x))$ for all $x \in (-r, r)$, then only even exponents appear in the Taylor series of f, it has the form $\sum_{k=0}^{\infty} a_{2k} x^{2k}$.

b) If f is an odd function $(f(x) = -f(-x))$ for all $x \in (-r, r)$, then only odd exponents appear in the Taylor series of f, it has the form $\sum_{k=0}^{\infty} a_{2k+1} x^{2k+1}$.

Exercise 3.14 Calculate the Taylor series of the function

$$f(x) = \begin{cases} e^{-\frac{1}{x^2}} & \text{if } x \neq 0 \\ 0 & \text{if } x = 0 \end{cases}$$

at $x_0 = 0$ and analyse the series for convergence. Justify the result!

Exercise 3.15 Calculate the Taylor series of the function arctan in $x_0 = 0$. Use the result for the approximate calculation of π. (Use for this for example $\tan(\pi/4) = 1$.)

Functions from $\mathbb{R}^n$ to $\mathbb{R}^m$

Exercise 3.16 Prove that the scalar product of a vector $\boldsymbol{x}$ with itself is equal to the square of its length (norm).

Exercise 3.17

a) Give a formal definition of the function $f : \mathbb{R} \to \mathbb{R}^+ \cup \{0\}$ with $f(x) = |x|$.

b) Prove that for all real numbers x, y $|x + y| \leq |x| + |y|$.

Exercise 3.18

a) In industrial production in the quality control, components are measured and the values $x_1, \ldots x_n$ determinated. The vector $\boldsymbol{d} = \boldsymbol{x} - \boldsymbol{s}$ indicates the deviation of the measurements to the nominal values $s_1, \ldots, s_n$. Now define a norm on $\mathbb{R}^n$ such that $||\boldsymbol{d}|| < \varepsilon$ holds, iff all deviations from the nominal value are less than a given tolerance ε.

b) Prove that the in a) defined norm satisfies all axioms of a norm.

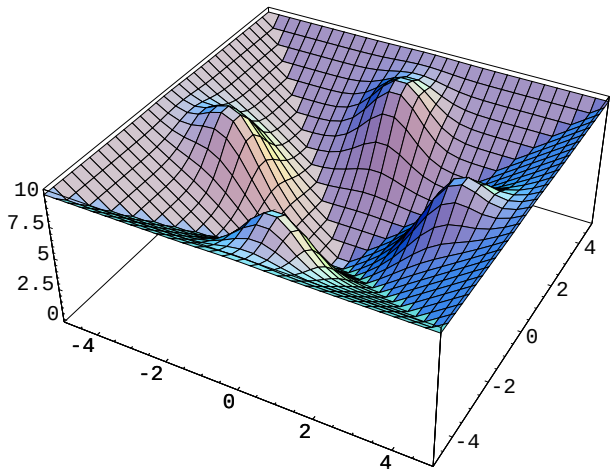
Exercise 3.19 Draw the graph of the following functions $f:\ \mathbb{R}^2 \to \mathbb{R}$ (first manually and then by the computer!):

$$f_1(x,y) = x^2 + y^3, \quad f_2(x,y) = x^2 + e^{-(10x)^2} \quad f_3(x,y) = x^2 + e^{-(5(x+y))^2} + e^{-(5(x-y))^2}$$

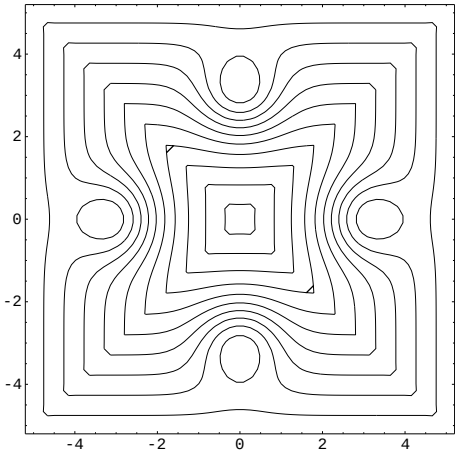
Exercise 3.20 Calculate the partial derivatives $\frac{\partial f}{\partial x_1}$, $\frac{\partial f}{\partial x_2}$, $\frac{\partial f}{\partial x_3}$ of the following functions $f:\mathbb{R}^3 \to \mathbb{R}$

a) $f(\boldsymbol{x}) = |\boldsymbol{x}|$ **b)** $f(\boldsymbol{x}) = x_1^{x_2} + x_1^{x_3}$ **c)** $f(\boldsymbol{x}) = x_1^{(x_2+x_3)}$
d) $f(\boldsymbol{x}) = \sin(x_1 + x_2)$ **e)** $f(\boldsymbol{x}) = \sin(x_1 + a\,x_2)$

Exercise 3.21 Build a function $f:\mathbb{R}^2 \to \mathbb{R}$, which generates roughly the following graph:




```
Plot3D[f[x,y], {x,-5,5},{y,-5,5},
PlotPoints -> 30]
```




```
ContourPlot[f[x,y], {x,-5,5},{y,-5,5},
PlotPoints -> 60, ContourSmoothing ->
True,ContourShading-> False]
```

Exercise 3.22 Calculate the derivative matrix of the function $\boldsymbol{f}(x_1,x_2,x_3) = \begin{pmatrix} \sqrt{x_1x_2x_3} \\ \sin(x_1x_2x_3) \end{pmatrix}$.

Exercise 3.23 For $\boldsymbol{f}(x,y) = \begin{pmatrix} \sqrt{x\,y} \\ \sin(e^x + e^y) \end{pmatrix}$, find the tangent plane at $\boldsymbol{x_0} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.

Exercise 3.24 Draw the graph of the function

$$f(x,y) = \begin{cases} y(1+\cos\frac{\pi x}{y}) & \text{for} |y| > |x| \\ 0 & \text{else} \end{cases}.$$

Show that f is continuous and partially differentiable in $\mathbb{R}^2$, but not in $\boldsymbol{0}$.

Exercise 3.25 Calculate the gradient of the function $f(x,y) = \dfrac{x^2+y^2}{1+x^4+y^4}$ and draw it as an arrow at different places in a contour lines image of f.

Exercise 3.26 The viscosity η of a liquid is to be determinated with the formula $K = 6\pi\eta v r$. Measured: $r = 3$cm, $v = 5$cm/sec, $K = 1000$dyn. Measurement error: $|\Delta r| \le 0.1$cm, $|\Delta v| \le 0.003$cm/sec, $|\Delta K| \le 0.1$dyn. Determine the viscosity η and its error $\Delta\eta$.

Extrema

Exercise 3.27 Examine the following function for extrema and specify whether it is a local, global, or an isolated extremum:

$$\begin{array}{lrcl} \text{a)} & f(x,y) & = & x^3y^2(1-x-y) \\ \text{b)} & g(x,y) & = & x^k + (x+y)^2 \qquad (k=0,3,4) \end{array}$$

Exercise 3.28 Given the function $f : \mathbb{R}^2 \to \mathbb{R}$, $f(x,y) = (y-x^2)(y-3x^2)$.

a) Calculate grad$f$ and show: grad$f(x,y) = 0 \Leftrightarrow x = y = 0$.

b) Show that (Hess$f$)(0) is semi-definite and that f has a isolated minimum on each line through 0.

c) Nevertheless, f has not an local extremum at 0 (to be shown!).

Exercise 3.29 Given the functions $\Phi(x,y) = y^2x - x^3$, $f(x,y) = x^2 + y^2 - 1$.

a) Examine Φ for extrema.

b) Sketch all contour lines $h = 0$ of Φ.

c) Examine Φ for local extrema under the constraint $f(x,y) = 0$.

Exercise 3.30 The function

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

has at (0,0) a discontinuity. This can be remedied easily by defining e.g. $f(0,0) := 3$.

a) Show that f is continuous on all $\mathbb{R}^2$ except at (0,0). Is it possible to define the function at the origin so that it is continuous?

b) Calculate all local extrema of the function f and draw (sketch) a contour line image (not easy).

c) Determine the local extrema under the constraint (not easy):

i) $x = 0.1$

ii) $y = 0.1$

iii) $x^2 + y^2 = 4$

Exercise 3.31 Show that grad$(f\,g) = g\,\text{grad}f + f\,\text{grad}g$.

Chapter 4

Statistics and Probability Basics

Based on **samples**, statistics deals with the derivation of general statements on certain **features**.
[1]

4.1 Recording Measurements in Samples

Discrete feature: finite amount of values.

Continuous feature: values in an interval of real numbers.

Definition 4.1 Let X be a feature (or random variable). A series of measurements $x_1, \ldots, x_n$ for X is called a sample of the length n.

Example 4.1 For the feature X (grades of the exam Mathematics I in WS 97/98) following sample has been recorded:
1.0 1.3 2.2 2.2 2.2 2.5 2.9 2.9 2.9 2.9 2.9 2.9 2.9 3.0 3.0 3.0 3.3 3.3 3.4 3.7 3.9 3.9 4.1 4.7
Let $g(x)$ be the absolute frequency of the value x. Then

$$h(x) = \frac{1}{n} g(x)$$

is called relative frequency or **empirical density** of X.

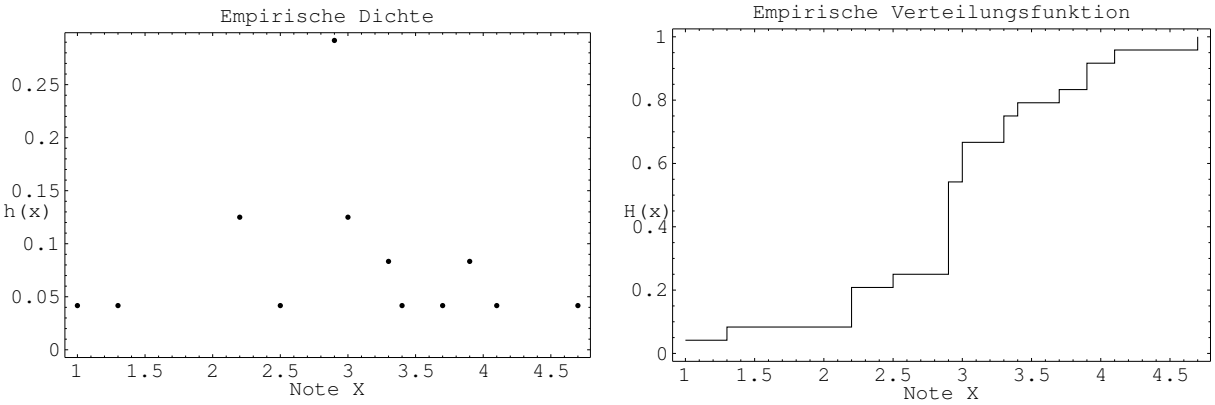
Grade X	Absolute frequency $g(x)$	Relative frequency $h(x)$
1.0	1	0.042
1.3	1	0.042
2.2	3	0.13
2.5	1	0.042
2.9	7	0.29
3.0	3	0.13
3.3	2	0.083
3.4	1	0.042
3.7	1	0.042
3.9	2	0.083
4.1	1	0.042
4.7	1	0.042

[1] The content of this chapter is strongly leaned on [**?**]. Therefore, [**?**] is the ideal book to read.

If $x_1 < x_2 < \ldots x_n$, then

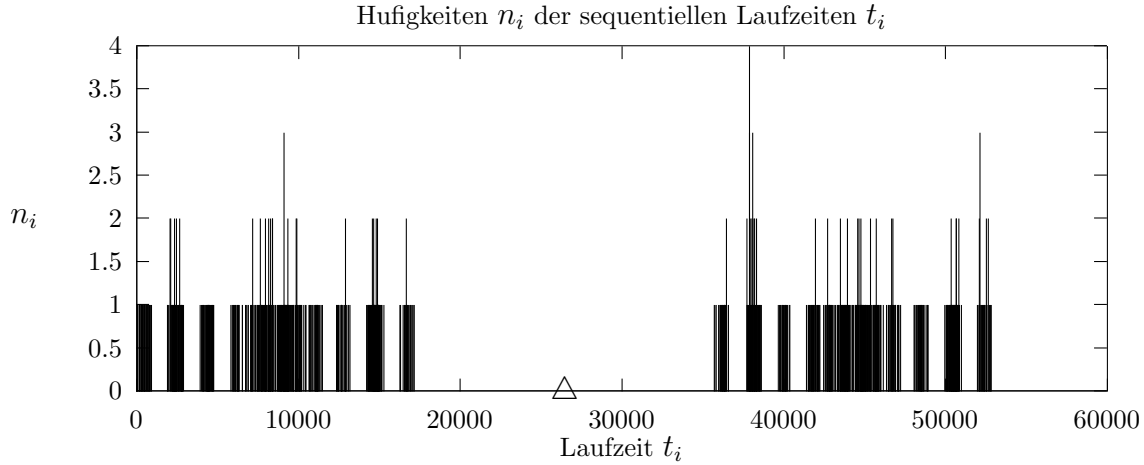
$$H(x) = \sum_{t \leq x} h(t)$$

is the **empirical distribution function**.
It is apparent from the data that 8.3 % of the participating students in the exam Mathematics 1 in WS 97/98 had a grade better than 2.0.
On the contrary, the following statement is an assumption: In the exam Mathematics 1, 8.3 % of the students of the HS RV-Wgt achieve a grade better than 2.0. This statemtent is a hypothesis and not provable.
However, under certain conditions one can determine the probability that this statement is true. Such computations are called **statistical induction**.

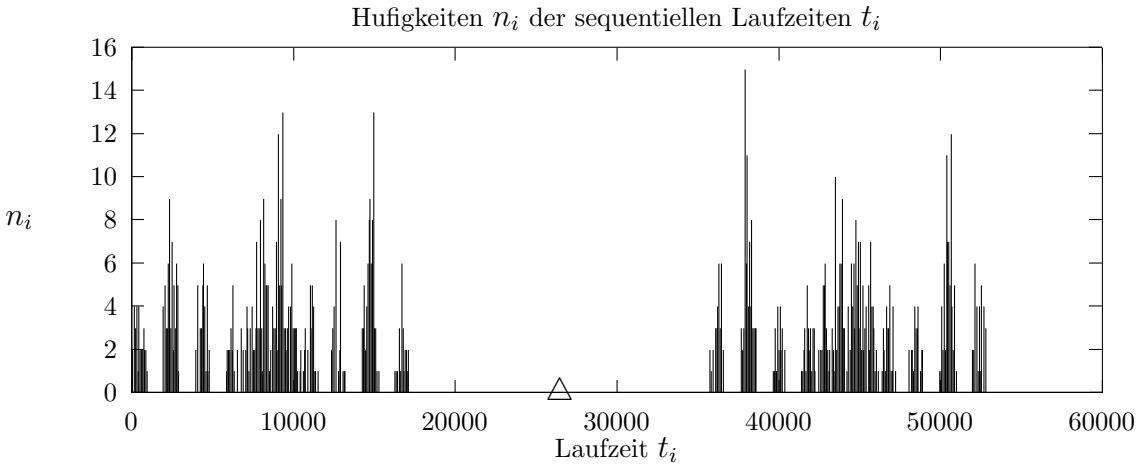


When calculating or plotting empirical density functions, it is often advantageous to group measured values to classes.

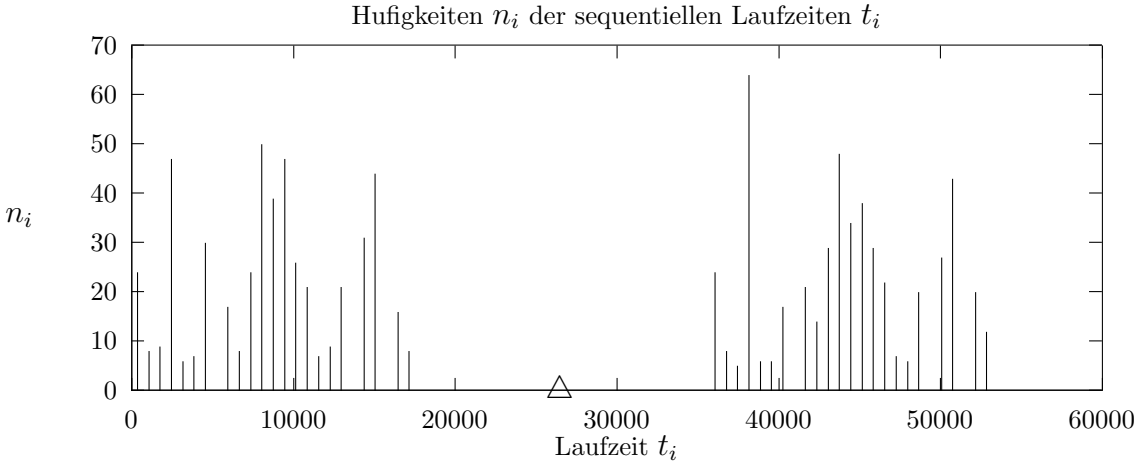
Example 4.2
Following frequency function has been determined from runtime measurements of a randomized program (automated theorem prover with randomized depth-first search and backtracking):



In this graphic, at any value $t_i \in \{1, \ldots, 60000\}$ a frequency in the form of a histogram is shown. One can clearly see the scattering effects due to low frequencies per time value t_i. In the next image, 70 values each have been summarized to a class, which results in 600 classes overall.



Summarizing 700 values each to a class one obtains 86 classes as shown in the third image. Here, the structure of the frequency distribution is not recognizable anymore.



The amount ℓ of the classes should neither be choosen too high nor too low. In [**?**] a rule of thumb $\ell \leq \sqrt{n}$ is given.

4.2 Statistical Parameters

The effort to describe a sample by a single number is fullfilled by following definition:

Definition 4.2 For a sample $x_1, x_2, \ldots x_n$ the term

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

is called **arithmetic mean** and if $x_1 < x_2 < \ldots x_n$, then the **sample median** is defined as

$$\tilde{x} = \begin{cases} x_{\frac{n+1}{2}} & \text{if } n \text{ odd} \\ \frac{1}{2}\left(x_{\frac{n}{2}} + x_{\frac{n}{2}+1}\right) & \text{if } n \text{ even} \end{cases}$$

In the example 4.2, the arithmetic mean is marked with the symbol $\triangle$. It is interesting that

the arithmetic mean minimizes the sum of squares of the distances

$$\sum_{i=1}^{n}(x_i - x)^2$$

whereas the median minimizes the sum of the absolut values of the distances

$$\sum_{i=1}^{n}|x_i - x|$$

(proof as exercise). Often, one does not only want to determine a mean value, but also a measure for the mean deviation of the arithmetic mean.

Definition 4.3 The number

$$s_x^2 := \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

is called **sample variance** and

$$s_x := \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

is called standard deviation

4.3 Multidimensional Samples

If not only grades from Mathematics 1, but for any student also the grades of Mathematics 2 and further courses are considered, one can ask if there is a statistical relationship between the grades of different courses. Therefore, a simple tool, the covariance matrix is introduced. For a multidimensional variable $(X_1, X_2, \ldots, X_k)$, a k-dimensional sample of the length n consists of a list of vectors

$$(x_{11}, x_{21}, \ldots, x_{k1}), (x_{12}, x_{22}, \ldots, x_{k2}), \ldots, (x_{1n}, x_{2n}, \ldots, x_{kn})$$

By extension of example 4.1, we obtain an example for 2 dimensions.

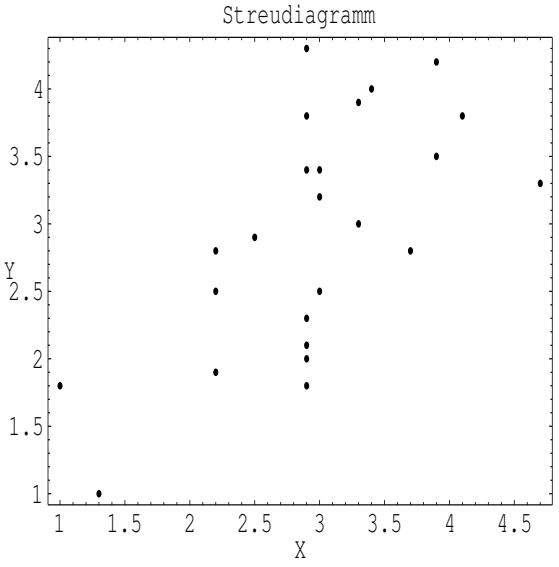
Example 4.3

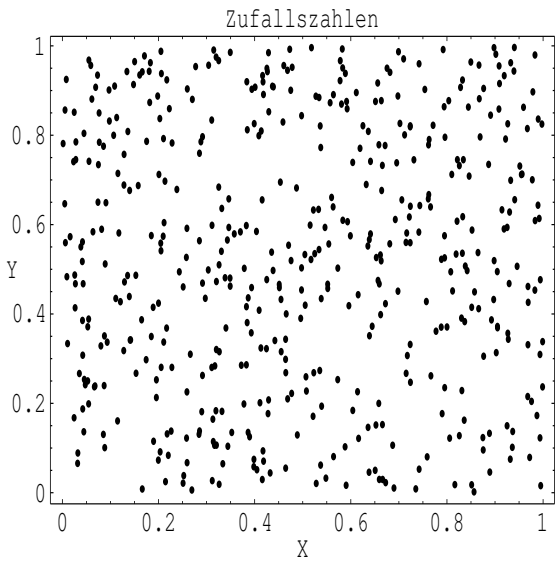
Grade X	Grade Y
1.0	1.8
1.3	1.0
2.2	1.9
2.2	2.8
2.2	2.5
2.5	2.9
2.9	3.8
2.9	4.3
2.9	2.3
2.9	3.4
2.9	2.0
2.9	1.8
2.9	2.1
3.0	3.4
3.0	2.5
3.0	3.2
3.3	3.0
3.3	3.9
3.4	4.0
3.7	2.8
3.9	3.5
3.9	4.2
4.1	3.8
4.7	3.3

If beside the grades of Mathematics 1 (X) the grades (Y) of Mathematics for computer science are considered, one could determine the 2-dimensional variable (X, Y) as per margin.
The question, if the variables X and Y are correlated can be answered by the covariance:

$$\sigma_{xy} = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$$

For the grades above we determine $\sigma_{xy} = 0.47$. That means that between these 2 variables a positive correlation exists, thus on average, a student being good in Mathematics 1 is also good in Mathematics for computer science.
This is also visible on the left of the following two scatter plots.





For the equally distributed random numbers in the right plot $\sigma_{xy} = 0.0025$ is determined. Thus, the two variables have a very low correlation.
If there are $k > 2$ variables, the data cannot easily be plotted graphically. But one can determine the covariances between two variables each in order to represent them in a **covariance matrix** σ:

$$\sigma_{ij} = \frac{1}{n-1} \sum_{\ell=1}^{n} (x_{i\ell} - \bar{x_i})(x_{j\ell} - \bar{x_j})$$

If dependencies among different variables are to be compared, a **correlation matrix** can be determined:

$$K_{ij} = \frac{\sigma_{ij}}{s_i \cdot s_j},$$

Here, all diagonal elements have the value 1.

Example 4.4 In a medical database of 473 patients[2]with a surgical removal of their appendix, 15 different symptoms as well as the diagnosis (appendicitis negative/positive) have been recorded.

[2]The data was obtained from the hospital 14 Nothelfer in Weingarten with the friendly assistance of Dr. Rampf. Mr. Kuchelmeister used the data for the development of an expert system in his diploma thesis.

```
Alter:                                              continuous.
gender_(1=m___2=w):                                 1,2.
pain_quadrant1_(0=nein__1=ja):                      0,1.
pain_quadrant2_(0=nein__1=ja):                      0,1.
pain_quadrant3_(0=nein__1=ja):                      0,1.
pain_quadrant4_(0=nein__1=ja):                      0,1.
guarding_(0=nein__1=ja):                            0,1.
rebound_tenderness_(0=nein__1=ja):                  0,1.
pain_on_tapping_(0=nein__1=ja):                     0,1.
vibration_(0=nein__1=ja):                           0,1.
rectal_pain_(0=nein__1=ja):                         0,1.
temp_ax:                                            continuous.
temp_re:                                            continuous.
leukocytes:                                         continuous.
diabetes_mellitus_(0=nein__1=ja):                   0,1
appendicitis_(0=nein__1=ja):                        0,1
```

The first 3 data sets are as follows:

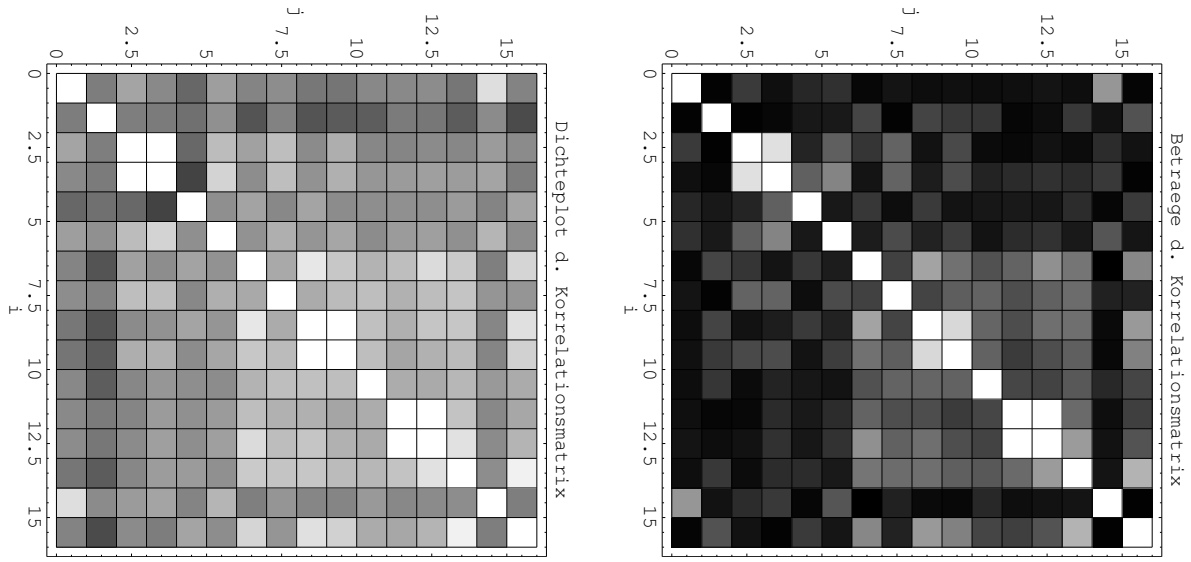
```
26  1  0  0  1  0  1  0  1  1  0  37.9  38.8  23100  0  1
17  2  0  0  1  0  1  0  1  1  0  36.9  37.4  8100   0  0
28  1  0  0  1  0  0  0  0  0  0  36.7  36.9  9600   0  1
```

The correlation matrix for the data of all 473 patients is:

1.	-0.009	0.14	0.037	-0.096	0.12	0.018	0.051	-0.034	-0.041	0.034	0.037	0.05	-0.037	0.37	0.012
-0.009	1.	-0.0074	-0.019	-0.06	0.063	-0.17	0.0084	-0.17	-0.14	-0.13	-0.017	-0.034	-0.14	0.045	-0.2
0.14	-0.0074	1.	0.55	-0.091	0.24	0.13	0.24	0.045	0.18	0.028	0.02	0.045	0.03	0.11	0.045
0.037	-0.019	0.55	1.	-0.24	0.33	0.051	0.25	0.074	0.19	0.087	0.11	0.12	0.11	0.14	-0.0091
-0.096	-0.06	-0.091	-0.24	1.	0.059	0.14	0.034	0.14	0.049	0.057	0.064	0.058	0.11	0.017	0.14
0.12	0.063	0.24	0.33	0.059	1.	0.071	0.19	0.086	0.15	0.048	0.11	0.12	0.063	0.21	0.053
0.018	-0.17	0.13	0.051	0.14	0.071	1.	0.16	0.4	0.28	0.2	0.24	0.36	0.29	-0.00013	0.33
0.051	0.0084	0.24	0.25	0.034	0.19	0.16	1.	0.17	0.23	0.24	0.19	0.24	0.27	0.083	0.084
-0.034	-0.17	0.045	0.074	0.14	0.086	0.4	0.17	1.	0.53	0.25	0.19	0.27	0.27	0.026	0.38
-0.041	-0.14	0.18	0.19	0.049	0.15	0.28	0.23	0.53	1.	0.24	0.15	0.19	0.23	0.02	0.32
0.034	-0.13	0.028	0.087	0.057	0.048	0.2	0.24	0.25	0.24	1.	0.17	0.17	0.22	0.098	0.17
0.037	-0.017	0.02	0.11	0.064	0.11	0.24	0.19	0.19	0.15	0.17	1.	0.72	0.26	0.035	0.15
0.05	-0.034	0.045	0.12	0.058	0.12	0.36	0.24	0.27	0.19	0.17	0.72	1.	0.38	0.044	0.21
-0.037	-0.14	0.03	0.11	0.11	0.063	0.29	0.27	0.27	0.23	0.22	0.26	0.38	1.	0.051	0.44
0.37	0.045	0.11	0.14	0.017	0.21	-0.00013	0.083	0.026	0.02	0.098	0.035	0.044	0.051	1.	-0.0055
0.012	-0.2	0.045	-0.0091	0.14	0.053	0.33	0.084	0.38	0.32	0.17	0.15	0.21	0.44	-0.0055	1.

The matrix structure is more apparent if the numbers are illustrated as density plot[3] In the left diagram, bright stands for positive and dark for negative. The right plot shows the absolute values. Here, white stands for a strong correlation between two variables and black for no correlation.

[3]The first to images have been rotated by 90^o. Therefore, the fields in the density plot correspond to the matrix elements.



It is clearly apparent that most of the variable pairs have no or only a very low correlation, whereas the two temperature variables are highly correlated.

4.4 Probability Theory

The purpose of probability theory is to determine the probability of certain possible events within an experiment.

Example 4.5 When throwing a die once, the probability for the event „throwing a six" is 1/6, whereas the probability for the event „throwing an odd number" is 1/2.

Definition 4.4 Let Ω be the set of possible outcomes of an experiment. Each $\omega \in \Omega$ stands for a possible outcome of the experiment. If the $w_i \in \Omega$ exclude each other, but cover all possible outcomes, they are called **elementary events**.

Example 4.6 When throwing a die once, $\Omega = \{1, 2, 3, 4, 5, 6\}$, because no two of these events can occur at the same time. Throwing an even number $\{2, 4, 6\}$ is not an elementary event, as well as throwing a number lower than 5 $\{1, 2, 3, 4\}$, because $\{2, 4, 6\} \cap \{1, 2, 3, 4\} = \{2, 4\} \neq \emptyset$.

Definition 4.5 Let Ω be a set of elementary events. $\bar{A} = \Omega - A = \{\omega \in \Omega | \omega \notin A\}$ is called the **complementary event** to A. A subset $\mathcal{A}$ of 2^Ω is called **event algebra** over Ω, if:

1. $\Omega \in \mathcal{A}$.
2. With A, $\bar{A}$ is also in $\mathcal{A}$.
3. If $(A_n)_{n \in \mathbb{N}}$ is a sequence $\mathcal{A}$, then $\cup_{n=1}^{\infty} A_n$ is also in $\mathcal{A}$.

Every event algebra contains the **sure event** Ω as well as the **impossible event** $\emptyset$.

At coin toss, one could choose $\mathcal{A} = 2^{\Omega}$ and $\Omega = \{1, 2, 3, 4, 5, 6\}$. Thus $\mathcal{A}$ contains any possible event by a toss.
If one is only interested in throwing a six, one would consider $A = \{6\}$ and $\bar{A} = \{1, 2, 3, 4, 5\}$ only, where the algebra results in $\mathcal{A} = \{\emptyset, A, \bar{A}, \Omega\}$.
The term of the probability should give us an as far as possible objective description of our „believe" or „conviction" about the outcome of an experiment. As numeric values, all real numbers in the interval $[0, 1]$ shall be possible, whereby 0 is the probability for the impossible event and 1 the probability for the sure event.

4.4.1 The Classical Probability Definition

Let $\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$ be finite. No elementary event is preferred, that means we assume a symmetry regarding the frequency of occurence of all elementary events. The probability $P(A)$ of the event A is defined by

$$P(A) = \frac{|A|}{|\Omega|} = \frac{\text{Amount of outcomes favourable to } A}{\text{Amount of possible outcomes}}$$

It is obvious that any elementary event has the probability $1/n$. The assumption of the same probability for all elementary events is called the **Laplace assumption**.

Example 4.7 Throwing a die, the probability for an even number is

$$P(\{2, 4, 6\}) = \frac{|\{2, 4, 6\}|}{|\{1, 2, 3, 4, 5, 6\}|} = \frac{3}{6} = \frac{1}{2}.$$

4.4.2 The Axiomatic Probability Definition

The classical definition is suitable for a finite set of elementary events only. For endless sets a more general definition is required.

Definition 4.6 Let Ω be a set and $\mathcal{A}$ an event algebra on Ω. A mapping

$$P : \mathcal{A} \to [0,1]$$

is called **probability measure** if:

1. $P(\Omega) = 1$.
2. If the events A_n of the sequence $(A_n)_{n\in\mathbb{N}}$ are pairwise inconsistent, i.e. for $i, j \in \mathbb{N}$ it holds $A_i \cap A_j = \emptyset$, then

$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i).$$

For $A \in \mathcal{A}$, $P(A)$ is called probability of the event A.

From this definition, some rules follow directly:

Theorem 4.1

1. $P(\emptyset) = 0$, i.e. the impossible event has the probability 0.
2. For pairwise inconsistent events A and B it holds $P(A \cup B) = P(A) + P(B)$.
3. For a finite amount of pairwise inconsistent events $A_1, A_2, \dots A_k$ it holds
$$P\left(\bigcup_{i=1}^{k} A_n\right) = \sum_{i=1}^{k} P(A_n).$$
4. For two each other complentary events A and $\bar{A}$ it holds $P(A) + P(\bar{A}) = 1$.
5. For any event A and B it holds $P(A \cup B) = P(A) + P(B) - P(A \cap B)$.
6. For $A \subseteq B$ it holds $P(A) \leq P(B)$.

Proof: as exercise.

4.4.3 Conditional Probabilities

Example 4.8 In the Doggenriedstraße in Weingarten the speed of 100 vehicles is measured. At each measurement it is recorded if the driver was a student or not. The results are as follows:

Event	Frequency	Relative frequency
Vehicle observed	100	1
Driver is a student (S)	30	0.3
Speed too high (G)	10	0.1
Driver is a student and speeding ($S \cap G$)	5	0.05

We now ask the following question: *Do students speed more frequently than the average person, or than non-students?*[4] The answer is given by the probability $P(G|S)$ for speeding under the condition that the driver is a student.

$$P(G|S) = \frac{|\text{Driver is a student and speeding}|}{|\text{Driver is a student}|} = \frac{5}{30} = \frac{1}{6}$$

Definition 4.7 For two events A and B, the *probability for A under the condition B* **(conditional probability)** is defined by
$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

[4] The determined probabilities can only be used for further statements if the sample (100 vehicles) is representative. Otherwise, one can only make a statament about the observed 100 vehicles.

At example 4.8 one can recognize that in the case of a finite event set the conditional probability $P(A|B)$ can be treated as the probability of A, when regarding only the event B, i.e. as

$$P(A|B) = \frac{|A \cap B|}{|B|}$$

Definition 4.8 If two events A and B behave as

$$P(A|B) = P(A),$$

then these events are called independent.

A and B are independent, if the probability of the event A is not influenced by the event B.

Theorem 4.2 From this definition, for the independent events A and B follows

$$P(A \cap B) = P(A) \cdot P(B)$$

Beweis: Proof:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = P(A) \quad \Rightarrow \quad P(A \cap B) = P(A) \cdot P(B)$$

Example 4.9 The probability for throwing two sixes with two dice is 1/36 if the dice are independent, because

$$\begin{aligned} P(\text{die 1} \equiv \text{six}) \cdot P(\text{die 2} \equiv \text{six}) &= \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36} \\ &= P(\text{die 1} \equiv \text{six} \cap \text{die 2} \equiv \text{six}), \end{aligned}$$

whereby the last equation applies only if the two dice are independent. If for example by magic power die 2 always falls like die 1, it holds

$$P(\text{die 1} \equiv \text{six} \cap \text{die 2} \equiv \text{six}) = \frac{1}{6}.$$

4.4.4 The Bayes Formula

Since equation (4.7) is symmetric in A and B, one can also write

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad \text{as well as} \quad P(B|A) = \frac{P(A \cap B)}{P(A)}.$$

Rearranging by $P(A \cap B)$ and equating results in the Bayes formula

$$\boxed{P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}}.$$

A very reliable alarm system warns at burglary with a certainty of 99%. So, can we infer from an alarm to burglary with high certainty?
No, because if for example $P(A|B) = 0.99, \quad P(A) = 0.1, \quad P(B) = 0.001$ holds, then the Bayes formula returns:

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)} = \frac{0.99 \cdot 0.001}{0.1} = 0.01.$$

4.5 Discrete Distributions

Definition 4.9 A random variable, which range of values is finite or countably infinite is called **discrete random variable**.

Example 4.10 Throwing a die, the number X is a discrete random variable with the values $\{1, 2, 3, 4, 5, 6\}$, this means in the example it holds $x_1 = 1, \ldots, x_6 = 6$. If the die does not prefer any number, then

$$p_i = P(X = x_i) = 1/6,$$

this means the numbers are *uniformly distributed*. The probability to throw a number ≤ 5 is

$$P(X \leq 5) = \sum_{i:x_i \leq 5} p_i = 5/6.$$

In general, one defines

Definition 4.10 The function, which assigns a probability p_i to each x_i of the random variable X is called the **discrete density function** of X.

Definition 4.11 For any real number x, a defined function

$$x \mapsto P(X \leq x) = \sum_{i:x_i \leq x} p_i$$

is called distribution function of X.

Such as the empirical distribution function, $P(X \leq x)$ is a monotonically increasing step function. Analogous to the mean value and variance of samples are the following definitions.

Definition 4.12 The number

$$E(X) = \sum_i x_i p_i$$

is called **expected value**. The **variance** is given by

$$Var(X) := E((X - E(X))^2) = \sum_i (x_i - E(X))^2 p_i$$

whereby $\sqrt{Var(x)}$ is called **standard deviation**.

It is easy to see that $Var(X) := E(X^2) - E(X)^2$ (exercise).

4.5.1 Binomial Distribution

Let a player's scoring probability at penalty kicking be = 0.9. The probability always to score at 10 independent kicks is

$$B_{10,0.9}(10) = 0.9^{10} \approx 0.35.$$

It is very unlikely that the player scores only once, the probability is

$$B_{10,0.9}(1) = 10 \cdot 0.1^9 \cdot 0.9 = 0.000000009$$

We might ask the question, which amount of scores is the most frequent at 10 kicks.

Definition 4.13 The distribution with the density function

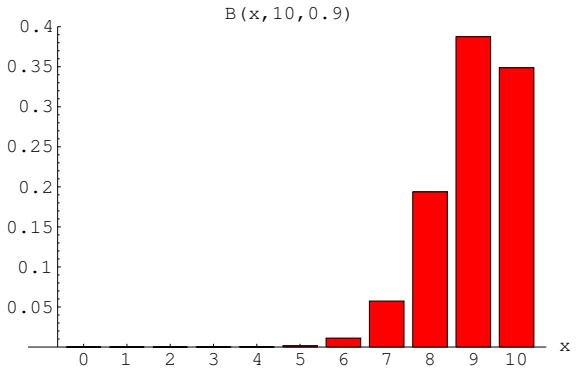
$$B_{n,p}(x) = \binom{n}{x} p^x (1-p)^{n-x}$$

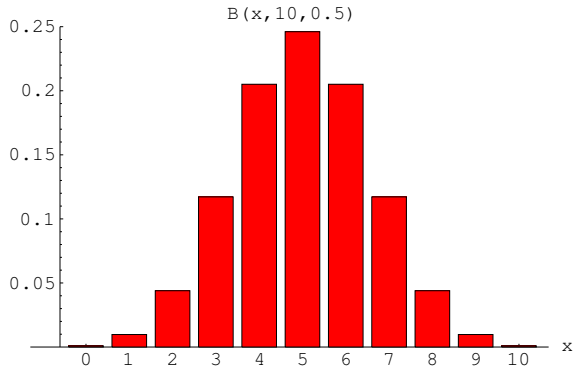
is called **binomial distribution**.

Thus, the binomial distribution indicates the probability that with n independent tries of a binary event of the probability p the result will be x times positive. Therefore, we obtain

$$B_{10,0.9}(k) = \binom{n}{k} 0.1^k \cdot 0.9^{n-k}$$

The following histograms show the densities for our example for $p = 0.9$ as well as for $p = 0.5$.





For the binomial distribution it holds

$$E(X) = \sum_{x=0}^{n} x \cdot \binom{n}{x} p^x (1-p)^{n-x} = np$$

and

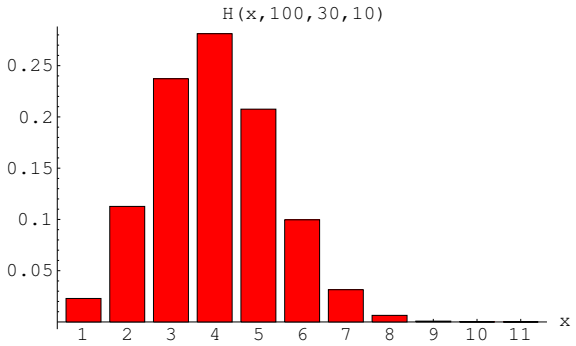
$$Var(X) = np(1-p).$$

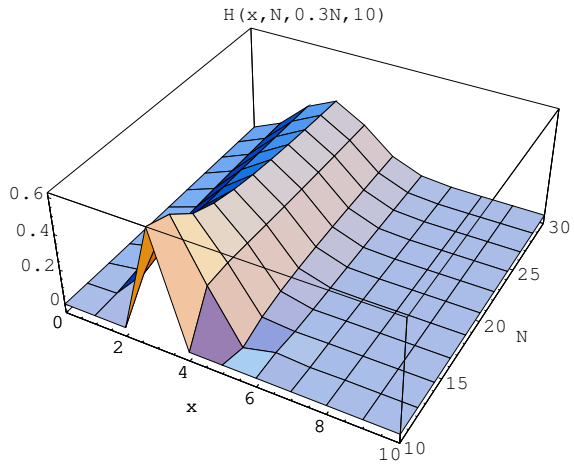
4.5.2 Hypergeometric Distribution

Let N small balls be placed in a box. K of them are black and $N-K$ white. When drawing n balls, the probability to draw x black is

$$H_{N,K,n}(x) = \frac{\binom{K}{x}\binom{N-K}{n-x}}{\binom{N}{n}}.$$

The left of the following graphs shows $H_{100,30,10}(x)$, the right one $H_{N,0.3N,10}(x)$. This corresponds to N balls in the box and 30% black balls. It is apparent, that for $N = 10$ the density has a sharp maximum, which becomes flatter with $N > 10$.





As expected, the expected value of the hypergeometric distribution is

$$E(X) = n \cdot \frac{K}{N}.$$

4.6 Continuous Distributions

Definition 4.14 A random variable X is called **continuous**, if its value range is a subset of the real numbers and if for the **density function** f and the distribution function F it holds

$$F(x) = P(X \leq x) = \int_{-\infty}^{x} f(t)dt.$$

With the requirements $P(\Omega) = 1$ and $P(\emptyset) = 0$ (see def. 4.6) we obtain

$$\lim_{x \to -\infty} F(x) = 0 \quad \text{sowie} \lim_{x \to \infty} F(x) = 1.$$

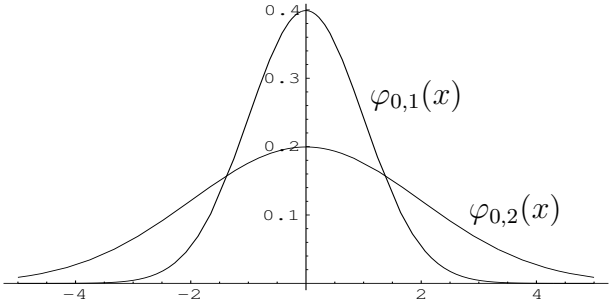
4.6.1 Normal Distribution

The most important continuous distribution for real applications is the **normal distribution** with the density

$$\varphi_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

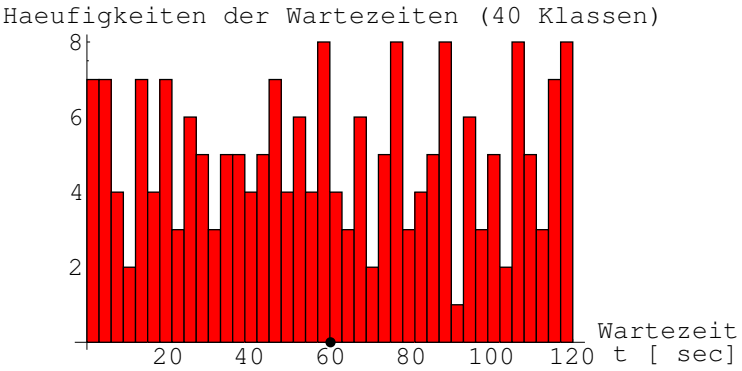
Theorem 4.3 For a normally distributed variable X with the density $\varphi_{\mu,\sigma}$ it holds $E(X) = \mu$ and $Var(X) = \sigma^2$.

For $\mu = 0$ and $\sigma = 1$ one obtains the **standard normal distribution** $\varphi_{0,1}$. With $\sigma = 2$ one obtains the flatter and broader density $\varphi_{0,2}$.



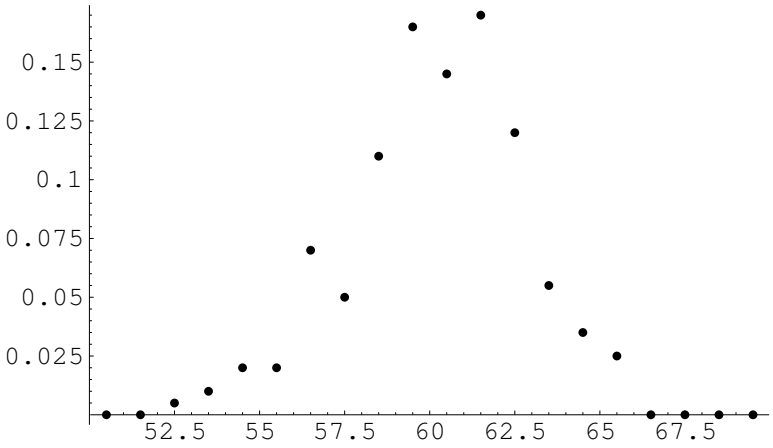
Example 4.11 Let the waiting times at a traffic light on a country road at lower traffic be uniformly distributed. We now want to **estimate** the mean waiting time by measuring the waiting time T 200 times.

The empirical frequency of the waiting times is shown opposite in the image. The mean value (•) lies at 60.165 seconds. The frequencies and the mean value indicate a uniform distribution of times between 0 und 120 sec.



Due to the finiteness of the sample, the mean value does not lie exactly at the expected value of 60 seconds. We now might ask the question, if the mean value is reliable, more precise with what probability such a measured mean differs from the expected value by a certain deviation. This will be investigated regarding the mean value from 200 times as random variable while recording a sample for the mean value. For example, we let 200 people independently measure the mean value from 200 records of the waiting time at a traffic light. We obtain the following result:

The empirical density function of the distribution of the mean value $\bar{t}$ shows a clear maximum at $t = 60$ seconds while steeply sloping at the borders at 0 and 120 seconds. It looks like a normal distribution.



The kind of relation between the distribution of the mean value and the normal distribution is shown by the following theorem:

Theorem 4.4 (Central Limit Theorem) If $X_1, X_2, \ldots, X_n$ are independent identically distributed random variables with $\sigma(X_i) < \infty$ and

$$S_n = X_1 + \ldots + X_n,$$

then S_n tends (for $n \to \infty$) to a normal distribution with the expected value $nE(X_1)$ and the standard deviation of $\sqrt{n}\sigma$. It holds

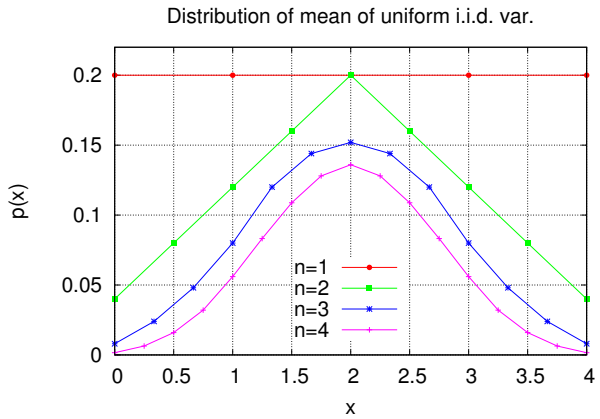
$$\lim_{n\to\infty} \sup\{|S_n(x) - \varphi_{nE(X_1),\sqrt{n}\sigma(X_1)}(x)| \,:\, x \in \mathbb{R}\} = 0.$$

This theorem has some important conclusions:

- The sum of independent identically distributed random variables asymptotically tends to a normal distribution.
- The mean of the n independent measurements of a random variable is approximately normally distributed. The approximation holds better, the more measurements are made.
- The standard deviation of a sum $X_1 + \ldots + X_n$ of identically distributed random variables is equal to $\sqrt{n}\sigma(X_1)$.

Example 4.12

The following diagram shows the (exact) distribution of the mean calculated from n i.i.d. (independent identically distributed) discrete variables, each uniformly distributed: $p(0) = p(1) = p(2) = p(3) = p(4) = 1/5$.



With the help of the central limit theorem we now want to determine the normal distribution of the mean value from example 4.11 in order to compare it with the empirical density of the mean value. The mean value $\bar{t}_n$ after n time measurements is

$$\bar{t}_n = \frac{1}{n}\sum_{i=1}^{n} t_i.$$

Following theorem 4.4, the sum $\sum_{i=1}^{n} t_i$ is normally distributed and has the density

$$\varphi_{nE(X_1),\sqrt{n}\sigma}(x) = \frac{1}{\sqrt{2\pi}\sqrt{n}\sigma} \exp\left(-\frac{(x - nE(T))^2}{2n\sigma^2}\right)$$

The mean value $\bar{t}_n$ has the density $\varphi_{E(T),\frac{\sigma}{\sqrt{n}}}$. [5] The variance σ^2 of the uniform distribution

[5] This is given by the following, easy to proof property of the variance: $Var(X/n) = 1/n^2 Var(X)$.

is still missing.

Definition 4.15 The density of the **uniform distribution over the interval** $(\mathbf{a}, \mathbf{b})$ (also called rectangular distribution) is

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{if } a \le x \le b \\ 0 & \text{if sonst} \end{cases}$$

One calculates

$$E(X) = \frac{1}{b-a}\int_a^b x\,dx = \frac{a+b}{2} \tag{4.1}$$

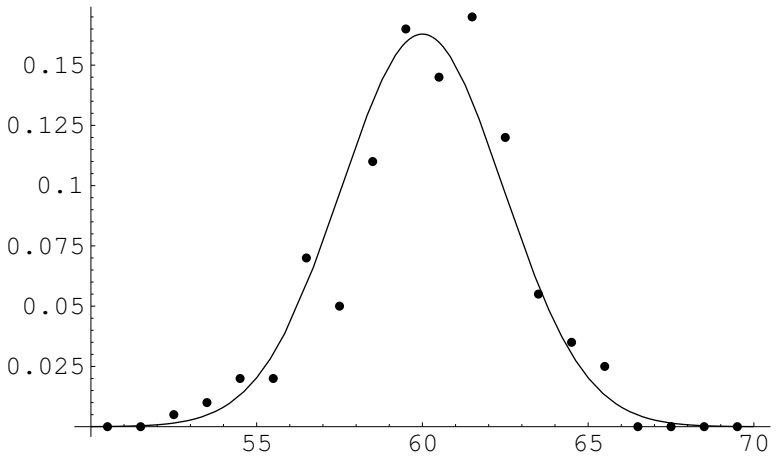
$$Var(X) = E(X^2) - E(X)^2 = \frac{1}{b-a}\int_a^b x^2\,dx - \left(\frac{a+b}{2}\right)^2 = \frac{(b-a)^2}{12} \tag{4.2}$$

Therefore, for the example one calculates

$$\frac{\sigma}{\sqrt{n}} = \frac{(b-a)}{\sqrt{12n}} = \frac{120}{\sqrt{12\cdot 200}} = \sqrt{6}$$

Thus, the density of the mean value of the traffic light waiting times should be approximated well by $\varphi_{60,\sqrt{6}}$ as it can be seen in the following image.

Density function of the distribution of the mean value with the density of the normal distribution $\varphi_{60,\sqrt{6}}$.



Since we now know the density of the mean value, it is easy to specify a symmetric interval in which the mean value (after our 200 measurements) lies with a probability of 0.95. In the image above ($\varphi_{60,\sqrt{6}}$) we have to determine the two points u_1 and u_2, which behave

$$P(u_1 \le \bar{t} \le u_2) = \int_{u_1}^{u_2} \varphi_{60,\sqrt{6}}(t)\,dt = 0.95$$

Because of

$$\int_{-\infty}^{\infty} \varphi_{60,\sqrt{6}}(t)\,dt = 1$$

it must behave

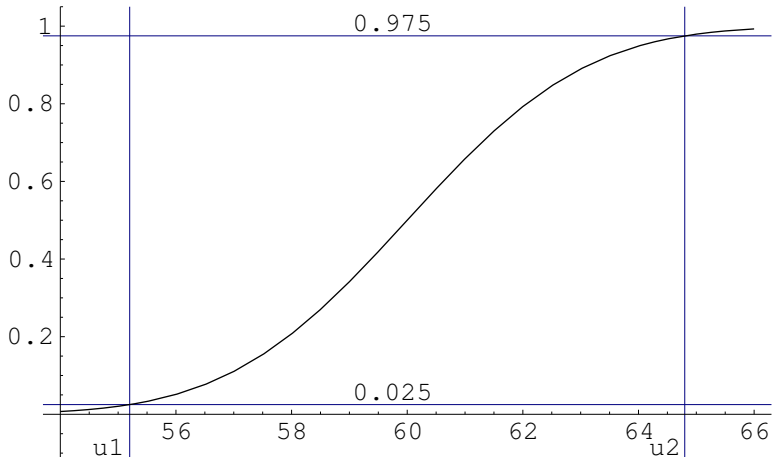
$$\int_{-\infty}^{u_1} \varphi_{60,\sqrt{6}}(t)\,dt = 0.025 \quad \text{und} \quad \int_{-\infty}^{u_2} \varphi_{60,\sqrt{6}}(t)\,dt = 0.975.$$

Graphically, we can find the two points u_1, u_2, searching for the x values to the level 0.025 and 0.975 in the graph of the distribution function of the normal distribution

$$\Phi_{60,\sqrt{6}}(x) = P(X \leq x) = \int_{-\infty}^{x} \varphi_{60,\sqrt{6}}(t)\, dt$$

From the image on the opposite we read out

$$u_1 \approx 55.2, \quad u_2 \approx 64.8.$$



We now know the following: After our sample of 200 time measurements the expexted value of our waiting time t lies in the interval $[55.2, 64.8]$ with a probability of 0.95.[6] This interval is called the **confidence interval** to the level 0.95.
In general, the confidence interval $[u_1, u_2]$ to the level $1 - \alpha$ has the following meaning. Instead of estimating a paramater Θ from sample measurements, we can try to determine an interval, that contains the value of Θ with high probability. For a given number α (in the example above, α was 0.05) two numbers u_1 and u_2 are sought which behave

$$P(u_1 \leq \Theta \leq u_2) = 1 - \alpha.$$

Not to be confused with the confidence interval are the quantiles of a distribution.

Definition 4.16 Let X be a continuous random variable and $\gamma \in (0, 1)$. A value x_γ is called **γ-quantile**, if it holds

$$P(X \leq x_\gamma) = \int_{-\infty}^{x_\gamma} f(t)\, dt = \gamma.$$

The 0.5 quantile is called **median**.

4.7 Exercises

Exercise 4.1

[6] This result is only exact under the condition that the standard deviation σ of the distribution of t is known. If σ is unknown too, the calculation is more complex.

a) Show that the arithmetic mean $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ minimizes the sum of the squared distances $\sum_{i=1}^{n}(x_i - x)^2$.

b) Show that the median

$$\tilde{x} = \begin{cases} x_{\frac{n+1}{2}} & \text{if } n \text{ odd} \\ \frac{1}{2}\left(x_{\frac{n}{2}} + x_{\frac{n+1}{2}}\right) & \text{if } n \text{ even} \end{cases}$$

minimizes the sum of the absolute values of the distances $\sum_{i=1}^{n} |x_i - x|$. (Hint: consider by an example how $\sum_{i=1}^{n} |x_i - x|$ is going to change if x deviates from the median.)

Exercise 4.2 As thrifty, hard-working Swabians we want to try to calculate whether the German lottery is worth playing. In German lottery, 6 balls are drawn out of 49. The 49 balls are numbered from 1-49. A drawn ball is not put back into the pot. In each lottery ticket field, the player chooses 6 numbers out of 49.

a) Calculate the number of possible draws in the lottery (6 of 49 / saturday night lottery), which result in having (exactly) three correct numbers. Then, what is the probability to have three correct numbers?
b) Give a formula for the probability of achieving n numbers in the lottery.
c) Give a formula for the probability of achieving n numbers in the lottery with the bonus number (the bonus number is determined by an additionally drawn 7th ball).
d) What is the probability that the (randomly) drawn "super number" (a number out of $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$) equals the last place of the serial number of the lottery ticket?
e) Calculate the average lottery prize if the following sums are payed out (s.n.: super number, b.n.: bonus number):

Winning class	I	II	III	IV	V	VI	VII
Correct numbers	6 with s.n.	6 without s.n.	5 with b.n.	5	4	3 with b.n.	3
Prize (6.12.1997)	4.334.833,80	1.444.944,60	135.463,50	10.478,20	178,20	108,70	11,00
Prize (29.11.1997)	12.085.335,80	1.382.226,80	172.778,30	12.905,90	192,30	82,30	12.10
Prize (22.11.1997)	7.938.655,30	3.291.767,70	141.075,70	11.018,40	157,50	79,20	10,10
Prize (15.11.1997)	3.988.534,00	2.215.852,20	117.309,80	9.537,30	130,70	60,80	8,70
Prize (8.11.1997)	16.141.472,80	7.288.193,60	242.939,70	14.798,30	190,10	87,70	10,90

Exercise 4.3 Show that for the variance the following rule holds

$$Var(X) = E(X^2) - E(X)^2.$$

Exercise 4.4

a) For pairwise inconsistent events A and B it holds $P(A \cup B) = P(A) + P(B)$. (Hint: consider, how the second part of definition 10.6 could be applied on (only) 2 events.)
b) $P(\emptyset) = 0$, i.e. the impossible event has the probability 0.
c) For two complementary events A and $\bar{A}$ it holds $P(A) + P(\bar{A}) = 1$.

d) For arbitrary events A and B it holds $P(A \cup B) = P(A) + P(B) - P(A \cap B)$.
e) For $A \subseteq B$ it holds $P(A) \leq P(B)$.

Exercise 4.5 Give an example for an estimator with 0 variance.

Exercise 4.6 Show that for the sample variance it holds:

$$s^2 = \frac{1}{n-1} \sum_{j=1}^{n} (x_j - \mu)^2 - \frac{n}{n-1} (\bar{x} - \mu)^2.$$

Chapter 5

Numerical Mathematics Fundamentals

5.1 Arithmetics on the Computer

5.1.1 Floating Point Numbers

The set of floating point numbers to base β, with t fractional digits and exponents between m and M, can be formally defined by

$$F(\beta, t, m, M) = \{d : d = \pm.d_1 d_2 \dots d_t \cdot \beta^e\} \cup \{0\} \subset \mathbb{Q}$$

with

$$\begin{aligned}
&\beta \in \mathbb{N} \\
&0 \leq d_i \leq \beta - 1 \qquad d_i : \text{digits}, \quad d_1 \neq 0 \\
&d_1, d_2, \dots, d_t : \text{mantissa} \\
&t : \text{mantissa length} \\
&e : \text{exponent with} \quad m \leq e \leq M \qquad m, M \in \mathbb{Z}
\end{aligned}$$

The floating point number $\pm.d_1 d_2 \dots d_t \cdot \beta^e$ has the value

$$d = \pm \left(d_1 \beta^{e-1} + d_2 \beta^{e-2} + \dots + d_t \beta^{e-t} \right)$$

Example 5.1 Let $\beta = 2$ and $t = 3$ given, that means we consider three-digit numbers in the binary system. The number $0.101 \cdot 2^{21}$ has the value

$$0.101 \cdot 2^{21} = 1 \cdot 2^{20} + 0 \cdot 2^{19} + 1 \cdot 2^{18} = 2^{20} + 2^{18}.$$

In the decimal system with $\beta = 10$ we need a six-digit mantissa ($t = 6$), to represent this number:

$$2^{20} + 2^{18} = 1310720 = 0.131072 \cdot 10^7.$$

5.1.1.1 Distribution of $F(\beta, t, m, M)$

$$|F(\beta, t, m, M)| = \underbrace{2}_{\pm} \underbrace{(M - m + 1)}_{\text{exponents}} \underbrace{(\beta^t - \beta^{(t-1)})}_{\text{mantissas}} + \underbrace{1}_{0}$$

Example 5.2 $F(2,3,-1,2)$

with the upper formula we get:

$$|F(2,3,-1,2)| = 2(4)(2^3 - 2^2) + 1 = 33$$

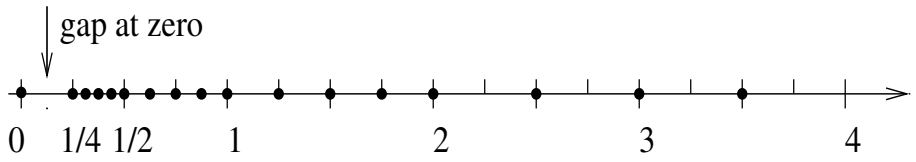
$\Rightarrow$ there are only the "0" and 32 different numbers between

$\pm 0.100 \cdot 2^{-1}$, the number with smallest absolute value
$\pm 0.111 \cdot 2^{2}$, the number with largest absolute value

The elements ≥ 0 of F(2,3,-1,2) are

$$0; \frac{1}{4}, \frac{5}{16}, \frac{3}{8}, \frac{7}{16}; \frac{1}{2}, \frac{5}{8}, \frac{3}{4}, \frac{7}{8}; 1, \frac{5}{4}, \frac{3}{2}, \frac{7}{4}; 2, \frac{5}{2}, 3, \frac{7}{2}$$

Distribution on the number line:



problems:

- Exponent overflow
- Exponent underflow
- Round-off error

5.1.2 Round-off Errors

5.1.2.1 Round-off and Truncation Errors (absolute)

Definition 5.1 $\mathrm{fl}_c, \mathrm{fl}_r : \left[-0.\alpha \ldots \alpha \cdot \beta^M, 0.\alpha \ldots \alpha \cdot \beta^M\right] \rightarrow F(\beta, t, m, M)$ with $\alpha = \beta - 1$

Round-off: $x \mapsto \mathrm{fl}_r(x)$ = nearest neighbor of x in $F(\beta, t, m, M)$

Truncate: $x \mapsto \mathrm{fl}_c(x) = \max\{y \in F(\beta, t, m, M) | y \leq x\}$

It holds:

$$\begin{aligned} \text{absolute value Round-off Errors} &= |\mathrm{fl}_r(x) - x| \leq \frac{1}{2}\beta^{e-t} \\ \text{absolute value Truncation Error} &= |\mathrm{fl}_c(x) - x| < \beta^{e-t} \end{aligned}$$

Example 5.3 $\underbrace{\beta = 10}_{\text{10er System}}$, $\overbrace{t = 2}^{\text{2stellige Mantisse}}$, $\underbrace{e = 3}_{\text{Exponent}}$

$x = 475$
$\mathrm{fl}_r(x) = 0.48 \cdot 10^3$ $\leftarrow$ round-off

$\text{fl}_c(x) = 0.47 \cdot 10^3 \quad \leftarrow$ truncate

$$|\text{fl}_r(x) - x| = |480 - 475| = 5 \leq \frac{1}{2} \cdot 10^{3-2} = 5$$
$$|\text{fl}_c(x) - x| = |470 - 475| = 5 < 10^{3-2} = 10$$

5.1.2.2 Round-off and Truncation Errors (relative)

$$\frac{|\text{fl}_r(x) - x|}{|x|} \leq \frac{1}{2}\beta^{1-t}$$
$$\frac{|\text{fl}_c(x) - x|}{|x|} < \beta^{1-t}$$

Example 5.4 relative round-off error

$$\frac{|480 - 475|}{|475|} = \frac{1}{95} \leq \frac{1}{2} \cdot 10^{-1} = \frac{1}{20}$$

$\rightarrow$ upper bound for the smallest number!

$$\frac{|110 - 105|}{|105|} = \frac{1}{21} < \frac{1}{20}$$

For fixed number of digits, the relative error gets bigger for smaller numbers!

Example 5.5 t=3, $\beta = 10$
$110 \cdot 105 = 11550 \neq 11600 = \text{fl}_r(11550)$
Achtung:
Field axioms violated!
$F(\beta, t, m, M)$ is not closed w.r.t. multiplication.

Let $\star \in \{+, -, \cdot, \text{div}\}$

$$\exists x, y \in F(\beta, t, m, M) : \text{fl}_r(x \star y) \neq x \star y$$

5.1.3 Cancellation

Example 5.6 Let $\beta = 10$ and $t = 8$

$$\begin{aligned}
a &= 0.1 \cdot 10^9 \\
b &= 0.1 \cdot 10^1 \\
c &= -0.1 \cdot 10^9 \\
a + b + c &= 0.1 \cdot 10^1 = 1 \\
\text{fl}_r\left(\text{fl}_r\left(a + b\right) + c\right) &= 0.1 \cdot 10^9 - 0.1 \cdot 10^9 = 0 \\
\text{fl}_r\left(a + \text{fl}_r\left(b + c\right)\right) &= 0.1 \cdot 10^9 - 0.1 \cdot 10^9 = 0 \\
\text{fl}_r\left(\text{fl}_r\left(a + c\right) + b\right) &= 0 + 0.1 \cdot 10^1 = 1
\end{aligned}$$

$\Rightarrow$ Associative law is not valid in $F(\beta, t, m, M)$

5.1.4 Condition Analysis

Example 5.7 Solve the linear system

$$\begin{aligned} x + ay &= 1 \\ ax + y &= 0 \end{aligned}$$

$$\begin{aligned} x - a^2 x &= 1 \\ x &= \frac{1}{1-a^2} \qquad \text{für } a \neq \pm 1 \end{aligned}$$

$a = 1.002$ = exact value
$\tilde{a} = 1.001$ = measurement or rounding-off error

$$\text{relative error:} \quad \left|\frac{\tilde{a} - a}{a}\right| = \frac{1}{1002}$$

solution:

$$x \approx -\frac{1}{0.004} \approx -249.75$$

$$\tilde{x} \approx -\frac{1}{0.002} \approx -499.75$$

$$\Rightarrow \text{relative error } \left|\frac{\tilde{x} - x}{x}\right| \approx \left|\frac{-250}{249.75}\right| = 1.001 \qquad (100\% \text{ error})$$

See Figure 5.1.

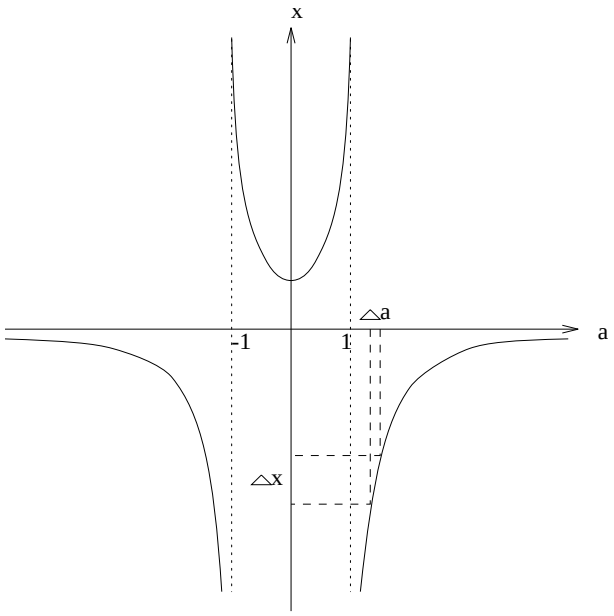


Figure 5.1: Gain of the input error under ill-condition.

Matrix $A = \begin{pmatrix} 1 & a \\ a & 1 \end{pmatrix}$ is singular for $a = 1$, i.e. $\begin{vmatrix} 1 & 1 \\ 1 & 1 \end{vmatrix} = 0$

Definition 5.2 Let P be the problem to calculate the function $f(x)$ with given input x. The condition number C_p is the factor by which a relative error $\frac{\Delta x}{x}$ in the input f will be increased, i.e.

$$\left|\frac{f(x+\Delta x)-f(x)}{f(x)}\right| = C_p \left|\frac{\Delta x}{x}\right|$$

It holds:

$$C_p = \left|\frac{(f(x+\Delta x)-f(x))/f(x)}{\Delta x/x}\right| \approx \left|\frac{f'(x)}{f(x)}x\right|$$

Example 5.8 Calculation of C_p

$$x = f(a) = \frac{1}{1-a^2} \qquad f'(a) = \frac{2a}{(1-a^2)^2}$$

$$C_p \approx \left|\frac{2a}{(1-a^2)^2}(1-a^2)a\right| = \left|\frac{2a^2}{1-a^2}\right| = 501.5$$

direct calculation (see above): $C_p \approx 1002$
Factor 2 due to linearization of f in a!

Definition 5.3 A problem is ill-conditioned (well-conditioned) if $C_p \gg 1$ ($C_p < 1$ oder $C_p \approx 1$)

Note: C_p depends on the input data!

5.2 Numerics of Linear Systems of Equations

see [1]

5.2.1 Solving linear equations (Gauß' method)

Linear System $Ax = b$**:**

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n \\ a_{ij} \in \mathbb{R} \qquad n \geq 1 & \end{aligned}$$

Questions:

- Is L solvable?
- Is there a unique solution?
- How to calculate the solutions?
- Is there an efficient algorithm?

5.2.1.1 Gaußian Elimination Method

$$
\begin{array}{ccccccccccccccc}
a_{11}x_1 & + & a_{12}x_2 & + & & & \cdots & & & & & + & a_{1n}x_n & = & b_1 \\
 & & a_{22}x_2 & + & & & \cdots & & & & & + & a_{2n}x_n & = & b_2 \\
 & & & & a_{kk}x_k & + & \cdots & + & a_{kj}x_j & + & \cdots & + & a_{kn}x_n & = & b_k \\
 & & & & \vdots & & & & \vdots & & & & \vdots & & \\
 & & & & a_{ik}x_k & + & \cdots & + & a_{ij}x_j & + & \cdots & + & a_{in}x_n & = & b_i \\
 & & & & \vdots & & & & \vdots & & & & \vdots & & \\
 & & & & a_{nk}x_k & + & \cdots & + & a_{nj}x_j & + & \cdots & + & a_{nn}x_n & = & b_n
\end{array}
$$

The algorithm:

```
for k=1,...,n-1
  search a_mk with |a_mk|=max{ |a_lk| : l >= k }
  if a_mk=0 print "singulaer"; stop
  swap lines m and k
  for i=k+1,...,n
    q_ik:=a_ik/a_kk
    for j=k+1,...,n
      a_ij:=a_ij - q_ik*a_kj
    end
    b_i:=b_i - q_ik*b_k
  end
end
```

Theorem 5.1 Complexity: The number of operations of the Gaußian elimination for large n is approximately equal to $\frac{1}{3}n^3$.

Proof:

$$
\begin{array}{rcl}
 & \overbrace{\text{lines}}\ \overbrace{\text{columns}} & \\
\text{1. step:} & \overbrace{(n-1)}\overbrace{(n-1+2)} & \text{operations} \\
\text{k-ter step:} & (n-k)(n-k+2) & \text{operations}
\end{array}
$$

total:

$$
T(n) = \sum_{k=1}^{n-1}(n-k)(n-k+2) \overbrace{=}^{l:=n-k} \sum_{l=1}^{n-1}(l(l+2))
$$

$$
= \sum_{l=1}^{n-1}(l^2+2l) = \frac{n^3}{3} - \frac{n^2}{2} + \frac{n}{6} + n(n-1)
$$

$$
= \frac{n^3}{3} + \frac{n^2}{2} - \frac{5}{6}n
$$

$$
\Rightarrow \quad \text{for large n:} \quad \frac{n^3}{3}
$$

Example 5.9 Computer with 1 GFLOPS

n	$T(n)$
10	$1/3 \cdot 10^3 \cdot 10^{-9}$ sec $\approx$ 0.3 μsec
100	$1/3 \cdot 100^3 \cdot 10^{-9}$ sec $\approx$ 0.3 msec
1000	$1/3 \cdot 1000^3 \cdot 10^{-9}$ sec $\approx$ 0.3 sec
10000	$1/3 \cdot 10000^3 \cdot 10^{-9}$ sec $\approx$ 300 sec = 5 min

Problems/Improvements:

1. long computing times for large n
   - better algorithms

     $$T(n) = C \cdot n^{2.38} \qquad \text{instead of} \qquad \frac{1}{3}\, n^3$$

   - Iterative method (Gauß-Seidel)
2. Round-off error
   - complete pivoting
   - Gauß-Seidel

Applications:

- Construction of curves through given points
- Estimation of parameters (least squares)
- Linear Programming
- Computer graphics, image processing (e.g. computer tomography)
- Numerical solving of differential equations

5.2.1.2 Backward Substitution

After n-1 elimination steps:

$$A'x = b \qquad \text{with} \qquad A' = \begin{pmatrix} a'_{11} & a'_{12} & \cdots & a'_{1n} \\ 0 & a'_{22} & \cdots & a'_{2n} \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & a'_{nn} \end{pmatrix}$$

Calculation of $x_1, \ldots, x_n$:

$$\begin{aligned} x_n &= \frac{b_n}{a'_{nn}} \\ x_{n-1} &= \frac{b_{n-1} - a'_{n-1,n} x_n}{a'_{n-1,n-1}} \end{aligned}$$

General:

$$x_i = \frac{b_i - \sum_{k=i+1}^{n} a'_{ik} x_k}{a'_{ii}}$$

$$i = n, n-1, \ldots, 1$$

Runtime:

- Divisions: n
- Number of additions and multiplications:

$$\sum_{i=1}^{n}(i-1) = \sum_{i=1}^{n-1} i = \frac{1}{2}n(n-1) \approx \frac{1}{2}n^2$$

⇒ Substitution is much faster than elimination!

5.2.1.3 Backward Elimination

A slight variant of the backward substitution is the backward elimination, where the upper right triangle of the matrix is being substituted similarly to the Gauß elimination. This variant is called Gauß-Jordan method. One application of this method is the computation of inverse matrices.

Theorem 5.2 Correctness: The Gaußian Method results in a unique solution $(x_1, \ldots, x_n)$ if and only if the linerar system L has a unique solution $(x_1, \ldots, x_n)$.

Proof: as exercise

5.2.2 Iterative improvement of the solution

Let $\bar{x}$ the calculated solution of $Ax = b$ with the Gauß method. In general $A\bar{x} = b - r$ with $r \neq \boldsymbol{0}$ (r: residual vector) because of $x = \bar{x} + \Delta x$.

$$\Rightarrow A\bar{x} = A(x - \Delta x) = b - r$$

$$A \cdot \Delta x = r$$

With this equation the correction $\Delta\bar{x}$ can be calculated. ⇒ better approximation for x:

$$x^{(2)} = \bar{x} + \Delta\bar{x}$$

Iterative Method:

$$x^{(1)} := \bar{x}$$

for $n = 1, 2, 3, \ldots$:

$$r^{(n)} = b - Ax^{(n)}$$

$$\text{calculate} \quad \Delta x^{(n)} \quad \text{nach} \quad A\Delta x^{(n)} = r^{(n)}$$

$$x^{(n+1)} = x^{(n)} + \Delta x^{(n)}$$

Note:

1. usually (A not very ill-conditionated) very few iterations (≈ 3) necessary.
2. Solving $A\Delta x^{(n)} = r^{(n)}$ is time-consuming: $O(\frac{1}{3}n^3)$. With LU decomposition (see 5.2.3) of A, $A\Delta x^{(n)} = r^{(n)}$ can be solved in $O(n^2)$ steps.
3. Must a system of equations be solved for more than one right hand side, all solutions will be calculated simultaneously (elimination necessary only once!)

5.2.3 LU-Decomposition

The Gaußian elimination (see algorithm) multiplies row i with the factor $q_{ik} := a_{ik}/a_{kk}$ for the elimination of each element a_{ik} in the k-th column below the diagonal. If we write all calculated q_{ik} in a lower triangular matrix, in which we add ones in the diagonal, we get

$$L := \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ q_{21} & 1 & 0 & & \vdots \\ q_{31} & q_{32} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ q_{n1} & q_{n2} & \dots & q_{nn-1} & 1 \end{pmatrix}.$$

Furthermore, let

$$U := A' = \begin{pmatrix} a'_{11} & a'_{12} & \cdots & a'_{1n} \\ 0 & a'_{22} & \cdots & a'_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a'_{nn} \end{pmatrix}$$

the upper triangular matrix after the elimination.

Theorem 5.3 Then $L \cdot U = A$ holds and the solution x of the system $Ax = b$ for any right hand side b can be calculated by solving the equation $L \cdot c = b$ for c and solving $U \cdot x = c$ for x.

The system $L \cdot c = b$ is solved by forward substitution and $U \cdot x = c$ by backward substitution.

Proof: We will show that $L \cdot U = A$. Then obviously it holds

$$A \cdot x = L \cdot U \cdot x = b.$$

Now we write $L \cdot U = A$ in detail:

$$L \cdot U = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ q_{21} & 1 & 0 & & \vdots \\ q_{31} & q_{32} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ q_{n1} & q_{n2} & \dots & q_{nn-1} & 1 \end{pmatrix} \cdot \begin{pmatrix} a'_{11} & a'_{12} & \cdots & a'_{1n} \\ 0 & a'_{22} & \cdots & a'_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a'_{nn} \end{pmatrix} = A$$

We now apply the Gaußian elimination on both sides and get

$$\begin{pmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{pmatrix} \cdot U = U$$

Thus $LU = A$. Because of the associativity of matrix multiplication only L has to be eliminated on the left side. □

Exercise 5.1 How could you factor A into a product UL, upper triangular times lower triangular? Would they be the same factors as in $A = LU$?

5.2.4 Condition Analysis for Matrices

$$Ax = b \qquad \text{with} \qquad A : \text{Matrix } (n \times n) \text{ and } x, b \in \mathbb{R}^n$$

What is the Norm of a matrix?

Vector Norm:

Definition 5.4 (p-Norm)

$$\forall x \in \mathbb{R}^n : \|x\|_p = (|x_1|^p + |x_2|^p + \cdots + |x_n|^p)^{\frac{1}{p}}$$

$$1 \leq p < \infty$$

Theorem 5.4 $\|x\|_p$ is a norm, i.e. it has the properties:

- $\forall x \neq 0 : \|x\|_p > 0 \quad ; \quad \|x\|_p = 0 \Leftrightarrow x = 0$
- $\forall \alpha \in \mathbb{R} : \|\alpha x\|_p = |\alpha| \cdot \|x\|_p$
- $\forall x, y \in \mathbb{R}^n : \|x + y\|_p \leq \|x\|_p + \|y\|_p$

Lemma 5.1 (Hölder inequality) For real numbers $p, q > 1$ with $\frac{1}{p} + \frac{1}{q} = 1$ and vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ we have

$$\|\boldsymbol{x}\boldsymbol{y}\|_1 \leq \|\boldsymbol{x}\|_p \|\boldsymbol{y}\|_q.$$

Proof: Since $\|\boldsymbol{x}\boldsymbol{y}\|_1 = \left|\sum_{i=1}^{n} x_i y_i\right| \leq \sum_{i=1}^{n} |x_i y_i|$ it remains to prove

$$\sum_{i=1}^{n} |x_i y_i| \leq \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} |y_i|^q\right)^{\frac{1}{q}}.$$

For real numbers $a, b > 0$ we have (proof as exercise)

$$ab \leq \frac{a^p}{p} + \frac{b^q}{q},$$

which we apply now to get

$$\sum_{i=1}^{n} \frac{|x_i y_i|}{\|\boldsymbol{x}\|_p \|\boldsymbol{y}\|_q} = \sum_{i=1}^{n} \frac{|x_i||y_i|}{\|\boldsymbol{x}\|_p \|\boldsymbol{y}\|_q} \leq \sum_{i=1}^{n} \left(\frac{1}{p} \frac{|x_i|^p}{\|\boldsymbol{x}\|_p^p} + \frac{1}{q} \frac{|y_i|^q}{\|\boldsymbol{y}\|_q^q} \right)$$
$$= \sum_{i=1}^{n} \frac{1}{p} \frac{|x_i|^p}{\|\boldsymbol{x}\|_p^p} + \sum_{i=1}^{n} \frac{1}{q} \frac{|y_i|^q}{\|\boldsymbol{y}\|_q^q} = \frac{1}{p\|\boldsymbol{x}\|_p^p} \sum_{i=1}^{n} |x_i|^p + \frac{1}{q\|\boldsymbol{y}\|_q^q} \sum_{i=1}^{n} |y_i|^q = \frac{1}{p} + \frac{1}{q} = 1$$

□

Proof of proposition 3 in Theorem 5.4: For the cases $p = 1$ and $p = \infty$ see exercises. For $1 < p < \infty$:

$$|x_i + y_i|^p = |x_i + y_i||x_i + y_i|^{p-1} \leq (|x_i| + |y_i|)|x_i + y_i|^{p-1} = |x_i||x_i + y_i|^{p-1} + |y_i||x_i + y_i|^{p-1}$$

Summation yields

$$\sum_{i=1}^{n} |x_i + y_i|^p \leq \sum_{i=1}^{n} |x_i||x_i + y_i|^{p-1} + \sum_{i=1}^{n} |y_i||x_i + y_i|^{p-1}. \tag{5.1}$$

Application of the Hölder inequality to both terms on the right hand sides gives

$$\sum_{i=1}^{n} |x_i||x_i + y_i|^{p-1} \leq \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} (|x_i + y_i|^{p-1})^q \right)^{\frac{1}{q}}$$

and

$$\sum_{i=1}^{n} |y_i||x_i + y_i|^{p-1} \leq \left(\sum_{i=1}^{n} |y_i|^p \right)^{\frac{1}{p}} \left(\sum_{i=1}^{n} (|x_i + y_i|^{p-1})^q \right)^{\frac{1}{q}}$$

what we substitute in Equation 5.1 to obtain

$$\sum_{i=1}^{n} |x_i + y_i|^p \leq \left(\left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}} + \left(\sum_{i=1}^{n} |y_i|^p \right)^{\frac{1}{p}} \right) \left(\sum_{i=1}^{n} |x_i + y_i|^p \right)^{\frac{1}{q}}.$$

In the rightmost factor we used $(p-1)q = p$. Now we divide by the rightmost factor, using $\frac{1}{p} = 1 - \frac{1}{q}$ and get the assertion

$$\left(\sum_{i=1}^{n} |x_i + y_i|^p \right)^{\frac{1}{p}} \leq \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}} + \left(\sum_{i=1}^{n} |y_i|^p \right)^{\frac{1}{p}}.$$

□

Lemma 5.2

$$\|x\|_\infty := \max_{1 \leq i \leq n} |x_i| = \lim_{p \to \infty} \|x\|_p$$

$\| \ \|_\infty$ is called maximum norm

In the following let $\|x\| = \|x\|_\infty$

maximum norm:

Definition 5.5 For any vector norm $\| \ \|$ the canonical matrix norm is defined as follows:

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}$$

Lemma 5.3 The matrix norm is a norm and for a $n \times m$ matrix A it holds

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{m} |a_{ij}|$$

$$\|Ax\| \leq \|A\| \cdot \|x\|$$

$$\|AB\| \leq \|A\| \cdot \|B\|$$

Condition of a matrix: Consequence of errors in the matrix elements of A or the right hand side b on errors in the solution x.

1. Error in b:

$$\begin{aligned} \tilde{b} &= b + \Delta b \\ \Rightarrow \tilde{x} &= x + \Delta x \\ \Rightarrow A(x + \Delta x) &= b + \Delta b \\ \Rightarrow \Delta x &= A^{-1} \Delta b \\ \Rightarrow \|\Delta x\| &= \|A^{-1} \Delta b\| \leq \|A^{-1}\| \cdot \|\Delta b\| \end{aligned}$$

$$b = Ax \quad \Rightarrow \quad \|b\| \leq \|A\| \cdot \|x\| \quad \Rightarrow \quad \frac{1}{\|x\|} \leq \frac{\|A\|}{\|b\|}$$

$$\Rightarrow \frac{\|\Delta x\|}{\|x\|} \leq \|A\| \cdot \|A^{-1}\| \frac{\|\Delta b\|}{\|b\|}$$

$$\frac{\frac{\|\Delta x\|}{\|x\|}}{\frac{\|\Delta b\|}{\|b\|}} \leq C_A$$

$$\text{with} \quad C_A = \|A\| \cdot \|A^{-1}\|$$

C_A: condition number of A

2. Error in A:

$$\begin{aligned} (A + \Delta A)(x + \Delta x) &= b \\ x + \Delta x &= (A + \Delta A)^{-1} b = (A + \Delta A)^{-1} Ax \\ \Delta x &= \left((A + \Delta A)^{-1} A - I \right) x \\ &= (A + \Delta A)^{-1} \left(A - (A + \Delta A) \right) x \\ &= (A + \Delta A)^{-1} \Delta A x \end{aligned}$$

$$\Rightarrow \|\Delta x\| \leq \|(A+\Delta A)^{-1}\| \cdot \|\Delta A\| \cdot \|x\|$$

$$\frac{\|\Delta x\|}{\|x\|} \leq \|(A+\Delta A)^{-1}\| \cdot \|A\| \cdot \frac{\|\Delta A\|}{\|A\|} \approx C_A \frac{\|\Delta A\|}{\|A\|} \approx \|A^{-1}\| \cdot \|\Delta A\|$$

$$C_A \quad \text{analogous to} \quad C_p: \qquad C_p = \left| \frac{f'(x)}{f(x)} x \right|$$

Example 5.10

$$\underbrace{\begin{pmatrix} 1 & a \\ a & 1 \end{pmatrix}}_{\text{A}} x = \underbrace{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}_{\text{b}}$$

$$A^{-1} = \frac{1}{1-a^2} \begin{pmatrix} 1 & -a \\ -a & 1 \end{pmatrix}$$

$$\|A\| = 1 + a, \qquad \|A^{-1}\| = \left| \frac{1+a}{1-a^2} \right| = \left| \frac{1}{1-a} \right| \qquad \text{for} \quad a > 0$$

$$\Rightarrow C_A = \|A\| \cdot \|A^{-1}\| = \left| \frac{(1+a)^2}{1-a^2} \right| = \left| \frac{1+a}{1-a} \right|$$

a=1.002:

$$\Rightarrow A = \begin{pmatrix} 1 & 1.002 \\ 1.002 & 1 \end{pmatrix} \quad \Delta A = \begin{pmatrix} 0 & -0.001 \\ -0.001 & 0 \end{pmatrix}$$

$$\Rightarrow C_A = 1001$$

$$\|\Delta A\| = 0.001, \quad \|A\| = 2.002$$

$$\frac{\|\Delta x\|}{\|x\|} \lessapprox 1001 \frac{0.001}{2.002} = 0.5$$

5.3 Roots of Nonlinear Equations

given: nonlinear equation $f(x) = 0$

sought: solution(s) (root(s))

5.3.1 Approximate Values, Starting Methods

Draw the graph of $f(x)$, value table

Example 5.11

$$f(x) = \left(\frac{x}{2}\right)^2 - \sin x$$

Table:

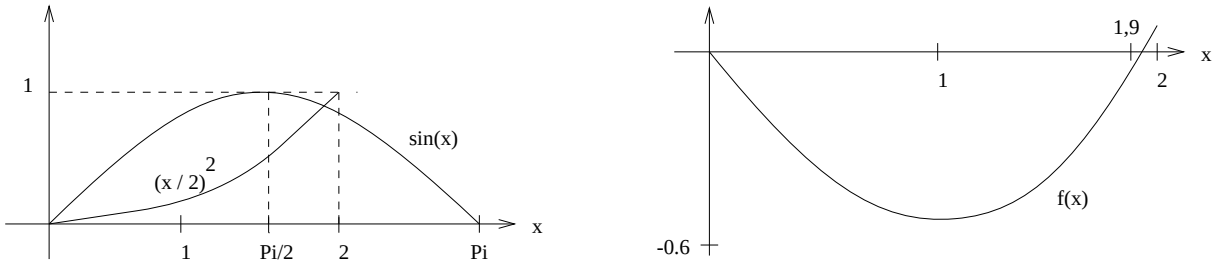


Figure 5.2: Graph to find the start value.

x	$(x/2)^2$	$\sin x$	$f(x)$
1,6	0,64	0.9996	< 0
1.8	0.81	0.974	< 0
2.0	1.00	0.909	> 0

$\Rightarrow$ Root in $[1.8; 2.0]$
in general: if f continuous and $f(a) \cdot f(b) < 0 \Rightarrow f$ has a root in $[a, b]$.

Interval bisection method

Requirements :

$f : [a, b] \to \mathbb{R}$ continuous and $f(a) \cdot f(b) < 0$.

Without loss of generality $f(a) < 0, f(b) > 0$ (otherwise take $-f(x)$)

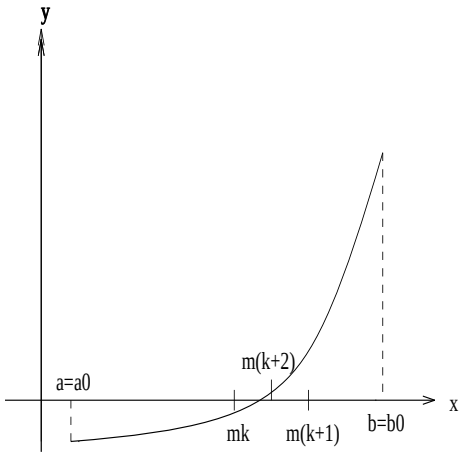


Figure 5.3: Root in the interval [a,b] can be determinated quickly by using the interval bisection method..

Algorithm:

$m_k = \frac{1}{2}(a_{k-1} + b_{k-1})$

$$(a_k, b_k) = \begin{cases} (m_k, b_{k-1}) & \text{if} \quad f(m_k) < 0 \\ (a_{k-1}, m_k) & \text{if} \quad f(m_k) > 0 \end{cases}$$

(Root found exactly if $f(m_k) = 0$)!

Theorem 5.5 Let $f : [a, b] \to \mathbb{R}$ continuous with $f(a) \cdot f(b) < 0$. Then the interval bisection method converges to a root $\bar{x}$ of f. After n steps $\bar{x}$ is determinated with a precision of $\frac{b-a}{2^n}$.

For the proof of theorem 5.5 the following definition and theorem are required:

Definition 5.6 A sequence (a_n) is a Cauchy sequence, if:

$$\forall \varepsilon > 0 : \exists N \in \mathbb{N} : \forall n, m \geq N : |a_m - a_n| < \varepsilon$$

Theorem 5.6 In $\mathbb{R}$ every Cauchy sequence converges.

Proof of theorem 5.5:

1. Speed of Convergence:
   n-th step:

$$(b_n - a_n) = \frac{1}{2}(b_{n-1} - a_{n-1}) = \ldots = \frac{1}{2^n}(b_0 - a_0) = \frac{1}{2^n}(b - a).$$

2. Convergence:

$$\bar{x} = m_{n+1} \pm \frac{1}{2}(b_n - a_n) = m_{n+1} \pm \frac{1}{2^{n+1}}(b - a)$$

   For $m \geq n + 1$ it holds

$$|a_m - a_n| \leq b_n - a_n = \frac{1}{2^n}(b - a) < \varepsilon \quad \text{for large enough } n.$$

   $\Rightarrow (a_n), (b_n)$ are Cauchy sequences $\Rightarrow (a_n), (b_n)$ converges with

$$\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n = \bar{x}$$

   because of $f(a_n) < 0 < f(b_n)$ and continuity of f.

$$\Rightarrow \left.\begin{array}{l} \lim_{n\to\infty} f(a_n) = f(\bar{x}) \leq 0 \\ \lim_{n\to\infty} f(b_n) = f(\bar{x}) \geq 0 \end{array}\right\} f(\bar{x}) = 0$$

Note:

1. for each step, the precision is doubled, respectively the distance to the solution halved. Thus for each step, the precision is improved by a binary digit.
   because of $10^{-1} \approx 2^{-3.3}$ about 3.3 steps are necessary to improve the precision by a decimal digit.
   $\Rightarrow$ slow convergence! (Example: for 12-digits precision, about 40 steps required)

2. slow convergence, **because only the sign of f is used**, $f(a_n), f(b_n)$ is never used!
   $\Rightarrow$ better methods use $f(x), f'(x), f''(x), \ldots$

3. interval bisection methods also applicable on discontinuous functions
   $\Rightarrow$ Exercise

4. discrete variants of interval bisection:
   Bisection Search (=efficient search method in ordered files)

$$T(n) \approx \log_2(n) \quad \text{instead of} \quad T(n) \approx n$$

   with n=number of entries in the file.

5. Why $\log_2(n)$ steps?
   Let $n = b - a$ the number of entries in the file.

$$\Rightarrow b_k - a_k \approx \frac{1}{2^k}(b-a) = \frac{n}{2^k}$$

   Number of steps to $b_k - a_k \leq 1$

$$\Rightarrow \frac{n}{2^k} \leq 1 \Rightarrow 2^k \geq n \Rightarrow k \geq \log_2 n$$

6. interval bisection methods globally convergent!

5.3.2 Fixed Point Iteration

Goal: Solution of equations of the form

$$x = f(x) \qquad \text{(Fixed Point Equation)}$$

Iterative Solution:

$$\begin{aligned} x_0 &= a \\ x_{n+1} &= f(x_n) \qquad (n = 0, 1, 2, \ldots) \end{aligned}$$

Example 5.12 In Figure 5.4 the solution of the fixed point equation $x = f(x)$ for various functions f is shown graphically.

Definition 5.7 A function $f : [a, b] \to [a, b] \in \mathbb{R}$ is called **a contraction** on [a,b], if a (Lipschitz) constant L with $0 < L < 1$ exists with $|f(x) - f(y)| \leq L|x - y|$ $\forall x, y \in [a, b]$

Lemma 5.4 If $f : [a, b] \to [a, b]$ is differentiable, then f is a contraction on $[a, b]$ with Lipschitz constant L if and only if holds:

$$\forall x \in [a, b] : \quad |f'(x)| \leq L < 1$$

Proof:
"→": let $|f(x) - f(y)| \leq L|x - y| \quad \forall x, y \in [a, b]$

$$\Rightarrow \forall x, y : \frac{|f(x) - f(y)|}{|x - y|} \leq L$$

$$\Rightarrow \lim_{x \to y} \frac{|f(x) - f(y)|}{|x - y|} = |f'(y)| \leq L$$

"←": (more difficult ⇒ omitted)

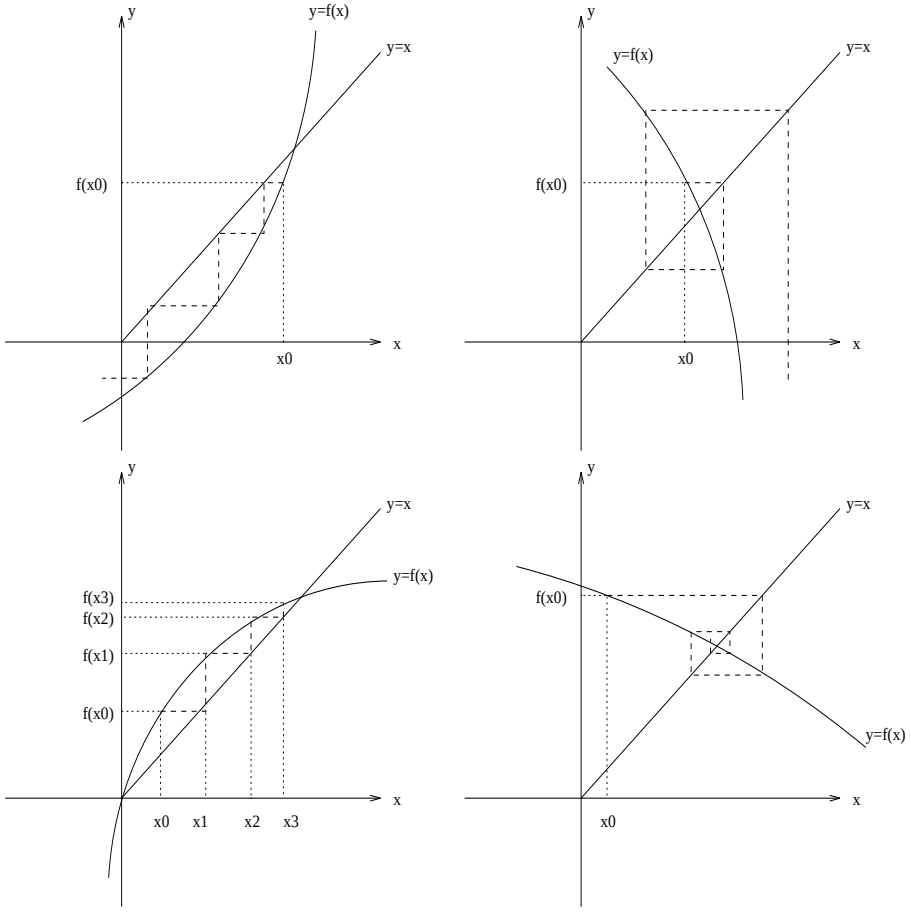


Figure 5.4: two examples of divergent and convergent iterations.

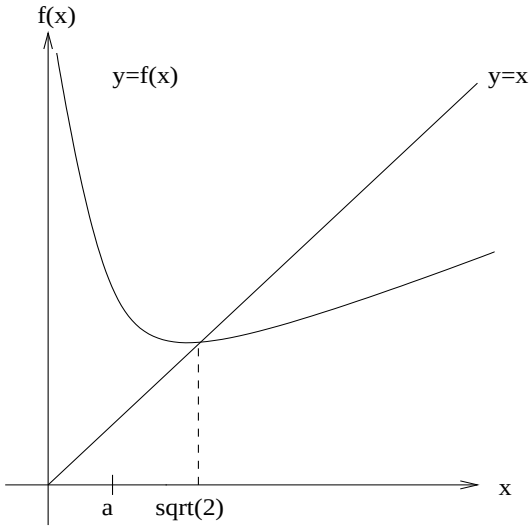


Figure 5.5: .

Example 5.13

$$f(x) = \frac{1}{2}\left(x + \frac{a}{x}\right)$$

$$f'(x) = \frac{1}{2} - \frac{a}{2x^2}$$

$$f'(x) > -1$$

$$\frac{1}{2} - \frac{a}{2x^2} > -1$$
$$\frac{3}{2} > \frac{a}{2x^2}$$
$$x > \sqrt{\frac{a}{3}}$$

a=2:

$$x > \sqrt{\frac{2}{3}} \approx 0.817$$

f is a contraction on $[\sqrt{\frac{a}{3}} + \varepsilon, \infty]$ for $\varepsilon > 0$.

Theorem 5.7 Banach Fixed Point Theorem: Let $f : [a, b] \to [a, b] \subset \mathbb{R}$ be a contraction. Then the following holds

1. f has exactly one fixed point $s \in [a, b]$.
2. For any initial value $x_0 \in [a, b]$ fixed point iteration converges to s.
3. The cutoff error can be estimated by:

$$|s - x_k| \leq \frac{L^{k-l}}{1 - L}|x_{l+1} - x_l| \quad \text{for} \quad 0 \leq l < k$$

For $l = 0$ we get

$$|s - x_k| \leq \frac{L^k}{1 - L}|x_1 - x_0| \quad \text{(a priori estimation)}$$

and for $l = k - 1$:

$$|s - x_k| \leq \frac{L}{1 - L}|x_k - x_{k-1}| \qquad \text{(a posteriori estimation).}$$

Proof:

$$\begin{aligned} |x_{k+1} - x_k| &= |f(x_k) - f(x_{k-1})| \leq L|x_k - x_{k-1}| \\ &= L|f(x_{k-1}) - f(x_{k-2})| \leq L^2|x_{k-1} - x_{k-2}| \\ &= \dots \\ &= L^{k-l}|x_{l+1} - x_l| \qquad \text{for} \qquad 0 \leq l \leq k \end{aligned}$$

for $l = 0$:

$$|x_{k+1} - x_k| \leq L^k|x_1 - x_0|$$

$$\begin{aligned}|x_{k+m} - x_k| &= |x_{k+m} \underbrace{-x_{k+m-1} + x_{k+m-1}}_{=0} \underbrace{- \ldots + \ldots}_{=0} - x_k| = \left| \sum_{i=k}^{k+m-1} x_{i+1} - x_i \right| \\ &\leq \sum_{i=k}^{k+m-1} |x_{i+1} - x_i| \leq L^k (L^{m-1} + L^{m-2} + \ldots + L + 1)|x_1 - x_0| \\ &= L^k \frac{1 - L^m}{1 - L} |x_1 - x_0| \to 0 \qquad \text{für} \qquad k \to \infty\end{aligned}$$

$\Rightarrow (x_k)$ Cauchy Sequence $\Rightarrow (x_k)$ converges

for $s = \lim_{n\to\infty} x_n$ we have $f(s) = f(\lim_{n\to\infty} x_n) = \lim_{n\to\infty} f(x_n) = \lim_{n\to\infty} x_{n+1} = s$.
Thus s is fixed point of f and s is unique, since for s_1, s_2 with $s_1 = f(s_1)$, $s_2 = f(s_2)$ it holds:

$$|s_1 - s_2| = |f(s_1) - f(s_2)| \leq L|s_1 - s_2| \qquad \text{because of} \qquad L < 1 \Rightarrow s_1 = s_2$$

Error estimation see [12] p. 188

Example 5.14

$$f(x) = \frac{1}{2}\left(x + \frac{a}{x}\right) \qquad a = 5, x_0 = 2$$

f contract on $[2, \infty]$ with $L = 0.5$.
Theorem 5.7 (3) with $l = k - 1$:

$$|s - x_k| \leq \frac{L}{1 - L} |x_k - x_{k-1}| \qquad \text{(a posteriori estimation)}$$

$$\Rightarrow |\sqrt{5} - x_k| \leq \frac{0.5}{1 - 0.5} |x_k - x_{k-1}| = |x_k - x_{k-1}|$$

n	x_n	$(x_n - x_{n-1}) \geq (\sqrt{5} - x_n)$
0	2	
1	2.25	0.25
2	2.2361111	0.0139
3	2.2360679779	0.000043
4	2.2360679775	0.00000000042

0.00000000042 (a posteriori)
0.031 (a priori)
Note:

Theorem 5.7 (3) gives estimation of the error **without knowing the limit!**

Example 5.15

$$f(x) = \exp(-x) = x$$
$$f : A \to A, \quad A = [0.5, 0.69]$$

$$\begin{aligned}L &= \max_{x \in A} |f'(x)| = \max_{x \in A} |-e^{-x}| \\ &= e^{-0.5} \approx 0.606531 < 1\end{aligned}$$

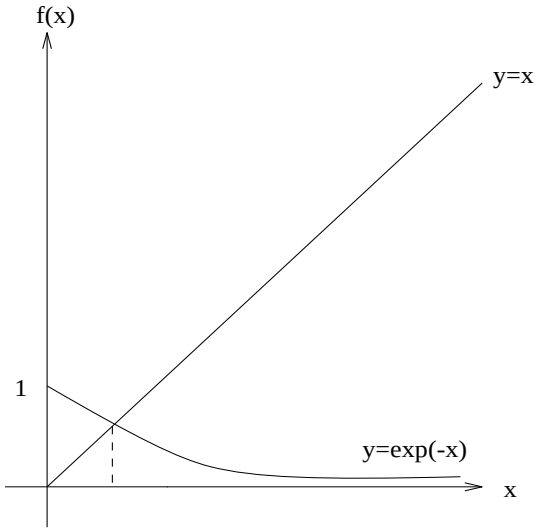


Figure 5.6: .

k	x_k
0	0.55
1	0.577
2	0.562
3	0.570
4	0.565
⋮	⋮
12	0.56712420
⋮	⋮
20	0.56714309
21	0.56714340
22	0.56714323

Theorem 5.7 (3) with $l = 0$:

$$|s - x_k| \leq \frac{L^k}{1-L}|x_1 - x_0| \quad \text{(a priori estimation)}$$

Calculation of k, if $|s - x_k| \leq \varepsilon = 10^{-6}$

$$k \geq \frac{\log\left(\frac{\varepsilon(1-L)}{|x_1-x_0|}\right)}{\log L} \approx 22.3$$

Error after 12 steps:

a priori: $|s - x_{12}| \leq 1.70 \cdot 10^{-4}$
a posteriori: $|s - x_{12}| \leq 8.13 \cdot 10^{-5}$ (better!)

Result:

The iteration in the first example converges much faster than in the second example.

5.3.3 Convergence Speed and Convergence Rate

Definition 5.8 $\varepsilon_k := x_k - s$ is called cutoff error

Fixed Point Theorem (f contract):

$$|\varepsilon_{k+1}| = |x_{k+1} - s| = |f(x_k) - f(s)| \leq L|x_k - s| = L|\varepsilon_k|$$

⇒ Error decreases in each step by factor L!

Theorem 5.8 If $f : [a, b] \to [a, b]$ satisfies the conditions of Theorem 5.7 and is continuously differentiable with $f'(x) \neq 0 \quad \forall x \in [a, b]$, then it holds:

$$\lim_{k\to\infty} \frac{\varepsilon_{k+1}}{\varepsilon_k} = f'(s)$$

Proof: as exercise
Conclusions:
$\varepsilon_{k+1} \approx q\varepsilon_k$ with $q := f'(s)$ (convergence rate)
(x_k) is called **linear convergent** with **convergence rate** $|q|$.
$\Rightarrow$ after m steps error $\varepsilon_{k+m} \approx \frac{1}{10}\varepsilon_k$
$m = ?$

$$\varepsilon_{k+m} \approx q^m \varepsilon_k = 10^{-1}\varepsilon_k$$
$$\Rightarrow m \log_{10} |q| \leq -1 \Rightarrow m \geq \frac{-1}{\log_{10} |q|}$$

$\lvert q\rvert = \lvert f'(s)\rvert$	0.316	0.562	0.75	0.891	0.944	0.972
m	2	4	8	20	40	80

Theorem 5.9 Let f be contracting with $f'(s) = 0$, $\forall x \in [a,b] f''(x) \neq 0$ and f'' continuous on $[a,b]$. Then it holds:

$$\lim_{k\to\infty} \frac{\varepsilon_{k+1}}{\varepsilon_k^2} = \frac{1}{2} f''(s)$$

Conclusion:

$$\text{for} \quad k \to \infty : \varepsilon_{k+1} \approx p\varepsilon_k^2 \quad \text{with} \quad p := \frac{1}{2} f''(s)$$

$\Rightarrow$ quadratic convergence (convergence with order=2)
Correct number of digits is doubled in each step (if $p \approx 1$), because

$$\begin{aligned} \varepsilon_{k+1} = p\varepsilon_k^2 \quad &\Leftrightarrow \quad \log \varepsilon_{k+1} = \log p + 2 \log \varepsilon_k \\ &\Leftrightarrow \quad \frac{\log \varepsilon_{k+1}}{\log \varepsilon_k} = \underbrace{\frac{\log p}{\log \varepsilon_k}}_{\approx 0} + 2 \end{aligned}$$

Example 5.16

$$\varepsilon_{k+1} = 10^{-8}, \varepsilon_k = 10^{-4}$$

Proof of Theorem 5.9:

$$\begin{aligned} \varepsilon_{k+1} &= x_{k+1} - s = f(x_k) - f(s) = f(s + \varepsilon_k) - f(s) \\ &= \underline{f(s)} + \underbrace{\varepsilon_k f'(s)}_{=0} + \frac{1}{2}\varepsilon_k^2 f''(s + \theta_k \varepsilon_k) - \underline{f(s)} \\ &= \frac{1}{2}\varepsilon_k^2 f''(s + \theta_k \varepsilon_k) \quad \text{with} \quad 0 < \theta_k < 1 \end{aligned}$$

because of $f''(x) \neq 0 \quad \forall x \in [a,b]$ and $x_0 \neq s$ it holds:

$$\forall k > 0 : x_k - s = \varepsilon_k \neq 0$$
$$\Rightarrow \frac{\varepsilon_{k+1}}{\varepsilon_k^2} = \frac{1}{2} f''(s + \theta_k \varepsilon_k) \qquad k = 0, 1, 2, \ldots$$
$$\lim_{k\to\infty} \frac{\varepsilon_{k+1}}{\varepsilon_k^2} = \frac{1}{2} \lim_{k\to\infty} f''(s + \theta_k \varepsilon_k) = \frac{1}{2} f''(s + \underbrace{\lim_{k\to\infty} (\theta_k \varepsilon_k)}_{=0}) = \frac{1}{2} f''(s)$$

5.3.4 Newtons method

sought: Solutions of $f(x) = 0$

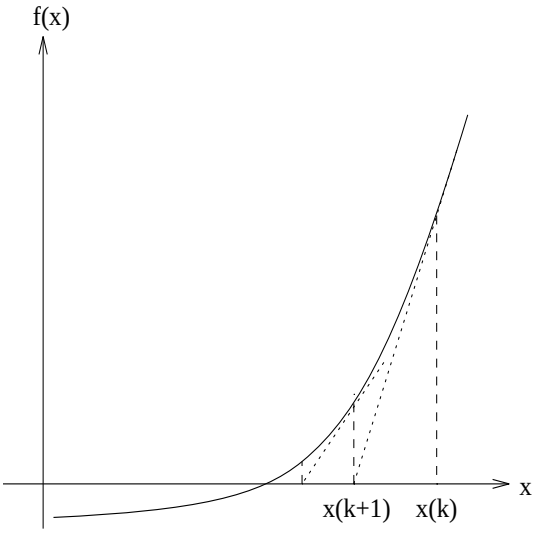


Figure 5.7: .

$$\text{The Tangent:} \quad T(x) = f(x_k) + (x - x_k)f'(x_k)$$
$$T(x_{k+1}) = 0 \Rightarrow f(x_k) + (x_{k+1} - x_k)f'(x_k) = 0$$
$$\Rightarrow (x_{k+1} - x_k)f'(x_k) = -f(x_k)$$

$$\boxed{x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}} \tag{5.2}$$

$$k = 0, 1, 2, \ldots$$

$$\text{with} \quad F(x) := x - \frac{f(x)}{f'(x)} \quad \text{is (5.2) for the fixed point iteration}$$
$$x_{k+1} = F(x_k) \quad \text{with} \quad F(s) = s \quad \text{(fixed point)}$$

Theorem 5.10 Let $f : [a, b] \to \mathbb{R}$ three times continuously differentiable and $\exists s \in [a, b] : f(s) = 0$, as well $\forall x \in [a, b] : f'(x) \neq 0$ and $f''(s) \neq 0$. Then there exists an interval $I = [s - \delta, s + \delta]$ with $\delta > 0$ on which $F : I \to I$ is a contraction. For each x_0,(x_k) is (according to 5.2) quadratically convergent.

Proof:

1. F **is a contraction** in the area of s, i.e. $|F'(x)| < 1$ for$s - \delta \leq x \leq s + \delta$

$$F'(x) = 1 - \frac{f'(x)^2 - f(x)f''(x)}{f'(x)^2} = \frac{f(x)f''(x)}{f'(x)^2} \tag{5.3}$$
$$\Rightarrow F'(s) = \frac{0f''(s)}{f'(s)^2} = 0.$$

Because of the continuity of F', $\delta > 0$ exists with

$$F'(x) \leq L < 1 \qquad \forall x \in [s-\delta, s+\delta] =: I$$

$\Rightarrow F$ is a contraction in I

$$\Rightarrow \lim_{k\to\infty} x_k = s$$

2. **Order of Convergence:**

   Application of Theorem 5.9 on F: $F'(s) = 0$

   from (5.3) we get:

$$F''(x) = \frac{f'(x)^2 f''(x) + f(x)f'(x)f'''(x) - 2f(x)f''(x)^2}{f'(x)^3}$$

$$\Rightarrow F''(s) = \frac{f'(s)^2 f''(s)}{f'(s)^3} = \frac{f''(s)}{f'(s)}$$

   According to Theorem 5.9 ,(x_k) is quadratically convergent on I if and only if $f''(s) \neq 0$. (otherwise even higher order of convergence)

5.4 Exercises

Exercise 5.2 Prove the triangular inequality for real numbers, i.e. that for any two real numbers x and y we have $|x+y| \leq |x| + |y|$.

Exercise 5.3

a) Calculate the p-norm $\|x\|_p$ of the vector $x = (1,2,3,4,5)$ for the values of $p = 1, 2, \ldots, 50$.

b) Draw the unit circles of various p-norms in $\mathbb{R}^2$ and compare them.

c) Prove that the p-norm is a norm for $p = 1, \infty$.

d) Show that for $x \geq 0$ and $0 < p < 1$ the inequality $x^p - px \leq 1 - p$ holds (hint: curve sketching of $x^p - px$).

e) Show by setting $x = a/b$ and $q = 1 - p$ in the above inequality, that for $a, b > 0$ the inequality $a^p b^q \leq pa + qb$ holds.

f) Show using the above result that for $a, b > 0$, $p, q > 1$ and $\frac{1}{p} + \frac{1}{q} = 1$ the inequality $ab \leq \frac{a^p}{p} + \frac{b^q}{q}$ holds.

Exercise 5.4 Prove Lemma 5.2, i.e. that $\|x\|_\infty = \lim_{p\to\infty} \|x\|_p$

Exercise 5.5

a) Write a Mathematica program using `LinearSolve`, which solves a linear system symbolically and apply it to a linear system with up to seven equations.

b) Show empirically that the length of the solution formula grows approximately exponentially with the number of equations.

Exercise 5.6 Show that the addition of the k-fold of row i of a square matrix A to another row j can be expressed as the product $G \cdot A$ with a square matrix G. Determine the matrix

G.

Exercise 5.7 Prove theorem 5.2, i.e. that the Gaussian method for solving linear systems is correct.

Exercise 5.8 Apply elimination to produce the factors L and U for

$$A = \begin{bmatrix} 2 & 1 \\ 8 & 7 \end{bmatrix}, \quad A = \begin{bmatrix} 3 & 1 & 1 \\ 1 & 3 & 1 \\ 1 & 1 & 3 \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 4 & 4 \\ 1 & 4 & 8 \end{bmatrix}$$

Exercise 5.9 Calculate for the matrix

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 0 & 1 \\ 2 & 1 & 1 \end{pmatrix}$$

the matrices L and U of the LU decomposition. Then determine the solutions of $Ax = b$ for the right sides $(1, 1, 1)^T$ and $(3, 1, 0)^T$.

Exercise 5.10 If $A = L_1 D_1 U_1$ and $A = L_2 D_2 U_2$, prove that $L_1 = L_2$, $D_1 = D_2$ and $U_1 = U_2$. If A is invertible, the factorization is unique.

a) Derive the equation $L_1^{-1} L_2 D_2 = D_1 U_1 U_2^{-1}$ and explain why one side is lower triangular and the other side is upper triangular.

b) Compare the main diagonals in that equation, and then compare the off-diagonals.

Exercise 5.11 For the calculation of $\sqrt{a}$, the iteration of $x_{n+1} = a/x_n$ with $a > 0$, $x_0 > 0$ can be tried.

a) Visualize the iteration sequence.

b) Explain on the basis of drawing why the sequence does not converge.

c) Prove that this sequence does not converge.

d) How to change the iteration formula $x_{n+1} = a/x_n$, so that the sequence converges?

Exercise 5.12

a) What means *convergence of a sequence* $(x_n)_{n\in\mathbb{N}}$? (Definition!)

b) Give a convergent, divergent, alternating convergent and alternating divergent sequence.

c) Give at least one simple convergence criterion for sequences.

Exercise 5.13 Apply the interval bisection method to the function

$$f(x) = \frac{x(1-x)}{1-x^2}$$

with the initial interval $[-4, -1/2]$. Calculate the limit of the sequence with at least 4 digits. Give reasons for the surprising result.

Exercise 5.14 Sought are the solutions of the equation

$$\tan x = \cos x \tag{5.4}$$

in the interval $[0, \pi/2]$.

a) Show that the equation (5.4) in $[0, \pi/2]$ has exactly one solution.

b) In the following, the equation (5.4) is to be solved by fixed pointiteration. Therefore use the form:

$$x = f(x) := \arctan(\cos x) \tag{5.5}$$

Give the smallest possible Lipschitz bound for f and a corresponding sub-interval of $[0, \pi/2]$.

c) Determine an a priori estimation for the number of iterations for a precision of at least 10^{-3}.

d) Calculate the iteration sequence (x_n) of the fixed-point iteration with the initial v alue $x_0 = \pi/4$ to $n = 10$.

e) Determine an interval in which the root is for sure using the a posteriori estimation after 8 steps.

f) Why is the transformation of the equation (5.4) to $x = \arccos(\tan x)$ less favorable than those used above?

g) Write a simple as possible Mathematica program (3-4 commands!), which calculates the iteration sequence and stores it in a table.

Exercise 5.15 Prove theorem 5.8, i.e. if $f : [a, b] \to [a, b]$ is a contraction and is continuously differentiable with $f'(x) \neq 0 \ \ \forall x \in [a, b]$, then it holds:

$$\lim_{k \to \infty} \frac{\varepsilon_{k+1}}{\varepsilon_k} = f'(s)$$

Exercise 5.16

a) Prove that any contracting function $f : [a, b] \to [a, b] \in \mathbb{R}$ is continuous.

b) Prove that not all contracting functions $f : [a, b] \to [a, b] \in \mathbb{R}$ are differentiable.

c) Prove that any differentiable function $f : D \to \mathbb{R}$, ($D \subset \mathbb{R}$ open) is continuous.

Chapter 6

Function Approximation

6.1 Polynomial Interpolation

Example 6.1 Linear interpolation (see figure Figure 6.1)

When there were no calculators, using logarithms for practical purposes was done with tables of logarithms. Only integers were mapped, intermediate values were determined by linear interpolation.

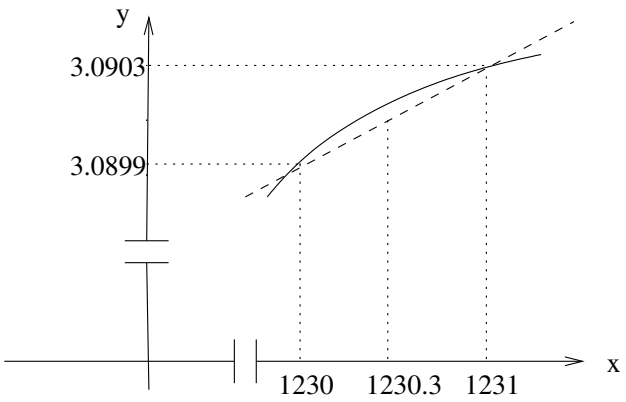


Figure 6.1: Determination of $lg(1230.3)$ using linear interpolation.

$$\begin{aligned} \lg(1230) &= 3.0899 \\ \lg(1231) &= 3.0903 \\ \lg(1230.3) &= ? \\ \lg(1230.3) &\approx 3.0899 + 4 \cdot 0.0001 \cdot 0.3 = 3.09002 \end{aligned}$$

6.1.1 Motivation

- Higher order interpolation (quadratic,...)
- Tools for numerical methods (functional approximation, numerical differentiation, integration ,...)

6.1.2 The Power Series Approach

Given: Table (x_k, y_k) for $(k = 1, \ldots, n)$

Sought: Polynomial p with $p(x_i) = y_i$ for $(i = 1, \ldots, n)$

Ansatz: $p(x) = a_1 + a_2 x + \cdots + a_n x^{n-1}$

$$\Rightarrow a_1 + a_2 x_i + a_3 x_i^2 + \cdots + a_n x_i^{n-1} = y_i \quad \text{for} \quad (i = 1, \ldots, n)$$

$$\Rightarrow A \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \text{with} \quad A = \underbrace{\begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & & & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{pmatrix}}_{\text{Vandermonde matrix}}$$

Theorem 6.1 If $x_1, \ldots, x_n$ are distinct, then for any $y_1, \ldots, y_n$ there is a unique polynomial p of degree $\leq n-1$ with $p(x_i) = y_i$ for $(i = 1, \ldots, n)$.

Proof:
To show that equation $A\boldsymbol{a} = \boldsymbol{y}$ is uniquely solvable, we show that the nullspace of A is $\boldsymbol{0}$, i.e. $A\boldsymbol{a} = \boldsymbol{0} \Rightarrow \boldsymbol{a} = \boldsymbol{0}$:

$$\begin{aligned} A\boldsymbol{a} = \boldsymbol{0} &\Rightarrow \forall i = 1, \ldots, n : p(x_i) = 0 \\ &\Rightarrow p(\boldsymbol{x}) \equiv 0 \quad \text{(zero polynomial)} \\ &\Rightarrow \boldsymbol{a} = \boldsymbol{0} \end{aligned}$$

Example 6.2 Interpolation of $\sin(x)$
Table of values in $\{-m, -m+1, \ldots, 0, 1, 2, \ldots, m\}$

$$\begin{aligned} \sin(0.5) &= 0.479426 \\ p(0.5) &= 0.479422 \quad \text{(m=3, i.e. n=7 points)} \\ p(0.5) &= 0.469088 \quad \text{(m=2, i.e. n=5 points)} \end{aligned}$$

$\sin(x)$ is well approximated by the interpolating polynomial, even at relatively small number of given points (n=5,7), as can be seen in Figure 6.2, Figure 6.3 and Figure 6.4.

Example 6.3 Interpolation of $f(x)$ in the interval [-1,1]:

$$f(x) = \frac{1}{1 + 25x^2}$$

Figure 6.5 clearly shows the poor approximation particulary in the margin areas. Idea: more given points in the margin areas
Improvement: Chebyshev interpolation

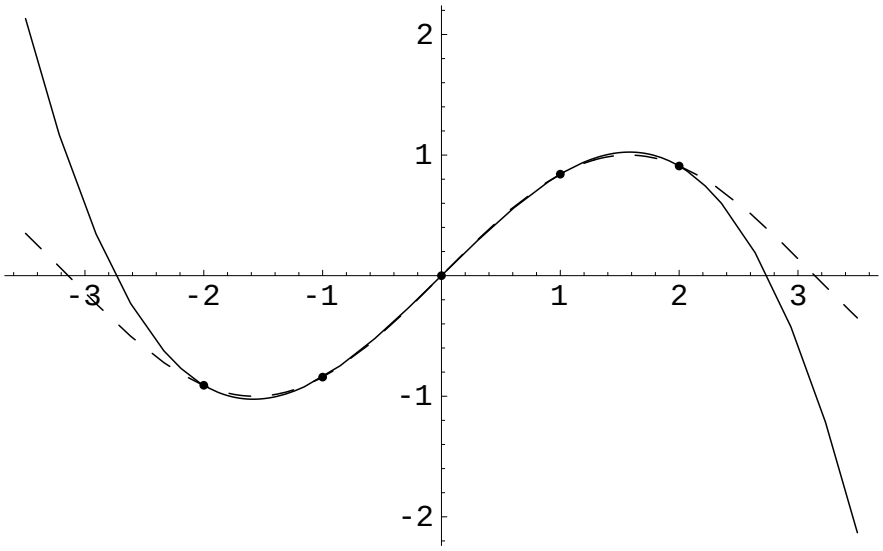


Figure 6.2: Interpolation of sin(x) with $n = 5$ given points.

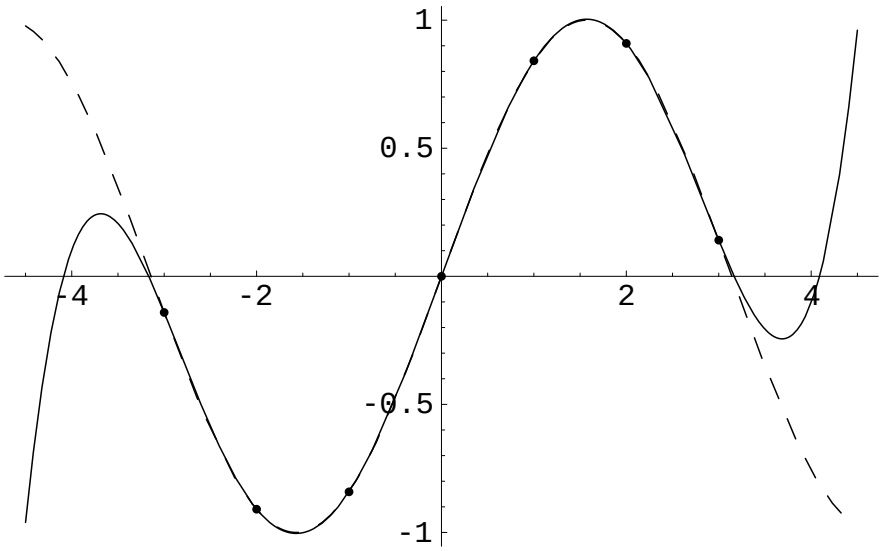


Figure 6.3: Interpolation of sin(x) with $n = 7$ given points.

Definition 6.1 For any $f : [a, b] \to \mathbb{R}$ we define $\|f\|_\infty := \max_{x\in[a,b]} |f(x)|$

Theorem 6.2 Let $f : [a, b] \to \mathbb{R}$ be n-times continuously differentiable. Let $a = x_1 < x_2 < \ldots < x_{n-1} < x_{n+1} = b$ and p the interpolating polynomial of degree n with $p(x_i) = f(x_i)$ for $(i = 1, \ldots, n)$. Then

$$f(x) - p(x) = \frac{f^{(n+1)}(z)}{(n+1)!}(x - x_1)(x - x_2)\cdots(x - x_{n+1})$$

for a point $z \in [a, b]$.

Note:

- remainder term is the same as in Taylor's theorem for $x_1 = x_2 = \cdots = x_{n+1}$

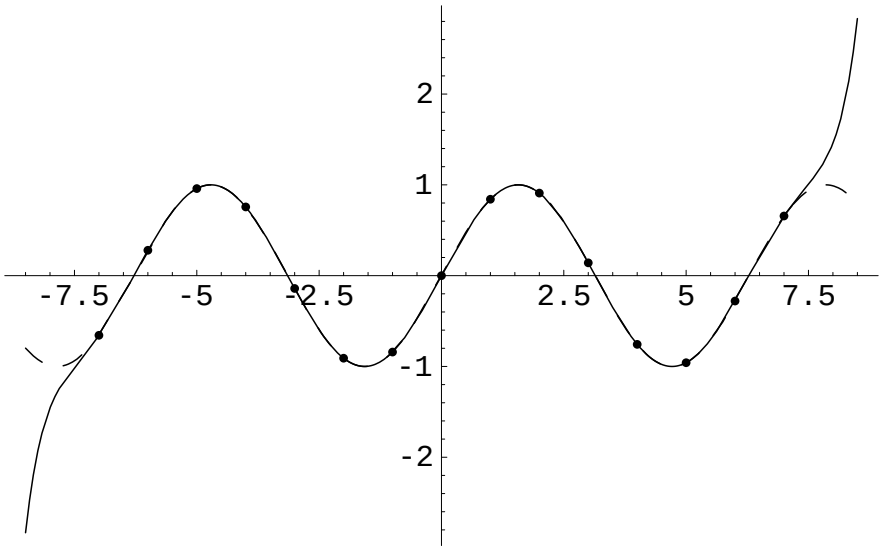


Figure 6.4: Interpolation of sin(x) with $n = 15$ given points.

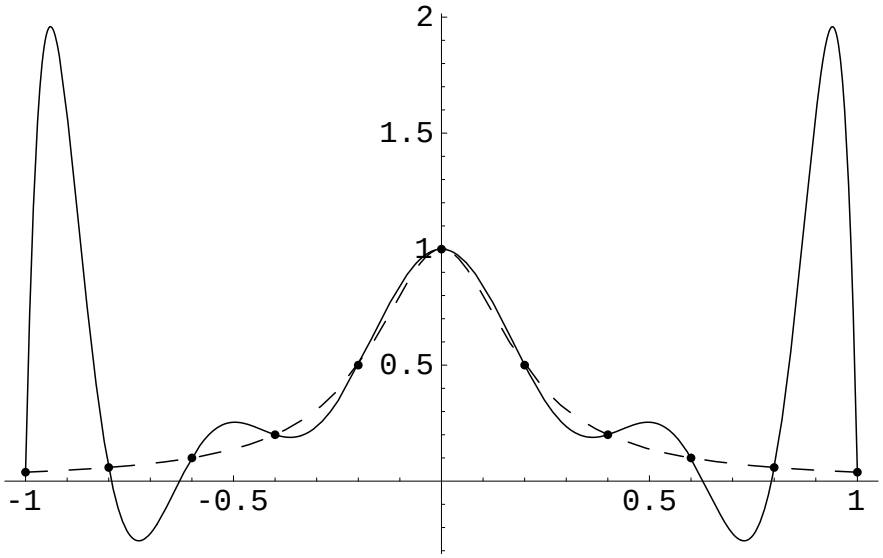


Figure 6.5: Interpolation with 11 given points.

- right hand side equals zero for $x = x_i$ (i.e. in all given points)

Question: How should the given points $x_1, \ldots, x_{n+1}$ be distributed, to minimize (for constant n) the maximum error?

Answer: Chebyshev interpolation

Theorem 6.3 Let $f : [-1, 1] \to \mathbb{R}$ and p the interpolating polynomial at the given points $-1 \leq x_1 < \cdots < x_n \leq 1$.
The approximation error $\|f - p\|_\infty = \max_{x \in [-1,1]} |f(x) - p(x)|$ is minimal for

$$x_k = -\cos\left(\frac{2k-1}{n} \cdot \frac{\pi}{2}\right) \quad (k = 1, \ldots, n)$$

The values x_k are called **Chebyshev abscissas**.

Example 6.4 Let n=6. the Chebyshev abscissas are (see also Figure 6.6).

k	1	2	3	4	5	6
2k-1	1	3	5	7	9	11
-cos $\left(\frac{\pi}{12}(2k-1)\right)$	-0.966	-0.707	-0.259	0.259	0.707	0.966

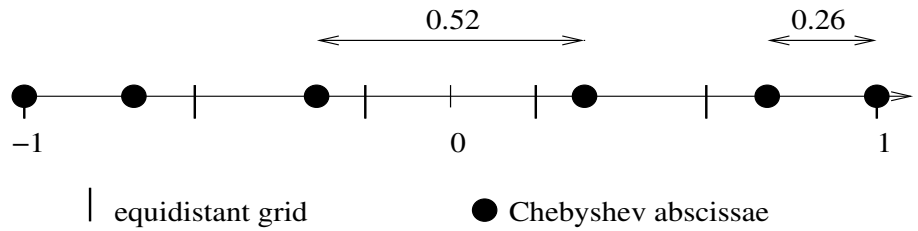


Figure 6.6: Distribution of the given points.

Example 6.5 Figure 6.7 shows a significant reduction in the maximum norm of the error when Chebyshev interpolation is applied.

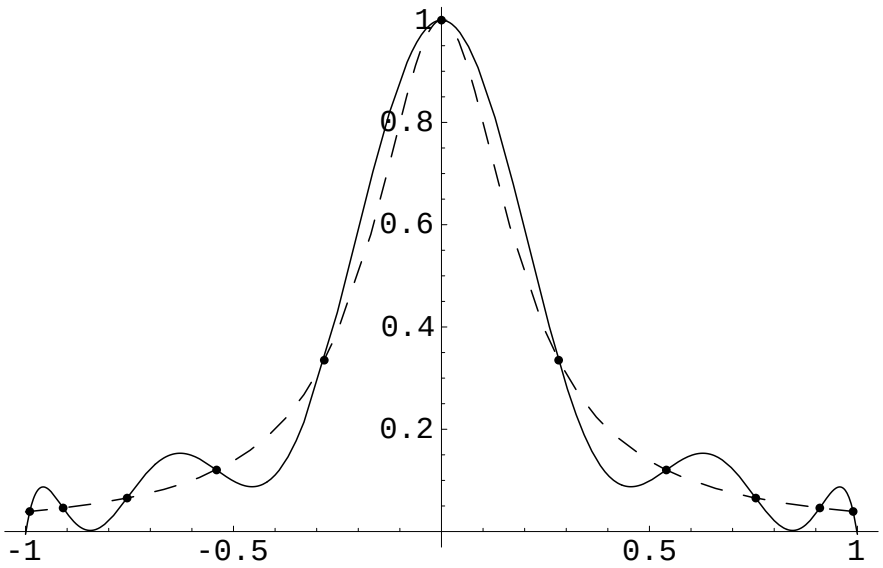


Figure 6.7: Chebyshev interpolation with 11 given points.

Corollar 6.1.1 Theorem 6.3 can be applied easily to functions $f : [a, b] \to \mathbb{R}$, by calculating the given points t_k for $k = 1, \ldots, n$ out of the Chebyshev abscissas x_k by

$$t_k = \frac{1}{2}(a + b) + \frac{1}{2}(b - a)x_k$$

.

Additional notes:

1. Are polynomials suitable for approximating a given function f?
   Polynomials are not suitable for functions alternating between strong and weak curvature or poles.
   Possibly: piecewise approximation by polynomials ($\Rightarrow$ spline approximation) or approximation by rational functions.

2. Is a polynomial well defined by the value table's data?
   equidistant given points → Chebyshev abscissas or choose smaller degree of the polynomial ⇒ overdetermined system of linear equations (degree(p) $\leq 2 \cdot \sqrt{n}$ in which n=Number of given points).

6.1.3 The Horner scheme

By using the following scheme, computing time will be saved in the evaluation of polynomials:

$$\begin{aligned} p(x) &= \sum_{k=1}^{n} a_k x^{k-1} = a_1 + a_2 x + \ldots + a_n x^{n-1} \\ &= a_1 + x(a_2 + x(a_3 + x(\ldots + x(a_{n-1} + x a_n)\ldots))) \end{aligned}$$

Iteration:

$$\begin{aligned} y_0 &:= a_n \\ y_k &:= y_{k-1} x + a_{n-k} \qquad k = 1, \ldots, n-1 \end{aligned}$$

$$\Rightarrow p(x) = y_{n-1}$$

Computing time:
(n-1) Additions + Multiplications
naive evaluation: ($x^k = \underbrace{x \cdot x \cdot \ldots \cdot x \cdot x}_{\text{k-times}}$)
(n-1) additions, (n-2)-times potentiate, (n-1) multiplications

$$\sum_{k=0}^{n-1} k = \frac{n(n-1)}{2} = \frac{1}{2}(n^2 - n) \quad \text{multiplications}$$

6.1.4 Function Approximation vs. Interpolation

In **interpolation** n points (x_k, y_k) with $(k = 1, \ldots, n)$ are given and a function p (e.g., a polynomial of degree n-1) is sought with $p(x_k) = y_k$ for $(k = 1, \ldots, n)$.

In the **approximation** of functions, a function $f : [a, b] \rightarrow \mathbb{R}$ is given (symbolically by a formula or a value table with possibly noisy values) and the task is to find the "'simplest"' possible function p, which approximates f as good as possible with respect to a norm (e.g. maximum norm). The function p can be a polynomial but also a linear combination of basis functions such as $p(x) = a_1 \sin x + a_2 \sin 2x + a_3 \sin 3x + \cdots + a_n \sin nx$ where $a_1, \ldots, a_n$ are to be determinated). **Interpolation** can be used as a tool for function **approximation**.

6.2 Spline interpolation

6.2.1 Interpolation of Functions

Given: Value table (x_k, y_k) with $k = 0, 1, \ldots, n$

Sought: Interpolating (function) $s(x)$ with $s(x_k) = y_k$, and $s(x)$ must be two times continuously differentiable.

Ansatz: piecewise cubic polynomials

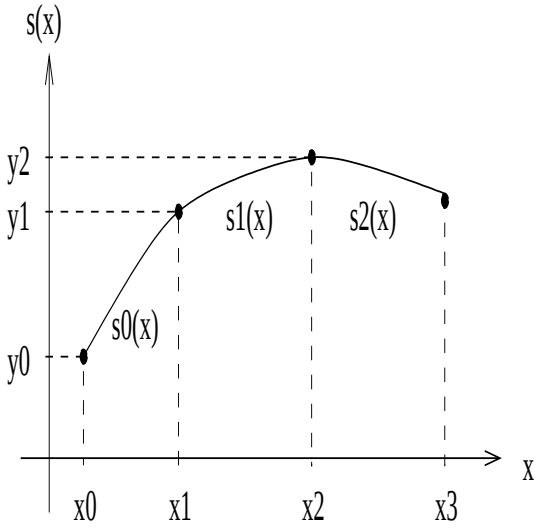


Figure 6.8: natural cubic spline through 4 points.

The property of $s(x)$ to be two times continuously differentiable implies:

$s'(x)$ continuous , $s''(x)$ continuous at all inner interval limits.

$\Rightarrow$ 2 additional conditions for each cubic polynomial

$\Rightarrow$ the n subpolynomials uniquely determined by 2 points + 2 derivation conditions.

$$\begin{aligned}
\textbf{ansatz:} \quad & \text{for (i=0,...,n-1) let} \\
s(x) &= s_i(x) = a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i && (6.1) \\
\textbf{requirements:} & \\
s_i(x_i) &= y_i \qquad \text{i=0,...,n-1} && (6.2) \\
s_{n-1}(x_n) &= y_n && (6.3) \\
s_i(x_{i+1}) &= s_{i+1}(x_{i+1}) \qquad \text{i=0,...,n-2} && (6.4) \\
s_i'(x_{i+1}) &= s_{i+1}'(x_{i+1}) \qquad \text{i=0,...,n-2} && (6.5) \\
s_i''(x_{i+1}) &= s_{i+1}''(x_{i+1}) \qquad \text{i=0,...,n-2} && (6.6)
\end{aligned}$$

$$\Rightarrow n + 1 + 3(n-1) \;=\; 4n - 2 \text{ linear equations for } 4n \text{ unknowns}$$

$\Rightarrow$ 2 conditions are missing

Additional condition (natural spline):

$$s''(x_0) = 0, \quad s''(x_n) = 0 \tag{6.7}$$

substitution:

$$h_i = x_{i+1} - x_i \tag{6.8}$$

$$\begin{aligned}
(6.1),(6.2) &\Rightarrow s_i(x_i) = d_i = y_i && (6.9)\\
(6.1),(6.2),(6.4) &\Rightarrow s_i(x_{i+1}) = a_i h_i^3 + b_i h_i^2 + c_i h_i + d_i = y_{i+1} && (6.10)\\
(6.1) &\Rightarrow s_i'(x_i) = c_i && (6.11)\\
(6.1) &\Rightarrow s_i'(x_{i+1}) = 3a_i h_i^2 + 2b_i h_i + c_i && (6.12)\\
(6.1) &\Rightarrow s_i''(x_i) = 2b_i =: y_i'' && (6.13)\\
(6.1) &\Rightarrow s_i''(x_{i+1}) = 6a_i h_i + 2b_i = s_{i+1}''(x_{i+1}) = y_{i+1}'' && (6.14)
\end{aligned}$$

$$\begin{aligned}
(6.13),(6.14) \Rightarrow a_i &= \frac{1}{6h_i}(y_{i+1}'' - y_i'')\\
(6.13) \Rightarrow b_i &= \frac{1}{2}y_i''\\
(6.9),(6.10),(6.13),(6.14) \Rightarrow c_i &= \frac{1}{h_i}(y_{i+1} - y_i) - \frac{h_i}{6}(y_{i+1}'' + 2y_i'')\\
(6.9) \Rightarrow d_i &= y_i
\end{aligned} \tag{6.16}$$

if y_i'' are known, then also a_i, b_i, c_i, d_i are known.

(6.16) in (6.12):

$$s_i'(x_{i+1}) = \frac{1}{h_i}(y_{i+1} - y_i) + \frac{h_i}{6}(2y_{i+1}'' + y_i'')$$

$$i \to i-1: \quad s_{i-1}'(x_i) = \frac{1}{h_{i-1}}(y_i - y_{i-1}) + \frac{h_{i-1}}{6}(2y_i'' + y_{i-1}'') \tag{6.17}$$

$$\begin{aligned}
\text{because of} \quad s_{i-1}'(x_i) &= s_i'(x_i) \qquad (\textit{Requirement } (6.5))\\
\text{and} \quad s_i'(x_i) &= c_i = \frac{1}{h_i}(y_{i+1} - y_i) - \frac{h_i}{6}(y_{i+1}'' + 2y_i'')
\end{aligned}$$

follows

$$\frac{1}{h_{i-1}}(y_i - y_{i-1}) + \frac{h_{i-1}}{6}(2y_i'' + y_{i-1}'') = \frac{1}{h_i}(y_{i+1} - y_i) - \frac{h_i}{6}(y_{i+1}'' + 2y_i'')$$

Sorting of the y''-variables to the left results in

$$\Rightarrow h_{i-1}y_{i-1}'' + 2(h_{i-1} + h_i)y_i'' + h_i y_{i+1}'' = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}) \tag{6.19}$$

for $i = 1, 2, \ldots, n-1$.

linear system for $y_1'', y_2'', \ldots, y_{n-1}''$
y_0'', y_n'' arbitrarily chooseable!
$y_0'' = y_n'' = 0$: natural spline

Example 6.6 $n = 5$

$$\begin{pmatrix} 2(h_0+h_1) & h_1 & 0 & 0 \\ h_1 & 2(h_1+h_2) & h_2 & 0 \\ 0 & h_2 & 2(h_2+h_3) & h_3 \\ 0 & 0 & h_3 & 2(h_3+h_4) \end{pmatrix} \cdot \begin{pmatrix} y_1'' \\ y_2'' \\ y_3'' \\ y_4'' \end{pmatrix} = \boldsymbol{r}$$

$$\text{with} \quad r_i = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1})$$

coefficient matrix is tridiagonal

Example 6.7 We determine a natural spline interpolant through the points $(0,0), (1,1), (2,0), (3,1)$. It holds $n = 3$ and $h_0 = h_1 = 1$. The coefficient matrix reads

$$\begin{pmatrix} 2(h_0+h_1) & h_1 \\ h_1 & 2(h_1+h_2) \end{pmatrix} = \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix}$$

with the right hand side

$$\begin{aligned} r_1 &= 6(y_2 - y_1) - 6(y_1 - y_0) = -12 \\ r_2 &= 6(y_3 - y_2) - 6(y_2 - y_1) = 12 \end{aligned}$$

yielding

$$\begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix} \begin{pmatrix} y_1'' \\ y_2'' \end{pmatrix} = \begin{pmatrix} -12 \\ 12 \end{pmatrix}$$

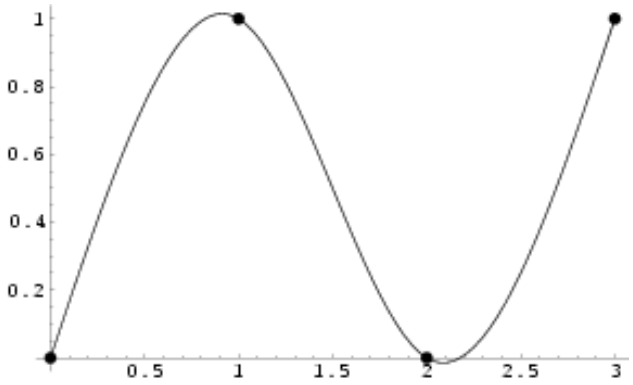
with the solution

$$\mathbf{y_1'' = -4}, \quad \mathbf{y_2'' = 4}, \quad \mathbf{y_0'' = y_3'' = 0}$$

Inserting in (6.16) gives

$$\begin{aligned} s_0(x) &= -2/3\,x^3 + 5/3\,x \\ s_1(x) &= 4/3\,x^3 - 6\,x^2 + 23/3\,x - 2 \\ s_2(x) &= -2/3\,x^3 + 6\,x^2 - 49/3\,x + 14. \end{aligned}$$

with the graph



6.2.2 Correctness and Complexity

Definition 6.2 A $n \times n$ matrix A is called **diagonally dominant**, if

$$|a_{ii}| > \sum_{\substack{k=1 \\ k \neq i}}^{n} |a_{ik}|$$

for $i = 1, 2, \ldots, n$

Theorem 6.4 A linear system $A \cdot x = b$ is uniquely solvable, if A is diagonally dominant. In the Gaussian Elimination neither row nor column swapping is needed.

Theorem 6.5 The computation time for the Gaussian elimination method for a tridiagonal matrix A is linear in the lenght n of A.

Proof: (see Exercises)

Theorem 6.6 Spline-Interpolation: Let $x_0 < x_1 < \ldots < x_n$. There is a unique cubic spline interpolant $s(x)$ with $y_0'' = y_n'' = 0$ (natural Spline). It can be calculated in linear time (O(n)) by the method described above (by using the tridiagonal matrix algorithm, see exercise).

The Tridiagonal Algorithm

$$\begin{pmatrix} b_1 & c_1 & 0 & \cdots & 0 \\ c_1 & \ddots & \ddots & 0 & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & 0 & \ddots & \ddots & c_{n-1} \\ 0 & \cdots & 0 & c_{n-1} & b_n \end{pmatrix} \cdot x = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}$$

Elimination:

$$\left.\begin{array}{rcl} m & := & c_{k-1}/b_{k-1} \\ b_k & := & b_k - m \cdot c_{k-1} \\ d_k & := & d_k - m \cdot d_{k-1} \end{array}\right\} k = 2, \ldots, n$$

Backward substitution:

$$\left.\begin{array}{rcl} d_n & := & d_n/b_n \\ d_k & := & (d_k - c_k d_{k+1})/b_k \\ x_k & = & d_k \end{array}\right\} k = n-1, \ldots, 1$$

Proof:

1. Existence and uniqueness
   Let $x_0 < x_1 < \ldots < x_n \Rightarrow h_i = x_{i+1} - x_i > 0$
   $\Rightarrow 2(h_{i-1} + h_i) > h_{i-1} + h_i$
   $\Rightarrow$ matrix diagonally dominant and uniquely solvable
   $\Rightarrow a_i, b_i, c_i, d_i$ uniquely determined
   $\Rightarrow$ spline interpolant uniquely determined
2. Computation time (see Exercises)

Other conditions:

1. $y_0'' = y_n'' = 0$ (natural spline)
2. $y_0'' = s''(x_0), y_n'' = s''(x_n)$ (s'' given)
3. $y_0'' = y_1'', y_n'' = y_{n-1}''$ (s'' constant on the border)
4. s' given at the border (best choice if $s'(x_0), s'(x_n)$ is known)
5. if $y_0 = y_n : y_0' = y_n', y_0'' = y_n''$ (periodic condition)

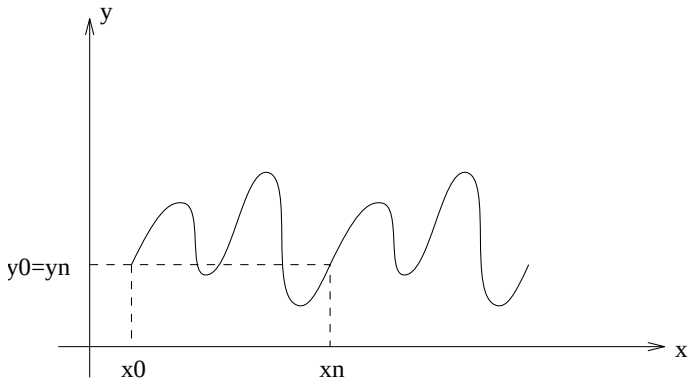


Figure 6.9: periodic condition at spline interpolation.

6.2.3 Interpolation of arbitrary curves

Example 6.8 Airfoil:

given: value table

k	x_k	y_k
1	x_1	y_1
2	x_2	y_2
⋮	⋮	⋮
n	x_n	y_n

The curve is **not a function**, therefore, naive interpolation is not applicable.
$\Rightarrow$ Parameter representation (parameter t)

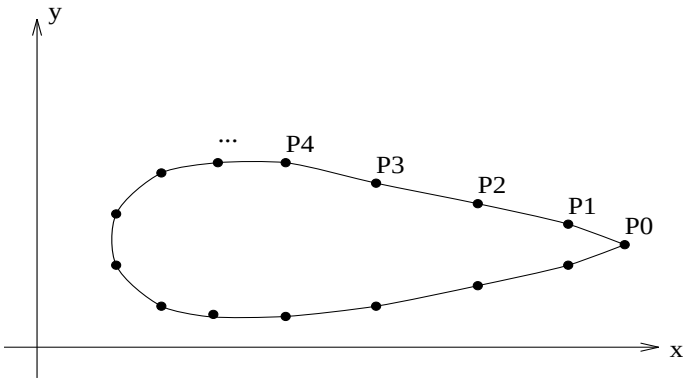
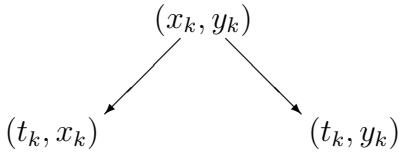


Figure 6.10: parametric plot of the given value pairs.



$(t_k, x_k), (t_k, y_k)$ unique, if (t_k) for $k = 1, \ldots, n$ monotonically increasing!

Simplest choice of t_k: $t_k = k$

k	t_k	x_k	t_k	y_k
0	0	x_0	0	y_0
1	1	x_1	1	y_1
2	2	x_2	2	y_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	n	x_n	n	y_n

ideal choice of t_k: arc length
good choice of t_k:

$$\begin{aligned} t_0 &= 0, \\ t_k &= t_{k-1} + ||P_k - P_{k-1}|| \\ &= t_{k-1} + \sqrt{(x_k - x_{k-1})^2 + (y_k - y_{k-1})^2} \qquad k = 1, 2, \ldots, n \end{aligned}$$

Calculation of the spline curve

1. Computation of the spline function for $(t_k, x_k) \Rightarrow x(t)$
2. Computation of the spline function for $(t_k, y_k) \Rightarrow y(t)$
3. spline curve defined by:
   $x = x(t)$
   $y = y(t)$
   for $0 \leq t \leq t_n$

6.3 Method of Least Squares and Pseudoinverse

6.3.1 Minimization according to Gauss

Given: n measurements, i.e. value pairs $(x_1, y_1), \ldots, (x_n, y_n)$
function $f(x, a_1, \ldots, a_k) = f(x) \quad k \leq n$

Sought: Values for $a_1, \ldots, a_k$ such, that

$$E(f(x_1) - y_1, \ldots, f(x_n) - y_n) = \sum_{i=1}^{n} (f(x_i) - y_i)^2$$

gets minimal!

Simplification: f is a linear combination of functions

$$f(x, a_1, \ldots, a_k) = a_1 f_1(x) + a_2 f_2(x) + \cdots + a_k f_k(x) \tag{6.20}$$

$$\text{E extremal} \Rightarrow \forall j = 1, \ldots, k : \frac{\partial E}{\partial a_j} = 0$$

$$E(\ldots) = \sum_{i=1}^{n} (a_1 f_1(x_i) + \cdots + a_k f_k(x_i) - y_i)^2$$

$$\frac{\partial E}{\partial a_j} = 2 \sum_{i=1}^{n} \left(\sum_{l=1}^{k} a_l f_l(x_i) - y_i \right) f_j(x_i)$$

$$\frac{\partial E}{\partial a_j} = 0 \Rightarrow \sum_{i=1}^{n} \sum_{l=1}^{k} a_l f_l(x_i) f_j(x_i) = \sum_{i=1}^{n} y_i f_j(x_i)$$

$$\Leftrightarrow \sum_{l=1}^{k} a_l \underbrace{\sum_{i=1}^{n} f_l(x_i) f_j(x_i)}_{A_{jl}} = \underbrace{\sum_{i=1}^{n} y_i f_j(x_i)}_{b_j}$$

$$\Leftrightarrow \sum_{l=1}^{k} A_{jl} a_l = b_j \quad \text{for} \quad (j = 1, \ldots, k) \tag{6.21}$$

linear system of equations for the parameters $a_1, \ldots a_k$ (**Normal equations!**)
Solving of the normal equations gives $a_1, \ldots, a_k$.
Note: normal equations are usually (not always) uniquely solvable (see Theorem 6.7).

Example 6.9 With the method of least squares the coefficients a_1, a_2, a_3 of the function $f(x) = a_1 x^2 + a_2 x + a_3$ using the given points $(0, -1), (2, 0), (3, 2), (4, 1)$ are to be determined. First, we set up the normal equations:

$$\sum_{l=1}^{k} A_{jl} a_l = b_j \quad \text{for} \quad (j = 1, \ldots, k)$$

with

$$A_{jl} = \sum_{i=1}^{n} f_l(x_i) f_j(x_i), \qquad b_j = \sum_{i=1}^{n} y_i f_j(x_i).$$

It follows:

$$A = \begin{pmatrix} \sum_{i=1}^n x_i^4 & \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^2 \\ \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i & \sum_{i=1}^n 1 \end{pmatrix} = \begin{pmatrix} 353 & 99 & 29 \\ 99 & 29 & 9 \\ 29 & 9 & 4 \end{pmatrix}$$

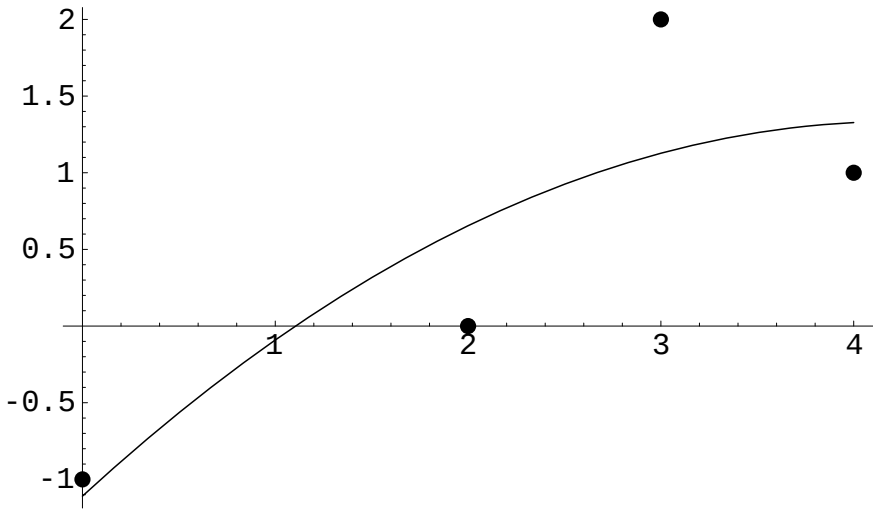
and

$$b = \begin{pmatrix} \sum_{i=1}^n y_i x_i^2 \\ \sum_{i=1}^n y_i x_i \\ \sum_{i=1}^n y_i \end{pmatrix} = \begin{pmatrix} 34 \\ 10 \\ 2 \end{pmatrix}$$

The solution of this linear system is $a_1 = -3/22$, $a_2 = 127/110$, $a_3 = -61/55$, because

$$\begin{pmatrix} 353 & 99 & 29 \\ 99 & 29 & 9 \\ 29 & 9 & 4 \end{pmatrix} \begin{pmatrix} -\frac{3}{22} \\ \frac{127}{110} \\ -\frac{61}{55} \end{pmatrix} = \begin{pmatrix} 34 \\ 10 \\ 2 \end{pmatrix}$$

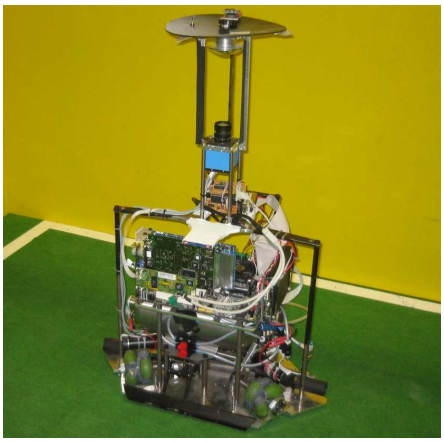
The resulting parabola has the following form:



6.3.2 Application: rectification of photos

In RoboCup, so-called "OmniCams" are used. These are digital cameras that take a 360-degree picture via a parabolic mirror (see fig. 6.11).
The mirror distorts the image considerably. With the Formula of mirror curvature a formula for conversion of pixel coordinates into real distances on the field can be derived. Because this formula critically depends on adjustments of the camera, the mirror, the image can not be rectified completely. Therefore, to determine the transformation of pixel distances into real distances we approximate an polynomial interpolation. White markings are pasted on the field at a distance of 25cm (fig. 6.12) and the pixels distances to the center are measured. This gives the following value table:

dist. d [mm]	0	250	500	750	1000	1250	1500	1750	2000	2250	2500	2750	3000	3250	3500	3750	4000	4250
pixel dist. x	0	50	108	149	182	209	231	248	263	276	287	297	305	313	319	325	330	334



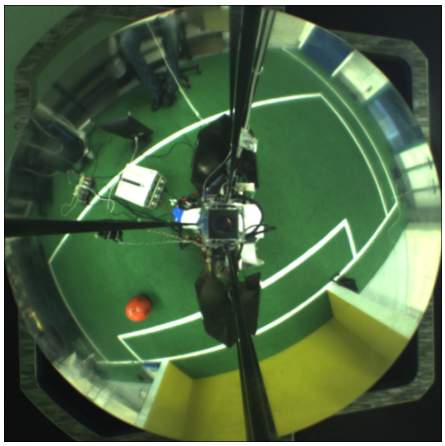
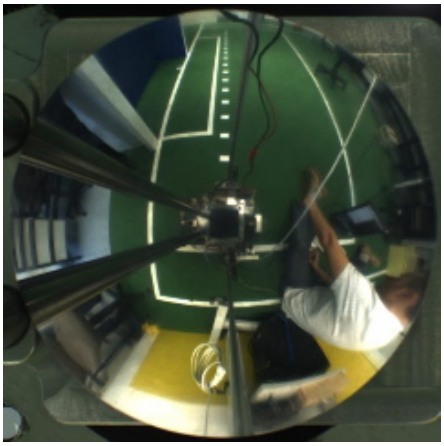


Figure 6.11: The RoboCup robot Kunibert with upward-pointing camera and mirror (left) and a distorted picture of the field.



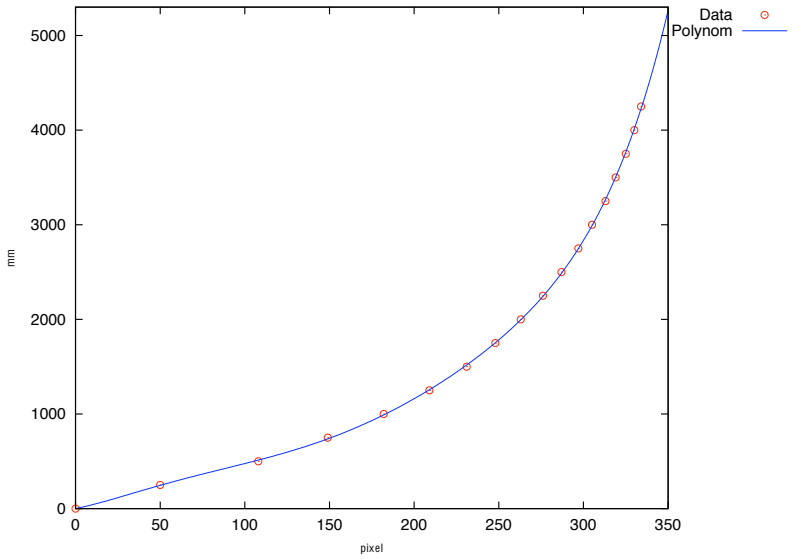


Figure 6.12: The markers for the interpolation on the field are shown in the left and the graph of the interpolating polynomial d(x) is in the right diagram.

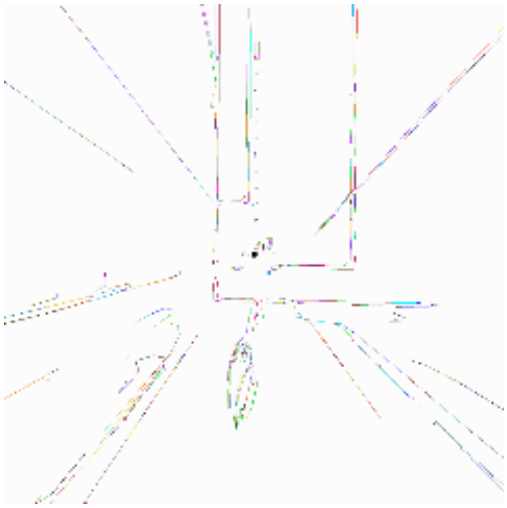


Figure 6.13: Modified image after the edge detection (left) and the rectified image after application of the transformation (right).

Now a polynomial of degree 6 (calculated with the method of least squares) is fitted to the points. We get:

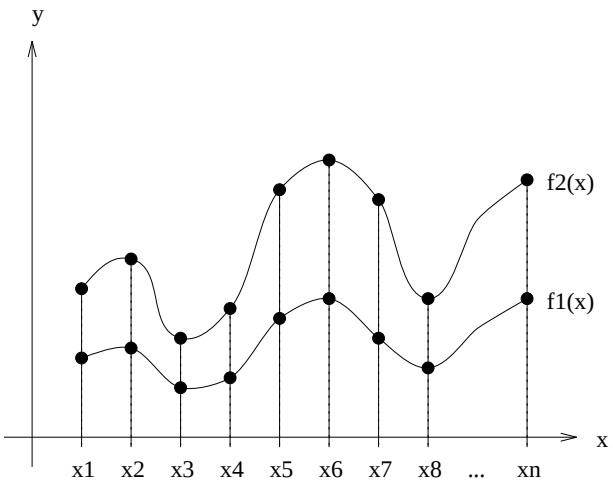
$$\begin{aligned} d(x) &= 3.02 \cdot 10^{-11} \cdot x^6 - 2.57 \cdot 10^{-8} \cdot x^5 + 8.36 \cdot 10^{-6} \cdot x^4 - 1.17 \cdot 10^{-3} \cdot x^3 + \\ & \quad 6.85 \cdot 10^{-2} \cdot x^2 + 3.51 \cdot x + 6.79 \cdot 10^{-1} \end{aligned}$$

Fig. 6.13 shows the image before and after the transformation.

Theorem 6.7 The normal equations are uniquely solvable if and only if the vectors

$$\begin{pmatrix} f_1(x_1) \\ \vdots \\ f_1(x_n) \end{pmatrix}, \ldots, \begin{pmatrix} f_k(x_1) \\ \vdots \\ f_k(x_n) \end{pmatrix}$$

are linearly independent.



$$\begin{pmatrix} f_2(x_1) \\ \vdots \\ f_2(x_n) \end{pmatrix} = 2 \begin{pmatrix} f_1(x_1) \\ \vdots \\ f_1(x_n) \end{pmatrix}$$

$\Rightarrow f_1$ and f_2 are **nicht** linearly independent on the grid $(x_1, \ldots, x_n)$.

Proof:
Normal equations uniquely solvable $\Leftrightarrow$ A non-singular

$$A_{jl} = \sum_{i=1}^{n} f_l(x_i) f_j(x_i)$$

$$\Leftrightarrow A = F^T F \quad \text{mit} \quad F = \begin{pmatrix} f_1(x_1) & \cdots & f_k(x_1) \\ \vdots & & \vdots \\ f_1(x_n) & \cdots & f_k(x_n) \end{pmatrix}$$

Assumption: $F^T F$ is singular $\Rightarrow \exists z \neq 0 : F^T F z = 0$

$$\Rightarrow z^T F^T F z = \|Fz\|_2^2 = 0$$

$$\Rightarrow Fz = 0 \Rightarrow \sum_{i=1}^{k} \boldsymbol{a}_i z_i = 0 \quad (\boldsymbol{a}_i\text{=i-th colunm of F})$$

$\Rightarrow$ columns of F are linearly dependent
$\Rightarrow$ contradiction to the assumption of Theorem 6.7

Example 6.10 We now show that the method of least squares is actually applicable in the example 6.9 and that the coefficients are uniquely determined.
According to Theorem 6.7 the following vectors must be linearly independent:

$$v_1 = \begin{pmatrix} f_1(x_1) \\ \vdots \\ f_1(x_4) \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \\ 9 \\ 16 \end{pmatrix}, \quad v_2 = \begin{pmatrix} f_2(x_1) \\ \vdots \\ f_2(x_4) \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 3 \\ 4 \end{pmatrix}, \quad v_3 = \begin{pmatrix} f_3(x_1) \\ \vdots \\ f_3(x_4) \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

If v_1, v_2, v_3 are linear independent, there must be real numbers $a, b, c \neq 0$, so that

$$a \begin{pmatrix} 0 \\ 4 \\ 9 \\ 16 \end{pmatrix} + b \begin{pmatrix} 0 \\ 2 \\ 3 \\ 4 \end{pmatrix} + c \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = 0.$$

Assume there are such Numbers a, b, c. Then it follows immediately $c = 0$ out of which

$$a \begin{pmatrix} 0 \\ 4 \\ 9 \\ 16 \end{pmatrix} + b \begin{pmatrix} 0 \\ 2 \\ 3 \\ 4 \end{pmatrix} = 0.$$

follows. But this means, v_1 must be a multiple of v_2. This is obviously not the case. So v_1, v_2, v_3 are linear independent.

6.3.3 Special Case: Straight Line Regression

regression line $f(x, a, b) = ax + b$

$$E = \sum_{i=1}^{n} (ax_i + b - y_i)^2$$

$$\begin{aligned} \frac{\partial E}{\partial a} &= 2 \sum_{i=1}^{n} (ax_i + b - y_i) x_i = 0 \\ \frac{\partial E}{\partial b} &= 2 \sum_{i=1}^{n} (ax_i + b - y_i) = 0 \end{aligned}$$

$$\begin{aligned} a \sum_{i=1}^{n} x_i^2 + b \sum_{i=1}^{n} x_i &= \sum_{i=1}^{n} x_i y_i \\ a \sum_{i=1}^{n} x_i + nb &= \sum_{i=1}^{n} y_i \end{aligned}$$

Solution:

$$\begin{aligned} a &= \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2} \\ b &= \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{n \sum x_i^2 - (\sum x_i)^2} \end{aligned}$$

Remains to be shown: The solution $\binom{a}{b}$ of degree E=0 is a minimum!

6.3.4 Statistical Justification

The method of least squares can be justified well with statistical methods. Here this is done only for one special case. Let $f(x) = c$ be the constant function and c be sought.

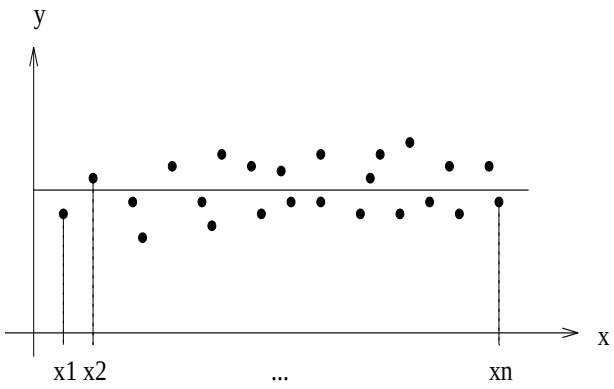


Figure 6.14: Mean over all function values.

$$E = \sum_{i=1}^{n} (f(x_i) - y_i)^2 = \sum_{i=1}^{n} (c - y_i)^2$$

$$\begin{aligned} \frac{\partial E}{\partial c} &= 2\sum_{i=1}^{n}(c - y_i) = 2\left(\sum_{i=1}^{n} c - \sum_{i=1}^{n} y_i\right) \\ &= 2\left(nc - \sum_{i=1}^{n} y_i\right) = 0 \end{aligned}$$

$$\Rightarrow nc = \sum_{i=1}^{n} y_i$$

$$\boxed{c = \frac{1}{n}\sum_{i=1}^{n} y_i} \text{ arithmetic mean}$$

Errors of the coefficients a_i
Because of measurement errors in (x_i, y_i), the coefficients $a_1, \ldots, a_k$ are erroneous.Calculation of the errors $\Delta a_1, \ldots, \Delta a_k$ out of $\Delta y_1, \ldots, \Delta y_n$ with the law of error propagation (maximum error).[1]

$$\Delta a_i = \sum_{j=1}^{n} \left|\frac{\partial a_i}{\partial y_j}\right| \Delta y_j$$

For many measurements, the formula for the maximum error gives a too large value. A better approximation is obtained by the formula for the mean Error

[1] Δy_i is the absolute value of the maximum expected measurement error of variable y_i.

$$\Delta a_i = \sqrt{\sum_{j=1}^{n} \left(\frac{\partial a_i}{\partial y_j}\right)^2 (\Delta y_j)^2}$$

Special Case Straight Line Regression:

$$\begin{aligned} \frac{\partial a}{\partial y_j} &= \frac{1}{N}\left(n x_j - \sum_{i=1}^{n} x_i\right) \\ \frac{\partial b}{\partial y_j} &= \frac{1}{N}\left(\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right) x_j\right) \\ \text{with} \quad N &= n\sum x_i^2 - \left(\sum x_i\right)^2 \end{aligned}$$

$$\begin{aligned} \Delta a &= \sum_{j=1}^{n} \left|\frac{\partial a}{\partial y_j}\right| \Delta y_j \\ \Delta b &= \sum_{j=1}^{n} \left|\frac{\partial b}{\partial y_j}\right| \Delta y_j \end{aligned}$$

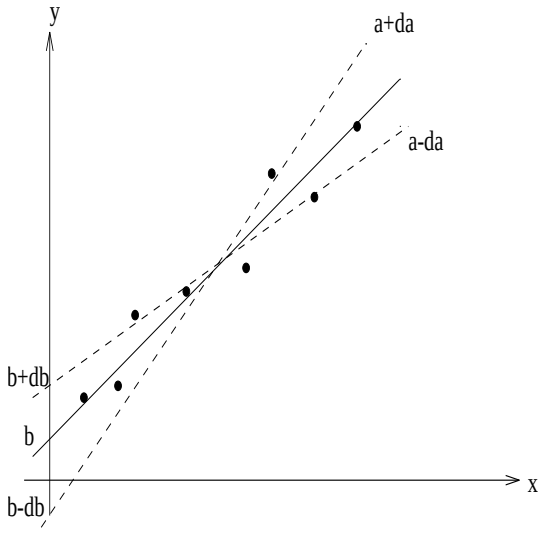


Figure 6.15: regression line through value pairs.

Nonlinear Regression (Examples): Power function:

$$\begin{aligned} v &= c \cdot u^d \qquad \text{Constants } c, d \text{ sought!} \\ \log v &= \log c + d \log u \end{aligned}$$

$$y := \log v, x := \log u \Rightarrow a_1 = \log c, a_2 = d$$

$$y = a_1 + a_2 x$$

Exponential function:

$$\begin{aligned} v &= Ae^{bu} \qquad A, b \text{ sought} \\ \ln v &= \ln A + bu \end{aligned}$$

$$y := \ln v, \quad x := u, \quad \Rightarrow \quad a_1 = \ln A, \quad a_2 = b$$
$$y = a_1 + a_2 x$$

6.3.5 Multidimensional Least Squares

The method presented so far is good for the approximation of functions $f : \mathbb{R} \to \mathbb{R}$, i.e. for one-dimensional functions with one-dimensional argument. In the setting of Equation 6.20 we determine the coefficients $a_1, \ldots, a_k$ of a linear combination of one-dimensional basis functions $f_1, \ldots, f_k$:

$$f(x) = a_1 f_1(x) + \cdots + a_k f_k(x) = \boldsymbol{a}^T \boldsymbol{f}(x). \tag{6.22}$$

Now, there is a very easy generalization of this ansatz to multidimensional input. We just replace the one-dimensional x by a vector $\boldsymbol{x}$ to obtain

$$f(\boldsymbol{x}) = a_1 f_1(\boldsymbol{x}) + \cdots + a_k f_k(\boldsymbol{x}) = \boldsymbol{a}^T \boldsymbol{f}(\boldsymbol{x}).$$

In the derivation of the normal equations, proof, etc. there are no changes other than replacing x by a vector.
A different way to get into the multidimensional world is the ansatz

$$f(\boldsymbol{x}) = a_1 x_1 + \cdots + a_k x_k = \boldsymbol{a}^T \boldsymbol{x}.$$

The advantage here is that we do not have to worry about the selection of the basis functions f_i. But there is no free lunch. The drawback is the very limited power of the linear approximation.

6.3.6 A More General View

We still want to fit a function

$$f(\boldsymbol{x}) = a_1 f_1(\boldsymbol{x}) + \cdots + a_k f_k(\boldsymbol{x}) = \boldsymbol{a}^T \boldsymbol{f}(\boldsymbol{x})$$

with k unknown parameters $a_1, \ldots, a_k$ through the n data points $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$. If we substitute all the points into the ansatz, requiring our function to hit all n points, i.e.

$$f(\boldsymbol{x}_i) = y_i,$$

we get the linear system

$$\begin{aligned} a_1 f_1(\boldsymbol{x}_1) + \ldots + a_k f_k(\boldsymbol{x}_1) &= y_1 \\ \vdots \qquad\qquad\quad \vdots \qquad\quad &\vdots \\ a_1 f_1(\boldsymbol{x}_n) + \ldots + a_k f_k(\boldsymbol{x}_n) &= y_n. \end{aligned} \tag{6.23}$$

If we define the $n \times k$-matrix M as

$$M_{ij} = f_j(\boldsymbol{x}_i),$$

Equation 54 reads

$$M \cdot \boldsymbol{a} = \boldsymbol{y}.$$

For $n > k$ the system is overdetermined and normally has no solution. In the next section, we will show how to find an approximate solution by using the method of least squares.
For the case $n = k$ we may get a unique solution, because here M is a square matrix. If we use for $j = 0, \ldots, k$ the basis functions

$$f_j(x) = x^j,$$

we end up with the Vandermonde matrix from Section 6.1.2.

6.3.7 Solving Overdetermined Linear Systems

The linear System

$$\begin{array}{ccccccc} x_1 & + & x_2 & + & x_3 & = & 1 \\ x_1 & + & x_2 & & & = & 1 \\ x_1 & + & & & x_3 & = & 1 \\ & & x_2 & + & x_3 & = & 1 \end{array}$$

is not solvable, because it is overdetermined. Even though we have to accept this fact, we can ask, which vektor x fulfills the linear system best. This can be formalized as follows:
Given, an overdetermined linear system

$$M\boldsymbol{x} = \boldsymbol{y}$$

with n equations and $k < n$ unknowns $x_1, \ldots x_k$. M is a $n \times k$ matrix, $\boldsymbol{x} \in \mathbb{R}^k$ and $\boldsymbol{y} \in \mathbb{R}^n$. Obviously, in general, there is no vector $\boldsymbol{x}$, for which $M\boldsymbol{x} = \boldsymbol{y}$. Therfore we are looking for a vector $\boldsymbol{x}$, which makes the left side as good as possible equal to the right side. That is, for which $M\boldsymbol{x} \approx \boldsymbol{y}$, or for which

$$||M\boldsymbol{x} - \boldsymbol{y}||_2 = \sqrt{(M\boldsymbol{x} - \boldsymbol{y})^2}$$

gets minimal. It also follows that $(M\boldsymbol{x} - \boldsymbol{y})^2$ gets minimal. So

$$\sum_{i=1}^{n}((M\boldsymbol{x})_i - y_i)^2 = \sum_{i=1}^{n}\left(\sum_{l=1}^{k} M_{il}x_l - y_i\right)^2$$

must be minimal. To determine the minimum we set all partial derivatives equal to zero:

$$\frac{\partial}{\partial x_j}\sum_{i=1}^{n}\left(\sum_{l=1}^{k} M_{il}x_l - y_i\right)^2 = 2\sum_{i=1}^{n}\left(\sum_{l=1}^{k} M_{il}x_l - y_i\right)M_{ij} = 0$$

and get after multiplying out

$$\sum_{i=1}^{n}\sum_{l=1}^{k} M_{il}M_{ij}x_l = \sum_{i=1}^{n} M_{ij}y_i$$

or

$$\sum_{l=1}^{k}\left(\sum_{i=1}^{n} M_{ji}^T M_{il}\right)x_l = \sum_{i=1}^{n} M_{ji}^T y_i$$

or as a vector equation

$$M^T M\boldsymbol{x} = M^T\boldsymbol{y}. \tag{6.24}$$

Therewith we have derived the following theorem

Theorem 6.8 Let an overdetermined linear system $M\boldsymbol{x} = \boldsymbol{y}$ with $\boldsymbol{x} \in \mathbb{R}^k$, $\boldsymbol{y} \in \mathbb{R}^n$ $(n > k)$ and the $n \times k$ matrix M be given. The solution $\hat{\boldsymbol{x}}$ with least squared error can be determined by solving the linear system

$$M^T M\hat{\boldsymbol{x}} = M^T\boldsymbol{y}.$$

This system has a unique solution if and only if the matrix M has full rank (This proposition is equivalent to theorem 6.7.).

Please note that Equation 6.24 is identical to the normal equations (Equation 6.21, proof as exercise.) This linear system can be rewritten into

$$\hat{\boldsymbol{x}} = (M^T M)^{-1}M^T\boldsymbol{y}.$$

If M is invertible and the system $M\boldsymbol{x} = \boldsymbol{y}$ is uniquely solvable, then the solution $\boldsymbol{x}$ can be calculated by

$$\boldsymbol{x} = M^{-1}\boldsymbol{y}.$$

Comparing this equation with the above for $\hat{\boldsymbol{x}}$, it is clear why the square matrix

$$(M^T M)^{-1}M^T$$

is called **pseudoinverse** of M. The matrix $M^T M$ is the so called Gram matrix of M. Now we apply the theorem to the example at the beginning of the section which reads

$$\begin{pmatrix} 111 \\ 110 \\ 101 \\ 011 \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

Application of Theorem 6.8 delivers

$$M^T M = \begin{pmatrix} 1\,1\,1\,0 \\ 1\,1\,0\,1 \\ 1\,0\,1\,1 \end{pmatrix} \cdot \begin{pmatrix} 1\,1\,1 \\ 1\,1\,0 \\ 1\,0\,1 \\ 0\,1\,1 \end{pmatrix} = \begin{pmatrix} 3\,2\,2 \\ 2\,3\,2 \\ 2\,2\,3 \end{pmatrix}$$

and the equation

$$\begin{pmatrix} 3\,2\,2 \\ 2\,3\,2 \\ 2\,2\,3 \end{pmatrix} \cdot \hat{\boldsymbol{x}} = \begin{pmatrix} 3 \\ 3 \\ 3 \end{pmatrix}$$

with the solution $\hat{\boldsymbol{x}} = (\frac{3}{7}, \frac{3}{7}, \frac{3}{7})^T$.

6.3.8 Solving Underdetermined Linear Systems

Let

$$M\boldsymbol{x} = \boldsymbol{y}$$

be an underdetermined linear system with n equations and $k > n$ unknowns $x_1, \ldots x_k$. So M is a $n \times k$ matrix, $\boldsymbol{x} \in \mathbb{R}^k$ and $\boldsymbol{y} \in \mathbb{R}^n$. Obviously, there are in general infinitely many vectors $\boldsymbol{x}$, with $M\boldsymbol{x} = \boldsymbol{y}$. So we can choose any one of these vectors. One way for this choice is to choose out of the set of solution vectors $\boldsymbol{x}$ one with minimal square norm $\|\boldsymbol{x}\|^2$.

The task to determine such a vector, can also be formulated as constrained extremum problem. A minimun of $\|\boldsymbol{x}\|^2$ under n constraints $M\boldsymbol{x} = \boldsymbol{y}$ is sought. With the method of Lagrange parameters it is

$$\|\boldsymbol{x}\|^2 + \boldsymbol{\lambda}^T(\boldsymbol{y} - M\boldsymbol{x})$$

For this scalar function, the gradient must become zero:

$$\nabla(\|\boldsymbol{x}\|^2 + \boldsymbol{\lambda}^T(\boldsymbol{y} - M\boldsymbol{x})) = 2\boldsymbol{x} - M^T\boldsymbol{\lambda} = 0$$

Multiplying the second equation from the left with M results in

$$2M\boldsymbol{x} - MM^T\boldsymbol{\lambda} = 0.$$

Insertion of $M\boldsymbol{x} = \boldsymbol{y}$ leads to

$$2\boldsymbol{y} = MM^T\boldsymbol{\lambda}$$

and

$$\boldsymbol{\lambda} = 2(MM^T)^{-1}\boldsymbol{y}.$$

With $2\boldsymbol{x} = M^T\boldsymbol{\lambda}$, we get

$$\boldsymbol{x} = 1/2\, M^T\boldsymbol{\lambda} = M^T(MM^T)^{-1}\boldsymbol{y}. \tag{6.25}$$

The matrix $M^T(MM^T)^{-1}$ is now a new **pseudoinverse**.

6.3.9 Application of the Pseudoinverse for Function Approximation

Let k basis functions $f_1, \ldots, f_k$ and n data points $(x_1, y_1), \ldots, (x_n, y_n)$ be given. We want to determine parameters $a_1 \ldots a_k$ for

$$f(x) = a_1 f_1(x) + \ldots a_k f_k(x),$$

such that for all x_i the equation $f(x_i) = y_i$ is fulfilled "as good as possible". For the three cases $n < k$, $n = k$ and $n > k$ we present examples. First, we determine the seven coefficients of the polynomial

$$f(x) = a_1 + a_2x + a_3x^2 + a_4x^3 + a_5x^4 + a_6x^5 + a_7x^6$$

with the help of the points $(1,1)$, $(2,1)$, $(3,1)$, $(4,1)$, $(5,4)$. Inserting the points results the underdetermined system of equations

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 & 16 & 32 & 64 \\ 1 & 3 & 9 & 27 & 81 & 243 & 729 \\ 1 & 4 & 16 & 64 & 256 & 1024 & 4096 \\ 1 & 5 & 25 & 125 & 625 & 3125 & 15625 \end{pmatrix} \cdot \boldsymbol{a} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 4 \end{pmatrix}.$$

Computing the pseudoinverse and solving for $\boldsymbol{a}$ yields

$$\boldsymbol{a^T} = (0.82, 0.36, -0.092, -0.23, 0.19, -0.056, 0.0055).$$

The result is shown in Figure 6.16, left. We recognize that here, despite the relatively high degree of the polynomial a very good approximation is achieved, (why?).

Reducing the degree of the polynomial to four, gives a quadratic matrix. It consists of the first five columns of the matrix above and the system becomes uniquely solvable with

$$\boldsymbol{a}^{\boldsymbol{T}} = (4., -6.25, 4.38, -1.25, 0.125).$$

In Figure 6.16 (middle) oscillations can be seen, which are due to significantly larger absolute values of the coefficients.
After a further reduction of the polynomial to two, only the first three columns of the matrix remain and the solution via pseudoinverse delivers the least squares parabola with the coefficients

$$\boldsymbol{a}^{\boldsymbol{T}} = (2.8, -1.97, 0.429)$$

as shown on the right in fig. 6.16.

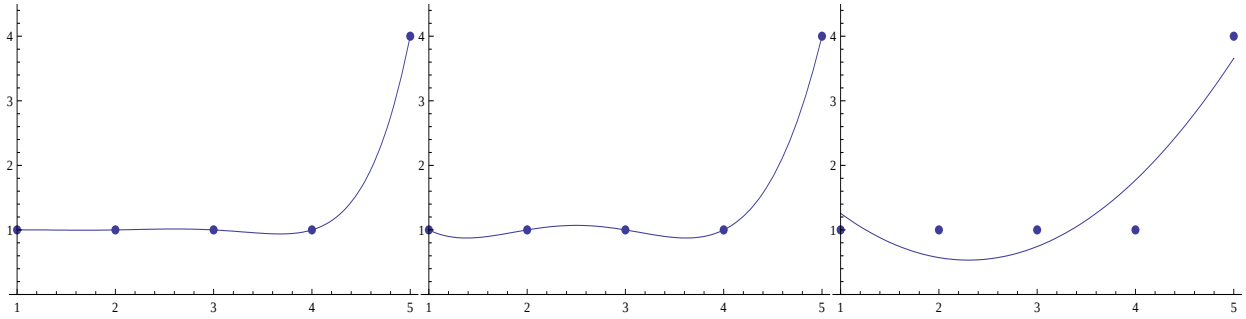


Figure 6.16: Polynomial fitted to data points in the underdetermined ($k = 7,\ n = 5$, left), unique ($k = 5,\ n = 5$, center) and overdetermined ($k = 3,\ n = 5$, right) case.

We see that the work with underdetermined problems can be quite interesting and can lead to good results. Unfortunately this is not always the case. If we try for example, like in the example of the polynomial interpolation of fig. 6.7 with fixed number of 11 given points, to increase the degree of the polynomial, then, unfortunately, the oscillations increase too, instead of decrease (see fig. 6.17). The parametric methods usually require some manual influencing. In the next section we describe Gaussian processes a method that works very elegantly and requires minimal manual adjustments.

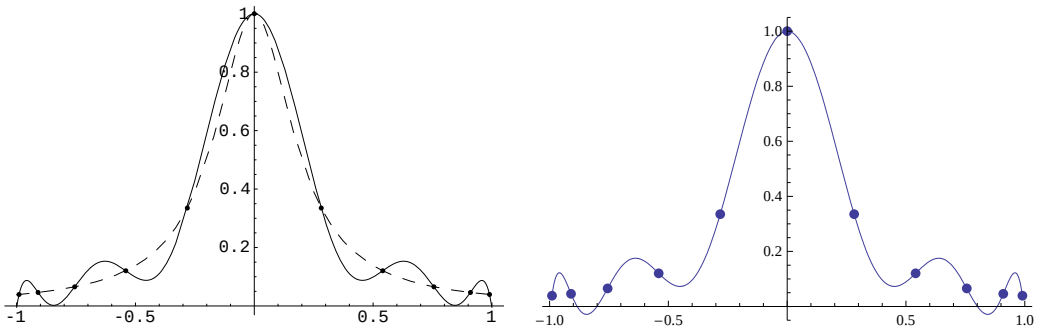


Figure 6.17: Ordinary Chebychef interpolation (left and Figure 6.7) with 11 points leading to a Polynomial of degree 10 and the solution of the underdetermined system for a polynomial of degree 12 with the same points (right) yielding somewhat higher error.

6.3.10 Summary

With the method of least error squares and minimizing the square of the solution $\boldsymbol{x}$, we have procedures to solve over and underdetermined linear systems. But there are also other methods. For example, in the case of underdetermined systems of equations, instead of determining $\|\boldsymbol{x}\|^2$, we could e.g. maximize the entropy

$$-\sum_{i=1}^{k} x_i \ln x_i$$

or determine an extremum of another function $\|\boldsymbol{x}\|$. The methods presented here are used mainly, because the equations to be solved remain linear.
The computing time for calculating the pseudoinverse can be estimated in underdetermined and in overdetermined case by $O(k^2n+k^3)$. Slightly faster than the calculation of $(MM^T)^{-1}$ it is using the QR decomposition or the Singular Value Decomposition (SVD). Then the time complexity is reduced to $O(k^2n)$. The here calculated pseudoinverses are so-called Moore-Penrose pseudoinverses. That is, in the case of a matrix M with real-valued coefficients, the pseudoinverse M^+ has the following features:

$$MM^+M = M$$
$$M^+MM^+ = M^+$$

Applied on M, MM^+ behaves indeed like an identity matrix.

6.4 Exercises

Polynomial Interpolation

Exercise 6.1

a) Let the points $(-1,1)$, $(0,0)$ and$(1,1)$ be given. Determine the interpolation polynomial through these three points.

b) Let the points $(-1,1)$, $(0,0)$, $(1,1)$ and $(2,0)$.be given. Determine the interpolation polynomial through these four points.

Exercise 6.2

a) Write a Mathematica program that calculates a table of all coefficients of the interpolating polynomial of degree n for any function f in any interval $[a,b]$. Pass the function name, the degree of the polynomial and the value table as parameters to the program. The Mathematica functions `Expand` and `Coefficient` may be useful.

b) Write for the value table generation a program for the equidistant case and one for the Chebyshev abscissas.

Exercise 6.3

a) Apply the program of exercise 6.2 to the interpolation of the function $f(x) := e^{-x^2}$ in the interval $[-2,10]$ and calculate the polynomial up to the 10th degree. The given points are to be distributed "equidistant".

Exercise 6.4

a) Calculate the maximum norm of the deviation between the interpolation polynomial p and f from exercise 6.3 on an equidistant grid with 100 given points.

b) Compare the Equidistant interpolation with the Chebyshev interpolation and with the Taylor series of f of degree 10 (expanded around $x_0 = 0$ and $x_0 = 4$, use the function (tt Series)) with respect to maximum norm of the approximation error.

Spline-Interpolation

Exercise 6.5 Given two points $(1, 1)$ and $(2, 0)$ for computing a cubic spline with natural constraints ($y_0'' = y_n'' = 0$).

a) How many lines and columns has the tri-diagonal matrix for computing the y''-variables?

b) Determine the spline by manually calculating the coefficients a_i, b_i, c_i, d_i

Exercise 6.6 The points $(-1, 1)$, $(0, 0)$ and$(1, 1)$ are given.

a) Determine the two cubic part splines with natural boundary conditions.

b) Why $s_0(x) = x^2$ and $s_1(x) = x^2$ is not a cubic spline function with natural boundary conditions? Argue **unrelated** to the correct solution.

Exercise 6.7 How does the coefficient matrix for the spline interpolation change, if instead of the boundary conditions $y_0'' = y_n'' = 0$, the boundary conditions $y_0'' = y_1''$, $y_n'' = y_{n-1}''$ (second derivative at the border) would be demanded? Change the coefficient matrix of example 7.1 accordingly.

Exercise 6.8 Program the tridiagonal matrix algorithm.

Exercise 6.9 Write a program to calculate a natural cubic spline out of a given value table.

Exercise 6.10 Apply the program from Exercise 6.9 on the interpolation of the function $f(x) := e^{-x^2}$ in the interval $[-2, 10]$ on a equidistant Grid with 11 points.

Exercise 6.11 Iterated Function Systems (IFS):

a) Calculate the value tables of the two sequences (x_n), (y_n) with

$$\begin{aligned} x_{n+1} &= a\, y_n + b \\ y_{n+1} &= c\, x_n + d \\ x_0 &= y_0 = 1 \end{aligned}$$

to $n = 20$, where use the parameter values $a = 0.9$, $b = -0.9$, $c = -0.9$, $d = 0.9$.

b) Connect the points $(x_0, y_0) \ldots (x_n, y_n)$ with a cubic natural spline. Select as parameter for the parametric representation the points euclidean distance.

Least Squares and Pseudoinverse

Exercise 6.12 With the method of least squares the coefficients a_1, a_2 of the function $f(x) = \frac{a_1}{x^2} + \frac{a_2}{(x-9)^2}$ using the given points $(1, 6), (2, 1), (7, 2), (8, 4)$ are to be determined.

a) Set up the normal equations.

b) Calculate the coefficients a_1, a_2.

c) Draw f in the interval $(0, 9)$ together with the points in a chart.

Exercise 6.13

a) Write a Mathematica program to determine the coefficients $a_1 \ldots a_k$ of a function

$$f(x) = a_1 f_1(x) + a_2 f_2(x) + \cdots + a_k f_k(x)$$

with the method of least squares. Parameters of the program are a table of data points, as well as a vector with the names of the base functions $f_1, \ldots, f_k$. Try to work without for loops and use the function (`LinearSolve`).

b) Test the program by creating a linear equation with 100 points on a line, and then use your program to determine the coefficients of the line. Repeat the test with slightly noisy data (add a small random number to the data values).

c) Determine the polynomial of degree 4, which minimizes the sum of the error squares of the following value table (see: http://www.hs-weingarten.de/~ertel/vorlesungen/mathi/mathi-ueb15.txt):

8	-16186.1	18	8016.53	28	10104.	38	41046.6
9	-2810.82	19	7922.01	29	15141.8	39	37451.1
10	773.875	20	4638.39	30	15940.5	40	37332.2
11	7352.34	21	3029.29	31	19609.5	41	29999.8
12	11454.5	22	2500.28	32	22738.	42	24818.1
13	15143.3	23	6543.8	33	25090.1	43	10571.6
14	13976.	24	3866.37	34	29882.6	44	1589.82
15	15137.1	25	2726.68	35	31719.7	45	-17641.9
16	10383.4	26	6916.44	36	38915.6	46	-37150.2
17	14471.9	27	8166.62	37	37402.3		

d) Calculate to c) the sum of the squares. Determine the coefficients of a parabola and calculate again the sum of the error squares. What difference do you see?

e) Which method allows you to determine experimentally, at several possible sets of basis functions, the "best"?

f) Find a function which creates an even smaller error.

Exercise 6.14 Given: $(0, 2)$, $(1, 3)$, $(2, 6)$. Determine with the method of least squares the coefficients c and d of the function $f(x) = c \cdot e^{d \cdot x}$. Note that the parameter d occurs nonlinear!

Exercise 6.15

a) Change the right hand side of the first system of equations at the beginning of Section 6.3.7, so that it gets uniquely solvable.

b) Which condition must hold, such that a linear system with n unknowns and $m > n$ equations is uniquely solvable?

Exercise 6.16 Use Theorem 6.8 to solve the system of equations $x_1 = 1$, $x_1 = 2$, $x_2 = 5$, $x_2 = 9$, $x_3 = -1$, $x_3 = 1$ by the method of least squares.

Exercise 6.17 Show that for the pseudoinverse M^+ of the sections 6.3.7 and 6.3.8 it holds $MM^+M = M$ and$M^+MM^+ = M^+$.

Exercise 6.18 Show that the computing time for the calculation of the pseudoinverse in sections 6.3.7 and 6.3.8 can be estimated by $O(k^2 n + k^3)$.

Exercise 6.19 Prove that the equation $M^T M \boldsymbol{x} = M^T \boldsymbol{y}$ for the approximate solution of an overdetermined linear system $M\boldsymbol{x} = \boldsymbol{y}$ (Equation 6.24) is equivalent to the normal equations from the least squares method (Equation 6.21).

Exercise 6.20 Given M,

$$M = \begin{pmatrix} 8 & 2 & 2 \\ 2 & 4 & 1 \end{pmatrix} \tag{6.26}$$

a) Perform the SVD decomposition and write M in the form $M = U\Sigma V^T$.

b) Compute the pseudoinverse M^+ of M.

c) Show that M^+ is a valid (Moore-Penrose) pseudoinverse.

d) Show that the pseudoinverse of M, using the technique of the underdertemined system mentioned in section 6.3.8, is the same as the one computed by SVD.

Exercise 6.21 Given the following Matrix M,

$$M = \begin{pmatrix} 3 & 6 \\ 2 & 4 \\ 2 & 4 \end{pmatrix}$$

a) Show that the pseudoinverse of the matrix M, using the technique of the overdetermined system mentioned in section 6.3.7, is not applicable.

b) Perform the SVD decomposition and write M in the form $M = U\Sigma V^T$.

c) Compute the pseudoinverse M^+ of M.

d) Show that M^+ is a valid pseudoinverse.

Chapter 7

Statistics and Probability

7.1 Random Numbers

7.1.1 Applications of Random Numbers

- Randomized Algorithms
- Stochastic Simulation (Monte-Carlo simulation)
- Cryptography (e.g., key generation, one-time pad)

Literature: **Don Knuth** "The Art of Computer Programming" volume 2

In [19] U. Maurer gives a good definition of randomness:

> **Definition 7.1** A random bit generator is a device that is designed to output a sequence of statistically independent and symmetrically distributed binary random variables, i.e., that is designed to be the implementation of a so-called **binary symmetric source (BSS)**. In contrast, a pseudo-random bit generator is designed to deterministically generate a binary sequence that only appears as if it were generated by a BSS.

> **Definition 7.2** A binary variable is symmetrically distributed if the probability for both values is exactly 1/2.

A sequence is random, if f for any length ℓ the distribution of all strings of length ℓ has maximum entropy.

> **Definition 7.3** A **Pseudo Random Number Generator** (**PRNG**) is an algorithm that (after entering one or more seed numbers) deterministically generates a sequence of numbers.

For cryptographic applications very problematic!
Alternative:

Use of physical random events such as thermal noise or radioactive Decay: **True Random Numbers**
$\Rightarrow$ True Random Number Generator (**true RNG**).
philosophy:
Till recently it is unknown if **hidden parameters** are describing a seemingly random process deterministically. Physicist have proven that there are real random processes.

7.1.2 Kolmogorov Complexity

- If a (large) file can be compressed, then the content is not random.
- True random numbers can not be compressed!
- Is (31415926...) random?
- No, because $\pi = 3.1415926\ldots$ can be compressed
- Computer program can calculate any number of digits of π!

Definition 7.4 The **Kolmogorov complexity** of a (infinite) sequence is the length of a shortest program, that can compute (enumerate) the sequence's terms [28].

- π has finite Kolmogorov complexity.
- Any sequence of random numbers has infinite Kolmogorov complexity!
- Unsuitable in practice, since the Kolmogorov complexity is not computable!
- Each PRNG only produces sequences of finite Kolmogorov complexity. Such sequences are **not random**

7.1.3 Compression of Random Number Sequences

Theorem 7.1 No program can compress any files of at least n-bit ($n \geq 0$) whithout loss.

Example 7.1

length n	bit sequences of length n	number
0	ϵ	1
1	0, 1	2
2	00, 01, 10, 11	4
3	000, 001, 010, 011, 100, 101, 110, 111	8

8 sequences of length 3, but only seven shorter!

Proof: Suppose a program could do it. We compress with it (only!) all files of n-bit. The compressed files are not exceeding the size of $n-1$ bits. The number of compressed files from of size 0 bis $n-1$ bits is

$$1 + 2 + 4 + 8 + \ldots + 2^{n-1} = 2^n - 1.$$

Because there are 2^n files of size n bits, at least two files have to be compressed to the same file. Thus, the compression is not lossless. □

7.1.4 Pseudo Random Number Generators

Definition 7.5 Linear Congruence Generators are defined recursively by

$$x_n = (ax_{n-1} + b) \bmod m.$$

with parametrs a, b and m.

[29] recommends for 32-bit integers $a = 7141$, $b = 54773$ and $m = 259200$.
The period is not exceeding m. Why? (see exercise 7.3)

Theorem 7.2 The functional characteristics of a congruence generator lead to the following upper bounds for the period:

recursion scheme	period
$x_n = f(x_{n-1}) \bmod m$	$\leq m$
$x_n = f(x_{n-1}, x_{n-2}) \bmod m$	$\leq m^2$
$x_n = f(x_{n-1}, x_{n-2}, x_{n-3}) \bmod m$	$\leq m^3$
$\ldots$	

Proof: With the modulus m we have only m different values for x_n. Since f is deterministic, if $x_n = f(x_{n-1}) \bmod m$, after the first repeated value, all succeeding values will be repeated as well. Thus the period is $\leq m$. If f depends on two previous values, then there are m^2 combinations. Thus the period is bounded by m^2 and so on. □

Apparently, the more predecessors x_n depends on, the longer the period can become. So it seems natural to use as many predecessors as possible. We try it with the sum of all predecessors and get

$$x_0 = a, \qquad x_n = \left(\sum_{i=0}^{n-1} x_i \right) \bmod m,$$

which may even lead to a non-periodic sequence, because the number of used predecessors gets bigger with increasing m.
Let us first consider the specified sequence with $x_0 = 1$ non-modular:

$$1, 1, 2, 4, 8, 16, 32, 64, 128, 256, \ldots$$

Obviously this is an exponential sequence, hence

Theorem 7.3 The recursively defined formula $x_0 = 1, x_n = \sum_{i=0}^{n-1} x_i$ for $n \geq 1$ is equivalent to $x_n = 2^{n-1}$.

Proof: For $n \geq 2$ we have

$$x_n = \sum_{i=0}^{n-1} x_i = x_{n-1} + \sum_{i=0}^{n-2} x_i = x_{n-1} + x_{n-1} = 2 \cdot x_{n-1}.$$

For $n = 1$, $x_1 = x_0 = 1$. Now it can be shown easily by induction, that $x_n = 2^{n-1}$ for $n \geq 1$ (see exercise 7.3). □

For the modular sequence $x_0 = 1, x_n = \left(\sum_{i=0}^{n-1} x_i\right) \bmod m$ is equivalent to $x_n = 2^{n-1} \bmod m$ for $n \geq 1$. Thus x_n depends only on x_{n-1} and m is the periods upper bound.
The period of the sequence is even $\leq m - 1$, because when zero is reached, the result will remain zero.
Not only the period is important for the quality of a PRNG. The symmetry of the bits should as well be good.

7.1.5 The Symmetry Test

In principle, it is easy to test a bit sequence on symmetry. The mean of an n-bit sequences has to be calculated

$$M(X_1, \ldots, X_n) = \frac{1}{n} \sum_{i=1}^{n} x_i$$

and compared with the expected value $E(X) = 1/2$ of a true random bit sequence. If the deviation of the mean from the expected value is small enough, the sequence passes the test. Now we want to calculate a threshold for the tolerable deviation. The expected value of a true random bit X is $E(X) = 1/2$ and also its standard deviation $\sigma(X) = 1/2$ (see exercise 7.4). The mean of n true random numbers, will deviate less from the expected value, the larger n gets. The central limit theorem (Theorem 4.4) tells us that for n independent identically distributed random variables $X_1, X_2, \ldots, X_n$ with standard deviation σ, the standard deviation of the sum $S_n = X_1 + \ldots + X_n$ is equal to $\sqrt{n}\sigma$. Thus, the standard deviation σ_n of the mean

$$M(X_1, \ldots, X_n) = \frac{1}{n} \sum_{i=1}^{n} x_i$$

of n random bits is

$$\sigma_n = \frac{1}{n}\sqrt{n}\sigma(X_1) = \frac{1}{\sqrt{n}}\sigma(X_1)$$

Because for random bits $\sigma(X_i) = 1/2$, we get

$$\sigma_n = \frac{1}{2\sqrt{n}}.$$

A normally distributed random variable has a value in $[\mu - 2\sigma, \mu + 2\sigma]$ with probability 0.95. This interval is the confidence interval to the level 0.95. We define the test of randomness as passed, if the mean of the bit sequence is in the interval $[1/2 - 2\sigma_n, 1/2 + 2\sigma_n,]$.

7.1.5.1 BBS Generator (Blum Blum Shub)

Even polynomial congruential generators of the form

$$x_n = (a_k x_{n-1}^k + a_{k-1} x_{n-1}^{k-1} + \ldots + a_0) \bmod m.$$

can be cracked. Therefore, it is natural to look for better generators. A PRNG that generates bits of very high quality, is the so-called BBS generator (see [23]). Choose primes p and q with

$$p \equiv q \equiv 3 \bmod 4.$$

Calculate $n = p \cdot q$ and choose a random number s, with $\text{ggT}(s, n) = 1$.
Calculate the Seed

$$x_0 = s^2 \bmod n.$$

The generator then repeatedly computes (starting with $i = 1$)

$$\begin{aligned} x_i &= (x_{i-1})^2 \bmod n \\ b_i &= x_i \bmod 2, \end{aligned}$$

and outputs b_i as the i-th random bit.
BBS is considered very good, but:
A BBS operated One-Time-Pad is as safe as a cipher with a key length of $|s|$.

7.1.6 Linear Feedback Shift Registers

Definition 7.6

- A **shift register** of length n consists of a bit vector $(x_n, \ldots, x_1)$. In each step, the bits are shifted one position to the right, i.e.

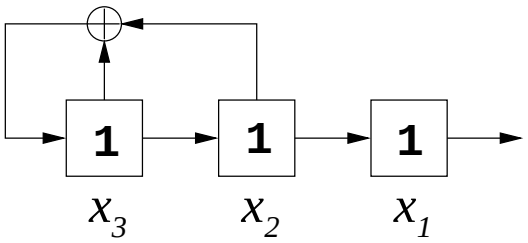
  $$x_n \mapsto x_{n-1}\,, \ldots,\; x_2 \mapsto x_1$$

  and a new bit *In* will be inserted to the left and the last bit *Out* will be output:

  $$In \mapsto x_n,\;\; x_1 \mapsto Out.$$

- A **Linear Feedback Shift Register (LFSR)** computes the new input (*In*) by modulo 2 addition of certain bits of the register.

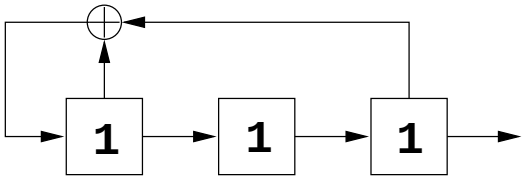
Example 7.2 LFSR_1:



x_3	x_2	x_1	*Out*
1	1	1	
0	**1**	**1**	1
1	0	1	1
1	1	0	1
0	**1**	**1**	0

Period 3.

Example 7.3 LFSR_2 has the period 7:



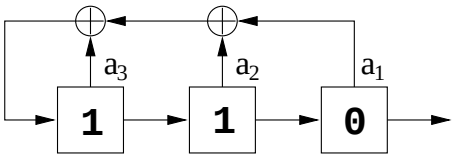
x_3	x_2	x_1	*Out*
1	**1**	**1**	
0	1	1	1
1	0	1	1
0	1	0	1
0	0	1	0
1	0	0	1
1	1	0	0
1	**1**	**1**	0

The maximum period of a LSFR of length n is $2^n - 1$. Why?

Example 7.4 Analysis of a LFSR of length 3
We look at the bit sequence

$$B = (01110010)$$

and search the parameters a_1, a_2, a_3.



The LFSR can be represented mathematically by mapping

$$(x_3, x_2, x_1) \mapsto (a_1 x_1 \oplus a_2 x_2 \oplus a_3 x_3, x_3, x_2),$$

repeatedly. The first three bits of the sequence B represent the state of the LFSR at a specific time, i.e. $x_1 = 0$, $x_2 = 1$, $x_3 = 1$
State of the LFSR: $(1, 1, 0)$
For each time unit later we get the state

$$(1,1,1) = (a_2 \oplus a_3, 1, 1) \tag{7.1}$$
$$(0,1,1) = (a_3 \oplus a_2 \oplus a_1, 1, 1) \tag{7.2}$$
$$(0,0,1) = (a_2 \oplus a_1, 0, 1) \tag{7.3}$$

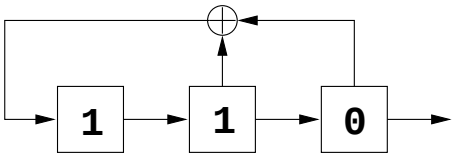
From (7.1), (7.2), (7.3) we obtain the equations

$$a_2 \oplus a_3 = 1 \tag{7.4}$$
$$a_3 \oplus a_2 \oplus a_1 = 0 \tag{7.5}$$
$$a_2 \oplus a_1 = 0 \tag{7.6}$$

and calculate

$$(7.4) \text{ in } (7.5) : 1 \oplus a_1 = 0 \Rightarrow a_1 = 1 \tag{7.7}$$
$$(7.7) \text{ in } (7.6) : a_2 \oplus 1 = 0 \Rightarrow a_2 = 1 \tag{7.8}$$
$$(7.8) \text{ in } (7.4) : a_3 = 0 \tag{7.9}$$

Thus the shift register has the form



and the sequence of states of a period of $LFSR_3$ is

1	**1**	**0**	
1	1	1	0
0	1	1	1
0	0	1	1
1	0	0	1
0	1	0	0
1	0	1	0
1	**1**	**0**	1
			0

Note that for analysis, only six bits of the Output sequence were used and that LFSR_3 has maximum period.

In general it can be shown, that for analysing a linear shift register at most $2n$ bits are required of the output sequence. (*Berlekamp-Massey-Algorithm*)

Definition 7.7 The **Linear Complexity** of a sequence is the length of the shortest LFSR that can generate the result.

If a key sequence has finite linear complexity n, then only $2n$ sequence bits are required to crack the code of the corresponding stream cipher.
$\Rightarrow$ *Kolmogorov Complexity.*

7.1.7 True Random Numbers

- Special Hardware
  - Physical Noise Source, AD converter, Amplifier, Filter, Test(?)
  - Special Hardware (Thermal Noise) for test purposes
  - Special Hardware for cryptographic applications are too expensive
- Intel: thermal noise of a resistor in the Pentium III processor
  - Frequency: 75000 bits per second [30]
- Maxtor: Noise of IDE Hard Drives
  - Frequency: 835 200 bits per second [21]

7.1.7.1 The Neumann Filter

John von Neumann, 1963, invented the following formula for repairing asymmetric sequences:

$$\begin{aligned} f: \quad 00 &\mapsto \epsilon \\ 11 &\mapsto \epsilon \\ 01 &\mapsto 0 \\ 10 &\mapsto 1, \end{aligned}$$

ϵ = the empty character string

Example 7.5 $10001101011100101110 \mapsto 10011$

Example 7.6 $11111111111111111111 \mapsto \epsilon$

Example 7.7 $10101010101010101010 \mapsto 1111111111$

Theorem 7.4 Are consecutive bits in a long ($n \to \infty$) bit sequence statistically independent, then after application of the Neumann Filter they are symmetrically distributed. The length of the bit sequence is shortened by the factor $p(1-p)$.

Proof:
If in a sequence the bits are independent and with probability p take the value "1", then the probability for a pair of "01" equals $p(1-p)$. The probability for the pair "10" is also $p(1-p)$. Thus, the probability p_n for the value "1" after the application of the Neumann Filter is given by

$$p_n = \frac{p(1-p)}{2p(1-p)} = 1/2.$$

For the proof of the reduction factor we refer to exercise 7.8. □

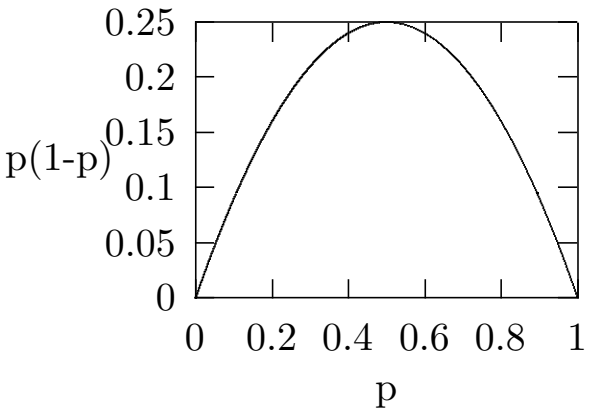


Figure 7.1: Influence of asymmetry on the yield of the Neumann Filter.

7.2 Calculation of Means - An Application for Functional Equations

7.2.1 Derivation of a suitable Speedup Formula

Task: Runtime comparsion of 3 computers

- Computer A: SUN-SPARC classic (to be compared with:)
- Computer B: PC Pentium 90 xxx
- Computer C: HP 9000/720

Example 7.8 Running time of Program 1

Computer	Time T_X	Speedup T_A/T_X
C_A	10.4 sec	1
C_B	8.1 sec	1.28
C_C	7.9 sec	1.32

Problem 1 *Result is not representative.*

Example 7.9 Running time of Program 2

Computer	Time T_X	Speedup T_A/T_X
C_A	2.7 sec	1
C_B	4.3 sec	0.63
C_C	2.6 sec	1.04

Solution 1 *Measure running times on a representative set of benchmarks (based on statistics of the applications of a typical user)*

Example 7.10 Benchmarks I_1, I_2, I_3

Computer	I_1	I_2	I_3	$\bar{T}$
C_A	1	2	100	34.3
C_B	2	4	47	17.7

- Speedup $\bar{T}_A/\bar{T}_B = 1.93$
- $\Rightarrow C_B$ is almost twice as fast as C_A?
- No, only for benchmark I_3!

Problem 2 *Speedup $S_1 = \bar{T}_A/\bar{T}_B$ is a relative measure, but in the previous example, S is determinated only by Benchmark I_3 (largest value).*

Definition 7.8 Let $x_1, \ldots, x_n \in \mathbb{R}$, then $A : \mathbb{R}^n \to \mathbb{R}$ with

$$A(x_1, \ldots, x_n) = \frac{1}{n} \sum_{k=1}^{n} x_k$$

is the Arithmetic Mean of $x_1, \ldots, x_n$.

Definition 7.9 Let $\alpha_1, \ldots, \alpha_n$ ($\beta_1, \ldots, \beta_n$) be the running times of Computer A (Computer B) on the Benchmarks $I_1, \ldots, I_n$. Then the Speedup S_1 is defined as:

$$S_1(C_A, C_B) = \frac{A(\alpha_1, \ldots, \alpha_n)}{A(\beta_1, \ldots, \beta_n)} = \frac{\sum_{k=1}^{n} \alpha_k}{\sum_{k=1}^{n} \beta_k}$$

Solution 2 *Calculate the sum of the ratios instead of the ratio of the sums!*

Definition 7.10

$$S_2(C_A, C_B) = A\left(\frac{\alpha_1}{\beta_1}, \ldots, \frac{\alpha_n}{\beta_n}\right) = \frac{1}{n} \sum_{k=1}^{n} \frac{\alpha_k}{\beta_k}$$

Application of S_2 on the previous example:

$$\begin{aligned} S_2(C_A, C_B) &= A(\frac{1}{2}, \frac{1}{2}, \frac{100}{47}) = \frac{\frac{1}{2} + \frac{1}{2} + \frac{100}{47}}{3} = 1.04 \\ S_2(C_B, C_A) &= A(2, 2, \frac{47}{100}) = \frac{2 + 2 + 0.47}{3} = 1.49 \end{aligned}$$

$\Rightarrow C_A$ faster than C_B, or C_B faster than C_A?

Problem 3 $S_2(C_A, C_B) \neq \dfrac{1}{S_2(C_B, C_A)}$

Example 7.11 Calculation of the Speedup

Computer	Runtime. of Benchm. I_1	Runtime. of Benchm. I_2
C_A	1	10
C_B	10	1

$$S_2(C_A, C_B) = S_2(C_B, C_A) = \left(\frac{1}{10} + 10\right) \cdot \frac{1}{2} = 5.05$$

Expected: $S_2 = 1$!

Conjecture: Geometric Mean solves the problem

Definition 7.11 $G : (\mathbb{R}\backslash\{0\})^n \to \mathbb{R}$ with

$$G(x_1, \ldots, x_n) = \sqrt[n]{x_1 \cdot \ldots \cdot x_n}$$

is the Geometric Mean of $x_1, \ldots, x_n$.

Definition 7.12

$$S_3(C_A, C_B) = G\left(\frac{\alpha_1}{\beta_1}, \ldots, \frac{\alpha_n}{\beta_n}\right) = \sqrt[n]{\prod_{k=1}^{n} \frac{\alpha_k}{\beta_k}}$$

is called User Speedup.

Remark: S_3 solves problems 3 !

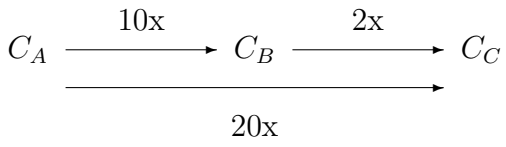
$$S_3(C_A, C_B) = \sqrt[n]{\prod_{k=1}^{n} \frac{\alpha_k}{\beta_k}} = \prod_{k=1}^{n} \frac{\alpha_k^{1/n}}{\beta_k^{1/n}} = \frac{1}{\prod_{k=1}^{n} \frac{\beta_k^{1/n}}{\alpha_k^{1/n}}} = \frac{1}{S_3(C_B, C_A)}$$

7.2.2 Requirements for a speedup function $M : \mathbb{R}_+^n \to \mathbb{R}_+$

A speedup function of relative quantities must fulfill the following functional equations:

1. $M(x, \ldots, x) = x$
2. $M(x_1, \ldots, x_n) \cdot M(y_1, \ldots, y_n) = M(x_1 y_1, \ldots, x_n y_n)$
3. $M(x_1, \ldots, x_k) = M(x_{\pi(1)}, \ldots, x_{\pi(k)})$ for each permutation π on $\{1, \ldots, k\}$

Explanation of requirement 2:



$$
\begin{aligned}
& S(C_A, C_B) \cdot S(C_B, C_C) = S(C_A, C_C) \\
\Leftrightarrow\ & M(\frac{\alpha_1}{\beta_1}, \ldots, \frac{\alpha_n}{\beta_n}) \cdot M(\frac{\beta_1}{\gamma_1}, \ldots, \frac{\beta_n}{\gamma_n}) = M(\frac{\alpha_1}{\gamma_1}, \ldots, \frac{\alpha_n}{\gamma_n}) \\
\Leftrightarrow\ & M(x_1, \ldots, x_n) \cdot M(y_1, \ldots, y_n) = M(x_1 y_1, \ldots, x_n y_n)
\end{aligned}
$$

Theorem 7.5 The Geometric Mean $G(x_1, \ldots, x_n)$ is the one and only function $M : \mathbb{R}_+^n \to \mathbb{R}_+$, which fulfills the requirements 1,2 and 3.

Proof:

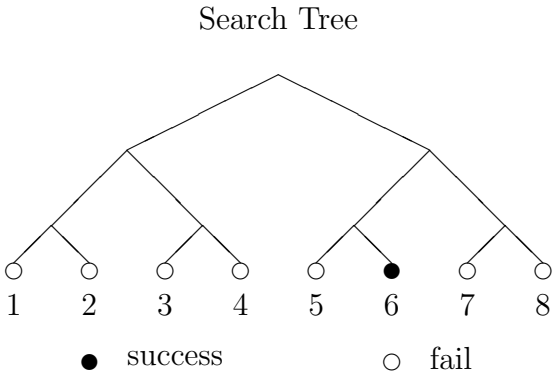
$$
\begin{aligned}
M(x_1, \ldots, x_n)^n &= M(x_1, \ldots, x_n) \cdot M(x_2, \ldots, x_n, x_1) \cdot \ldots \cdot M(x_n, x_1, \ldots, x_{n-1}) \\
&= M(x_1 \cdot \ldots \cdot x_n, \ldots, x_1 \cdot \ldots \cdot x_n) \\
&= x_1 \cdot \ldots \cdot x_n
\end{aligned}
$$

7.2.3 Application / Case Study: Randomized Depth-First Search

Randomized Algorithms

Definition 7.13 An algorithm A which gets in addition to its input I a sequence of random numbers is called randomized algorithm.

Note: In general the runtime of A (for fixed inputs I) depends on the random numbers.



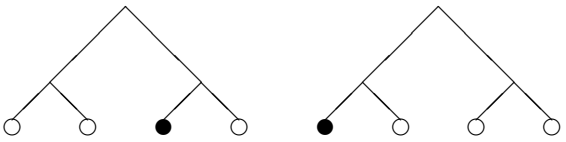

```
DEPTH-FIRST-SEARCH(Node, Goal)
If GoalReached(Node, Goal) Return("Solution found")
NewNodes = Successors(Node)
While NewNodes ≠ ∅
    Result = DEPTH-FIRST-SEARCH(First(NewNodes), Goal)
    If Result = "Solution found" Return("Solution found")
    NewNodes = Rest(NewNodes)
Return("No solution")
```

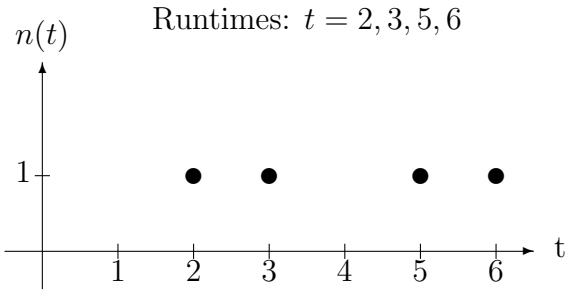
Figure 7.2: The algorithm for depth-first search. The function "First" returns the first element of a list, and "Rest" the rest of the list.

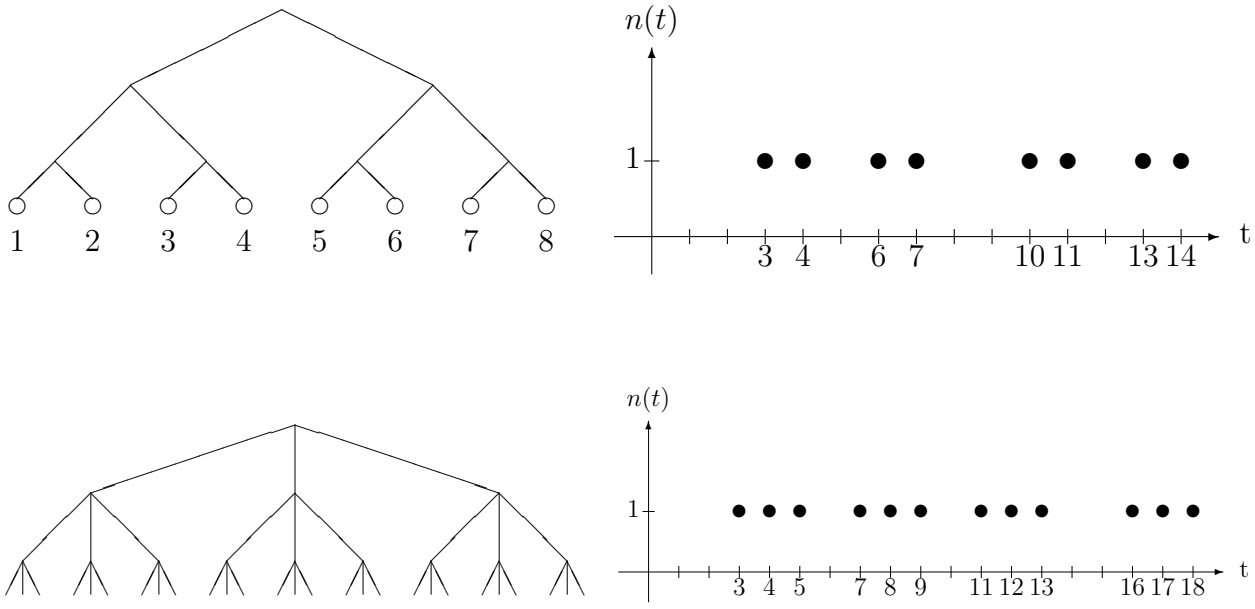
7.2.4 Depth-First Search

Depth-first-search searches the binary tree recursively until one solution was found.
Randomized Depth-first-search: random choice of left/right successor.
many different possible runtimes (runtime distribution) for fixed tree.

Example 7.12 4 different trees each with a solution:







7.2.5 How to measure speedup for such randomized algorithms?

$$\Rightarrow S_3(C_1, C_p) = G\left(\frac{\alpha_1}{\beta_1}, \ldots, \frac{\alpha_n}{\beta_n}\right) \textbf{?}$$

not meaningful since assignment $\alpha_i \leftrightarrow \beta_i$ does not exist!
but:

$$S_3(C_A, C_B) = G\left(\frac{\alpha_1}{\beta_1}, \ldots, \frac{\alpha_1}{\beta_m}; \frac{\alpha_2}{\beta_1}, \ldots, \frac{\alpha_2}{\beta_m}; \ldots; \frac{\alpha_n}{\beta_1}, \ldots, \frac{\alpha_n}{\beta_m}\right)$$

- All possible ratios are calculated.
- Proceeding as above, but requirement is meaningless.
- New axioms, thus different (more difficult) proof.

7.3 Exercises

Exercise 7.1 Define the term "random number generator" in analogy to the term "random bit generator". Instead of bits we now allow numbers from a finite set N.

Exercise 7.2 Can the the Kolmogorov complexity of a sequence S be measured in practice? Dioscuss this questions with:

a) Write pseudocode of an program that finds the shortest C-program that outputs the given sequence S. Based on the grammar of the language C this program generates all C-programs of length $1, 2, 3, \ldots$. Each generated C-program now is executed and the produced sequence compared with S.

b) Which problems appear with this program?

c) Modify the program such that it approximates the Kolmogorov complexity of a given sequence S.

Exercise 7.3

a) Why is the period of a Linear Congruential Generator bounded above by the value of the modulus m? How could the generator be modified (for fixed m), to increase the period significantly?

b) Experiment with generators of the form $x_n = (x_{n-1} + x_{n-2}) \bmod m$ and find cases for x_0 and m, where the period is longer than m.

c) Consider generators of the form $x_n = (x_{n-1} + x_{n-2} + x_{n-3}) \bmod m$ and find cases for x_0 and m, where the period is longer than m^2.

d) Analyse the generators of the form $x_n = (x_{n-1} + x_{n-2} + \ldots + x_0) \bmod m$ with $x_0 = 1$ on periodicity.

e) Prove by induction: If $x_1 = x_0 = 1$ and $x_n = 2 \cdot x_{n-1}$ for $n \geq 2$, then it also holds $x_n = 2^{n-1}$ for $n \geq 1$.

Exercise 7.4

a) Calculate the expected value and standard deviation of a true binary random variable.

b) Draw the density function of a sum of 10, 100, 1000, 10 000 good random bits. Use the built-in Mathematica function `Random` or the Octave function `rand`. Then determine for each of the sums the sample standard deviation.

Exercise 7.5

a) Implement the mentioned linear congruential generator of the form $x_n = (ax_{n-1} + b) \bmod m$ with $a = 7141$, $b = 54773$ and $m = 259200$ in a programming language of your choice.

b) Test this generator on symmetry and periodicity.

c) Repeat the test after applying the Neumann Filter.

Exercise 7.6

a) Show that the bit sequence `11011011010101010101` passes the symmetry test.

b) Would you accept this sequence as a random bit sequence? Why?

c) Why is the symmetry test not sufficient to test the quality of a random number generator?

d) Suggest different randomness tests and apply them to the sequence.

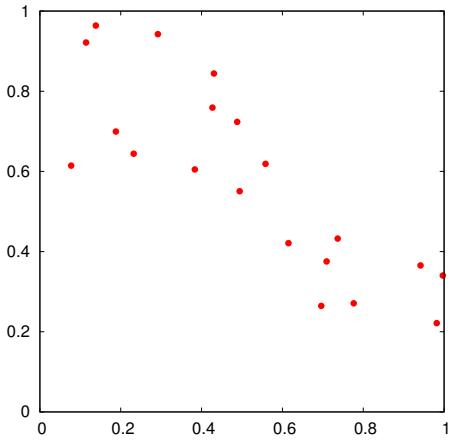
Exercise 7.7 What can you say theoretically about the period of the BBS generator?

Exercise 7.8

Show that the length of a finite bit sequence $(a_n)_{n \in \{0,1\}}$ with independent bits gets shortened by applying the Neumann-filter by approximately the factor $p(1-p)$, if the relative proportion of ones is equal to p. (Theorem 7.4)

7.4 Principal Component Analysis (PCA)

In multideminsional data sets quite often some variables are correlated or even redundant, as shown in the 2-dim. scatterplot beside. We may then for example reduce the dimensionality of the data. We follow chapter 12 in [7].



Given is a set of data points $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$, each $\boldsymbol{x}_n$ being a vector of D dimensions. We want to project the points into a lower dimensional space with $M < D$ dimensions. We start with looking for the direction in D-dim. space with highest variance of the data. Let $\boldsymbol{u}_1$ a unit vector in this direction, i.e. $\boldsymbol{u}_1^T \boldsymbol{u}_1 = 1$. We project the data points $\boldsymbol{x}_n$ onto this direction yielding the scalar value $\boldsymbol{u}_1^T \boldsymbol{x}_n$. The mean of the projected data is

$$\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{u}_1^T \boldsymbol{x}_n = \boldsymbol{u}_1^T \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n = \boldsymbol{u}_1^T \bar{\boldsymbol{x}}$$

and their variance

$$\frac{1}{N-1} \sum_{n=1}^{N} (\boldsymbol{u}_1^T \boldsymbol{x}_n - \boldsymbol{u}_1^T \bar{\boldsymbol{x}})^2 = \boldsymbol{u}_1^T \boldsymbol{S} \boldsymbol{u}_1$$

To see this, the definition of the covariance of two scalar variables x_i and x_j is

$$\boldsymbol{S}_{ij} = \frac{1}{N-1} \sum_{n=1}^{N} (x_{ni} - \bar{x}_i)(x_{nj} - \bar{x}_j)$$

where x_{ni} is the i-th component of the n-th data sample. The covariance matrix is

$$\boldsymbol{S} = \frac{1}{N-1} \sum_{n=1}^{N} (\boldsymbol{x}_n - \bar{\boldsymbol{x}})(\boldsymbol{x}_n - \bar{\boldsymbol{x}})^T$$

Thus

$$\begin{aligned} \boldsymbol{u}_1^T \boldsymbol{S} \boldsymbol{u}_1 &= \frac{1}{N-1} \sum_{n=1}^{N} \boldsymbol{u}_1^T (\boldsymbol{x}_n - \bar{\boldsymbol{x}})(\boldsymbol{x}_n - \bar{\boldsymbol{x}})^T \boldsymbol{u}_1 = \frac{1}{N-1} \sum_{n=1}^{N} \boldsymbol{u}_1^T (\boldsymbol{x}_n - \bar{\boldsymbol{x}}) \boldsymbol{u}_1^T (\boldsymbol{x}_n - \bar{\boldsymbol{x}}) \\ &= \frac{1}{N-1} \sum_{n=1}^{N} (\boldsymbol{u}_1^T \boldsymbol{x}_n - \boldsymbol{u}_1^T \bar{\boldsymbol{x}})(\boldsymbol{u}_1^T \boldsymbol{x}_n - \boldsymbol{u}_1^T \bar{\boldsymbol{x}}) = \frac{1}{N-1} \sum_{n=1}^{N} (\boldsymbol{u}_1^T \boldsymbol{x}_n - \boldsymbol{u}_1^T \bar{\boldsymbol{x}})^2 \end{aligned}$$

In order to find the vector $\boldsymbol{u}_1$ which produces maximum variance $\boldsymbol{u}_1^T \boldsymbol{S} \boldsymbol{u}_1$, we will maximize this quantity by deriving it w.r.t. $\boldsymbol{u}_1$. To prevent $\|\boldsymbol{u}_1\| \to \infty$ we have to use the normalization condition $\boldsymbol{u}_1^T \boldsymbol{u}_1 = 1$ as a constraint, which yields the Lagrangian

$$L = \boldsymbol{u}_1^T \boldsymbol{S} \boldsymbol{u}_1 + \lambda_1 (1 - \boldsymbol{u}_1^T \boldsymbol{u}_1).$$

and the necessary condition for a maximum is

$$\frac{\partial L}{\partial \boldsymbol{u}_1} = 2\boldsymbol{S}\boldsymbol{u}_1 - 2\lambda_1 \boldsymbol{u}_1 = 0,$$

yielding

$$\boldsymbol{S}\boldsymbol{u}_1 = \lambda_1 \boldsymbol{u}_1,$$

which is the eigenvalue equation for the covariance matrix $\boldsymbol{S}$. Obviously, if we choose λ_1 as the largest eigenvalue, we will obtain highest variance, i.e.

$$\boldsymbol{u}_1^T \boldsymbol{S}\boldsymbol{u}_1 = \boldsymbol{u}_1^T \lambda_1 \boldsymbol{u}_1 = \lambda_1.$$

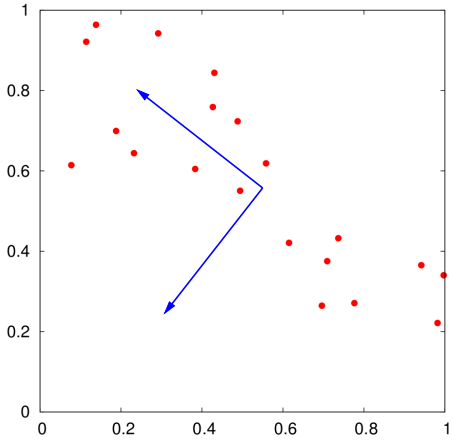
From this we now can conclude

Theorem 7.6 The variance of the data points is maximal in the direction of the eigenvector $\boldsymbol{u}_1$ to the largest eigenvalue of the covariance matrix S. This maximal eigenvector is called the **principal component**.

Application to the above data points yields the two eigenvectors

$$\boldsymbol{u}_1 = \begin{pmatrix} -0.788 \\ 0.615 \end{pmatrix} \qquad \boldsymbol{u}_2 = \begin{pmatrix} -0.615 \\ -0.788 \end{pmatrix}$$

with the corresponding eigenvalues $\lambda_1 = 0.128$ and $\lambda_2 = 0.011$. The graph shows that the principal component $\boldsymbol{u}_1$ points in the direction of highest variance.



After finding the direction with highest variance, we partition the D-dimensional space into $\boldsymbol{u}_1$ and its orthogonal complement. In the resulting $(D-1)$-dimensional space we again determine the principal component. This procedure will be repeated until we have M principal components. The simple result is

Theorem 7.7 The eigenvectors $\boldsymbol{u}_1 \ldots \boldsymbol{u}_M$ to the M largest eigenvalues of the $\boldsymbol{S}$ determine the M orthogonal directions of highest variance of the data set $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$.

Proof: by induction:
For $M = 1$ we refer to theorem 7.6. Now assume, the M directions with highest variance are already determined. Since $\boldsymbol{u}_{M+1}$ has to be orthogonal to $\boldsymbol{u}_1 \ldots \boldsymbol{u}_M$, we will require the constraints

$$\boldsymbol{u}_{M+1}^T \boldsymbol{u}_1 = \boldsymbol{u}_{M+1}^T \boldsymbol{u}_2 = \ldots = \boldsymbol{u}_{M+1}^T \boldsymbol{u}_M = 0.$$

Similarly to the above procedure we will determine $\boldsymbol{u}_{M+1}$ by maximizing the variance of the data in the remaining space. As above, the variance of the data in the direction $\boldsymbol{u}_{M+1}$ is $\boldsymbol{u}_{M+1}^T \boldsymbol{S}\boldsymbol{u}_{M+1}$. Together with the above M orthogonality constraints and the normality constraint $\boldsymbol{u}_{M+1}^T \boldsymbol{u}_{M+1} = 1$ we have to find a maximum of the new Lagrangian

$$L = \boldsymbol{u}_{M+1}^T \boldsymbol{S}\boldsymbol{u}_{M+1} + \lambda_{M+1}(1 - \boldsymbol{u}_{M+1}^T \boldsymbol{u}_{M+1}) + \sum_{i=1}^{M} \eta_i \boldsymbol{u}_{M+1}^T \boldsymbol{u}_i$$

with respect to $\boldsymbol{u}_{M+1}$. It turns out (exercise 7.11) that the solution $\boldsymbol{u}_{M+1}$ has to fulfill

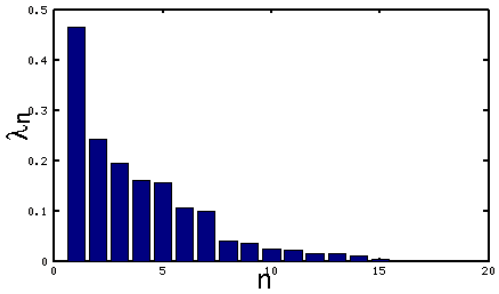
$$\boldsymbol{S}\boldsymbol{u}_{M+1} = \lambda_{M+1}\boldsymbol{u}_{M+1}$$

i.e. it is again an eigenvector of S. Obviously we have to select among the $D - M$ not yet selected eigenvectors the one with the largest eigenvalue. □

We now apply PCA to the Lexmed data from example 4.4 in section 4.3. Some raw data samples are:

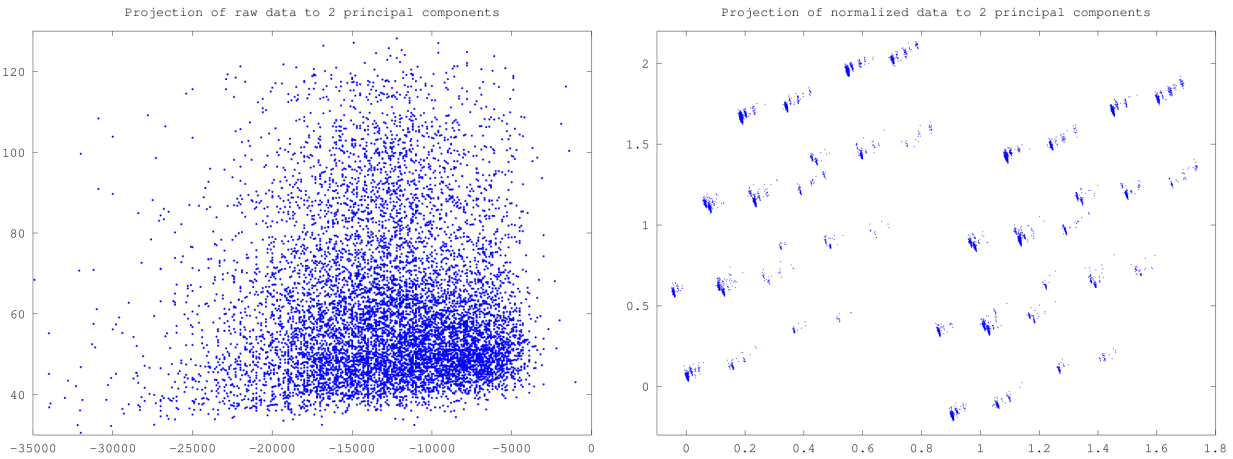
19	1	0	0	1	0	1	0	1	1	0	362	378	13400	0
13	1	0	0	1	0	1	0	1	1	1	383	385	18100	0
18	2	0	0	1	1	0	0	0	0	0	362	370	9300	0
73	2	1	0	1	1	1	0	1	1	1	376	380	13600	1
36	1	0	0	1	0	1	0	1	1	0	372	382	11300	0
18	2	0	0	1	0	1	0	1	0	0	366	378	13000	0
19	2	0	0	1	0	0	0	0	0	1	372	378	6400	0
62	1	0	0	1	0	1	0	1	1	0	376	390	22000	0

After normalization of the data to the interval $[0, 1]$ we obtain the eigenvalues:



0.47 0.24 0.19 0.16 0.16 0.11 0.10 0.039 0.036 0.023 0.023 0.016 0.016 0.01 0.004

Due to the step after the 7-th largest eigenvalue, a transformation of the data to the 7-dimensional space spanned by the eigenvectors of the 7 largest eigenvalues may be considered. If for visualization we plot the data to the two principal components (eigenvectors to the two largest eigenvalues), we get for the raw data the left and for the normalized data the right diagram:



The corresponding two eigenvectors for the raw data are:

$$(-1, 0.2, -0.03, -0.02, -0.003, -0.06, -0.3, -0.04, -0.2, -0.2, -0.1, -3, -4, \mathbf{-10000}, -0.004) \cdot 10^{-4}$$
$$(\mathbf{100}, -0.10, 0.16, 0.05, -0.04, 0.17, 0.27, 0.06, 0.09, 0.08, -0.03, \mathbf{3.34}, \mathbf{5.66}, -0.02, 0.17) \cdot 10^{-2}$$

The first vector projects on the leukocyte value and the second on a combination of the age and the fever values. Why?

7.4.1 Applications of PCA

- Dimensionality reduction
- Data compression
- Extraction of features from pixel images
- Data visualization

An Image compression example[1]

- 5000 gray-scale images with $32 \times 32 = 1024$ pixels each.
- Application of PCA with 100 principal components.
- I.e. projection on 100-dimensional subspace.
- Transformation of compressed images back into original space.

100 Images[2]

[1] From Andrew Ng's excellent lecture "Machine Learning": ml-class.org.
[2] From ml-class.org.

Original and Recovered Images[3]

Bill Clinton[4]

36 Principal components[5]

Scalability

- Would this work with 1 Megapixel images also?
- No! Why?

[3] From ml-class.org.
[4] From ml-class.org.
[5] From ml-class.org.

- $D = 10^6$ dimensional space!
- 5000 images = 5000 data points in 10^6-dimensional space.
- $N = 5000$ data points define a 4999-dimensional hyperplane.
- Thus we need: $M \ll N - 1 = 4999$.
- Otherwise: Underdetermined problem!
- Compression by a factor of $10^6/5000 = 200$.

Back to Andrew Ng's Example

- $D = 1024$.
- 5000 images = 5000 data points in 1024-dimensional space.
- 5000 points in $M = 100$ dim. space.
- $M = 100 \ll 4999 = N - 1$.
- Structure of data can be conserved.

7.5 Estimators

Estimators & Properties
This chapter covers the estimation of unknown parameters. Most often a parameterized distribution given, but with unknown true parameters. The goal is to estimate these parameters with the help of samples $\boldsymbol{x}$ (from the true distribution). We collect all parameters of interest in the variable $\boldsymbol{\gamma}$ First we start with the definition of an estimator followed by some easy examples and come back to this later when we talk about maximum likelihood estimators.
An estimator T_γ is used to infer the value of an unknown parameter $\boldsymbol{\gamma}$ in a statistical model. It is a *function* defined as:

$$T_\gamma : \mathcal{X} \mapsto \Gamma$$

where $\mathcal{X}$ is a sample space with elements $\boldsymbol{x} := \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\} \in \mathcal{X}$
Normally we will not be able to estimate the true parameter exactly and so we have to define some properties that assures a certain quality of the estimations found with the help of T. The true parameter is unknown and so we have to look for other reasonable criteria. For example the expected value of the estimator should be the parameter to estimate.
Desireable properies are:

- *unbiasedness:* $\mathbb{E}[T_\gamma] = \gamma$
- *minimum variance:* An unbiased estimator T_γ^* has minimum variance if

  $$\text{var}[T_\gamma^*] \leq \text{var}[T_\gamma]$$

  for all unbiased estimators T.

Sample Mean & Sample Variance
We can formulate the calculation of the sample mean and variance in terms of estimators: Let the x_j be samples from a distribution with mean μ and variance σ^2

- The function $\bar{x} : \mathbb{R}^n \mapsto \mathbb{R}$

$$\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_j$$

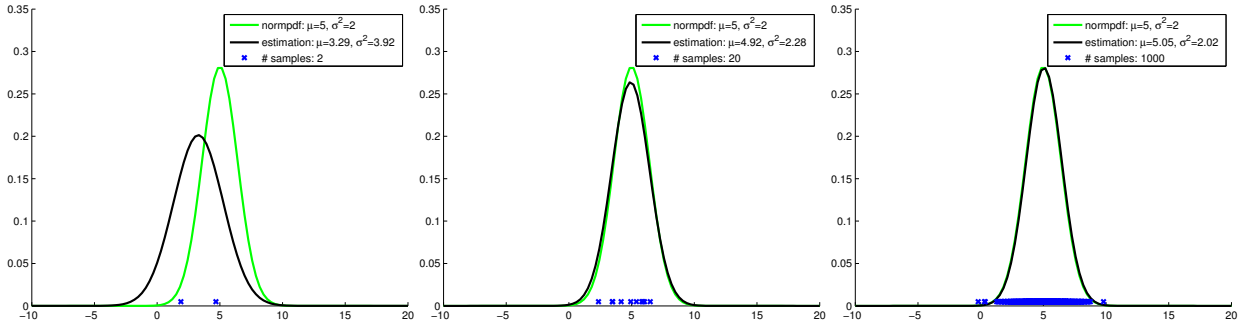
  is called the *sample mean*

- The function $s^2 : \mathbb{R}^n \mapsto \mathbb{R}$

$$s^2 = \frac{1}{n-1}\sum_{j=1}^{n}(x_j - \bar{x})^2$$

  is called the *sample variance*

Example: Sample Mean & Sample Variance
Sampling from a Gaussian distribution with mean $\mu = 5$ and variance $\sigma^2 = 2$. The black line is a plot of the true Gaussian and the green line is a Gaussian were the mean and the variance is calculated with $\bar{x}$ and s^2 respectively.



As expected the estimation becomes better the more samples are used.

Unbiasedness of Sample Mean
As mentioned before there are some properties we want for an estimator to hold. We are going to proof the unbiasedness and leave the proof for the minimum variance criterion as an exercise to the reader.
Proof:

$$\mathbb{E}[\bar{x}] = \frac{1}{n}\sum_{j=1}^{n}\mathbb{E}[x_j] = \mu$$

□

Unbiasedness of Sample Variance

Proof: We can rewrite s^2 as:

$$s^2 = \frac{1}{n-1}\sum_{j=1}^{n}(x_j - \mu)^2 - \frac{n}{n-1}(\bar{x} - \mu)^2 \quad \text{then}$$

$$\begin{aligned}\mathbb{E}[s^2] &= \frac{1}{n-1}\sum_{j=1}^{n}\mathbb{E}[(x_j - \mu)^2] - \frac{n}{n-1}\mathbb{E}[(\bar{x} - \mu)^2] \\ &= \frac{1}{n-1}\sum_{j=1}^{n}\mathrm{var}[x_j] - \frac{n}{n-1}\mathrm{var}[\bar{x}] \\ &= \frac{n}{n-1}\sigma^2 - \frac{n}{n-1}\frac{\sigma^2}{n} = \frac{n-1}{n-1}\sigma^2 = \sigma^2\end{aligned}$$

□

Sample Mean & Sample Variance (variances)
We can not only calculate the expected value of estimators, but also their variance. It is an exercise to proof the following:
The variance $var\bar{x}$ of the estimator $\bar{x}$ is given by

$$\mathrm{var}[\bar{x}] = \frac{1}{n}\sigma^2$$

$$\mathrm{var}[s^2] = \frac{2}{n-1}\sigma^4$$

Expectations and Covariances
The *expectation* of some function $f(x)$ under a probability distribution $p(x)$ is given by

$$\mathbb{E}[f] = \int p(x)f(x)\,dx$$

and the *variance* of $f(x)$ is defined by

$$\begin{aligned}\mathrm{var}[f] &= \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2] \\ &= \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2\end{aligned}$$

For two random variables x and y, the *covariance* is defined by

$$\begin{aligned}\mathrm{cov}[x, y] &= \mathbb{E}[(x - \mathbb{E}[x])(y - \mathbb{E}[y])] \\ &= \mathbb{E}[xy] - \mathbb{E}[x]\mathbb{E}[y]\end{aligned}$$

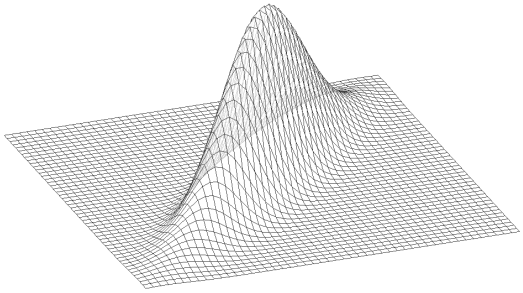
Covariance and Independence
Remember, that for two independent variables x and y we have $p(x, y) = p(x) \cdot p(y)$. Thus

$$\begin{aligned}\mathbb{E}[xy] &= \int\int p(x, y)xy\,dx\,dy = \int\int p(x)p(y)xy\,dx\,dy \\ &= \int p(y)y\,dy \int p(x)x\,dx = \mathbb{E}[x]\mathbb{E}[y]\end{aligned}$$

and we get for independent variables

$$\mathrm{cov}[x, y] = 0$$

7.6 Gaussian Distributions

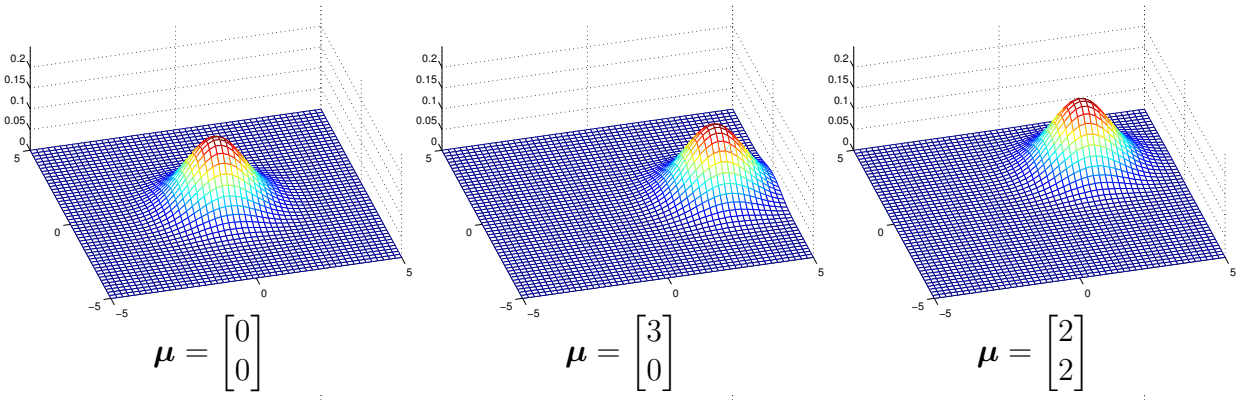


Definition 7.14 A **Gaussian distribution** is fully specified by a D-dimensional **mean vector** $\boldsymbol{\mu}$ and $D \times D$ **covariance matrix** $\boldsymbol{\Sigma}$ with the density function

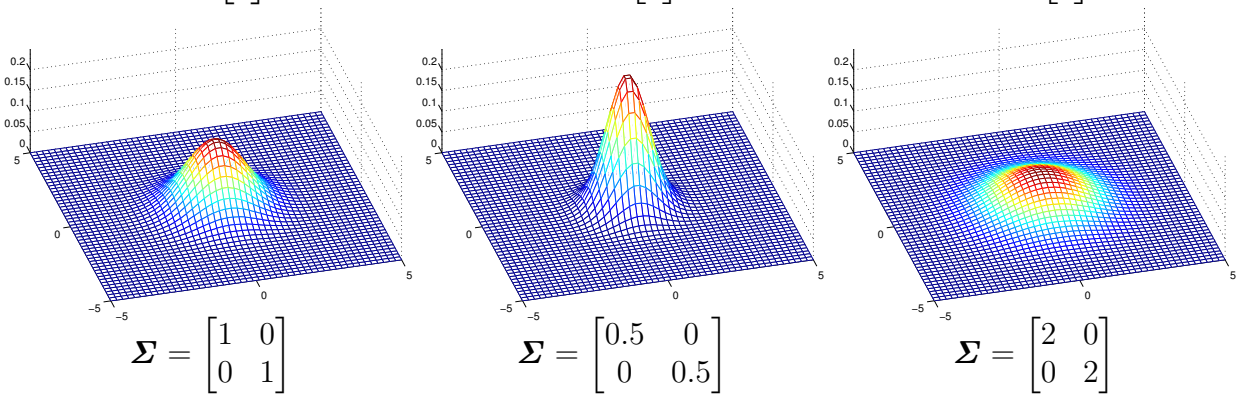
$$p(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{D}{2}} \, |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

- That $\boldsymbol{\mu}$ is the mean and $\boldsymbol{\Sigma}$ the covariance matrix of the normal distribution, has to be proven!
- If the variables $x1, \ldots, x_D$ are all independent, then $\boldsymbol{\Sigma}$ is diagonal! Why?

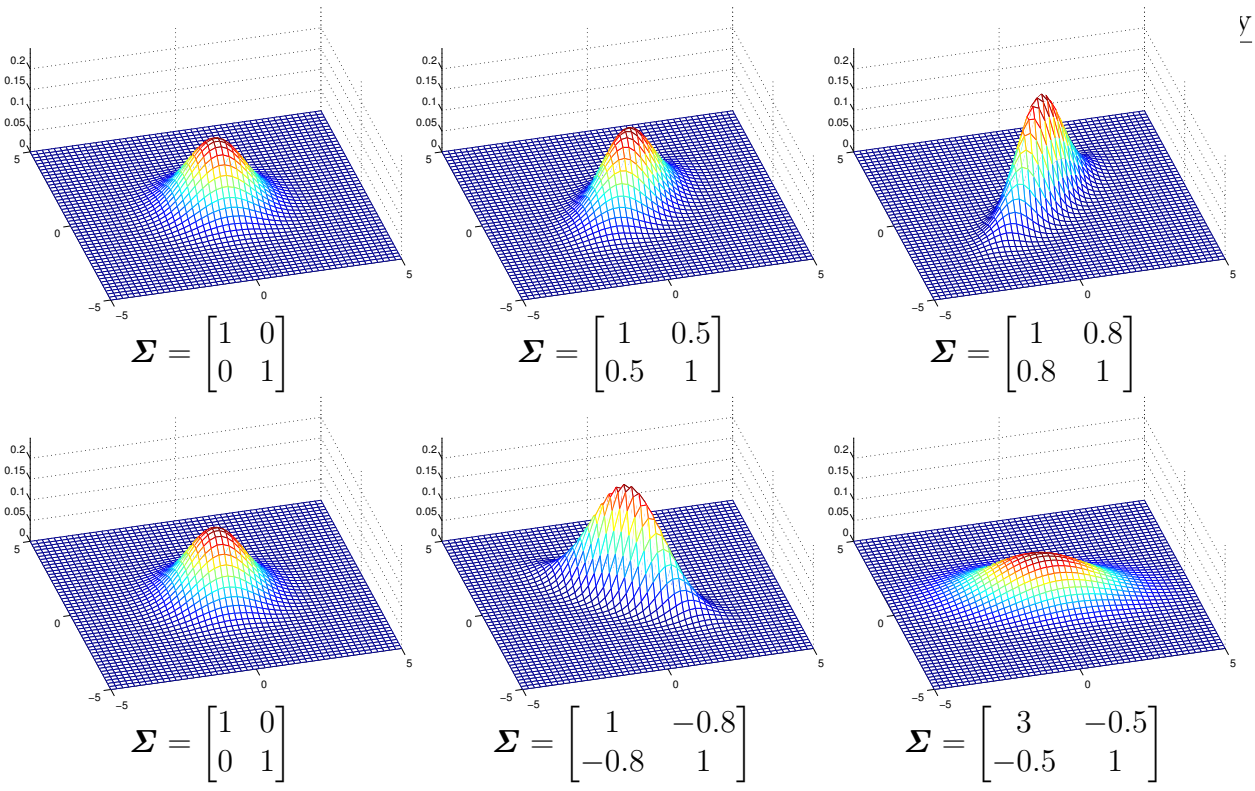
Examples: Mean Vector and Covariance Matrix



$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\quad$ $\boldsymbol{\mu} = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$ $\quad$ $\boldsymbol{\mu} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$



$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $\quad$ $\boldsymbol{\Sigma} = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$ $\quad$ $\boldsymbol{\Sigma} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$



Covariance Matrix Properties
The covariance matrix $\boldsymbol{\Sigma}$ is *symmetric*

- $\boldsymbol{\Sigma}$ is invertible and $\boldsymbol{\Sigma}^{-1}$ is symmetric
- All eigenvalues are real
- All eigenvectors are orthogonal
- Eigenvectors point in the direction of principal axes of the ellipsoid.

The covariance matrix $\boldsymbol{\Sigma}$ is *positive definite*

- $\implies x^T \boldsymbol{\Sigma} x > 0 \quad \forall x \in \mathbb{R}^n \setminus \{0\}$
- All eigenvalues are positive
- $\boldsymbol{\Sigma}$ is invertible and $\boldsymbol{\Sigma}^{-1}$ is positive definite

Diagonalization of the Covariance Matrix
Let $\boldsymbol{u}_1 \dots \boldsymbol{u}_D$ the eigenvectors of $\boldsymbol{\Sigma}$. Then the transformation $\boldsymbol{x} \mapsto \boldsymbol{y}$ with

$$y_i = \boldsymbol{u}_i^T (\boldsymbol{x} - \boldsymbol{\mu})$$

makes all variables y_i pairwise independent with diagonal covariance matrix $\boldsymbol{\Sigma'}$ and zero mean.

Product of Gaussian Distributions

The *product* of two Gaussian distributions is given by

$$\mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a) \cdot \mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b) = z_c \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$$

where

$$\boldsymbol{\mu}_c = \boldsymbol{\Sigma}_c \left(\boldsymbol{\Sigma}_a^{-1}\boldsymbol{\mu}_a + \boldsymbol{\Sigma}_b^{-1}\boldsymbol{\mu}_b\right) \quad \text{and} \quad \boldsymbol{\Sigma}_c = \left(\boldsymbol{\Sigma}_a^{-1} + \boldsymbol{\Sigma}_b^{-1}\right)^{-1}$$

Marginal Gaussian Distribution

Recall, in general, the *marginal distribution* for a joint random variable $p(\boldsymbol{x}, \boldsymbol{y})$ is given by

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}, \boldsymbol{y})\, d\boldsymbol{y}$$

Given a joint distribution

$$p(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{N}\left(\begin{bmatrix} a \\ b \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix}\right)$$

the *marginal Gaussian distribution* is simply given by

$$p(\boldsymbol{x}) = \mathcal{N}(a, A)$$

Conditional Gaussian Distribution

The *conditional distribution*, in general, is given by

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{y})}$$

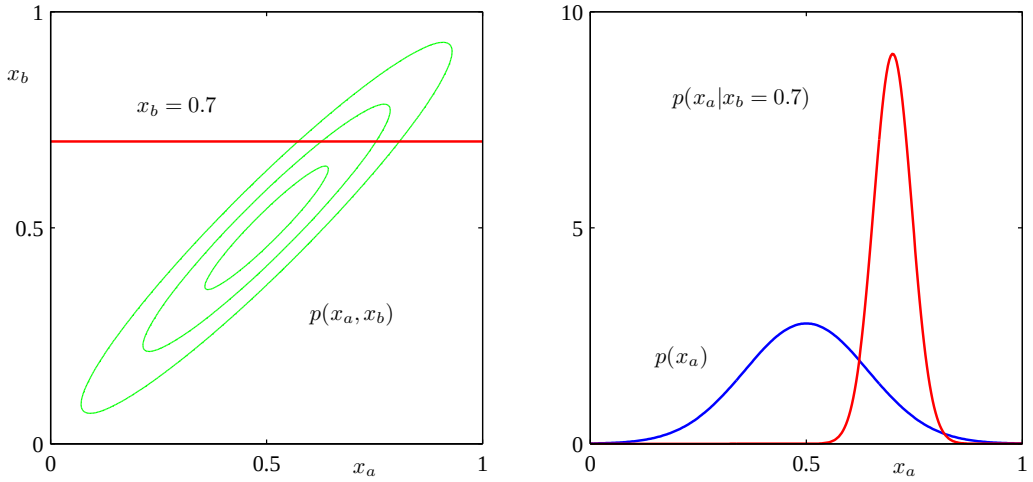
Given a joint distribution

$$p(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{N}\left(\begin{bmatrix} a \\ b \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix}\right)$$

the *conditional Gaussian distribution* is given by

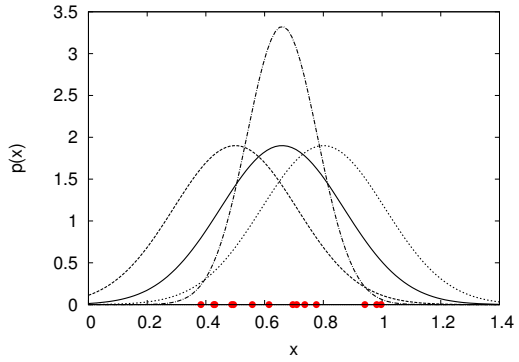
$$p(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}\left(b + CA^{-1}(\boldsymbol{x} - a), B - CA^{-1}C^T\right)$$

Marginal & Conditional Gaussian Distribution



7.7 Maximum Likelihood

Which one of the following normal distributions maximizes the probability for independently observing the given data points?



Maximum Likelihood for Gaussian distributions
Let $x_1, \ldots, x_n$, be i.i.d (independently and identically distributed) according to $\mathcal{N}(\mu, \sigma^2)$ and $x := \{x_1, \cdots, x_n\}$, then the joint density is:

$$L_x(\mu, \sigma^2) = p(x|\mu, \sigma^2) = \prod_{j=1}^{n} p(x_j|\mu, \sigma^2)$$
$$= \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{(x_j - \mu)^2}{\sigma^2}\right)$$

The log likelihood function is given by

$$\ln L_x(\mu, \sigma^2) = -\frac{1}{2\sigma^2}\sum_{j=1}^{n}(x_j - \mu)^2 - \frac{n}{2}\ln\sigma^2 - \frac{n}{2}\ln(2\pi)$$

Maximizing $\ln L_x(\mu, \sigma^2)$ with respect to μ, we obtain the maximum likelihood solution given by

$$\mu_{ML} = \frac{1}{n}\sum_{j=1}^{n} x_j$$

what we recognize as the sample mean.

Maximizing $\ln L_x(\mu, \sigma^2)$ with respect to σ^2 leads to

$$\sigma^2_{ML} = \frac{1}{n}\sum_{j=1}^{n}(x_j - \mu_{ML})^2$$

which is different from the sample variance and therefore *biased*.

The Likelihood Function
The "Maximum likelihood estimator" is a mapping from samples to parameter values for which the likelihood function becomes a maximum. The formal definition of a likelihood function is:
Let Γ be the parameter space and p_γ the joint density w.r.t. γ, then the *likelihood function* L_x is defined as:

$$L_x : \Gamma \mapsto \mathbb{R}_+$$

$$L_x(\boldsymbol{\gamma}) := p(x|\boldsymbol{\gamma}) \quad \forall \boldsymbol{\gamma} \in \Gamma \quad \forall x := \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\} \in \mathcal{X}$$

The likelihood function is a function of the parameters $\boldsymbol{\gamma}$, where as the joint density is a function of x !
The difference is, that we normally have a probability distribution $p_{\boldsymbol{\gamma}}(x)$ with parameters $\boldsymbol{\gamma}$ given and we evaluate this function at various inputs x. We now assume, that we do know the parameters, but that have given some samples x from the true underlying distribution and our goal is to estimate these parameters. We do this by searching for some parameter values that maximize the likelihood function (and so maximize also the probability density).

Maximum Likelihood
We call the estimator T *maximum likelihood estimator* (ML estimator) if

$$T : \mathcal{X} \mapsto \Gamma \quad \text{with}$$

$$L_x(T(x)) = \sup_{\gamma \in \Gamma} L_x(\gamma), \quad \forall x := \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\} \in \mathcal{X}$$

In many cases it is possible to derive the likelihood function and set its derivative with respect to the parameters to zero. Sometimes it is also easier to maximize the so called *log likelihood* $l_x(\gamma) := \ln L_x(\gamma)$

Bernoulli Distribution
Outcome is either a "success" or "failure" (e.g. coin flipping with heads = 1 and tails = 0)

$$p_\mu(x = 1) = \mu$$
$$p_\mu(x = 0) = 1 - \mu$$

Bernoulli Distribution:

$$\text{Bern}_\mu(x) = \mu^x (1 - \mu)^{1-x}$$
$$\mathbb{E}[x] = \mu$$
$$\text{var}[x] = \mu(1 - \mu)$$

Example: ML for Bernoulli distributions
Let $x_j, j \in N_n$, be i.i.d according to $\text{Bern}_\mu(x_j)$ with $p(x_j = 1) = \mu$ and $p(x_j = 0) = 1 - \mu$. Then the joint probability is given by

$$p(x|\mu) = \mu^{\sum x_j} (1 - \mu)^{n - \sum x_j}, \quad x = (x_1, \cdots, x_n) \in \{0, 1\}^n$$

Solving the equation:

$$\frac{\partial}{\partial \mu} \ln L_x(\mu) = \frac{1}{\mu} \sum_{j=1}^{n} x_j - \frac{1}{1 - \mu}(n - \sum_{j=1}^{n} x_j) = 0$$

leads to

$$\mu_{ML} = \frac{1}{n} \sum_{j=1}^{n} x_j$$

7.8 Linear Regression

Maximum Likelihood Linear Regression
Assumption:

$$y_i = \boldsymbol{a}^T \boldsymbol{f}(x_i) + \varepsilon_i$$

where ε_i are i.i.d according to $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$

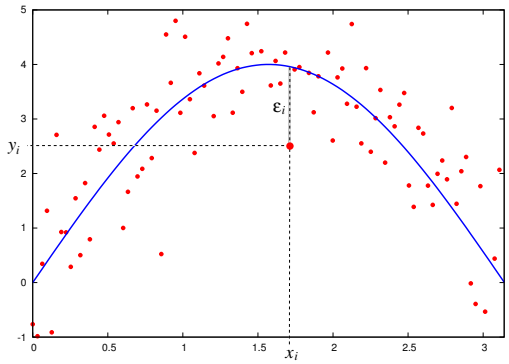


Figure 7.3: Example of sample points drawn from a function $a_1 \sin x + a_2 \cos x$ with added gaussian noise ε_i.

$$p(\varepsilon_i; \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\varepsilon_i)^2}{2\sigma^2}\right)$$

This *implies* that

$$p(y_i | x_i; \boldsymbol{a}, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \boldsymbol{a}^T \boldsymbol{f}(x_i))^2}{2\sigma^2}\right)$$

ML Linear Regression | Likelihood Function
Given $\boldsymbol{X}$ (the design matrix, which contains all the x_i's) and $\boldsymbol{y}$ (containing all the y_i's)

$$p(\boldsymbol{y} | X; \boldsymbol{a}, \sigma^2) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_j - \boldsymbol{a}^T \boldsymbol{f}(x_j))^2}{2\sigma^2}\right)$$

$$\ln p(\boldsymbol{y} | X; \boldsymbol{a}, \sigma^2) = -\frac{1}{\sigma^2} \frac{1}{2} \sum_{j=1}^{n} (y_j - \boldsymbol{a}^T \boldsymbol{f}(x_j))^2 - \frac{n}{2} \ln \sigma^2 - \frac{n}{2} \ln(2\pi)$$

Note: maximizing $\ln p(y | X; a, \sigma^2)$ w.r.t. a is the same as minimizing

$$\frac{1}{2} \sum_{i=1}^{n} (y_i - \boldsymbol{a}^T \boldsymbol{f}(x_i))^2$$

ML linear regression = Least squares solution!

ML Linear Regression | Determining $\boldsymbol{a}_{ML}$

$$\nabla_a \ln p(\boldsymbol{y}|X;\boldsymbol{a},\sigma^2) = \frac{1}{\sigma^2}\sum_{j=1}^{n}(y_j - \boldsymbol{a}^T\boldsymbol{f}(x_j))\boldsymbol{f}(x_j)$$

setting this to zero

$$\begin{aligned} 0 &= \sum_{j=1}^{n} y_j\boldsymbol{f}(x_j) - \sum_{j=1}^{n}(\boldsymbol{a}^T\boldsymbol{f}(x_j))\boldsymbol{f}(x_j) \\ &= \sum_{j=1}^{n} y_j\boldsymbol{f}(x_j) - \sum_{j=1}^{n}(\boldsymbol{f}(x_j)\boldsymbol{f}(x_j)^T)\boldsymbol{a} \\ &= \underbrace{\sum_{j=1}^{n} y_j\boldsymbol{f}(x_j)}_{\boldsymbol{F}^T\boldsymbol{y}} - \underbrace{\sum_{j=1}^{n}(\boldsymbol{f}(x_j)\boldsymbol{f}(x_j)^T)\boldsymbol{a}}_{\boldsymbol{F}^T\boldsymbol{F}\boldsymbol{a}} \end{aligned}$$

ML Linear Regression | Determining $\boldsymbol{a}_{ML}$

$$0 = \underbrace{\sum_{j=1}^{n} y_j\boldsymbol{f}(x_j)}_{\boldsymbol{F}^T\boldsymbol{y}} - \underbrace{\sum_{j=1}^{n}(\boldsymbol{f}(x_j)\boldsymbol{f}(x_j)^T)\boldsymbol{a}}_{\boldsymbol{F}^T\boldsymbol{F}\boldsymbol{a}}$$

with

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{f}(x_1)^T \\ \boldsymbol{f}(x_2)^T \\ \vdots \\ \boldsymbol{f}(x_n)^T \end{bmatrix}$$

Remember: The matrix $\boldsymbol{F}$ is equal to the matrix $\boldsymbol{M}$ we know from section 6.3 on least squares.

ML Linear Regression | Determining $\boldsymbol{a}_{ML}$
We see, that $\boldsymbol{a}_{ML}$ is given by

$$\boldsymbol{a}_{ML} = (\boldsymbol{F}^T\boldsymbol{F})^{-1}\boldsymbol{F}^T \cdot \boldsymbol{y}$$

Furthermore we notice, that *maximizing the likelihood* (under Gaussian noise assumption) is *equivalent to solving least squares!*

ML Linear Regression | Determining σ^2
Approach:

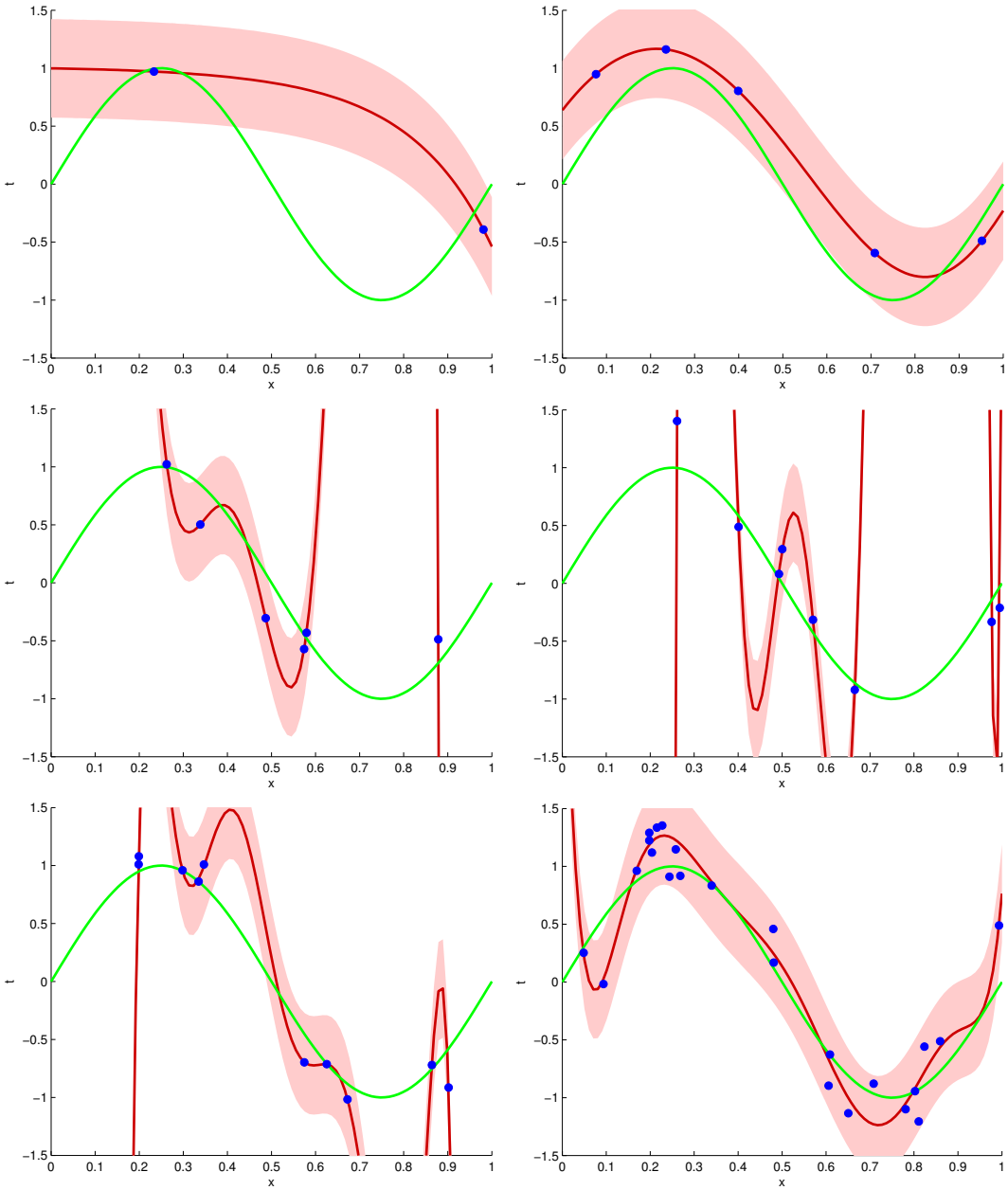
- determined the ML solution of the weights denoted by $\boldsymbol{a}_{ML}$
- subsequently use $\boldsymbol{a}_{ML}$ to find σ^2_{ML} by

$$\sigma^2_{ML} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \boldsymbol{a}^T_{ML}\boldsymbol{f}(x_i))^2$$

ML Linear Regression | Predictive Distribution
The probabilistic model we have now, leads us to the *predictive distribution*. For some new *prediction input* values x_*, the *prediction output* y_* is distributed according to

$$p(y_*|x_*; \boldsymbol{a}_{ML}, \sigma^2_{ML}) = \mathcal{N}\left(\boldsymbol{a}^T_{ML}\boldsymbol{f}(x_*), \sigma^2_{ML}\right)$$

Fitting a 9th order polynomial to samples of the function $\sin(2\pi x)$ with Gaussian noise.



Bayesian Inference
Towards a more Bayesian treatment:

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

we have to do the following steps:

- define *prior* distribution over the parameters a as $p(\boldsymbol{a})$
- obtain the *likelihood* $p(y|X, \boldsymbol{a})$
- calculate the *posterior* $p(\boldsymbol{a}|X, y) \propto p(y|X, \boldsymbol{a})p(\boldsymbol{a})$

Example: Bayesian Inference

- The *likelihood* function was given by

$$p(y|X, \boldsymbol{a}; \sigma^2) = \prod_{i=1}^{n} \mathcal{N}\left(y_i|\boldsymbol{a}^T\boldsymbol{f}(x_i), \sigma^2\right)$$

- For simplicity we assume a zero-mean isotropic Gaussian *prior* over $\boldsymbol{a}$ with parameter α

$$p(\boldsymbol{a}; \alpha) = \mathcal{N}(\boldsymbol{a}|0, \alpha I)$$

- The corresponding *posterior* distribution over $\boldsymbol{a}$ is then given by

$$p(\boldsymbol{a}|y, X, \alpha; \sigma^2) = \mathcal{N}(\boldsymbol{a}|m_n, S_n)$$

  where

$$m_n = \frac{1}{\sigma^2} S_n \boldsymbol{F}^T y \quad \text{and} \quad S_n^{-1} = \frac{1}{\alpha}\boldsymbol{I} + \frac{1}{\sigma^2}\boldsymbol{F}^T\boldsymbol{F}$$

Example: Bayesian Inference
The log of the posterior distribution is given by the sum of the log likelihood and the log of the prior as:

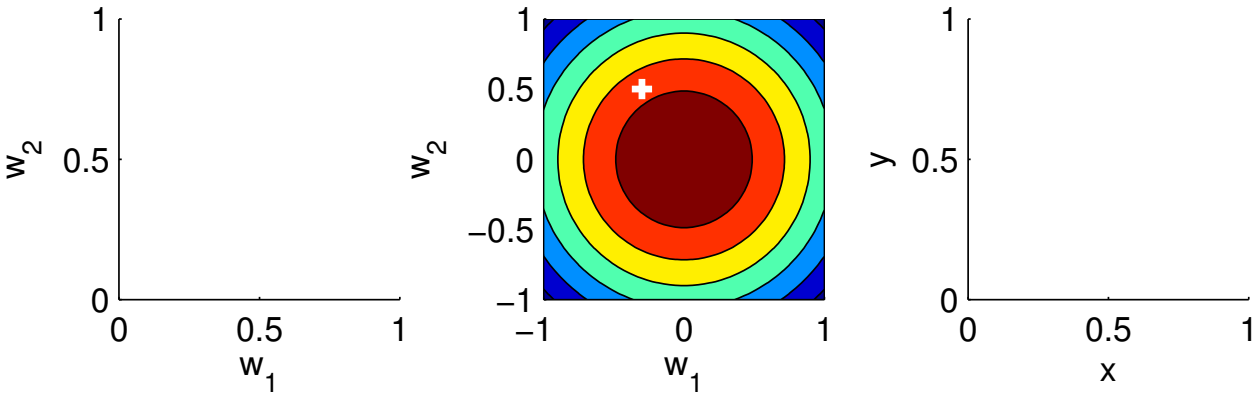
$$\ln p(\boldsymbol{a}|X, y) = -\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i - \boldsymbol{a}^T\boldsymbol{f}(x_i)\right)^2 - \frac{1}{2\alpha}\boldsymbol{a}^T\boldsymbol{a} + \text{const}$$

Maximizing the posterior (MAP) distribution w.r.t. $\boldsymbol{a}$ leads to

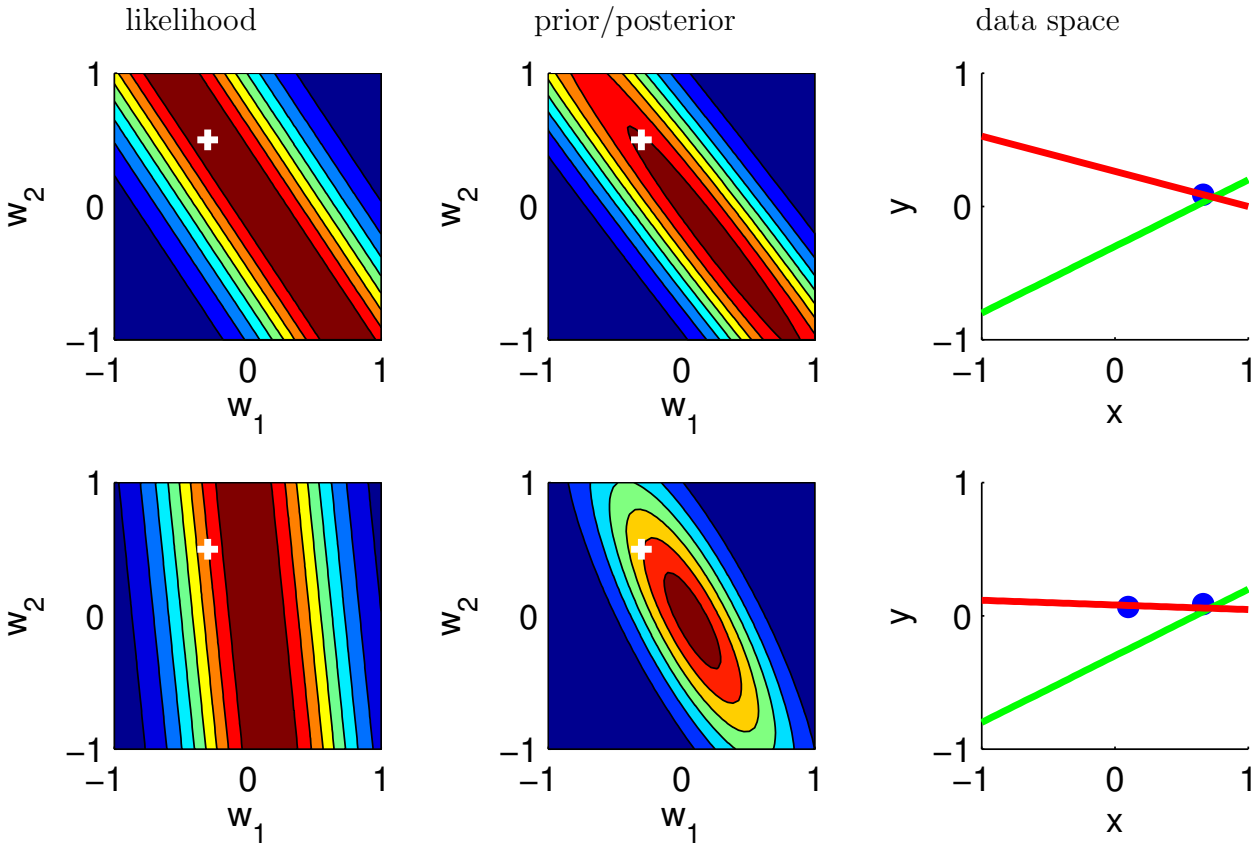
$$\boldsymbol{a}_{MAP} = (\lambda I + \boldsymbol{F}^T\boldsymbol{F})^{-1}\boldsymbol{F}^T y \quad \text{with} \quad \lambda = \frac{\sigma^2}{\alpha}$$

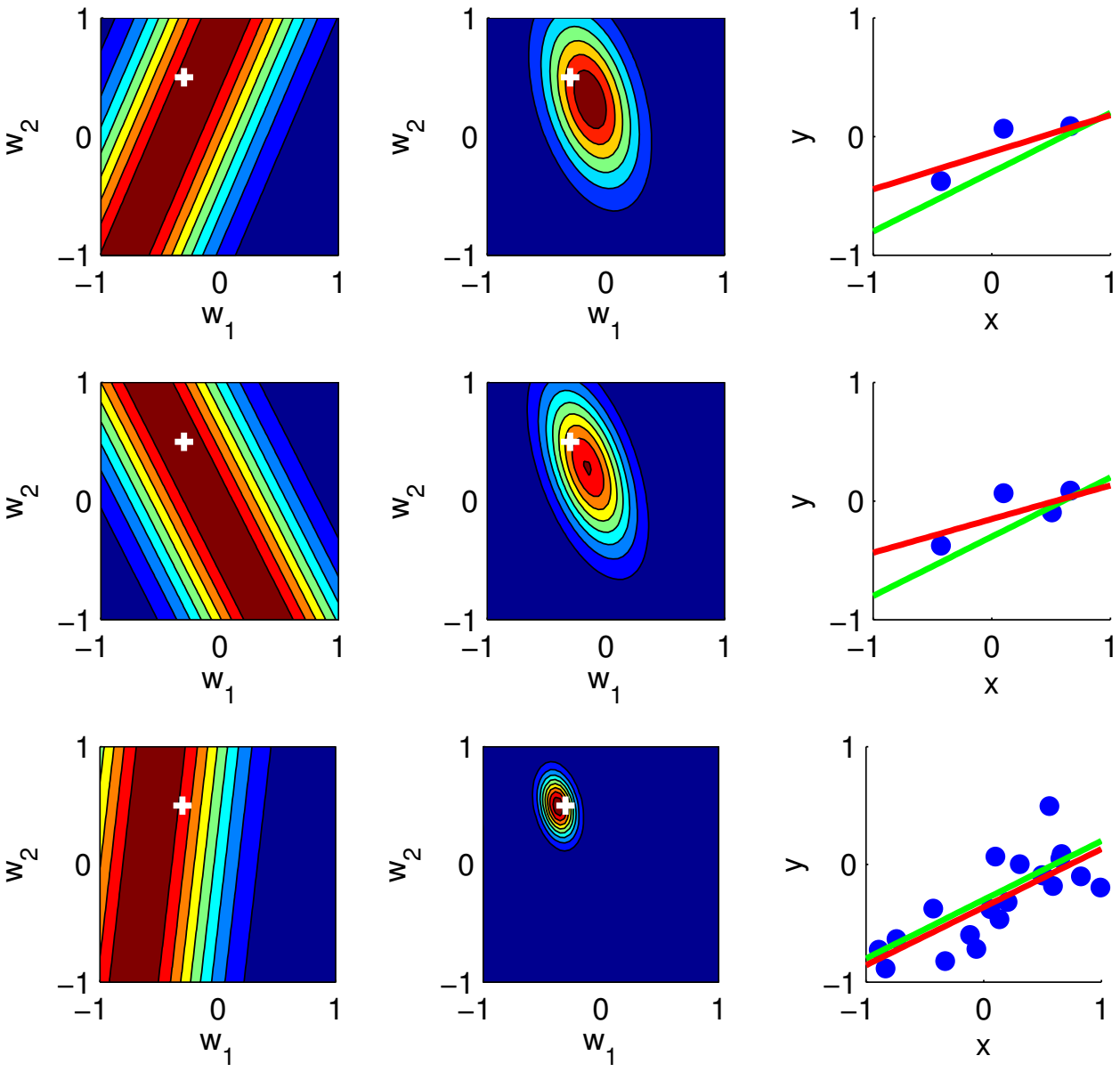
Example: MAP
In this example we *fit a straight line* to data coming from $y = 0.5x - 0.3$ with $\mathcal{N}(0, 0.04)$ noise. We can directly plot the *parameter space*:

With $\alpha = 0.5$, the parameter prior is



Example: MAP
Now we sequentially receive some data





Reminder: Conditional Probabilities

discrete Variables:

$$p(A) = \sum_B p(A, B)$$

continuous variables:

$$p(x, y) = \int p(x, a, y) da$$

conditioning:

$$\begin{aligned} p(x|y) &= \frac{p(x, y)}{p(y)} = \int \frac{p(x, a, y)}{p(y)} da \\ &= \int \frac{p(x, a, y)}{p(a, y)} \frac{p(a, y)}{p(y)} da = \int p(x|a, y) p(a|y) da \end{aligned}$$

Bayesian Linear Regression

In practice, we want to make predictions of t for new values of x_*. This requires to evaluate the *predictive distribution* defined by

$$p(y_*|x_*, y, X, \alpha; \sigma^2) = \int p(y_*|x_*, y, X, \boldsymbol{a}; \sigma^2) p(\boldsymbol{a}|y, X, \alpha; \sigma^2)\, d\boldsymbol{a}$$

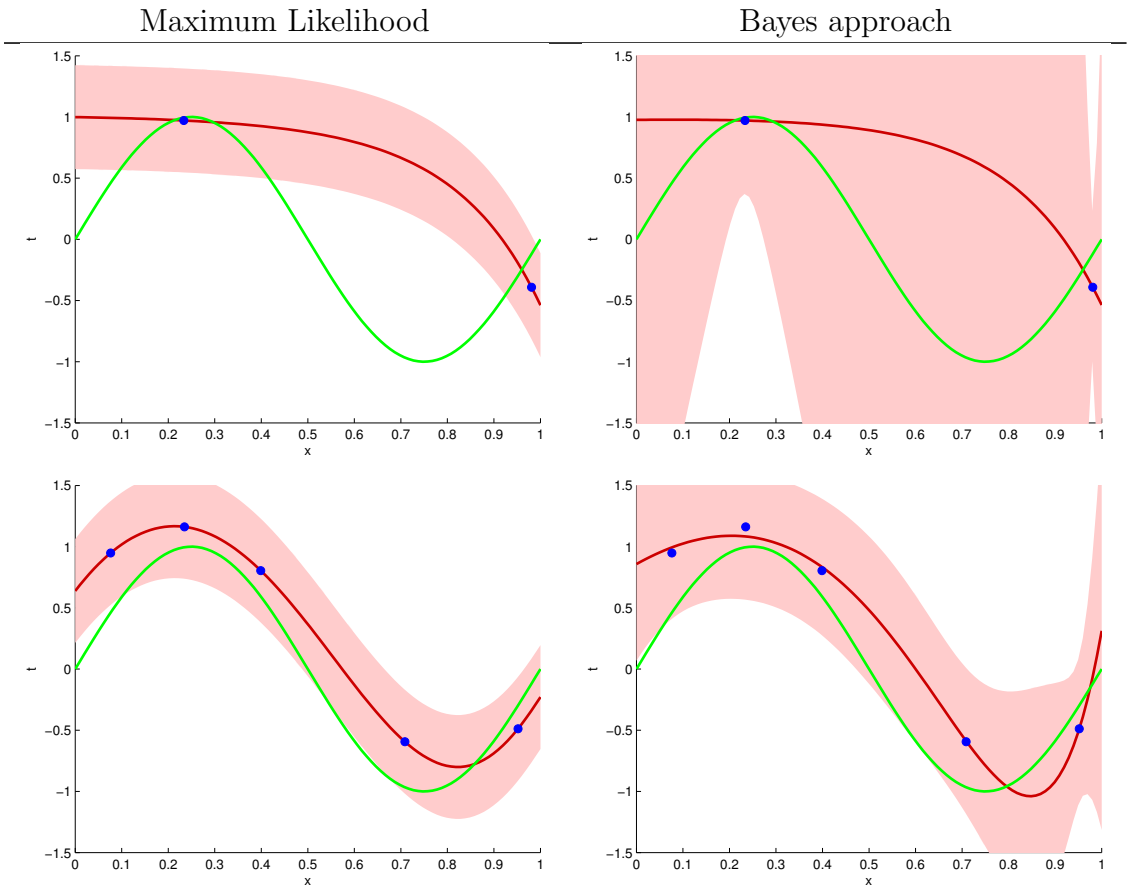
The convolution is a Gaussian with

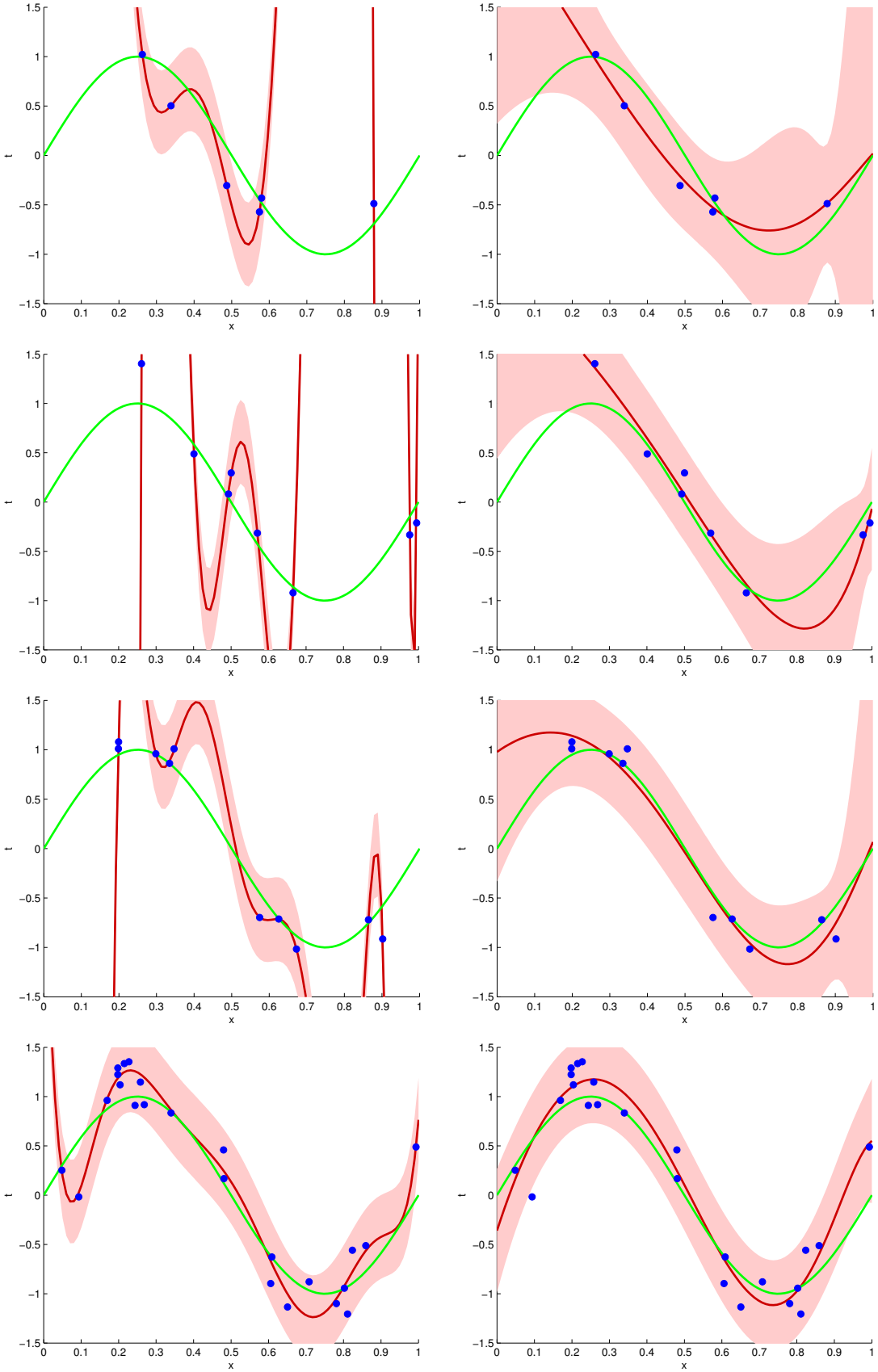
$$p(y_*|x_*, y, X, \alpha; \sigma^2) = \mathcal{N}\left(\boldsymbol{m}_n^T \boldsymbol{f}(x_*), \sigma_n^2(x_*)\right)$$

where

$$\sigma_n^2(x_*) = \sigma^2 + \boldsymbol{f}(x_*)^T S_n \boldsymbol{f}(x_*)$$

Example: Comparison between ML and Bayesian approach
Fitting a 9th order polynomial to samples of the function $\sin(2\pi x)$ with Gaussian noise.





Final Comments

In a fully Bayesian setting we should introduce priors over both, α and σ^2, but this is analytically *intractable*:

$$p(y_*|y) = \iiint p(y_*|\boldsymbol{a}, \sigma^2) p(\boldsymbol{a}|y, \alpha, \sigma^2) p(\alpha, \sigma^2)\, d\boldsymbol{a}\, d\alpha\, d\sigma^2$$

Have a look at

- *Empirical Bayes*: maximizing the marginal likelihood
- *Laplace approximation*: local Gaussian approximation of the posterior
- *Expectation maximization* (EM)

7.9 Exercises

Exercise 7.9 Calculate the probability distribution of the mean of n independant identically distributed discrete random variables $X_1, \ldots, X_n$, with

$$p(X_i = 0) = p(X_i = 1) = p(X_i = 2) = p(X_i = 3) = p(X_i = 4) = 1/5$$

for $n = 1, 2, 3, 4$.

Exercise 7.10 Prove the following identities for derivatives w.r.t. vectors:

a) $\dfrac{\partial(\boldsymbol{a}^T\boldsymbol{x})}{\partial\boldsymbol{x}} = \boldsymbol{a}$.

b) $\dfrac{\partial(\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x})}{\partial\boldsymbol{x}} = (\boldsymbol{A} + \boldsymbol{A}^T)\boldsymbol{x}$.

Exercise 7.11 To complete the proof of theorem Theorem 7.7, find a maximum of the variance $\boldsymbol{u}_{M+1}^T\boldsymbol{S}\boldsymbol{u}_{M+1}$ with respect to $\boldsymbol{u}_{M+1}$ under the constraints $\boldsymbol{u}_{M+1}^T\boldsymbol{u}_{M+1} = 1$ and $\boldsymbol{u}_{M+1}^T\boldsymbol{u}_1 = \boldsymbol{u}_{M+1}^T\boldsymbol{u}_2 = \ldots = \boldsymbol{u}_{M+1}^T\boldsymbol{u}_M = 0$.

Exercise 7.12 Apply PCA to the Lexmed data. The data file `appraw1-15.m` with the variables number 1 to 15 (variable number 16 removed) can be downloaded from the course website.

a) Determine the the eigenvalues and eigenvectors for the raw data.

b) Normalize the data to the interval $[0, 1]$ and repeat PCA.

c) Explain the differences.

d) Select the largest eigenvalues and give the transformation matrix for transforming the data into a lower dimensional space.

Exercise 7.13 Plot various two-dimensional normal distributions $\mathcal{N}(\mu, \Sigma)$ and validate empirically the following propositions. You may use for example

$$\mathcal{N}\left(\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}10 & 0\\0 & 1\end{bmatrix}\right) \quad \text{and} \quad \mathcal{N}\left(\begin{bmatrix}-1\\-1\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 10\end{bmatrix}\right).$$

a) The sum of two normal distributions is not a normal distribution.

b) The maximum of two normal distributions is not a normal distribution.

c) The product of two normal distributions is a normal distribution.

Exercise 7.14 Show that $\mathrm{cov}[x, y] = \mathbb{E}[xy] - \mathbb{E}[x]\mathbb{E}[y]$.

Exercise 7.15 Give an example for an estimator with 0 variance.

Exercise 7.16 Show that

a) $E[\bar{x}] = \mu$.

b) $\mathrm{var}[\bar{x}] = \frac{1}{n}\sigma^2$.

c) for the sample variance it holds:

$$s^2 = \frac{1}{n-1}\sum_{j=1}^{n}(x_j - \mu)^2 - \frac{n}{n-1}(\bar{x} - \mu)^2.$$

Exercise 7.17 Give an example for an unbiased estimator for the mean with higher variance than the sample mean.

Exercise 7.18 Let $U(a, b)$ the uniform distribution over the interval $[a; b] \subset \mathbb{R}$ with $a < b$. Further $x := (x_1, \cdots, x_n) \in \mathbb{R}^n$ are ordered samples from an unknown $U(a, b)$ s.t. $x_1 \leq \cdots \leq x_n$. The parameter space is denoted by $\Gamma = \{(a, b) \in \mathbb{R}^2 | a < b\}$. Define a density function $u_{a,b}$ of $U(a, b)$ and the likelihood function. Determine a maximum likelihood estimator for (a, b).

Exercise 7.19 Show that the expectation of a variable x that is Gaussian distributed with $\mathcal{N}(\mu, \sigma^2)$ is μ or in other words: $\mathbb{E}[x] = \int_{-\infty}^{\infty} \mathcal{N}(\mu, \sigma^2) x dx = \mu$. You can use the fact, that a Gaussian is a probability distribution and therefore integrates to 1 and that for an odd function f the following holds true: $\int_{-a}^{a} f(x)dx = 0$.

Exercise 7.20 Show that estimating the maximum posterior (MAP) with Gaussian likelihood and Gaussian prior (as in the lecture) with

$$a_{MAP} = (\lambda I + F^T F)^{-1} F^T y$$

is equal to "regularized least squares" which is the original least squares formulation plus some penalty term for high parameter values:

$$E(a) = \frac{1}{2} \sum_{i=1}^{n} (a^T f(x_i) - y_i)^2 + \frac{\lambda}{2} ||a||^2$$

Hint: Calculate the derivative of E with respect to a and set it to zero.

Exercise 7.21 Prove that the expected value is linear, i.e. that $E[ax + b] = aE[x] + b$ for
a) discrete variables.

b) continuous variables.

Chapter 8

Function Approximation

8.1 Linear Regression – Summary

We want to fit a function

$$f(\boldsymbol{x}) = a_1 f_1(\boldsymbol{x}) + \cdots + a_k f_k(\boldsymbol{x}) = \boldsymbol{a}^T \boldsymbol{f}(\boldsymbol{x})$$

with k unknown parameters $a_1, \ldots, a_k$ through the n data points $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$. If we substitute all the points into the ansatz, requiring our function to hit all n points, i.e.

$$f(\boldsymbol{x}_i) = y_i,$$

we get the linear system

$$\begin{array}{ccc} a_1 f_1(\boldsymbol{x}_1) + \ldots + a_k f_k(\boldsymbol{x}_1) = & y_1 \\ \vdots \qquad\qquad\qquad \vdots & \vdots \\ a_1 f_1(\boldsymbol{x}_n) + \ldots + a_k f_k(\boldsymbol{x}_n) = & y_n. \end{array}$$

In matrix notation we get

$$\boldsymbol{M} \cdot \boldsymbol{a} = \boldsymbol{y} \quad \text{with} \quad \boldsymbol{M}_{ij} = f_j(\boldsymbol{x}_i),$$

$n > k$ the system is overdetermined and normally has no solution.

$n < k$ the system is underdetermined and normally has infinitely many solutions.

We examined different solutions for the linear regression problem:

Overdetermined case:

- Least Squares / Pseudoinverse
- Maximum Likelihood
- Bayesian Linear Regression

Underdetermined case:

- Pseudoinverse

Methods for solving $\boldsymbol{M} \cdot \boldsymbol{a} = \boldsymbol{y}$

Overdetermined case:

Least Squares / Pseudoinverse:

minimize $\|\|\boldsymbol{M}\,\boldsymbol{a} - \boldsymbol{y}\|\|_2$	$\hat{\boldsymbol{a}} = (\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{y}$

Maximum Likelihood:

maximize $p(\boldsymbol{X}\|\boldsymbol{a})$	$\hat{\boldsymbol{a}} = (\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{y}$

Bayesian lin. Regression (MAP = maximum posterior probab.):

maximize $p(\boldsymbol{a}\|\boldsymbol{X})$	$\hat{\boldsymbol{a}} = (\lambda\boldsymbol{I} + \boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{y}$

Regularized Least Squares:

minimize $\|\|\boldsymbol{M}\,\boldsymbol{a} - \boldsymbol{y}\|\|_2 + \lambda\|\|\boldsymbol{a}\|\|_2$	$\hat{\boldsymbol{a}} = (\lambda\boldsymbol{I} + \boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{y}$

design matrix $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$

Methods for solving $\boldsymbol{M} \cdot \boldsymbol{a} = \boldsymbol{y}$

Underdetermined case:

- minimize $\|\|\boldsymbol{a}\|\|_2$ under the constraint $\boldsymbol{M}\,\boldsymbol{a} - \boldsymbol{y} = \boldsymbol{0}$. Solution: $\boxed{\hat{\boldsymbol{a}} = (\boldsymbol{M}\,\boldsymbol{M}^T)^{-1}\boldsymbol{M}\,\boldsymbol{y}}$
- compare (AI lecture)[**?**]: maximize Entropy of probability distribution under given contraints

8.2 Radial Basis Function Networks

8.2.1 Introduction

Radial basis function networks (RBFs) are a form of supervised learning techniques that are used to model or estimate an unknown function between a set of input-output pairs. The idea of RBFs has been presented as a solution for non-linear classification problems. The theory of RBFs had initiated from Cover whereby his theorem proved that a classification problem is more likely to be linearly separated in a high dimensional space rather than in a low dimensional space. Further discussion about the Cover's theorem accompanied with a detailing example will be presented in the next section.
Radial basis function networks are considered to be linear models with non-linear activation functions. Linear approximation models had been studied in statistics for about 200 years, and the theory is applicable to radial basis function networks (RBF) which are just one particular type of linear models. The idea of radial basis function networks is similar to that of multi layer perceptron neural networks with differences such as:

- Radial basis function as an activation function, rather than a sigmoid function.
- Three layer network with an input, one hidden and an output layer.
- No back propagation is included in solving for the output weights.

There are two main applications for radial basis functions. The first is a solution of classification problems which will be briefly mentioned in the next section so to explain the Cover's theorem. The other idea of interest is utilizing RBFs as a solution for an approximation problem (i.e estimating a function that maps sets of input-output pairs) will be further discussed and detailed.

8.2.2 RBFs for function approximation

Now the focus will be shifted to the form of RBFs used for function approximation. In other ways answering the question of supervised learning problem, which could be stated as:
Given a set of input output pairs, attain the unknown function mapping the latter set.
To have a detailed idea of the subject a brief introduction to supervised learning will be mentioned.

8.2.2.1 Supervised Learning

A problem in statistics with applications in many areas is to guess or to estimate a function from a sample of input-output pairs with a little or no knowledge of the form of the function. So common is the problem that it has different names in different disciplines (e.g. nonparametric regression, function approximation, system identification, inductive learning).
In machine learning terminology, the problem is called supervised learning. The function is learned from examples, which a teacher supplies. The set of examples, or *training set*, contains elements which consist of paired values of the independent (input) variable x the dependent (output) variable y. Mathematically given a vector n-patterns of a p-dimensional vector $\boldsymbol{x}$ The training set (pairs of input and outputs) is given as:

$$T = \{(\boldsymbol{x}_i, \hat{y}_i)\}_{i=1}^{n} \tag{8.1}$$

This training set reflects that the outputs y are corrupted by noise. In other words the correct value to the input $\boldsymbol{x}_i$, namely y_i, is unknown. The training set only specifies $\hat{y}_i$ which is y_i, plus a small amount of noise.

$$\hat{y}_i = f(\boldsymbol{x}_i) + \epsilon \tag{8.2}$$

Where ϵ is some form of Gaussian noise with zero mean and some covariance.

In real applications the independent variable values in the training set are often also affected by noise. This type of noise is more difficult to model and we shall not attempt it. In any case, taking account of noise in the inputs is approximately equivalent to assuming noiseless inputs but an increased amount of noise in the outputs.

8.2.2.2 Nonparametric Regression

In regression problems there are two deviations the *parametric* and *nonparametric* approach. Parametric regression is a form of regression whereby the functional relation of the input-output pairs is assumed to be known, but may contain unknown parameters. This case is not of interest, because it has a main disadvantage that the functional topology should be known in advance to solve such a problem. This prior knowledge, is difficult to be found especially in the case of complicated and highly nonlinear systems. Therefore the focus will be shifted to the nonparametric approach, where no prior knowledge of the functional mapping is required. Radial basis function networks are a form of nonparametric regression,

that aim to find an underlying relation between inputs and outputs[24]. In other words the goal of the radial basis function network is to fit the best values of some weights in order to minimize a certain error defined by an *error function*.

8.2.2.3 Linear Models

A linear model of a function $f(\boldsymbol{x})$ takes the form:

$$f(\boldsymbol{x}) = \sum_{j=1}^{m} a_j f_j(\boldsymbol{x}) \tag{8.3}$$

The model f is expressed as a linear combination of a set of m fixed functions (often called basis functions by analogy to the concept of a vector being composed of a linear combination of basis vectors). The aim of any network is to find the best possible weights a_j so to minimize the sum of the squared error that is often defined by the error function.

Activation Functions Before going into the details on how to solve for the weights we discuss the activation functions (f_j).There are several types of activation functions that are used in neural networks, but the functions of interest are the radial functions. Radial functions are a special class of functions. There characteristic features is that there response decreases (or increases) monotonically with the distance from the central point.

- Gaussian which is the most commonly used:

$$f_j(\boldsymbol{x}) = \exp \frac{-(||\boldsymbol{x}_i - \boldsymbol{c}_j||^2)}{2\sigma^2} \tag{8.4}$$

  The Gaussian function decreases monotonically with the distance from the center as shown in figure 8.1.

- Multiquadric:

$$f_j(||\boldsymbol{x} - \boldsymbol{c}||) = \sqrt{(||\boldsymbol{x_i} - \boldsymbol{c_j}||)^2 + b^2} \tag{8.5}$$

- Inverse Multiquadrics:

$$f_j(||\boldsymbol{x} - \boldsymbol{c}||) = \frac{1}{\sqrt{(||\boldsymbol{x_i} - \boldsymbol{c_j}||)^2 + b^2}} \tag{8.6}$$

8.2.2.4 Radial Basis Function Networks

Radial functions are simply a class of functions. In principle, they could be employed in any sort of model (linear or nonlinear) and any sort of network (single-layer or multilayer). However, since Broomhead and Lowe's[25], radial basis function networks (RBF networks) have traditionally been associated with three layers as follows, see figure 8.2 :

- Input layer of dimensions representing the n patterns of the p-dimensional input vector $\boldsymbol{x}$
- Hidden layer containing the activation radial functions (such as Gaussian) with number m

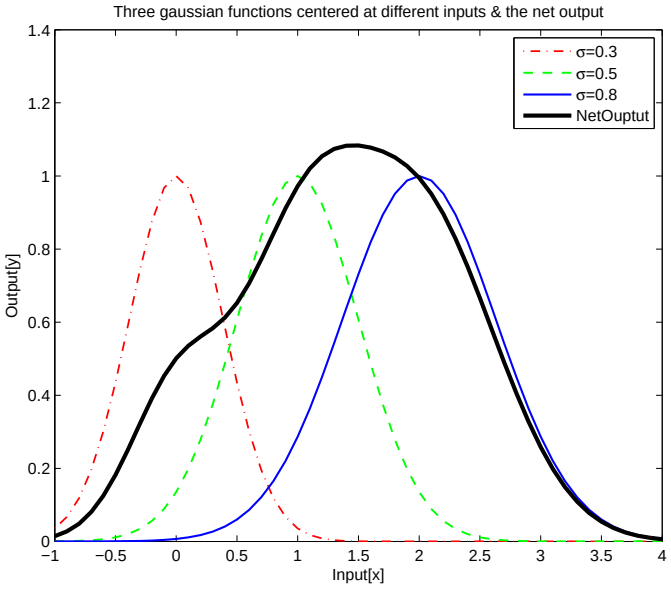


Figure 8.1: Weights were set to $a_1 = 0.4$,a_2=0.7 and $a_3 = 0.9$

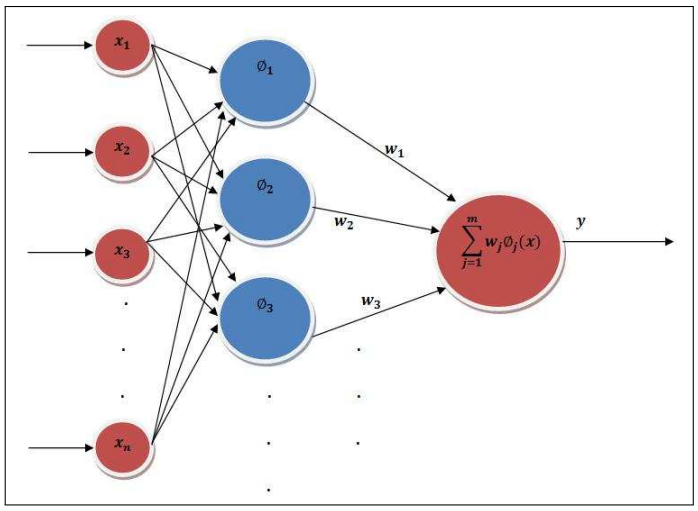


Figure 8.2: Structure of a RBF network

- Linear output unit

The unknowns in the case of a linear RBF model are the weights a_j's that need to be found and solved for. In order to solve for the weights the problem should be reformulated in a sums of squared errors form.

Least Square Problem An error function of the weights should be defined and then an optimization procedure will be used to attain them. Lets consider the overall picture again, given a data set $T = \{(\boldsymbol{x}_i, \hat{y}_i)\}_{i=1}^{n}$ we have to estimate a function between these input and output pairs. From figure 8.2 it can be seen that the output function is:

$$f(\boldsymbol{x}) = \sum_{j=1}^{m} a_j f_j(\boldsymbol{x}) \tag{8.7}$$

Then we define the error function as the sum of the squared errors, between the real valued y_i's and the predicted ones from the RBF network as follows:

$$E(y, f(\boldsymbol{x})) = \frac{1}{2}\sum_{i=1}^{n}(y_i - f(\boldsymbol{x_i}))^2$$
$$E(y, f(\boldsymbol{x})) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{m} a_j f_j(\boldsymbol{x_i}))^2 \tag{8.8}$$

The objective now is to find the best set of the a_j that minimizes the error function E of equation (8.8). Mathematically formulated the above idea could be described as follows:

$$a_j = \arg \min_{(a_1,\ldots,a_j,\ldots,a_m)} E(y, f(\boldsymbol{x})) \tag{8.9}$$

Several algorithms had been suggested for such an evaluation[6], and maybe the most common is the gradient descent algorithm. This algorithm might have some problems like convergence, getting stuck in a local minimum and so on.
Therefore, it would be better if there was a way to represent the above equation in a matrix form, and then a single step to solve for the weights would be utilized[2]. For this formulation consider the following:

- Let $\boldsymbol{y} = \begin{pmatrix} y_1 & y_2 & \ldots & y_n \end{pmatrix}^T$ represent the desired outputs.
- Let $\boldsymbol{a} = \begin{pmatrix} a_1 & a_2 & \ldots & a_m \end{pmatrix}^T$ represent the weights that have to be determined.
- Let the matrix $\boldsymbol{M} = \begin{pmatrix} f_1(\boldsymbol{x_1}) & f_2(\boldsymbol{x_1}) & \ldots & f_m(\boldsymbol{x_1}) \\ f_1(\boldsymbol{x_2}) & f_2(\boldsymbol{x_2}) & \ldots & f_m(\boldsymbol{x_2}) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(\boldsymbol{x_n}) & f_2(\boldsymbol{x_n}) & \ldots & f_m(\boldsymbol{x_n}) \end{pmatrix}$ be the matrix of the RBFs operating at the input points.

There for the above system could be transformed into the following form:

$$\boldsymbol{M}\,\boldsymbol{a} = \boldsymbol{y} \tag{8.10}$$

Therefore solving for the weights after this formulation is straight forward and requires only the inversion of the $\boldsymbol{M}$ matrix.
Assuming the $\boldsymbol{M}$ is nonsingular and $\boldsymbol{M}^{-1}$ exists then the weights could be calculated using the following equation as:

$$\boldsymbol{a} = \boldsymbol{M}^{-1}\boldsymbol{y} \tag{8.11}$$

A special case of this solution is when the number of the hidden layer units (i.e Gaussian Functions) is equal to that of the number of samples present by the training set T. In other words the $\boldsymbol{M}$-matrix is an n by n matrix, and there normal inversion exists in the case the latter matrix was non-singular. On the other hand, if this matrix was not a square one which is the most general case, whereby the number of hidden units m is less than that of the training sample n, then the $\boldsymbol{M}^{-1}$-matrix could not be attained normally. Rather the pseudo-inverse has to be calculated. To do this there are different methods some of which are:

- QR-decomposition
- Single Value Decomposition (SVD)

Concrete Example Consider the following three points $(1, 3)$,$(2, 2.1)$,$(3, 2.5)$ to be approximated by a function. The RBFs used are Gaussian centered at each input point. The objective of this example is to illustrate the effect of the choice of σ on the underlying function being approximated.

It is clear from figure 8.3, that having a small σ causes overfitting, and the choice of a big

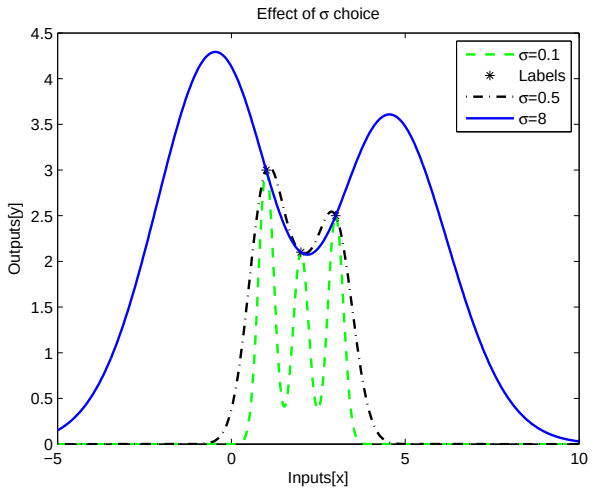


Figure 8.3: The effect of sigma

σ caused very high and low overshoots. The latter case could be explained by the fact that choosing a high value for the σ consequently leads to attaining very high positive and negative values of the weights fitting the required points, so that the function thus approximated could pass through all the points presented.

8.2.3 Over-fitting Problem

Consider that we have chosen the number of the basis functions to be the same number as the training examples T, moreover we have chosen the centers of the radial basis function networks to be the input points. This leads to the so-called problem of overfitting. As clear

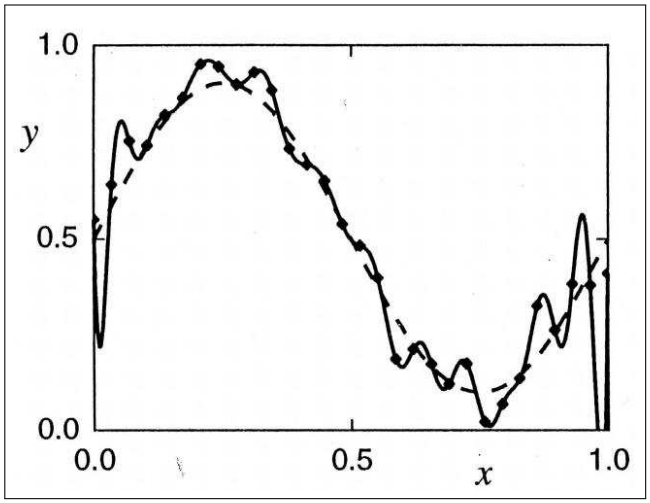


Figure 8.4: Overfitting effect

from figure 8.4, the function which was supposed to be approximated is the one represented

by the dashed line, but due to the latter configuration of the RBF it rather tended to approximate the bold line, which is not the intended mapping.
The network described in this example, is a specific type of RBFs used solely for interpolation. The problems of such a scheme are:

1. Poor performance on noisy data:
   - As already known, we do not usually want the network's outputs to pass through all the data points when the data is noisy, because that will be a highly oscillatory function that will not provide good generalization.
2. Computationally inefficient:
   - The network requires one hidden unit (i.e. one basis function) for each training data pattern, and so for large data sets the network will become very costly to evaluate. The matrix inversion cost is typically $O(n^3)$ for n data points.

8.2.3.1 Improving RBFs

In order to improve the RBF networks such that it doesn't conduct solely exact interpolation, the following points could be taken into account:

1. The number m of basis functions (hidden units) should be less than n.
2. The centers of the basis functions do not need to be defined as the training data input vectors. They can instead be determined by a training algorithm.
3. The basis functions need not all have the same width parameter σ. These can also be determined by a training algorithm.
4. We can introduce bias parameters into the linear sum of activations at the output layer. These will compensate for the difference between the average value over the data set of the basis function activations and the corresponding average value of the targets.

The most general approach to overcome overfitting is to assume that the centers and the width of the Gaussian functions are unknown, and apply a supervised learning algorithm to solve for all the variables. This approach also includes a regularization term that thus form the so called regularization network. The latter idea lies behind the fact that if we add a regularization term for the network being the gradient of the function intended in approximation, will form a network that does not rely only on interpolation, rather on both interpolation and smoothing as follows:

$$\begin{aligned} E_{new} &= E_{normal} + \lambda E_{reg} \\ E_{new} &= \frac{1}{2}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{m} a_j f_j(\boldsymbol{x})\right)^2 + \frac{1}{2}\lambda||\nabla F||^2 \end{aligned} \tag{8.12}$$

This approach will not be discussed here, rather a clustering algorithm to choose the centers is represented.
As mentioned above, that the correct choice of the centers affects critically the performance of the network and the function thus approximated. For that sake the correct choice of the centers for the radial basis functions being approximated is critical. The upcoming section will clarify a specific clustering algorithm for the choice of the centers and the widths.

8.2.3.2 Autonomous determination of center

The choice of the centers of the radial basis functions could be done using a *K-means Clustering*, and could be described as follows:

- The algorithm partitions data points into K disjoint subsets (K is predefined).
- The clustering criteria are:
  - the cluster centers are set in the high density regions of data
  - a data point is assigned to the cluster with which it has the minimum distance to the center

Mathematically this is equivalent to minimizing the sum of square clustering function defined as :

$$E = \sum_{j=1}^{k} \sum_{n \in S_j} ||\boldsymbol{x}^n - \boldsymbol{c}_j||^2$$
$$\boldsymbol{c}_j = \frac{1}{N_j} \sum_{n \in S_j} \boldsymbol{x}^n \tag{8.13}$$

Where S_j is the j'th cluster with N_j points.
After achieving the centers, now the values of the σ could be set according to the diameters of the clusters previously attained. For further information about the *K-means clustering* please refer to [**?**].

8.3 Clustering

If we search in a search engine for the term “mars”, we will get results like “the planet mars” and “Chocolate, confectionery and beverage conglomerate” which are semantically quite different. In the set of discovered documents there are two noticeably different **clusters**. Google, for example, still lists the results in an unstructured way. It would be better if the search engine separated the clusters and presented them to the user accordingly because the user is usually interested in only one of the clusters.
The distinction of **clustering** in contrast to supervised learning is that the training data are unlabeled. Thus the pre-structuring of the data by the supervisor is missing. Rather, finding structures is the whole point of clustering. In the space of training data, accumulations of of data such as those in Figure 8.5 are to be found. In a cluster, the distance of neighboring points is typically smaller than the distance between points of different clusters. Therefore the choice of a suitable distance metric for points, that is, for objects to be grouped and for clusters, is of fundamental importance. As before, we assume in the following that every data object is described by a vector of numerical attributes.

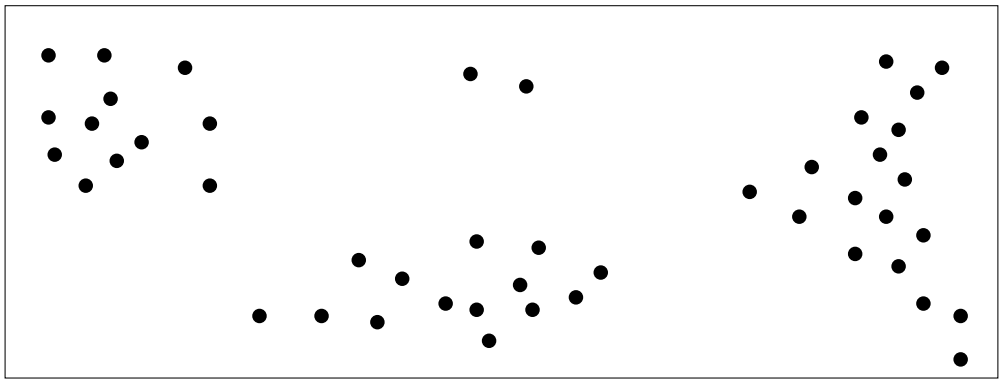


Figure 8.5: Simple two-dimensional example with four clearly separated clusters.

8.3.1 Distance Metrics

Accordingly for each application, the various distance metrics are defined for the distance d between two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{R}^n$ The most common is the euclidean distance

$$d_e(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}.$$

Somewhat simpler is the sum of squared distances

$$d_q(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{n} (x_i - y_i)^2,$$

which, for algorithms in which only distances are compared, is equaivalent to the euclidean distance. Also used are the aforementioned Manhattan distance

$$d_m(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{n} |x_i - y_i|$$

as well as the distance of the maximum component

$$d_\infty(\boldsymbol{x}, \boldsymbol{y}) = \max_{i=1,\ldots,n} |x_i - y_i|$$

which is based on the maximum norm. During text classification, the normalized projection of the two vectors on each other, that is, the normalized scalar product

$$\frac{\boldsymbol{x}\ \boldsymbol{y}}{|\boldsymbol{x}|\ |\boldsymbol{y}|}$$

is frequently calculated, where $|\boldsymbol{x}|$ is the euclidian norm of $\boldsymbol{x}$. Because this formula is a metric for the similarity of the two vectors, as a distance metric the inverse

$$d_s(\boldsymbol{x}, \boldsymbol{y}) = \frac{|\boldsymbol{x}|\ |\boldsymbol{y}|}{\boldsymbol{x}\ \boldsymbol{y}}$$

can be used, or "$>$" and "$<$" can be swapped for all comparisons. In the search for a text, the attributes $x_1, \ldots, x_n$ are calculated similarly to naive Bayes as components of the vector $\boldsymbol{x}$ as follows. For a dictionary with 50,000 words, the value x_i equals the frequency of the i-th dictionary word in the text. Since normally almost all components are zero in such a vector, during the calculation of the scalar product, nearly all terms of the summation are zero. By exploiting this kind of information, the implementation can be sped up significantly.

8.3.2 k-Means and the EM Algorithm

Whenever the number of clusters is already known in advance, the **k-Means** algorithm can be used. As its name suggests, k clusters are defined by their average value. First the k cluster midpoints $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$ are randomly or manually initialized. Then the following two steps are repeatedly carried out:

- Classification of all data to its nearest cluster midpoint
- Recomputation of the cluster midpoint.

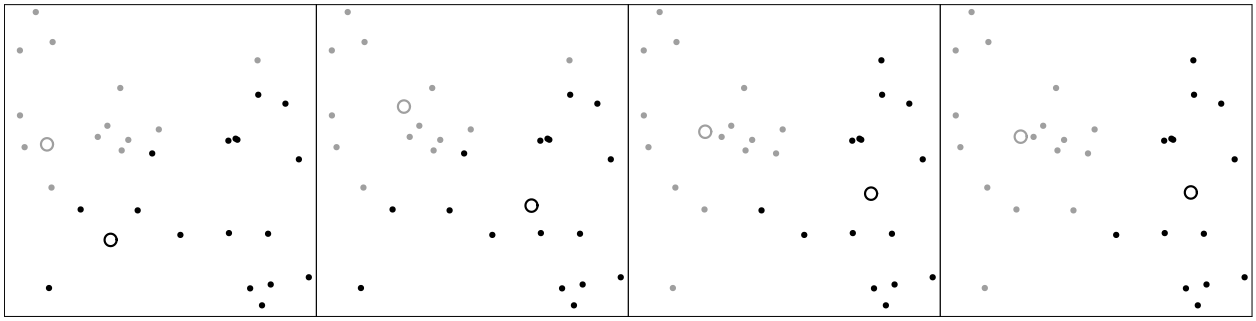
The following scheme results as an algorithm:

K-MEANS($\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, k$)
initialize $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$ (e.g. randomly)
Repeat
    classify $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ to each's nearest $\boldsymbol{\mu}_i$
    recalculate $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$
Until no change in $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$
Return($\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$)

The calculation of the cluster midpoint $\boldsymbol{\mu}$ for points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_l$ is done by

$$\boldsymbol{\mu} = \frac{1}{l} \sum_{i=1}^{l} \boldsymbol{x}_i.$$

The execution on an example is shown in Figure 8.6 for the case of two classes. We see how after three iterations, the class centers, which were first randomly chosen, stabilize. While this algorithm does not guarantee convergence, it usually converges very quickly. This means that the number of iteration steps is typically much smaller than the number of data points. Its complexity is $O(ndkt)$, where n is the total number of points, d the dimensionality of the feature space, and t the number of iteration steps.



random initialization $t = 1$ $t = 2$ $t = 3$

Figure 8.6: k-means with two classes ($k = 2$) applied to 30 data points. Far left is the data set with the initial centers, and to the right is the cluster after each iteration. After three iterations convergence is reached.

In many cases, the necessity of giving the number of classes in advance poses an inconvenient limitation. Therefore we will next introduce an algorithm which is more flexible.
Before that, however, we will mention the **EM algorithm**, which is a continuous variant of k-means, for it does not make a firm assignment of the data to classes, rather, for each point it returns the probability of it belonging to the various classes. Here we must assume that the type of probability distribution is known. Often the normal distribution is used. The task of the EM algorithm is to determine the parameters (mean $\boldsymbol{\mu}_i$ and covariance matrices Σ_i of the k multidimentional normal distributions) for each cluster. Similarly to k-means, the two following steps are repeatedly executed:

Expectation: For each data point the probability $P(C_j|\boldsymbol{x}_i)$ that it belongs to each cluster is calculated.

Maximization: Using the newly calculated probabilities, the parameters of the distribution are recalculated.

Thereby a softer clustering is achieved, which in many cases leads to better results. This alternation between expectation and maximization gives the algorithm its name. In addition to clustering, for example, the EM algorithm is used to learn Bayesian networks. [**?**].

8.3.3 Hierarchical Clustering

In hierarchical clustering we begin with n clusters consisting of one point each. Then the nearest neighbor clusters are combined until all points have been combined into a single cluster, or until a termination criterion has been reached. We obtain the scheme

HierarchicalClustering($\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$)
initialize $C_1 = \{\boldsymbol{x}_1\}, \ldots, C_n = \{\boldsymbol{x}_n\}$
Repeat
    Find two clusters C_i and C_j with the smallest distance
    Combine C_i and C_j
Until Termination condition reached
Return(tree with clusters)

The termination condition could be chosen as, for example, a desired number of clusters or a maximum distance between clusters. In Figure 8.7 this algorithm is represented schematically as a binary tree, in which from bottom to top in each step, that is, at each level, two subtrees are connected. At the top level all points are unified into one large cluster.

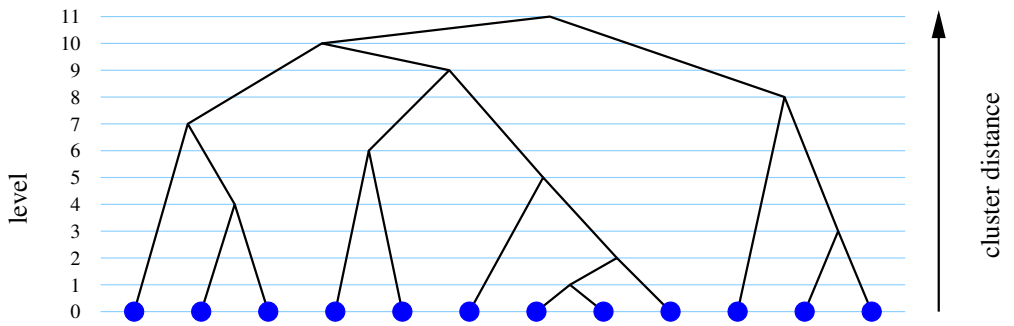


Figure 8.7: In hierarchical clustering, the two clusters with the smallest distance are combined in each step.

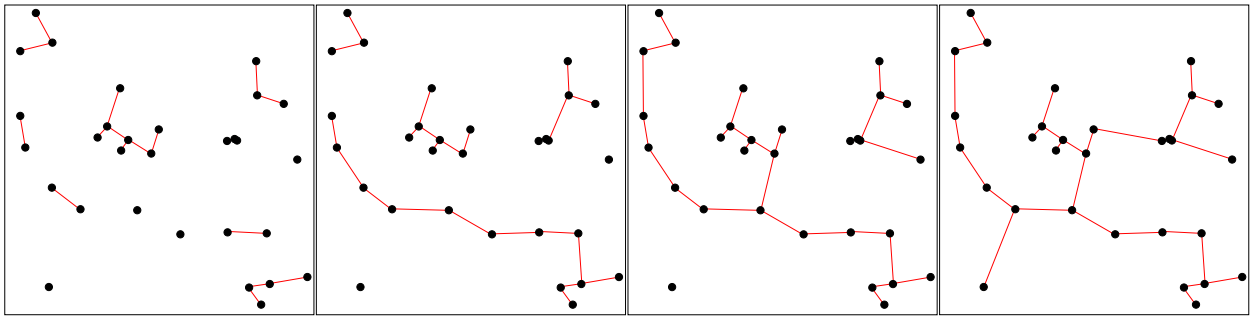
It is so far unclear how the distances between the clusters are calculated. Indeed, in the previous section we defined various distance metrics for points, but these cannot be used on clusters. A convenient and often used metric is the distance between the two closest points in the two clusters C_i und C_j:

$$d_{min}(C_i, C_j) = \min_{\boldsymbol{x} \in C_i,\ \boldsymbol{y} \in C_j} d(\boldsymbol{x}, \boldsymbol{y}).$$

Thus we obtain the **nearest neighbor algorithm**, whose application is shown in Figure 8.8. We see that this algorithm generates a minimum spanning tree.[1] The example furthermore shows that the two described algorithms generate quite different clusters. This tells us that for graphs with clusters which are not clearly separated, the result depends heavily on the algorithm or the chosen distance metric.
For an efficient implementation of this algorithm, we first create an adjacency matrix in which the distances between all points is saved, which requires $O(n^2)$ time and memory. If the number of clusters does not have an upper limit, the loop will iterate $n - 1$ times and the asymptotic computation time becomes $O(n^3)$.
To calculate the distance between two clusters, we can also use the distance between the two farthest points

$$d_{max}(C_i, C_j) = \max_{\boldsymbol{x} \in C_i, \boldsymbol{y} \in C_j} d(\boldsymbol{x}, \boldsymbol{y}).$$

[1] A minimum spanning tree is an acyclic, undirected graph with the minimum sum of edge lengths.



$d_{min} \geq 1.2$ $d_{min} \geq 1.6$ $d_{min} \geq 2$ min. spanning tree

Figure 8.8: The nearest neighbor algorithm applied to the data from Figure 8.6 at different levels with 12, 6, 3, 1 clusters.

and obtain the **farthest neighbor algorithm**. Alternatively, the distance of the cluster's midpoint $d_\mu(C_i, C_j) = d(\boldsymbol{\mu_i}, \boldsymbol{\mu_j})$ is used. Besides the clustering algorithm presented here, there are many others, for which we direct the reader to [**?**] for further study.

8.4 Singular Value Decomposition and the Pseudo-Inverse

In Theorem 6.8 we have seen that for the computation of the pseudoinverse of an overdetermined matrix $\boldsymbol{M}$ the square matrix $\boldsymbol{M}^T\boldsymbol{M}$ must be invertible. Analogously, due to Equation 6.25, for an underdetermined matrix $\boldsymbol{M}$ the square matrix $\boldsymbol{M}\boldsymbol{M}^T$ has to be invertible. In both cases, the resulting square matrix is invertible if the matrix $\boldsymbol{M}$ has full rank.
We will now present an even more general method for determining a pseudoinverse even if $\boldsymbol{M}$ has not full rank.

Reminder: Linear Algebra
Recommended preparation: Gilbert Strang Video Lectures

- Lecture 21: Eigenvalues and eigenvectors
- Lecture 25: Symmetric matrices and positive definiteness

on http://ocw.mit.edu/courses/mathematics/18-06-linear-algebra-spring-2010

Definition 8.1 Two vectors $\boldsymbol{x}_i, \boldsymbol{x}_j$ are called orthonormal if

$$\boldsymbol{x}_i^T \boldsymbol{x}_j = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{else} \end{cases} .$$

A matrix $\boldsymbol{A}$ is called orthogonal, if its columns are orthonormal.

Some basic facts:

- For any orthogonal matrix $\boldsymbol{A}$ we have $\boldsymbol{A}^T\boldsymbol{A} = I$.

- No eigenvalues of an invertible $n \times n$ matrix are zero.
- If all eigenvalues of an $n \times n$ matrix are pairwise different, then the eigenvectors are linearly independent.
- A symmetric matrix has only real eigenvalues.
- The eigenvectors of a symmetric matrix are orthogonal. They can be chosen to be orthonormal.

Diagonalization of symmetric matrices
Eigenvalue equations:

$$\boldsymbol{A}\boldsymbol{x}_1 = \lambda_1 \boldsymbol{x}_1 \quad \dots \quad \boldsymbol{A}\boldsymbol{x}_n = \lambda_n \boldsymbol{x}_n$$

Combing all n equations yields

$$\boldsymbol{A}\,(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = (\lambda_1 \boldsymbol{x}_1, \dots, \lambda_n \boldsymbol{x}_n) = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix}$$

With $\boldsymbol{Q} = (\boldsymbol{x}_1, \dots \boldsymbol{x}_n)$ and $\boldsymbol{\Lambda} = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix}$ we get

$$\boldsymbol{A}\boldsymbol{Q} = \boldsymbol{Q}\boldsymbol{\Lambda} \quad \text{and} \quad \boldsymbol{A} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T.$$

Theorem 8.1 (Spectral theorem)

- Every symmetric matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ has the factorization $\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$.
- The columns of $\boldsymbol{Q}$ are the eigenvectors.
- The eigenvectors are orthogonal.
- $\boldsymbol{\Lambda}$ is diagonal with the eigenvalues as elements.

Singular Value Decomposition
Gilbert Strang writes in [1]:

> "I give you my opinion directly. The SVD is the climax of this linear algebra course. I think of it as the final step in the Fundamental Theorem. First come the dimensions of the four subspaces. Then their orthogonality. Then the orthonormal bases which diagonalize A. It is all in the formula $\boldsymbol{M} = U\Sigma V^T$. **You have made it to the top.**"

- $\boldsymbol{M} \in \mathbb{R}^{m \times n}$ hat not full rank.
- $\boldsymbol{M}^T\boldsymbol{M}$ is symmetric, but not invertible.

Eigenvalue equation: $\boxed{\boldsymbol{M}^T\boldsymbol{M}\,\boldsymbol{v}_i = \sigma_i^2\boldsymbol{v}_i}$

$$\boldsymbol{v}_i^T\boldsymbol{M}^T\boldsymbol{M}\,\boldsymbol{v}_i = ||\boldsymbol{M}\,\boldsymbol{v}_i||^2 = \sigma_i^2\boldsymbol{v}_i^T\boldsymbol{v}_i = \sigma_i^2 \geq 0.$$

Thus $\boldsymbol{M}^T\boldsymbol{M}$ is positive definite and $||\boldsymbol{M}\,\boldsymbol{v}_i|| = \sigma_i \geq 0$. Now

$$\boldsymbol{M}\,\boldsymbol{M}^T\boldsymbol{M}\,\boldsymbol{v}_i = \sigma_i^2\boldsymbol{M}\,\boldsymbol{v}_i$$

shows that $\boldsymbol{M}\,\boldsymbol{M}^T$ has the same eigenvalues σ_i^2 with the unit eigenvectors

$$\boldsymbol{u}_i = \boldsymbol{M}\,\boldsymbol{v}_i/\sigma_i.$$

This leads to (r = rank of $\boldsymbol{M}$) $\boxed{\boldsymbol{M}\,\boldsymbol{v}_1 = \sigma_1\boldsymbol{u}_1 \quad \ldots \quad \boldsymbol{M}\,\boldsymbol{v}_r = \sigma_r\boldsymbol{u}_r}$
and

$$\boldsymbol{M}\begin{pmatrix}\boldsymbol{v}_1 \ldots \boldsymbol{v}_r\end{pmatrix} = \begin{pmatrix}\boldsymbol{u}_1 \ldots \boldsymbol{u}_r\end{pmatrix}\begin{pmatrix}\sigma_1 & & 0\\ & \ddots & \\ 0 & & \sigma_r\end{pmatrix}.$$

Adding orthonormal vectors $\boldsymbol{v}_i$ from the nullspace of $\boldsymbol{M}$ and orthonormal vectors $\boldsymbol{u}_i$ from the nullspace of $\boldsymbol{M}^{\boldsymbol{T}}$:

$$\boldsymbol{M}\begin{pmatrix}\boldsymbol{v}_1 \ldots \boldsymbol{v}_r \ldots \boldsymbol{v}_n\end{pmatrix} = \begin{pmatrix}\boldsymbol{u}_1 \ldots \boldsymbol{u}_r \ldots \boldsymbol{u}_m\end{pmatrix}\begin{pmatrix}\sigma_1 & & & 0\ldots 0\\ & \ddots & & \\ & & \sigma_r & \\ & & 0 & \vdots\ \vdots\\ & & & \ddots\\ 0 & & & 0\ldots 0\end{pmatrix}.$$

The dimensions of these matrices are

$$(m \times n)\ (n \times n) = (m \times m)(m \times n).$$

Written im matrix notation, we get $\boldsymbol{M}\,\boldsymbol{V} = \boldsymbol{U}\,\boldsymbol{\Sigma}$ with the orthogonal matrices $\boldsymbol{V}$ and $\boldsymbol{U}$ and

$$\boldsymbol{M} = \boldsymbol{U}\,\boldsymbol{\Sigma}\,\boldsymbol{V}^T = \boldsymbol{u}_1\sigma_1\boldsymbol{v}_1^T + \ldots + \boldsymbol{u}_r\sigma_r\boldsymbol{v}_r^T.$$

The pseudoinverse of M now can easily be computed by

$$\boldsymbol{M}^+ = \boldsymbol{V}\,\boldsymbol{\Sigma}^+\,\boldsymbol{U}^T = \boldsymbol{v}_1\frac{1}{\sigma_1}\boldsymbol{u}_1^T + \ldots + \boldsymbol{v}_r\frac{1}{\sigma_r}\boldsymbol{u}_r^T. \tag{8.14}$$

with the $n \times m$ matrix

$$\boldsymbol{\Sigma}^+ = \begin{pmatrix}1/\sigma_1 & & & & 0\\ & \ddots & & & \\ & & 1/\sigma_r & & \\ & & & 0 & \\ & & & & \ddots\\ 0 & & \ldots & & 0\\ \vdots & & & & \vdots\\ 0 & & \ldots & & 0\end{pmatrix}.$$

Summary
The simplest way to compute the SVD is

- $\boldsymbol{U} \in \mathbb{R}^{m \times m}$: Eigenvector matrix of $\boldsymbol{M}\boldsymbol{M}^T$.
- $\boldsymbol{\Sigma} \in \mathbb{R}^{m \times n}$ being the positive square roots of the eigenvalues of either $\boldsymbol{M}\boldsymbol{M}^T$ or $\boldsymbol{M}^T\boldsymbol{M}$.
- $\boldsymbol{V} \in \mathbb{R}^{n \times n}$: Eigenvector matrix of $\boldsymbol{M}^T\boldsymbol{M}$.
- Substitute $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{\Sigma}$ in equation 8.14 to get $\boldsymbol{M}^+$.

Regularized Version of SVD

- after applying SVD we get $\boldsymbol{M}^+ = \boldsymbol{V}\boldsymbol{\Sigma}^+\boldsymbol{U}^T$.
- To solve $\boldsymbol{M} \cdot \boldsymbol{a} = \boldsymbol{y}$ for $\boldsymbol{a}$ we approximate $\boxed{\hat{\boldsymbol{a}} = \boldsymbol{M}^+\boldsymbol{y}}$

With regularization term:
choose a parameter $\gamma > 0$ and solve

$$\boxed{\hat{\boldsymbol{a}} = (\gamma \boldsymbol{I} + \boldsymbol{M}^+\boldsymbol{M})^{-1}\boldsymbol{M}^+\boldsymbol{y}}$$

Example
Find the SVD decomposition of the matrix $\boldsymbol{M} = \begin{pmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{pmatrix}$.

$$\boldsymbol{M}\boldsymbol{M}^T = \begin{pmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \end{pmatrix} \cdot \begin{pmatrix} 3 & 2 \\ 2 & 3 \\ 2 & -2 \end{pmatrix} = \begin{pmatrix} 17 & 8 \\ 8 & 17 \end{pmatrix} \tag{8.15}$$

The characteristic polynomial is the determinant $|\boldsymbol{M}\boldsymbol{M}^T - \lambda I|$. Thus we first have to calculate $\boldsymbol{M}\boldsymbol{M}^T - \lambda I$,

$$\boldsymbol{M}\boldsymbol{M}^T - \lambda I = \begin{pmatrix} 17 - \lambda & 8 \\ 8 & 17 - \lambda \end{pmatrix} \tag{8.16}$$

The determinant is

$$|\boldsymbol{M}\boldsymbol{M}^T - \lambda I| = \lambda^2 - 34\lambda + 225 = (\lambda - 25)(\lambda - 9) \tag{8.17}$$

The eigenvalues of MM^T are $\sigma_1^2 = 25$ and $\sigma_2^2 = 9$. This means in Σ we have $\sigma_1 = \sqrt{25} = 5$ and $\sigma_2 = \sqrt{9} = 3$. To obtain the eigenvector of $\boldsymbol{M}\boldsymbol{M}^T$ for $\sigma_1^2 = 25$ solve $(\boldsymbol{M}\boldsymbol{M}^T - \lambda I)\boldsymbol{u_1} = 0$,

$$(\boldsymbol{M}\boldsymbol{M}^T - \lambda_1 I)\boldsymbol{u_1} = \begin{pmatrix} -8 & 8 \\ 8 & -8 \end{pmatrix} \boldsymbol{u_1} = 0 \tag{8.18}$$

An obvious eigenvector of the previous matrix is $(1\ 1)^T$. Normalizing this vector we attain $\boldsymbol{u_1} = (\frac{1}{\sqrt{2}}\ \frac{1}{\sqrt{2}})^T$. For the second eigenvalue $\sigma_2^2 = 9$, we proceed in the same way and we will find that $\boldsymbol{u_2} = (\frac{1}{\sqrt{2}}\ -\frac{1}{\sqrt{2}})^T$, is the second eigenvector of $\boldsymbol{M}\boldsymbol{M}^T$. Till now we have found the matrix U and Σ in equation **??**. To solve for V use $\boldsymbol{M}^T\boldsymbol{M}$. The eigenvalues of

$\boldsymbol{M}^T\boldsymbol{M}$ are 25, 9 and 0, and since $\boldsymbol{M}^T\boldsymbol{M}$ is symmetric we know that the eigenvectors will be orthogonal.
For $\lambda = 25$, we have

$$\boldsymbol{M}^T\boldsymbol{M} - 25I = \begin{pmatrix} -12 & 12 & 2 \\ 12 & -12 & -2 \\ 2 & -2 & -17 \end{pmatrix} \tag{8.19}$$

which row-reduces to $\begin{pmatrix} 1 & -1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$.
An eigenvector is $\boldsymbol{v_1} = (\frac{1}{\sqrt{2}} \ \frac{1}{\sqrt{2}} \ 0)^T$.
For $\lambda = 9$, we have

$$\boldsymbol{M}^T\boldsymbol{M} - 9I = \begin{pmatrix} 4 & 12 & 2 \\ 12 & 4 & -2 \\ 2 & -2 & -1 \end{pmatrix} \tag{8.20}$$

which row reduces to $\begin{pmatrix} 1 & 0 & -\frac{1}{4} \\ 0 & 1 & \frac{1}{4} \\ 0 & 0 & 0 \end{pmatrix}$. An eigenvector is $\boldsymbol{v_2} = (\frac{1}{\sqrt{18}} \ -\frac{1}{\sqrt{18}} \ \frac{4}{\sqrt{18}})^T$.
For the last eigenvector $\lambda_3 = 0$, we could find a unit vector perpendicular to $\boldsymbol{v_1}$ and $\boldsymbol{v_2}$ or solve $(\boldsymbol{M}^T\boldsymbol{M} - \lambda_3 I)\boldsymbol{v_3} = 0$, then we deduce that $\boldsymbol{v_3} = (\frac{2}{3} \ -\frac{2}{3} \ \frac{-1}{3})^T$. So the full SVD of our matrix $\boldsymbol{M}$ could now be written as,

$$\boldsymbol{M} = U\Sigma V^T = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 5 & 0 & 0 \\ 0 & 3 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{18}} & -\frac{1}{\sqrt{18}} & \frac{4}{\sqrt{18}} \\ \frac{2}{3} & -\frac{2}{3} & -\frac{1}{3} \end{pmatrix}$$

The pseudoinverse of $\boldsymbol{M}$ is

$$\boldsymbol{M}^+ = V\Sigma^+ U^T = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{18}} & \frac{2}{3} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{18}} & \frac{-2}{2} \\ 0 & \frac{4}{\sqrt{18}} & -\frac{1}{3} \end{pmatrix} \begin{pmatrix} \frac{1}{5} & 0 \\ 0 & \frac{1}{3} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

Linear Regression

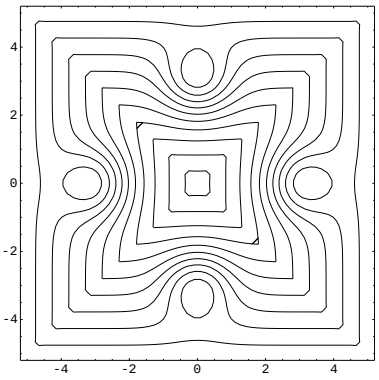
- Linear Regression: Estimate Parameters for

$$f(\boldsymbol{x}) = a_1 f_1(\boldsymbol{x}) + \cdots + a_k f_k(\boldsymbol{x}) = \boldsymbol{a}^T \boldsymbol{f}(\boldsymbol{x})$$

- Constraints: $f(\boldsymbol{x}_i) = \boldsymbol{a}^T \boldsymbol{f}(\boldsymbol{x}_i) = y_i$
- $\boldsymbol{M} \cdot \boldsymbol{a} = \boldsymbol{y}$ with $\boldsymbol{M}_{ij} = f_j(\boldsymbol{x}_i)$.
- Overdetermined! No Solution
- Minimize $E = ||\boldsymbol{M}\,\boldsymbol{a} - \boldsymbol{y}||_2$
- Error E on data must become a Minimum: $\nabla_{\boldsymbol{a}} E = 0$
- Solution $\hat{\boldsymbol{a}} = (\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{y}$

Nonlinear Regression

- Error E on data must become a Minimum: $\nabla_{\boldsymbol{a}} E = 0$
- $\nabla_{\boldsymbol{a}} E = 0$ is nonlinear!
- Solution: Gradient descent!
- Adjust $\boldsymbol{a}$ in the direction of steepest descent!



8.5 Exercises

Exercise 8.1 Given M,

$$M = \begin{pmatrix} 8 & 2 & 2 \\ 2 & 4 & 1 \end{pmatrix}$$

a) Perform the SVD decomposition and write M in the form $M = U\Sigma V^T$.

b) Compute the pseudoinverse M^+ of M.

c) Show that M^+ is a valid (Moore-Penrose) pseudoinverse.

d) Show that the pseudoinverse of M, using the technique of the underdertemined system mentioned in section 6.3.8, is the same as the one computed by SVD.

Exercise 8.2 Given the following Matrix M,

$$M = \begin{pmatrix} 3 & 6 \\ 2 & 4 \\ 2 & 4 \end{pmatrix}$$

a) Show that the pseudoinverse of the matrix M, using the technique of the overdetermined system mentioned in section 6.3.7, is not applicable.

b) Perform the SVD decomposition and write M in the form $M = U\Sigma V^T$.

c) Compute the pseudoinverse M^+ of M.

d) Show that M^+ is a valid pseudoinverse.

Exercise 8.3 Prove:

a) $\boldsymbol{M}^+ = \boldsymbol{V}\boldsymbol{\Sigma}^+\boldsymbol{U}^T$ is a Moore-Penrose-Pseudoinverse of $\boldsymbol{M}$.

b) $\boldsymbol{\Sigma}^+$ is the pseudoinverse of $\boldsymbol{\Sigma}$, i.e. that $\boldsymbol{\Sigma}^+ = (\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}^T$.

Exercise 8.4 Repeat your function approximation experiments from exercise **??** using SVD. Report about your results.

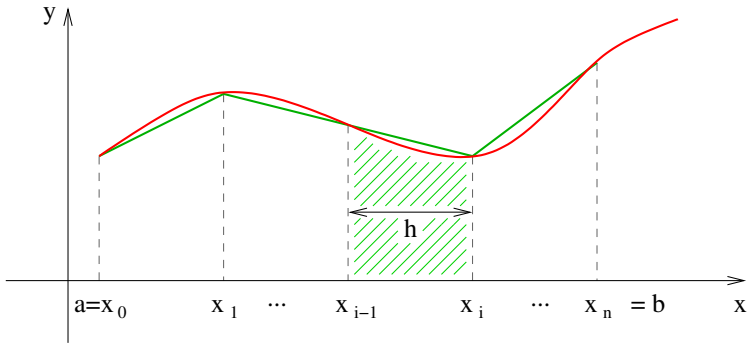
Chapter 9

Numerical Integration and Solution of Ordinary Differential Equations

9.1 Numerical Integration

Numerical integration is very important in applications, but the analytical (symbolic) integration is always preferable, if possible.

The Trapezoidal Rule



Equidistant partition of $[a,b]$ by $x_0 = a,\ x_1 = a+h,\ x_2 = a+2h, ..., x_n = a+nh = b$

Step size: $h = \dfrac{(b-a)}{n}$

Approximation : $\displaystyle\int_{x_{i-1}}^{x_i} f(x)\,dx \approx$Area of a trapezoid $= h \cdot \frac{f(x_{i-1})+f(x_i)}{2}$

Theorem 9.1 (Trapezoidal Rule) Let $f : [a,b] \to \mathbb{R}$ twice continuously differentiable . Then it holds

$$\int_a^b f(x)\,dx = \underbrace{h \cdot \Big(\frac{f(x_0)}{2} + f(x_1) + ... + f(x_{n-1}) + \frac{f(x_n)}{2}\Big)}_{T(h)} - \Delta T(h)$$

with $\quad |\Delta T(h)| \le \dfrac{(b-a)h^2}{12} \max\limits_{x\in[a,b]} \{|f''(x)|\}$

Proof: From Theorem 6.2 we know that the approximation error for polynomial interpolation of the function f on the $n+1$ points $x_0, \ldots, x_n$ by a polynomial p of degree n is given by

$$f(x) - p(x) = \frac{f^{(n+1)}(z)}{(n+1)!}(x - x_0)(x - x_1) \cdots (x - x_n)$$

for a point $z \in [a, b]$. For linear interpolation of f with two points x_{i-1}, x_i this yields

$$f(x) = p(x) + \frac{f''(z_i)}{2}(x - x_{i-1})(x - x_i)$$

for $z_i \in [x_{i-1}, x_i]$.Applying this to the error of the trapezoidal rule on **one sub-interval** $[x_{i-1}, x_i]$ **only** we get:

$$\begin{aligned}\varepsilon_i = \Delta T(h) &= T(h) - \int_{x_{i-1}}^{x_i} f(x)\,dx = T(h) - \int_{x_{i-1}}^{x_i} p(x)\,dx - \int_{x_{i-1}}^{x_i} \frac{f''(z_i)}{2}(x - x_{i-1})(x - x_i)\,dx \\ &= -\frac{f''(z_i)}{2} \int_{x_{i-1}}^{x_i} (x - x_{i-1})(x - x_i)\,dx.\end{aligned}$$

Substituting $x = x_{i-1} + ht$ we evaluate

$$\int_{x_{i-1}}^{x_i} (x - x_{i-1})(x - x_i)\,dx = h^3 \int_0^1 t(t-1)\,dt = -\frac{h^3}{6}$$

and get

$$\varepsilon_i = \frac{f''(z_i)h^3}{12}.$$

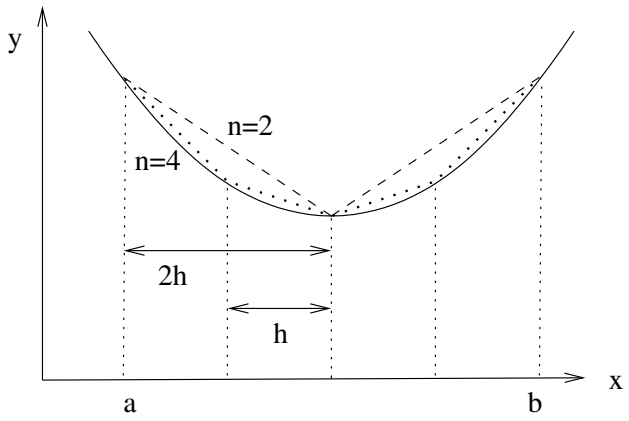
For the trapezoidal rule on the whole interval $[a, b]$ we get

$$\begin{aligned}|\Delta T(h)| &= \left|\sum_{i=1}^{n} \varepsilon_i\right| \le \sum_{i=1}^{n} |\varepsilon_i| = \sum_{i=1}^{n} \frac{|f''(z_i)|h^3}{12} \le \sum_{i=1}^{n} \frac{h^3}{12} \max_{x\in[a,b]}\{|f''(x)|\} \\ &= \frac{nh^3}{12} \max_{x\in[a,b]}\{|f''(x)|\} = \frac{(b-a)h^2}{12} \max_{x\in[a,b]}\{|f''(x)|\}\end{aligned}$$

and the proof is complete. □

Richardson Extrapolation

Note: Halving of h ($2h \to h$) doubles the computational effort ($2n$ function evaluations). The error is reduced by factor 4: $\Delta T(2h) \approx 4\Delta T(h)$



$$T(h) \approx \int_a^b f(x)dx + ch^2 = \int_a^b f(x)dx + \Delta T(h)$$
$$T(2h) \approx \int_a^b f(x)dx + \mathbf{4}ch^2 = \int_a^b f(x)dx + 4\Delta T(h)$$

$$T(2h) - T(h) \approx 3\Delta T(h) \quad \Rightarrow \quad \Delta T(h) \approx \frac{1}{3}\left(T(2h) - T(h)\right)$$

$$\Rightarrow \int_a^b f(x)dx = T(h) - \Delta T(h) \approx T(h) - \left[\frac{1}{3}\left(T(2h) - T(h)\right)\right]$$

$$\boxed{\int_a^b f(x)dx \approx \frac{4}{3}T(h) - \frac{1}{3}T(2h)}$$

This formula gives a better approximation than T(h) and is called Richardson Extrapolation.

Repeated Richardson Extrapolation
We can generalize the Richardson Extrapolation to any calculation where we know the asymptotic behaviour of some function F to be calculated for $h \to 0$ as

$$F(h) = a_0 + a_1 h^p + O(h^r),$$

where $a_0 = F(0)$ is the desired value, a_1 is unknown and $p < r$. Suppose we know F for h and qh:

$$F(h) = a_0 + a_1 h^p + O(h^r),$$
$$F(qh) = a_0 + a_1 (qh)^p + O(h^r),$$

Solving for a_0 yields

$$F(0) = a_0 = F(h) + \frac{F(h) - F(qh)}{q^p - 1} + O(h^r)$$

This formula leads to a reduction of the error from $O(h^p)$ to $O(h^r)$.

Theorem 9.2 If we know the complete expansion of F as

$$F(h) = a_0 + a_1 h^{p_1} + a_2 h^{p_2} + a_3 h^{p_3} + \ldots,$$

we recursively compute

$$F_1(h) = F(h) \quad \text{and} \quad F_{k+1}(h) = F_k(h) + \frac{F_k(h) - F_k(qh)}{q^{p_k} - 1}$$

Then $F_n(h) = a_0 + a_n^{(n)} h^{p_n} + a_{n+1}^{(n)} h^{p_{n+1}} + \ldots$.

An inductive proof can be found e.g. in [27].

The Rhomberg Method
It can be shown [27] that for the trapezoidal rule we have

$$T(h) = \int_a^b f(x)dx + a_1 h^2 + a_2 h^4 + a_3 h^6 + \ldots$$

We apply repeated Richardson extrapolation with $q = 2$:

$$T_1(h) = T(h)$$
$$T_{k+1}(h) = T_k(h) + \frac{\Delta_k}{2^{2k} - 1} \quad \text{with} \quad \Delta_k = T_k(h) - T_k(2h)$$

Example 9.1 We want to approximate

$$\int_0^{0.8} \frac{\sin x}{x} dx$$

and get

h	$T_1(h)$	$\Delta_1/3$	$T_2(h)$	$\Delta_2/15$	$T_3(h)$	$\Delta_3/63$	$T_4(h)$
0.8	0.758678						
		0.003360					
0.4	0.768757		0.77211714				
		0.000835		−0.00000133			
0.2	0.771262		0.77209711		0.772095771		
		0.000208		−0.00000008		$2.26 \cdot 10^{-10}$	
0.1	0.771887		0.77209587		0.7720957853		0.772095785485

The exact solution is $\int_0^{0.8} \frac{\sin x}{x} dx \approx 0.7720957854820$. We see that $T_4(0.1)$ is a much better approximation than $T_1(0.1)$.

Alternative Methods
We briefly sketch two alternative methods for approximating definite integrals. They are examples of the so called Monte-Carlo methods (they work with random numbers).
For many complex applications e.g. the modeling by Differential Equations is either not possible or too computationally intensive. A solution is the direct simulation of each process using a stochastic model. Such models are used in the areas

- Static Shysics (Many Particle Physics)
- Hydrodynamics
- Meteorology
- Road Traffic
- Waiting Queue Systems

We give two simple examples of randomized methods for approximating integrals.

Method 1
Calculating the area under a curve (see Figure 9.1)[1ex]

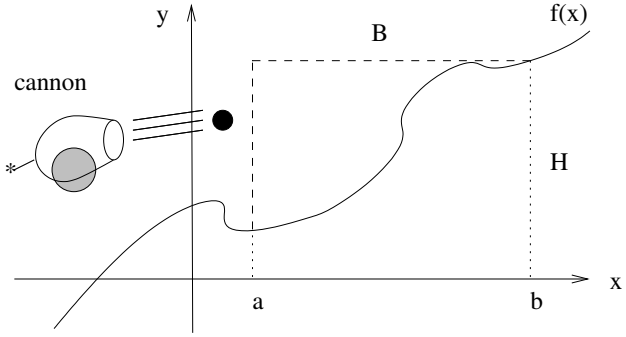


Figure 9.1: Area calculation using the Monte Carlo Method.

$$\int_a^b f(x)dx \approx \frac{\text{Number of hits under the curve}}{\text{Number of hits inside the rectangle}} \cdot B \cdot H$$

Method 2
Following the mean value theorem of integration it holds

$$\int_a^b f(x)\,dx = (b-a)\cdot M, \tag{9.1}$$

where M is the mean of f in the interval $[a,b]$. Now, we discretize the interval with the given points $x_1,\ldots,x_n$ and calculate the mean of f on the given points according to

$$A = \frac{1}{n}\sum_{i=1}^{n} f(x_i).$$

Due to the definition of the Riemann integral, only for fine discretization $A \approx M$ holds. Therewith M of (9.1) can be replaced by A yielding

$$\int_a^b f(x)\,dx = \frac{b-a}{n}\sum_{i=1}^{n} f(x_i).$$

The given points x_i should be chosen randomly. (why?)
For one-dimensional integrals both presented methods are clearly inferior to the trapezoidal rule. However, in higher dimensions, the advantages show up in the form of much shorter computing times.

9.2 Numerical Differentiation

First Derivative

- Goal: compute numerically $f'(a)$ at some point $x = a$
- Idea: approximate the derivative by a finite difference quotient (see Figure 9.2):

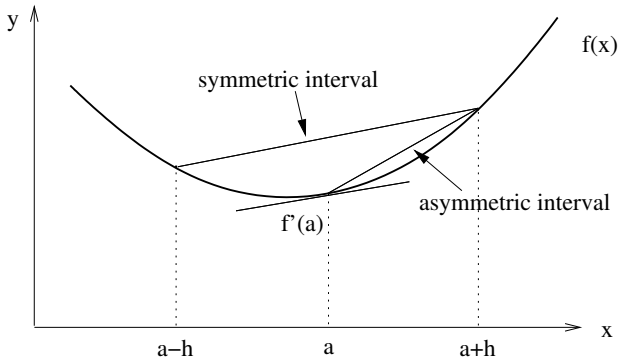


Figure 9.2: Central Difference.

$$f'(x) = \lim_{\substack{h \to 0 \\ h \neq 0}} \frac{f(x+h) - f(x)}{h} \approx \frac{f(x+h) - f(x)}{h}$$

First Derivative: Approximation Error

- How does the approximation error depend on h?

Taylor Expansion of f in $x_0 = a$:

$$f(a+h) = f(a) + f'(a)h + \frac{1}{2!}f''(a)h^2 + \frac{1}{3!}f'''(a)h^3 + \dots$$

Division by h gives

$$\frac{f(a+h) - f(a)}{h} = f'(a) + \frac{1}{2!}f''(a)h + \frac{1}{3!}f'''(a)h^2 + \dots = f'(a) + O(h)$$

thus proving

Theorem 9.3 Let $f : \mathbb{R} \to \mathbb{R}$ two times continuously differentiable. Then the error of the asymmetric difference decreases linearly with h, i.e.

$$\frac{f(a+h) - f(a)}{h} = f'(a) + O(h).$$

Central difference

$$f'(x) = \lim_{\substack{h\to 0\\ h\neq 0}} \frac{f(x+h)-f(x-h)}{2h} \approx \frac{f(x+h)-f(x-h)}{2h}$$

- Is the central difference asymptotically better?

Taylor Expansion of f in $x_0 = a$:

$$f(a+h) = f(a) + f'(a)h + \frac{1}{2!}f''(a)h^2 + \frac{1}{3!}f'''(a)h^3 + \dots \tag{9.2}$$

$$f(a-h) = f(a) - f'(a)h + \frac{1}{2!}f''(a)h^2 - \frac{1}{3!}f'''(a)h^3 + \dots \tag{9.3}$$

Subtracting (9.3) from (9.2) and dividing by $2h$ leads to

$$\begin{aligned}\frac{f(a+h)-f(a-h)}{2h} &= f'(a) + \frac{1}{3!}f'''(a)h^2 + \frac{1}{5!}f^{(5)}(a)h^4 + \frac{1}{7!}f^{(7)}(a)h^6 + \dots \\ &= f'(a) + O(h^2)\end{aligned}$$

thus proving

Theorem 9.4 Let $f : \mathbb{R} \to \mathbb{R}$ three times continuously differentiable. Then the error of the symmetric difference decreases quadratically with h, i.e.

$$\frac{f(a+h)-f(a-h)}{2h} = f'(a) + O(h^2).$$

Example 9.2 We compute the central difference with repeated Richardson Extrapolation on the function $f(x) = 1/x$ in $x = 1$ with $h = 0.8, 0.4, 0.2, 0.1, 0.05, 0.025$:

h	$F_1(h)$	$F_2(h)$	$F_3(h)$	$F_4(h)$	$F_5(h)$	$F_6(h)$
0.8	-2.777778					
0.4	-1.190476	-0.661376				
0.2	-1.041667	-0.992063	-1.01410935			
0.1	-1.010101	-0.999579	-1.00008017	-0.999857481		
0.05	-1.002506	-0.999975	-1.00000105	-0.999999799	-1.00000036	
0.025	-1.000625	-0.999998	-1.000000016	-0.99999999934	-1.0000000001	-0.9999999998

$\Delta_1/3$	$\Delta_2/15$	$\Delta_3/63$	$\Delta_4/255$	$\Delta_5/1023$
0.529101				
0.049603	-0.0220459			
0.010522	-0.0005010	0.000222685		
0.002532	-0.0000264	0.000001256	-0.0000005581	
0.000627	-0.0000016	0.000000016	-0.0000000008	0.0000000003

Second Derivative

$$f''(x) = \lim_{h\to 0} \frac{f'(x+\frac{h}{2}) - f'(x-\frac{h}{2})}{h} = \lim_{h\to 0} \frac{f(x+h) - f(x) - f(x) + f(x-h)}{h^2}$$
$$\approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

The approximation error can easily be shown to decrease quadratically with h by adding (9.3) to (9.2):

$$\frac{f(a+h) - 2f(a) + f(a-h)}{h^2} = +\frac{2}{2!}f''(a) + \frac{2}{4!}f^{(4)}(a)h^2 + \frac{2}{6!}f^{(6)}(a)h^4 + \dots$$

It can be shown ([27], chapter 7), that, if we (recursively) use symmetric formulas for higher derivatives, the approxiamtion error contains only even powers of h. As a consequence, the same Richardson extrapolation scheme can be applied.

9.3 Numerical Solution of Ordinary Differential Equations

We will use the common shorthand ODE for ordinary differential equation.

Initial Value Problems for Systems of ODEs
Given a function $f(x, y)$, we want to find a function $y(x)$ on an interval $[a, b]$ which is an approximate solution of the first order ODE

$$\frac{dy}{dx} = f(x, y) \quad \text{with the initial condition} \quad y(a) = c$$

The order of a differential equation is the degree of the highest derivative occuring in the equation. If f is linear, then there are symbolic solutions.
Many applications can be modelled by systems of first order ODEs

$$\frac{d\eta_i}{dx} = \phi_i(x, \eta_1, \dots, \eta_s) \quad (i = 1, \dots, s)$$

for the unknown functions $\eta_1(x), \dots, \eta_s(x)$ with the initial contitions

$$\eta_i(a_i) = \gamma_i \quad (i = 1, \dots, s)$$

Such a system can be written in vector form. With

$$\boldsymbol{y} = (\eta_1(x), \dots, \eta_s(x))^T$$
$$\boldsymbol{c} = (\gamma_1(x), \dots, \gamma_s(x))^T$$
$$\boldsymbol{f} = (\phi_1(x), \dots, \phi_s(x))^T$$

the systems reads

$$\frac{d\boldsymbol{y}}{dx} = \boldsymbol{f}(x, \boldsymbol{y}), \quad \boldsymbol{y}(\boldsymbol{a}) = \boldsymbol{c}.$$

Example 9.3 ODEs of higher order can be transformed into a system of first order ODEs. For the third order ODE

$$d^3y/dx^3 = g(x, y, dy/dx, d^2y/dx^2)$$

with the initial conditions

$$y(0) = \gamma_1, \quad y'(0) = \gamma_2, \quad y''(0) = \gamma_3$$

we substitute

$$\eta_1 = y, \quad \eta_2 = dy/dx, \quad \eta_3 = d^2y/dx^2$$

and get

$$\begin{aligned} d\eta_1/dx &= \eta_2, & \eta_1(0) &= \gamma_1 \\ d\eta_2/dx &= \eta_3, & \eta_2(0) &= \gamma_2 \\ d\eta_3/dx &= g(x, y, \eta_1, \eta_2, \eta_3), & \eta_3(0) &= \gamma_3 \end{aligned}$$

Theorem 9.5 Any system of ODEs can be transformed into an equivalent system of ODEs with derivatives of order one only.

The Euler Method

We discretize the interval $[a, b]$ into subintervals of width h by

$$x_i = a + ih \quad (i = 0, 1, \ldots) \quad \text{and} \quad \boldsymbol{y}_0 = \boldsymbol{y}(a) = \boldsymbol{c}$$

and we want to compute the values $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots$ as an approximation for the exact values $y(x_1), y(x_2), \ldots$. We approximate the system of ODEs by

$$\frac{d\boldsymbol{y}}{dx} \approx \frac{\boldsymbol{y_{n+1}} - \boldsymbol{y_n}}{h} = \boldsymbol{f}(x_n, \boldsymbol{y}_n)$$

yielding the recursion

$$\boxed{\boldsymbol{y}_0 = \boldsymbol{c}, \quad \boldsymbol{y}_{n+1} = \boldsymbol{y}_n + h\boldsymbol{f}(x_n, \boldsymbol{y}_n), \quad (n = 1, 2, \ldots)}$$

The approximation error of the Euler method can be estimated using the Taylor expansion

$$\boldsymbol{y}(x_{n+1}) = \boldsymbol{y}(x_n) + \boldsymbol{y}'(x_n) \cdot h + \frac{\boldsymbol{y}''}{2!}h^2 + \frac{\boldsymbol{y}'''}{3!}h^3 + \ldots$$

The error then is

$$\frac{\boldsymbol{y}(x_{n+1}) - \boldsymbol{y}(x_n)}{h} - \boldsymbol{y}'(x_n) = \frac{\boldsymbol{y}''}{2!}h + \frac{\boldsymbol{y}'''}{3!}h^2 + \ldots.$$

One can thus apply Richardson Extrapolation with $p_k = k$.

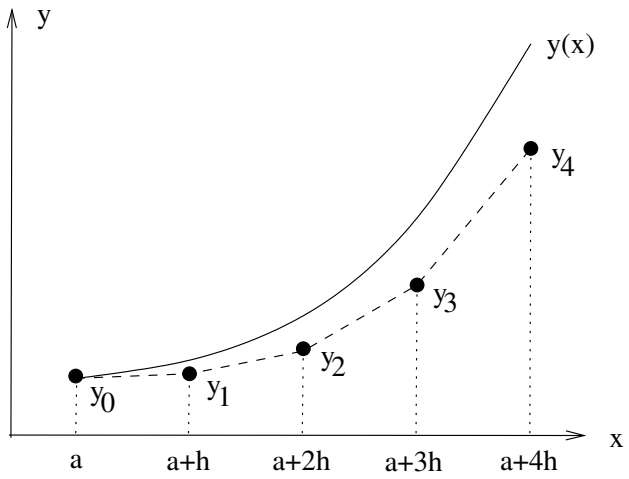


Figure 9.3: Solution polygon of the Euler method..

		$h = 0.1$		$h = 0.2$	
x_n	$y(x_n)$	y_n	error	y_n	error
0	1.00	1.00	0	1.00	0
0.1	1.105	1.1	0.005		
0.2	1.221	1.21	0.011	1.2	0.021
0.3	1.350	1.331	0.019		
0.4	1.492	1.464	0.028	1.44	0.052
0.5	1.649	1.611	0.038		
0.6	1.822	1.772	0.050	1.728	0.094

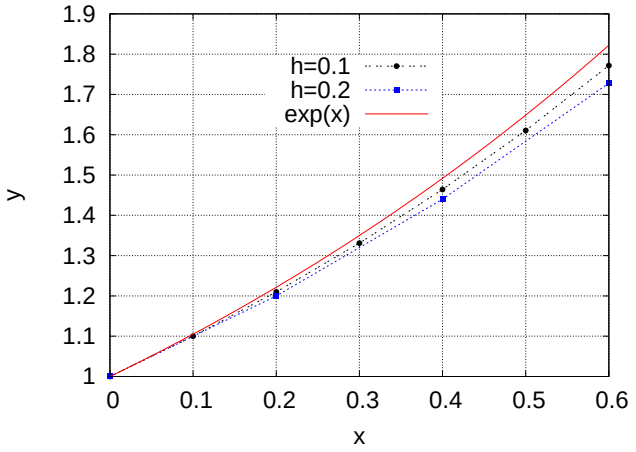


Figure 9.4: Results of the Euler method applied to the ODE $y' = y$ with $y(0) = 1$ for $h = 0.1$ and $h = 0.2$.

Runge-Kutta Methods

The error of the Euler method is due to the linear approximation of $\boldsymbol{y}(x)$ in x_n as can be seen in Figure 9.3. This can be improved by averaging over an appropriately chosen combination of values of the function $\boldsymbol{f}(x, \boldsymbol{y})$. The simplest formula of this type, the **Heun Method** uses a symmetric average of $\boldsymbol{f}(x_n)$ and $\boldsymbol{f}(x_{n+1})$ with the consequence that $(\boldsymbol{y}_{n+1} - \boldsymbol{y}_n)/h$ is effectively used as a symmetric approximation of $d\boldsymbol{y}/dx$ in $x_n + h/2$:

$$\frac{d\boldsymbol{y}}{dx} \approx \frac{\boldsymbol{y_{n+1}} - \boldsymbol{y_n}}{h} = \frac{1}{2}(\boldsymbol{f}(x_n, \boldsymbol{y}_n) + \boldsymbol{f}(x_{n+1}, \boldsymbol{y}_n + h\boldsymbol{f}(x_n, \boldsymbol{y}_n)))$$

Solving this for $\boldsymbol{y}_{n+1}$ leads to the recursion scheme

$$\begin{aligned} \boldsymbol{k_1} &= h\boldsymbol{f}(x_n, \boldsymbol{y}_n) \\ \boldsymbol{k_2} &= h\boldsymbol{f}(x_n + h, \boldsymbol{y}_n + \boldsymbol{k_1}) \\ \boldsymbol{y_{n+1}} &= \boldsymbol{y_n} + \frac{1}{2}(\boldsymbol{k_1} + \boldsymbol{k_2}) \end{aligned}$$

We use the notion $\boldsymbol{y}(x, h)$ for the numeric result with step width h obtained from applying the recursion scheme. We get a quadratic approximation error

$$\boldsymbol{y}(x, h) = \boldsymbol{y}(x) + \boldsymbol{c_2}(x)h^2 + \boldsymbol{c_3}(x)h^3 + \boldsymbol{c_4}(x)h^4 + \ldots$$

with the exponents $p_k = 2, 3, 4, 5, \ldots$ for Richardson extrapolation.
An even better scheme, known as **fourth order Runge Kutta** or *classical Runge Kutta* is

$$\begin{aligned}
\boldsymbol{k_1} &= h\boldsymbol{f}(x_n, \boldsymbol{y}_n) \\
\boldsymbol{k_2} &= h\boldsymbol{f}(x_n + \frac{1}{2}h, \boldsymbol{y}_n + \frac{1}{2}\boldsymbol{k_1}) \\
\boldsymbol{k_3} &= h\boldsymbol{f}(x_n + \frac{1}{2}h, \boldsymbol{y}_n + \frac{1}{2}\boldsymbol{k_2}) \\
\boldsymbol{k_4} &= h\boldsymbol{f}(x_n + h, \boldsymbol{y}_n + \boldsymbol{k_3}) \\
\boldsymbol{y_{n+1}} &= \boldsymbol{y_n} + \frac{1}{6}(\boldsymbol{k_1} + 2\boldsymbol{k_2} + 2\boldsymbol{k_3} + \boldsymbol{k_4})
\end{aligned}$$

with the approximation error

$$\boldsymbol{y}(x, h) = \boldsymbol{y}(x) + \boldsymbol{c_4}(x)h^4 + \boldsymbol{c_3}(x)h^5 + \ldots$$

and $p_k = 4, 5, 6, \ldots$
Figure 9.5 shows a comparison between the three yet presented methods for solving first order initial value problems. It clearly confirms the theoretical results wrt. the approximation error which are:
Euler method: $O(h)$, Heun method: $O(h^2)$, Runge Kutta $O(h^4)$

		Euler method		Heun method		Runge Kutta	
x_n	$y(x_n)$	y_n	error	y_n	error	y_n	error
0	1.00	1.00	0	1.00	0	1.00	0
0.1	1.10517	1.1	0.005	1.105	0.00017	1.10517	$8.5 \cdot 10^{-8}$
0.2	1.22140	1.21	0.011	1.22103	0.00038	1.22140	$1.9 \cdot 10^{-7}$
0.3	1.34986	1.33	0.019	1.34923	0.00063	1.34986	$3.1 \cdot 10^{-7}$
0.4	1.49182	1.46	0.028	1.4909	0.00092	1.49182	$4.6 \cdot 10^{-7}$
0.5	1.64872	1.61	0.038	1.64745	0.00127	1.64872	$6.3 \cdot 10^{-7}$
0.6	1.82212	1.77	0.051	1.82043	0.00169	1.82212	$8.4 \cdot 10^{-7}$

Figure 9.5: Comparison of Euler method, Heun method and Runge Kutta applied to the ODE $y' = y$ with $y(0) = 1$ and $h = 0.1$.

Often the selection of an appropriately small step size h is critical for good results of all described methods. This can be automatized with methods that adapt the step size (see [12]).

Example 9.4 We want to solve a classical predator prey system from biology. $y_1(t)$ may be a population of sheep and $y_2(t)$ a population of wolves. With no wolves the sheeps breed nicely. Breeding of the wolves increases monotonically with the number of wolves and sheep. But with no sheep, wolves will die out. The ODEs from Lotka-Volterra are [12]:

$$\begin{aligned}
\dot{y}_1(t) &= \alpha y_1(t)(1 - y_2(t)) \\
\dot{y}_2(t) &= y_2(t)(y_1(t) - 1)
\end{aligned}$$

With the Runge Kutta method we can easily compute the population dynamics for this system. A sample plot is shown in Figure 9.6.

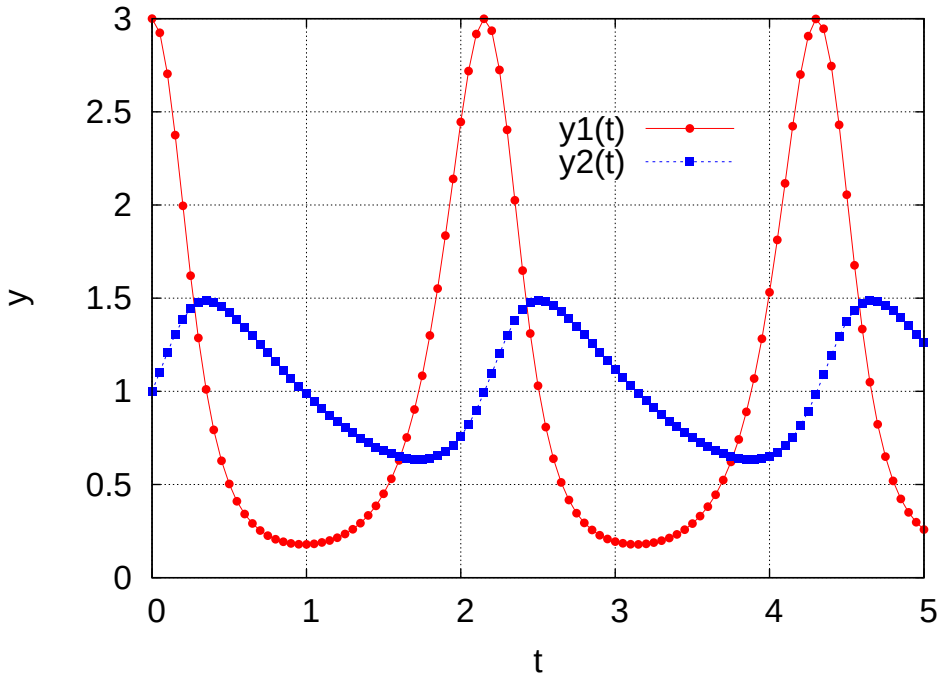


Figure 9.6: Population dynamics for $\alpha = 10$, $t = 0, \ldots, 5$ $h = 0.05$.

Boundary Value Problems for Second Order ODEs

As already mentioned in example 9.3, whenever a second order ODE can be written as

$$y'' = f(x, y, y'),$$

it can be transformed into a system of two first order ODEs and then be solved with the methods already described. We will now sketch ideas for a direct solution of scalar second order boundary value problems of the form

$$y'' = f(x, y, y') \text{ with the boundary conditions } y(a) = \alpha, y(b) = \beta.$$

We discretize the derivatives by

$$y'(x_n) \approx \frac{y_{n+1} - y_{n-1}}{2h} \quad \text{and} \quad y''(x_n) \approx \frac{y_{n+1} - 2y_n + y_{n-1}}{h^2}$$

on the interval $[a, b]$ with $b - a = mh$ and $x_i = a + ih$. y_i is the approximation of $y(x_i)$. We obtain the (typically nonlinear) system of equations

$$\begin{aligned} y_0 &= \alpha \\ y_{n+1} - 2y_n + y_{n-1} &= h^2 f(x_n, y_n, \frac{y_{n+1} - y_{n-1}}{2h}), \quad (n = 1, 2, 3, \ldots m-1) \\ y_m &= \beta. \end{aligned}$$

With $\boldsymbol{f} = (f_1, \ldots, f_{m-1})^T$ and

$$f_n = f(x_n, y_n, \frac{y_{n+1} - y_{n-1}}{2h})$$

we can write the system in matrix form

$$\boldsymbol{A}\boldsymbol{y} = h^2 \boldsymbol{f}(\boldsymbol{y}) - \boldsymbol{r} \tag{9.4}$$

with

$$\boldsymbol{A} = \begin{pmatrix} -2 & 1 & 0 & 0\cdots & 0 \\ 1 & -2 & 1 & 0\cdots & 0 \\ 0 & 1 & -2 & 1\cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0\cdots & 1 & -2 & 1 \\ 0 & 0\cdots & 0 & 1 & -2 \end{pmatrix}, \quad \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ \vdots \\ y_{m-1} \end{pmatrix}, \quad \boldsymbol{f}(\boldsymbol{y}) = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ \vdots \\ f_{m-1} \end{pmatrix}, \quad \boldsymbol{r} = \begin{pmatrix} \alpha \\ 0 \\ 0 \\ \vdots \\ 0 \\ \beta \end{pmatrix}.$$

If the differential equation is linear, this is a linear system that can be solved in linear time with the tridiagonal algorithm described in Section 6.2.2. Since we used symmetric approximation formulas for the derivatives, the approximation error is

$$\boldsymbol{y}(x,h) = \boldsymbol{y}(x) + \boldsymbol{c_1}(x)h^2 + \boldsymbol{c_2}(x)h^4 + \boldsymbol{c_3}(x)h^6 + \ldots$$

In the nonlinear case one can use the iterative approach

$$\boldsymbol{A}\boldsymbol{y}^{k+1} = h^2\boldsymbol{f}(\boldsymbol{y}^k) - \boldsymbol{r} \tag{9.5}$$

where $\boldsymbol{y}^k$ stands for the value of $\boldsymbol{y}$ after k iterations. As initial values one can use a linear interpolation between the two boundary values $y_0 = y(0) = \alpha, y_m = y(b) = \beta$:

$$\boldsymbol{y^0}_i = \alpha + (\beta - \alpha)\frac{i}{m}.$$

Multiplication of Equation 9.5 with $\boldsymbol{A}^{-1}$ gives

$$\boxed{\boldsymbol{y}^{k+1} = h^2\boldsymbol{A}^{-1}\boldsymbol{f}(\boldsymbol{y}^k) - \boldsymbol{A}^{-1}\boldsymbol{r}}$$

This is a fixed point iteration

$$\boldsymbol{y}^{k+1} = \boldsymbol{F}(\boldsymbol{y}^k)$$

for solving the fixed point equation

$$\boldsymbol{y} = \boldsymbol{F}(\boldsymbol{y}) \tag{9.6}$$

with

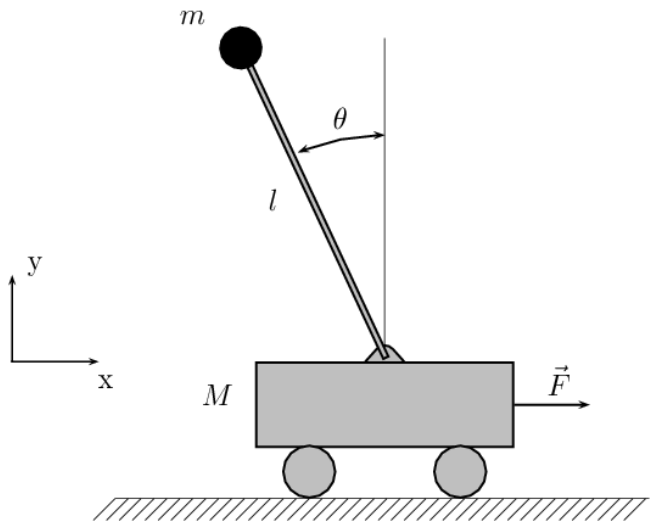
$$\boldsymbol{F}(\boldsymbol{y}) = h^2\boldsymbol{A}^{-1}\boldsymbol{f}(\boldsymbol{y}) - \boldsymbol{A}^{-1}\boldsymbol{r}.$$

A generalization of the Banach fixed point theorem from Section 5.3.2 can be applied here if F is a contraction. This means, if for any vectors $\boldsymbol{x}, \boldsymbol{y}$ there is a nonnegative real number $L < 1$ with

$$\|\boldsymbol{F}(\boldsymbol{x}) - \boldsymbol{F}(\boldsymbol{y})\| \leq L\|\boldsymbol{x} - \boldsymbol{y}\|,$$

the iteration converges to the unique solution of Equation 9.6 (or equivalently Equation 9.4).

The Cart-Pole-Problem



$$(M+m)\,\ddot{x} - ml\ddot{\theta}\cos\theta + ml\dot{\theta}^2\sin\theta = 0$$
$$ml(-g\sin\theta - \ddot{x}\cos\theta + l\ddot{\theta}) = 0$$

9.4 Linear Differential Equations with Constant Coefficients

To solve the one dimensional first order ODE[1]

$$\frac{dy}{dx} = \lambda y \quad \text{with the initial value } y(0)$$

we try

$$y(x) = ae^{\lambda x}$$

and get

$$y(x) = y(0)e^{\lambda x}$$

Systems of Linear Differential Equations with Constant Coefficients
To solve

$$\frac{d\boldsymbol{y}}{dx} = \boldsymbol{A}\boldsymbol{y} \quad \text{with the initial value } \boldsymbol{y}(0) \tag{9.7}$$

we try

$$\boldsymbol{y}(x) = \boldsymbol{u}e^{\lambda x}$$

Substitution leads to the Eigenvalue problem $\boldsymbol{A}\boldsymbol{u} = \lambda\boldsymbol{u}$

[1] We follow section 6.3 in [1]

Example
To solve

$$\frac{d\boldsymbol{y}}{dx} = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \boldsymbol{y} \quad \text{with} \quad \boldsymbol{y}(0) = \begin{pmatrix} 5 \\ 4 \end{pmatrix} \tag{9.8}$$

we have to solve $\boldsymbol{A}\boldsymbol{u} = \lambda\boldsymbol{u}$ and get the characteristic equation

$$(1-\lambda)(1-\lambda) - 4 = 0$$

with the solutions $\lambda_1 = 3$ and $\lambda_2 = -1$ and the eigenvectors

$$\boldsymbol{u}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \boldsymbol{u}_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

The particular solutions are:

$$\boldsymbol{y}_1(x) = \boldsymbol{u}_1 e^{\lambda_1 x} \quad \text{and} \quad \boldsymbol{y}_2(x) = \boldsymbol{u}_2 e^{\lambda_2 x}$$

The linear combinations

$$\boldsymbol{y}(x) = a_1\boldsymbol{u}_1 e^{\lambda_1 x} + a_2\boldsymbol{u}_2 e^{\lambda_2 x}$$

represent the subspace of all solutions of equation 9.7.
For $x = 0$ we get

$$\boldsymbol{y}(0) = a_1\boldsymbol{u}_1 + a_2\boldsymbol{u}_2 = (\boldsymbol{u}_1\boldsymbol{u}_2)\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}.$$

For the example (equation 9.8) this gives

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 5 \\ 4 \end{pmatrix}$$

or

$$\begin{aligned} a_1 + a_2 &= 5 \\ a_1 - a_2 &= 4 \end{aligned}$$

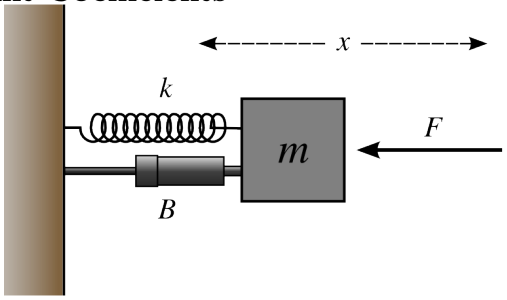
yielding $a_1 = 9/2$ and $a_2 = 1/2$ and the solution to our initial value problem is

$$\boldsymbol{y}(x) = \begin{pmatrix} 9/2 \\ 9/2 \end{pmatrix} e^{3x} + \begin{pmatrix} 1/2 \\ -1/2 \end{pmatrix} e^{-x}$$

Second order Linear Linear ODEs with Constant Coefficients

Many mechanical systems can be described by the second order linear ODE[2]

$$m\ddot{x} + b\dot{x} + kx = 0 \tag{9.9}$$



with

$\dot{x} = \frac{dx}{dt}$ = derivative wrt. time t

$m\ddot{x} =$ resulting force on point mass m (Newton's Law)

$-b\dot{x} =$ friction proportional to speed (damping)

$-kx =$ elastic restoring force (linear spring)

[2]Figure from http://en.wikipedia.org/wiki/File:Mass-Spring-Damper.png

Transformation to a system of first order ODEs

$$m\ddot{x} + b\dot{x} + kx = 0$$

We substitute $\dot{x} = v$ and thus $\ddot{x} = \dot{v}$ and get the first order system

$$\begin{aligned} \dot{x} &= v \\ m\dot{v} + bv + kx &= 0 \end{aligned} \qquad \text{or} \qquad \begin{aligned} \dot{x} &= v \\ m\dot{v} &= -kx - bv \end{aligned}$$

In matrix form:

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\alpha & -\beta \end{pmatrix} \cdot \begin{pmatrix} x \\ v \end{pmatrix} \tag{9.10}$$

with $\alpha = \frac{k}{m}$ and $\beta = \frac{b}{m}$.
Eigenvalue problem:

$$\left| \begin{pmatrix} -\lambda & 1 \\ -\alpha & -\beta - \lambda \end{pmatrix} \right| = 0$$

Characteristic equation:

$$-\lambda(-\beta - \lambda) + \alpha = \lambda^2 + \beta\lambda + \alpha = 0$$

with the solutions

$$\lambda_{1,2} = -\frac{\beta}{2} \pm \sqrt{\frac{\beta^2}{4} - \alpha}.$$

The corresponding eigenvectors are

$$\boldsymbol{u}_1 = \begin{pmatrix} 1 \\ \lambda_1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{u}_2 = \begin{pmatrix} 1 \\ \lambda_2 \end{pmatrix}.$$

The solutions for the ODE system (9.10) are

$$\begin{pmatrix} x \\ v \end{pmatrix} = a_1 \boldsymbol{u}_1 e^{\lambda_1 t} + a_2 \boldsymbol{u}_2 e^{\lambda_2 t} = a_1 \begin{pmatrix} 1 \\ \lambda_1 \end{pmatrix} e^{\lambda_1 t} + a_2 \begin{pmatrix} 1 \\ \lambda_2 \end{pmatrix} e^{\lambda_2 t}$$

We only look at the x-component:

$$x(t) = a_1 e^{\lambda_1 t} + a_2 e^{\lambda_2 t}$$

Eigenvalues may be complex: $\lambda = r + i\omega$. Then

$$e^{\lambda t} = e^{rt + i\omega t} = e^{rt} \cdot e^{i\omega t} = e^{rt} \cdot (\cos \omega t + i \sin \omega t)$$

Since

$$|e^{i\omega t}| = \sqrt{(\cos^2 \omega t + \sin^2 \omega t)} = 1,$$

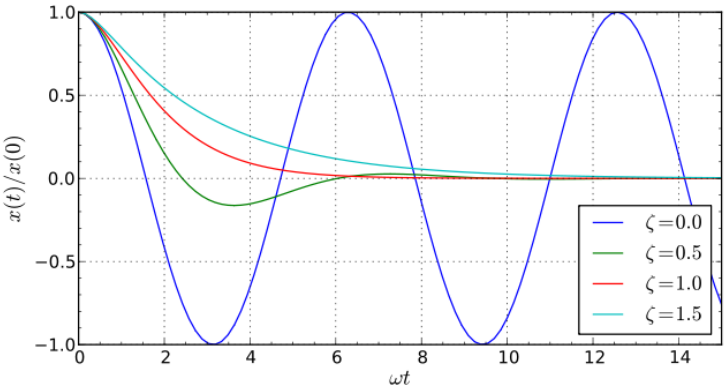
the real factor e^{rt} determines if the solution is **stable**.

Definition 9.1 We call a matrix A **stable** if all eigenvalues have negative real parts.

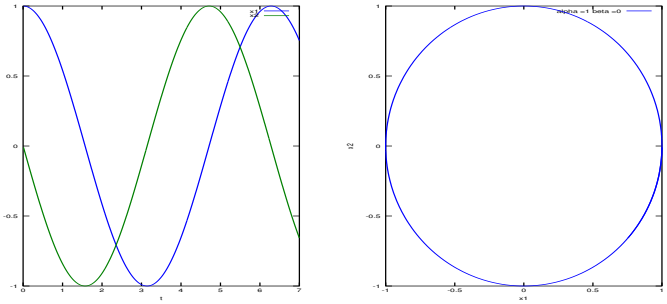
The complex part $\cos \omega t + i \sin \omega t$ produces oscillations.
Solution is exponential only, if the eigenvalues are real, i.e. if

$$\frac{\beta^2}{4} - \alpha > 0.$$

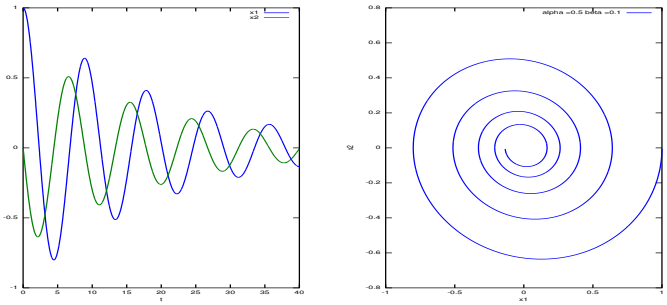
For $\alpha > 0$ and $\beta > 0$ this means $\beta > 2\sqrt{\alpha}$ or $b > 2\sqrt{km}$.
With $\xi = \frac{b}{2\sqrt{km}}$ we get the solution diagram[3]



In 2-dimensional x, v-space we get the solutions



Plot of $x(t)$, $v(t)$ (left) and the x, v phase diagram for $\alpha = 1$, $\beta = 0$ (right).

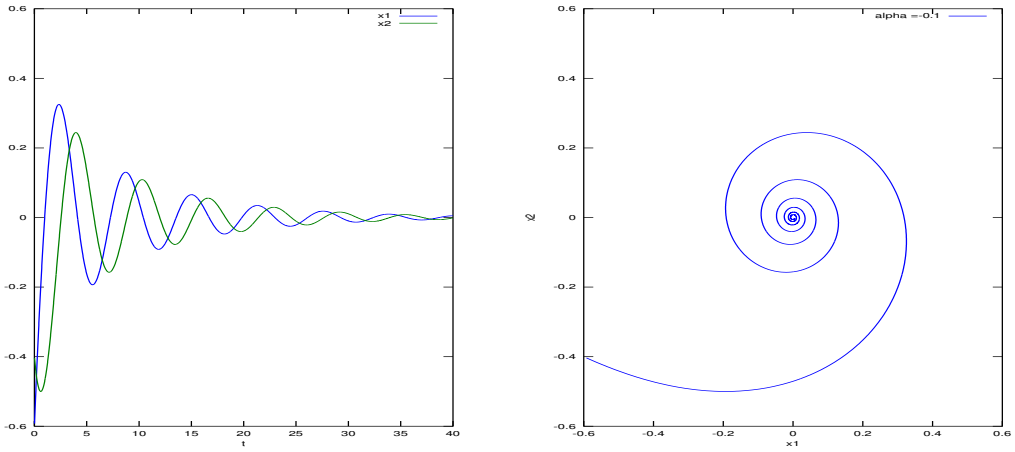


Plot of $x(t)$, $v(t)$ (left) and the x, v phase diagram for $\alpha = 0.5$, $\beta = 0.1$ (right).

Back to nonlinear ODEs
We consider the following system of two nonlinear ODEs:

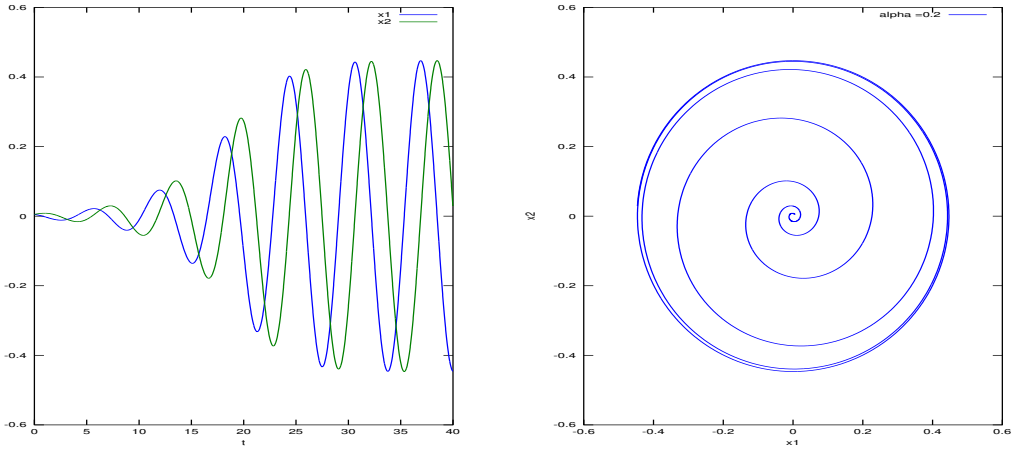
$$\dot{y}_1 = \alpha y_1 - y_2 - y_1(y_1^2 + y_2^2)$$
$$\dot{y}_2 = y_1 + \alpha y_2 - y_2(y_1^2 + y_2^2)$$

[3]Figure from http://en.wikipedia.org/wiki/Harmonic_oscillator



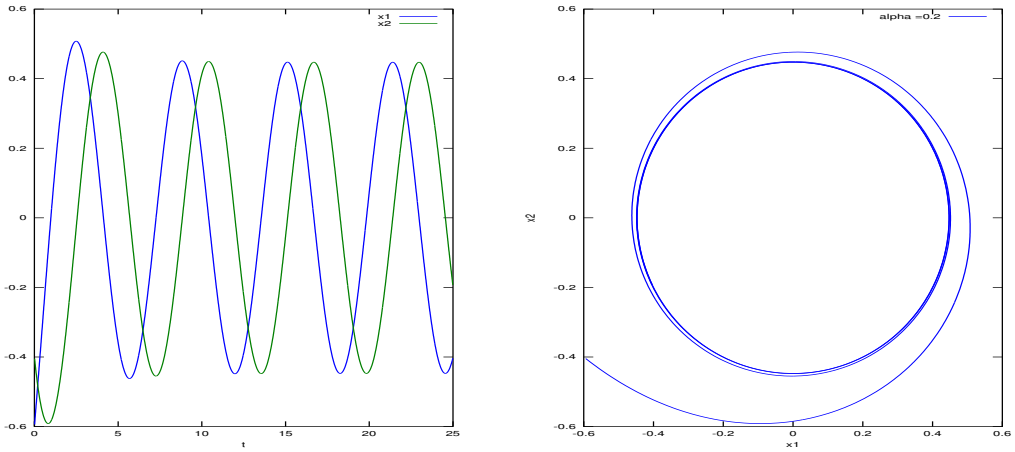
Plot of $y_1(t),\ y_2(t)$ (left) and the y_1, y_2 phase diagram for $\alpha = -0.1$ (right).

Hopf Bifurcation



$y_1(t),\ y_2(t)$ (left) and the y_1, y_2 phase diagram for $\alpha = 0.2$ (right).

Hopf Bifurcation



Same setting ($\alpha = 0.2$), but different initial values.

Hopf Bifurcation, Properties[4][5]

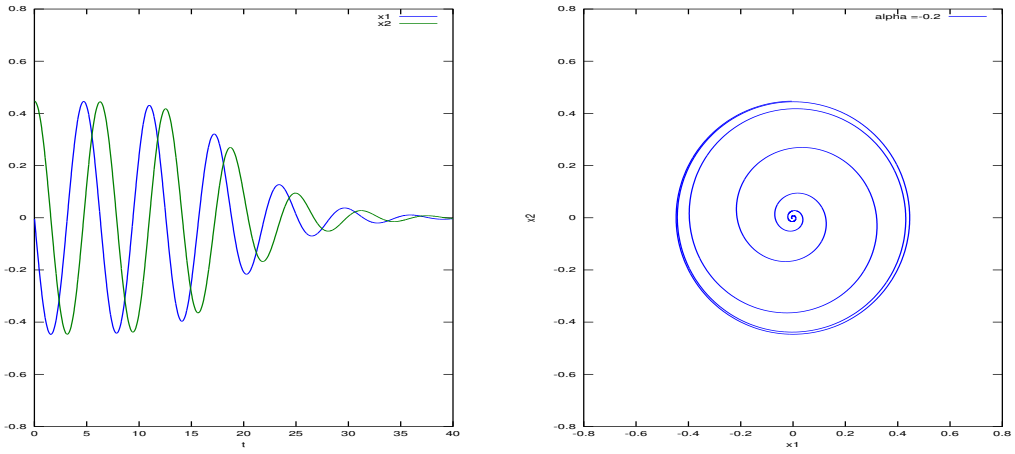
- Limit cycle is a *stable attractor*.
- Supercritical Hopf bifurcation.
- $\alpha < 0$: stable dynamics (converges to steady point).
- $\alpha \geq 0$: unstable dynamics.
- First Lyapunavo coefficient is negative.

Definition 9.2 The appearance or the disappearance of a periodic orbit through a local change in the stability properties of a steady point is known as **Hopf bifurcation.**

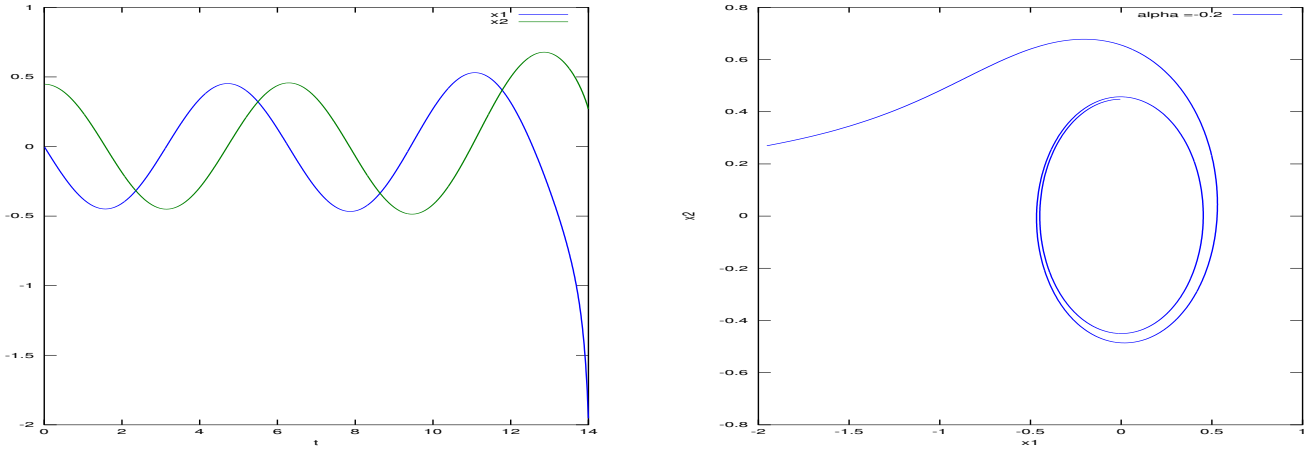
Unstable Attractor

We slightly modify the system of ODEs:

$$\dot{y}_1 = \alpha y_1 - y_2 + y_1(y_1^2 + y_2^2)$$
$$\dot{y}_2 = y_1 + \alpha y_2 + y_2(y_1^2 + y_2^2)$$



$\alpha = -0.2$ and $\boldsymbol{y}^T(0) = (0, 0.447)$.



$\alpha = -0.2$ and $\boldsymbol{y}^T(0) = (0, 0.448)$.

[4] www.scholarpedia.org/article/Andronov-Hopf_bifurcation
[5] en.wikipedia.org/wiki/Hopf_bifurcation

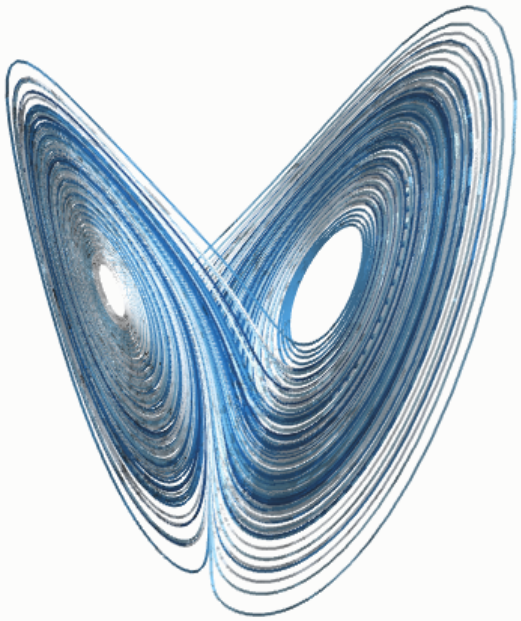
Unstable Attractor, Properties

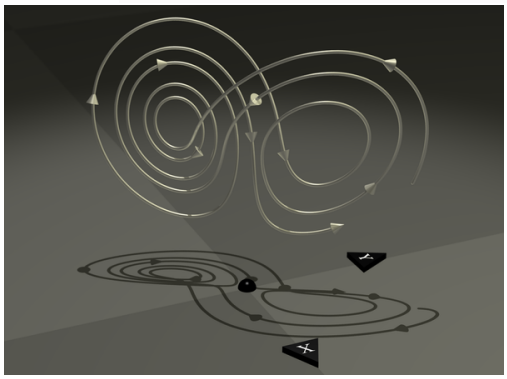
- Limit cycle is an *unstable attractor*.
- Subcritical Hopf bifurcation.
- $\alpha < 0$: the origin is a stable steady point.
- $\alpha \geq 0$: unstable dynamics (divergence).
- First Lyapunavo coefficient is positive.

The Lorenz Attractor[6]

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = x(\rho - z) - y$$
$$\dot{z} = xy - \beta z$$

- Simple model of atmospheric convection.
- Chaotic attractor.





The Logistic Equation

Similar chaotic dynamics as in the Lorenz attractor can be observed in the following discrete population model:

- Reproduction proportional to $q_r\, q_v\, X_n$.
- Animals die proportional to $q_d(C - X_n)$.

[6] en.wikipedia.org/wiki/Lorenz_attractor

- C = capacity of the habitate.

$$X_{n+1} = q_r \, q_v \, X_n \, (C - X_n).$$

Simplification ($C = 1$):

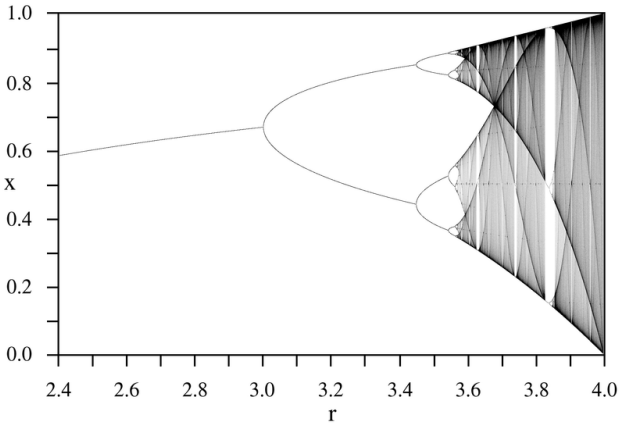
$$x_{n+1} = r \, x_n \, (1 - x_n).$$

The Logistic Equation, Values

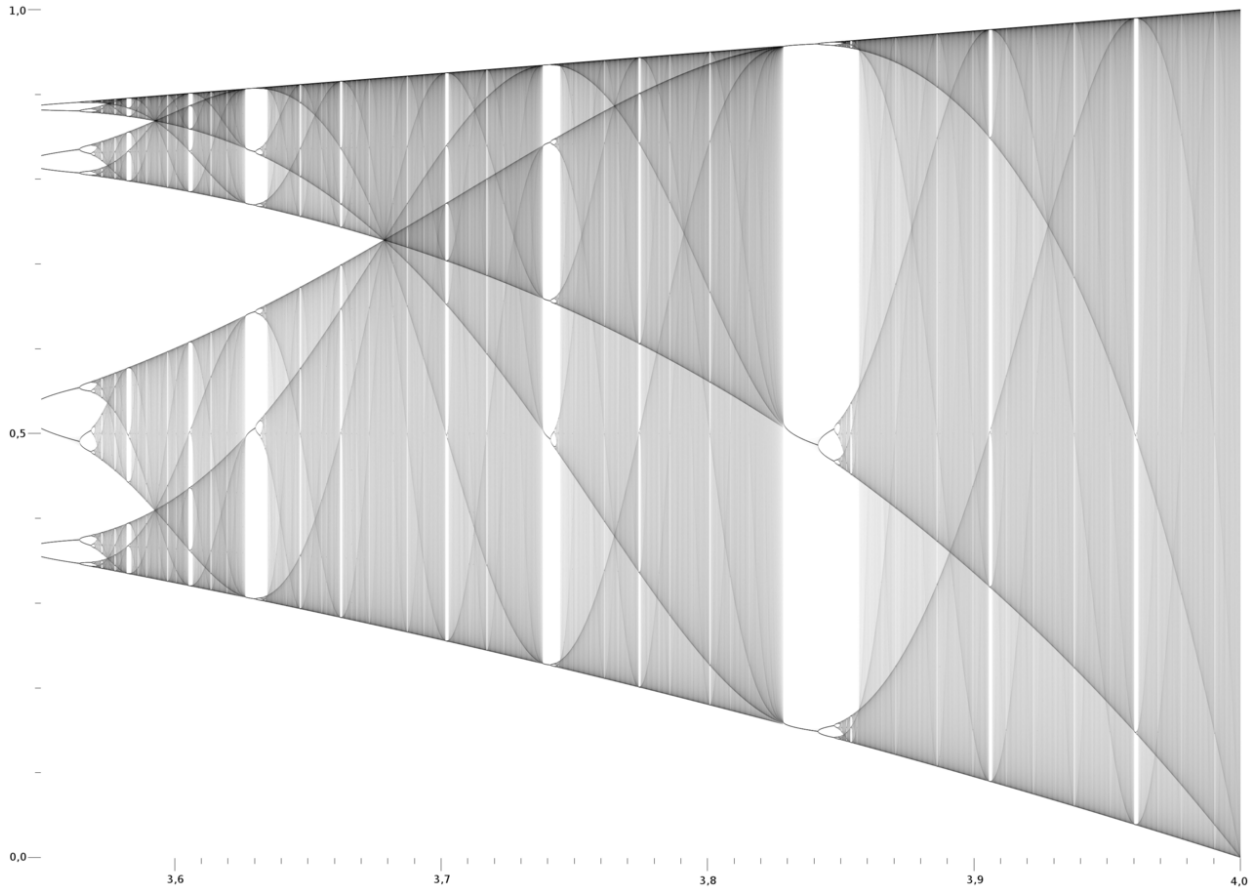
r = 2.2000: 0.10000 0.19800 0.34935 0.50007 0.55000 0.54450 0.54564 0.54542 0.54546 0.54545 **r = 3.2000:** 0.10000 0.28800 0.65618 0.72195 0.64237 0.73514 0.62307 0.75153 0.59754 0.76955 0.56749 ... 0.79945 0.51305 0.79946 0.51304 0.79946 0.51304 **r = 3.5000:** 0.10000 0.31500 0.75521 0.64703 0.79933 0.56140 0.86181 0.41684 0.85079 0.44431 0.86414 0.41090 0.84721 ... 0.50089 0.87500 0.38282 0.82694 0.50088 0.87500 0.38282 0.82694

The Feigenbaum Diagram[7]

In the following bifurcation diagram we see the limit values drawn over the parameter value r:



[7] de.wikipedia.org/wiki/Logistische_Gleichung



The End

- Thank you for attending the lectures!
- Thank you for working hard on the exercises!
- I wish you fun with Mathematics, with the exercises and with ...
- I wish you all the best for the exam!!!

9.5 Exercises

9.5.1 Numerical Integration and Differentiation

Exercise 9.1 Let $h = x_i - x_{i-1}$. Calculate the integral $\int_{x_{i-1}}^{x_i} (x - x_{i-1})(x - x_i)\,dx$ using the substitution $x = x_{i-1} + ht$ with the new variable t.

Exercise 9.2 Write a program for the numerical approximate computation of the integral of a function f in the interval $[a, b]$.

a) Write a function T for the computation of the integral with the trapezoidal rule on an equidistant grid with n equal sub intervals.

b) Apply the function T with n and $2n$ sub intervals to increase the accuracy with Richardson-extrapolation.

c) Apply your functions to $\int_0^1 e^{-x^2}\,dx$ and produce a table of the approximation error depending on the step size h $(1/20 \le h \le 1)$.

d) Show using the above table that the error decreases quadratically for $h \to 0$.

Exercise 9.3

a) Compute the area of a unit circle using both presented Monte-Carlo methods (naive and mean of function values) to an accuracy of at least 10^{-3}.

b) Produce for both methods a table of the deviations of the estimated value depending on the number of trials (random number pairs) and draw this function. What can you say about the convergence of this method?

c) Compute the volume of four dimensional unit sphere to a relative accuracy of 10^{-3}. How much more running time do you need?

Exercise 9.4

a) Compute the first derivative of the function $\cos x/x$ in $x = 2$ with the symmetric difference formula and $h = 0.1$.

b) Apply Richardson extrapolation to compute $F_4(h)$.

c) Compare the error of $F_4(h)$ with the theoretical estimate given in Theorem 9.2.

d) Use the table of function values of the function f given beside to approximate the derivative $f'(x)$. Apply repeated Richardson extrapolation to get $F_2(h)$, $F_3(h)$ and $F_4(h)$. Plot the resulting functions.

0.5	-3.75
0.75	-1.36607
1.	0.
1.25	0.729167
1.5	1.05
1.75	1.10795
2.	1.
2.25	0.793269
2.5	0.535714
2.75	0.2625
3.	0.

9.5.2 Differential Equations

Exercise 9.5

a) Write programs that implement the Euler-, Heun- and Runge Kutta methods for solving first order initial value problems.

b) Implement the Richardson extrapolation scheme for these methods.

Exercise 9.6 The initial value problem

$$\frac{dy}{dx} = \sin(xy) \qquad y_0 = y(0) = 1$$

is to be solved numerically for $x \in [0, 10]$.

a) Compare the Euler-, Heun- and Runge Kutta methods on this example. Use $h = 0.1$.

b) Apply Richardson extrapolation to improve the results in $x = 5$ for all methods. (attention: use the correct p_k for each method.)

Exercise 9.7 Apply the Runge Kutta method to the predator-prey example 9.4 and experiment with the parameter α and the initial values. Try to explain the population results biologically.

Exercise 9.8 Use Runge Kutta to solve the initial value problem

$$\frac{dy}{dx} = x\sin(xy) \qquad y_0 = y(0) = 1$$

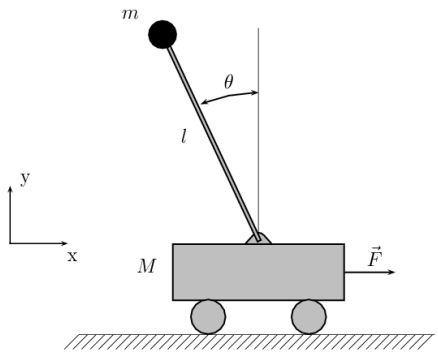
for $x \in [0, 20]$. Report about problems and possible solutions.

Exercise 9.9 The following table shows the differences between the approximations computed with Richardson extrapolation for some numeric algorithm. Determine from the table the convergence order of the algorithm for $h \to 0$ and all the exponents p_i in the taylor expansion for $F(h)$. (Hint: These differences are an approximation of the error on the respective approximation level,)

h					
1	-0.075433				
0.5	-0.018304	0.0001479			
0.25	-0.004542	$9.106 \cdot 10^{-6}$	$-3.492 \cdot 10^{-8}$		
0.125	-0.001133	$5.670 \cdot 10^{-7}$	$-5.409 \cdot 10^{-10}$	$1.208 \cdot 10^{-12}$	
0.0625	-0.000283	$3.540 \cdot 10^{-8}$	$-8.433 \cdot 10^{-12}$	$4.691 \cdot 10^{-15}$	$-6.847 \cdot 10^{-18}$

Exercise 9.10 (challenging)

The dynamics of the inverted pendulum – also called cart pole – system as shown beside can be described by the following two differential equations of second order. Here $\dot{x}$ $\ddot{x}$, etc. are the first and second derivatives wrt. the time t. A derivation of these equations can be found on Wikipedia (not required here).



$$(M+m)\,\ddot{x} - ml\ddot{\theta}\cos\theta + ml\dot{\theta}^2\sin\theta = 0 \qquad (9.11)$$

$$ml(-g\sin\theta - \ddot{x}\cos\theta + l\ddot{\theta}) = 0 \qquad (9.12)$$

a) Use the substitution $y_1 = x,\ y_2 = \dot{x},\ y_3 = \theta,\ y_4 = \dot{\theta}$ to obtain a system of 4 first order ODEs of the form $\dot{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{y})$. (hint: make sure, the right hand sides of the differential equations contain no derivatives!)

b) Apply the Runge Kutta method to solve the system for $g = 9.81$, $m = 1$, $m = 1$ with the initial condition $y_1(0) = 0,\ y_2(0) = 0,\ y_3(0) = 0.01,\ y_4(0) = 0$.

c) Plot the functions $y_1(t),\ y_2(t),\ y_3(t),\ y_4(t)$ and try to understand them.

d) Experiment with other initial conditions and other masses, e.g. $m = 1$, $M = 100000$ or $M = 1$, $m = 100000$.

Exercise 9.11 Prove that, if $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are solutions of the ODE $\boldsymbol{y}' = \lambda\boldsymbol{y}$, then any linear combination of $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ is also a solution.

Exercise 9.12 Prove that the eigenvectors of the matrix

$$\begin{pmatrix} 0 & 1 \\ -\alpha & -\beta \end{pmatrix}$$

from equation 9.10 with the eigenvalues λ_1 and λ_2 are $(1, \lambda_1)^T$ and $(1, \lambda_2)^T$.

Exercise 9.13

a) Solve the initial value problem $m\ddot{x} + b\dot{x} + kx = 0$ with $x(0) = 0$ and $\dot{x}(0) = -10m/s$ for the parameters: $m = 10kg$, $b = 2kg/s$, $k = 1kg/s^2$. Plot the resulting function $x(t)$.

b) The general solution involves a complex component $i\sin\omega t$. Does it make sense to have a complex sine-wave as solution for an ODE with real coefficients and real initial conditions? What is the natural solution for this problem?

Exercise 9.14 Linearize the Lotka-Volterra ODEs and show that this no good model for a predator prey system. To do this:

a) Calculate the Jacobian matrix of the right hand side of the ODEs at $\boldsymbol{y}(0)$ and set up the linearized ODEs.

b) Calculate the eigenvalues of the Jacobian and describe the solutions of the linearized system.

Exercise 9.15 Download the Octave/Matlab code for the Lorenz attractor from http://en.wikipedia.org/wiki/Lorenz_attractor. Modify the code to dynamically follow a trajectory and observe the chaotic dynamics of the system.

Bibliography

[1] G. Strang. *Introduction to linear algebra*. Wellesley Cambridge Press, 3rd edition, 2003. 1.1, 5.2, 63, 1

[2] Gilbert Strang. *Linear Algebra and its applications*. Harcourt Brace Jovanovich College Publishers, 1988. 8.2.2.4

[3] R. Hamming. *Numerical Methods for Scientists and Engineers*. Dover Publications, 1987.

[4] W. Cheney and D. Kincaid. *Numerical mathematics and computing*. Thomson Brooks/Cole, 2007.

[5] S.M. Ross. *Introduction to probability and statistics for engineers and scientists*. Academic Press, 2009.

[6] J. Nocedal and S.J. Wright. *Numerical optimization*. Springer Verlag, 1999. 8.2.2.4

[7] C.M. Bishop. *Pattern recognition and machine learning*. Springer New York:, 2006. 7.4

[8] M. Brill. *Mathematik für Informatiker*. Hanser Verlag, 2001. Sehr gutes Buch, das auch diskrete Mathematik beinhaltet.

[9] M. Knorrenschild. *Numerische Mathematik*. Hanser Verlag, 2005.

[10] F. Reinhardt and H. Soeder. *dtv–Atlas zur Mathematik, Band 1 und Band 2: Algebra und Grundlagen*. Deutscher Taschenbuchverlag, München, 1977.

[11] H. Späth. *Numerik*. Vieweg, 1994. Leicht verständlich, voraussichtlich werden grösere Teile der Vorlesung aus diesem Buch entnommen.

[12] H. R. Schwarz. *Numerische Mathematik*. Teubner Verlag, 1988. Gutes Buch, sehr ausführlich. 5.3.2, 105, 9.4

[13] S. Wolfram. *Mathematica, A System for Doing Mathematics by Computer*. Addison Wesley, 1991. Das Standardwerk des Mathematica-Entwicklers. Daneben gibt es viele andere Bücher über Mathematica.

[14] P. J. Fleming and J. J. Wallace. How not to Lie with Statistics: The Correct Way to Summarize Benchmark Results. *Comm. of the ACM*, 29(3):218–221, 1986.

[15] J.E. Smith. Characterizing Computer Performance with a Single Number. *Communications of the ACM*, 31(10):1202–1206, 1988.

[16] J. Aczél. *Lectures on Functional Equations and Their Applications*, pages 148–151, 240–244, 291. Academic Press, New York/London, 1966.

[17] W. Ertel. On the Definition of Speedup. In *PARLE'94, Parallel Architectures and Languages Europe*, Lect. Notes in Comp. Sci. 817, pages 289–300. Springer, Berlin/New York, 1994.

[18] D.E. Knuth. *Seminumerical Algorithms*, volume 2 of *The Art of Computer Programming*. Addison-Wesley, 3rd edition, 1997.

[19] U. Maurer. A universal statistical test for random bit generators. *Journal of Cryptography*, 5(2):89–105, 1992. 7.1.1

[20] G. Marsaglia. A current view of random number generators. In *Computer Science and Statistics: The Interface.*, pages 3–10. Elsevier Science, 1985.

[21] W. Ertel and E. Schreck. Real random numbers produced by a maxtor disk drive. http://www.hs-weingarten.de/~ertel/rrng/maxtor.html, 2000. 7.1.7

[22] J. von Neumann. Various techniques used in connection with random digits. In *von Neumann's Collected Works*, volume 5. Pergamon Press, 1963.

[23] L. Blum, M. Blum, and M. Shub. A simple unpredictable pseudo-random number generator. *SIAM Journal of Computing*, 15(2):364–383, 1986. 7.1.5.1

[24] M.J.D Powell. Radial basis functions for multivariable interpolation: a review. IMA conference on Algorithms for the Approximation of Function and Data, 1985. 8.2.2.2

[25] Broomhead D.S and Lowe D. Multivariable functional interpolation and adaptive networks. Complex Systems 2, 1988. 8.2.2.4

[26] Wolfgang Ertel. *Grundkurs Künstlische Intelligenz*. Vieweg and Teubner, 2009.

[27] T. Tierney, G. Dahlquist, and A. Björck. *Numerical Methods*. Dover Publication Inc., 2003. 83, 84, 94

[28] M. Li and P. Vitanyi. Two decades of applied kolmogorov complexity. In *3rd IEEE Conference on Structure in Complexity theory*, pages 80–101, 1988. 7.4

[29] B. Schneier. *Angewandte Kryptogrphie*. Addison-Wesley, 1996. Deutsche Übersetzung. 7.1.4

[30] B. Jun and P. Kocher. The intel random number generator (white paper). http://developer.intel.com/design/security/rng/rngppr.htm, 1999. 7.1.7

[31] Carl Edward Rasmussen and Christopher Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[32] J. Shawe Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[33] David J. C. MacKay. *Bayesian Methods for Adaptive Models*. PhD thesis, California Institute of Technology, 1991.