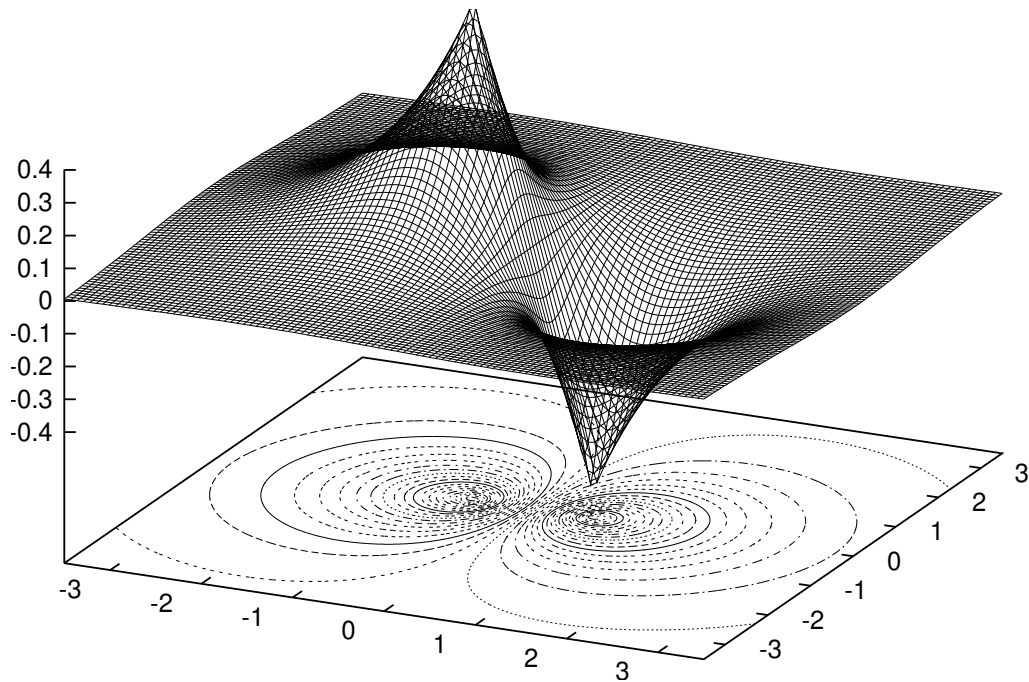


Einführung in die mathematische Behandlung der Naturwissenschaften

CAS Python edition

Prof. Dr. H Ebert



*The general theory of quantum mechanics is now almost complete. [...] The underlying physical laws necessary for **the mathematical theory of a large part of physics and the whole of chemistry** are thus completely known, [...]*

P. A. M. Dirac, Proc. Roy. Soc. London A123 (1929) Seite 714-33

Inhaltsverzeichnis

Inhaltsverzeichnis	1
1 Grundlagen	4
1.1 Literatur	4
1.2 Mengen	6
1.3 Verknüpfungen und Gruppen	14
1.4 Funktionen	19
1.5 Zahlen	24
1.5.1 Natürliche Zahlen, Prinzip der vollständigen Induktion	24
1.5.2 Ganze, rationale und reelle Zahlen	26
1.5.3 Komplexe Zahlen	31
2 Analytische Geometrie und Lineare Algebra	39
2.1 Vektoralgebra	39
2.1.1 Einführung des Vektorbegriffs in Anlehnung an die Euklidische Geometrie	39
2.1.2 Gleichheit, Addition, Subtraktion und Multiplikation mit einem Skalar	42
2.1.3 Das skalare Produkt zweier Vektoren	48
2.1.4 Das Kreuzprodukt und Spatprodukt	52
2.1.5 Geraden- und Ebenengleichung	55
2.1.6 Lineare Vektorräume	58
2.2 Matrizen	64
2.2.1 Definitionen	64
2.2.2 Gleichheit, Addition, Subtraktion und Multiplikation mit einem Skalar	66
2.2.3 Matrixmultiplikation	67
2.2.4 Elementare Umformungen	68
2.2.5 Rang einer Matrix	74
2.2.6 Inverse von quadratischen Matrizen	77
2.3 Determinanten	83
2.3.1 Definitionen	83
2.3.2 Eigenschaften von Determinanten	86
2.3.3 Berechnung einer Determinanten	93
2.4 Lineare Gleichungssysteme	95
2.4.1 Definitionen	95

2.4.2	Lösbarkeit eines LGS's	96
2.5	Basistransformation und Symmetrioperationen	99
2.5.1	Koordinaten eines Vektors bezüglich einer festen Basis	99
2.5.2	Basistransformation	99
2.5.3	Symmetrioperationen	101
2.6	Eigenwertprobleme	105
2.6.1	Definitionen	105
2.6.2	Lösung eines Eigenwertproblems	105
2.6.3	Eigenschaften eines Eigenwertproblems	109
3	Funktionen einer Variablen	116
3.1	Grundbegriffe	116
3.1.1	Definitionen	116
3.1.2	Darstellung einer Funktion	117
3.1.3	Umkehrfunktion	122
3.1.4	Verknüpfung von Funktionen	124
3.2	Grenzwerte	128
3.2.1	Zahlenfolgen	128
3.2.2	Grenzwert einer Funktion	130
3.2.3	Methoden der Grenzwertbestimmung	135
3.3	Stetigkeit	138
3.3.1	Definitionen	138
3.3.2	Eigenschaften von stetigen Funktionen	141
3.4	Elementare Funktionen	143
3.4.1	Ganze rationale Funktionen	143
3.4.2	Gebrochen rationale Funktionen	148
3.4.3	Potenz- und Wurzelfunktionen	149
3.4.4	Trigonometrische Funktionen	150
3.4.5	Exponential- und Logarithmusfunktion	156
3.5	Differentialrechnung	162
3.5.1	Definitionen und geometrische Deutung	162
3.5.2	Ableitungen elementarer Funktionen	164
3.5.3	Differentiationsregeln	167
3.5.4	Ableitungen höherer Ordnung	172
3.5.5	Kurvendiskussion	174
3.5.6	Eigenschaften differenzierbarer Funktionen	180
3.5.7	Anwendungen	181
3.6	Integralrechnung	186
3.6.1	Geometrische Deutung und Definitionen	186
3.6.2	Eigenschaften des bestimmten Integrals	190
3.6.3	Stammfunktionen und unbestimmtes Integral	191
3.6.4	Integrationsregeln	195
3.6.5	Integration gebrochen rationaler Funktionen	200
3.6.6	Uneigentliche Integrale	203
3.6.7	Interpolation und numerische Integration	207

3.7	Potenzreihenentwicklung von Funktionen	212
3.7.1	Taylorentwicklung	212
3.7.2	Potenzreihen	216
4	Funktionen mehrerer Veränderlicher	223
4.1	Definition, Einführung	223
4.2	Der Begriff der Stetigkeit	233
4.3	Mehrdimensionale Differentialrechnung	236
4.3.1	Differenzierbarkeit	236
4.3.2	Partielle Ableitungen	239
4.3.3	Partielle Ableitungen höherer Ordnung	246
4.3.4	Extremwerte	247
4.3.5	Kettenregeln für die partielle Differentiation	251
4.4	Mehrdimensionale Integralrechnung	257
4.4.1	Differentialformen	257
4.4.2	Wegintegrale (2. Art)	259
4.4.3	Wegintegrale für Gradientenfelder	264
4.4.4	Mehrfachintegrale	270
	Index	277
	Liste der CAS Beispiele	281

Kapitel 1

Grundlagen

1.1 Literatur

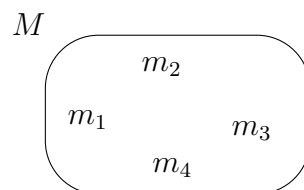
- L. Papula
Mathematik für Chemiker
Enke Verlag, Stuttgart
ISBN 3-432-88133-9, vergriffen
Signatur: Ausgabe 1982: VC 6000 P218(2),
Ausgabe 1991: VC 6000 P218(3) (10 Exemplare)
- L. Papula
Übungen und Anwendungen
zur Mathematik für Chemiker
Enke Verlag, Stuttgart
ISBN 3-432-88953-4, vergriffen
Signatur: Ausgabe 1988: VC 6000 P218 U2(2),
Ausgabe 1992: VC 6000 PC218 U2(3)
- L. Zachmann
Mathematik für Chemiker
Verlag Wiley-VCH
ISBN 3-527-29224-1, EUR 52,95
Signatur: Ausgabe 1972: VC 6000 Z16,
Ausgabe 1974: A 813,
Ausgabe 1977: VC 6000 Z16(3),
Ausgabe 1984: VC 6000 Z16(4)+2,
Ausgabe 1990: VC 6000 Z16(4)
- N. Rösch
Mathematik für Chemiker
Springer Verlag, Berlin
ISBN 3-540-56824-7, EUR 24,95
Signatur: Ausgabe 1993: VC 6000 R718 (12 Exemplare)

- M. Stockhausen
Mathematik für Chemiker
Steinkopff Verlag
ISBN 3-7985-1025-3, vergriffen
bestellbar über Zentralbibliothek
- K. Jug
Mathematik in der Chemie
Springer Verlag, Berlin
ISBN 3-540-55771-7, vergriffen
Signatur: Ausgabe 1993: VC 6000 J93(2)
- I. N. Bronstein, K. A. Semendjajew
Taschenbuch der Mathematik
B. G. Teubner Verlagsgesellschaft, Stuttgart, EUR 29,95
Signatur: Ausgabe 1964: SK 110 B869(4)
- G. Brunner
Mathematik für Chemiker
Spektrum Verlag
Band I und Band II, ca EUR 31,00
- E.A. Reinsch
Mathematik für Chemiker
Teubner, Wiesbaden 2004 ISBN 3-519-00443-7
Signatur: Ausgabe 2004: VC 6000 R374 +4
- D. Guedj
Das Theorem des Papageis
Lübbe, 2001
ISBN 3-4550-2546-3, EUR 9,95
- S. Singh
Fermats letzter Satz
Taschenbuch, DTV, München 2000
ISBN 3-4233-3052x, EUR 10,00
bestellbar über Zentralbibliothek

Leider sind die Bücher von Papula vergriffen und daher nur in der Bibliothek erhältlich. Die letzten beiden Bücher sind zwei Beispiele dafür, daß Mathematik durchaus unterhaltend und spannend sein kann.

1.2 Mengen

D "Unter einer **Menge** M versteht man eine Zusammenfassung von bestimmten, wohl-unterschiedenen Objekten m unserer Anschauung oder unseres Denkens zu einem Ganzen." (Georg Cantor 1845-1918)



N

$M = \{m_1, m_2, m_3, m_4\}$
 $M = \{m_3, m_4, m_1, m_2\}$ d.h. die Reihenfolge ist beliebig
 oder $M = \{m_i | i = 1, \dots, 4\}$
 oder $M = \{m_i | m_i \text{ erfüllt die Eigenschaft } \dots\}$
 speziell $M = \{\}$
 oder $M = \emptyset$ bezeichnet die **leere Menge**

D Die Bestandteile oder Objekte in einer Menge nennt man **Elemente**. Für "m ist Element der Menge M" schreibt man: $m \in M$. Gehört ein Objekt n nicht zur Menge M, so sagt man "n ist nicht Element von M": $n \notin M$.

Beispiel: $M = \{a, b, c\} : a \in M, d \notin M$

CAS-Beispiel



Wenn Sie das Tool Jupyter Notebook verwenden, führen Sie diesen Befehl aus, um interaktive Plots zu erhalten. Ansonsten können Sie den Befehl ignorieren.

```
%matplotlib notebook
```

Definition der Mengen M und N . Vergleiche M und N .


```
M = {'a', 'b', 'c', 'd'}
N = {'a', 'b', 'c', 'd', 'd', 'a'}
```

```
M
```

```
N
```

Offensichtlich sind beide Mengen gleich, da die Elemente d und a in N nicht zweifach zu berücksichtigen sind.

Die Abfrage, ob die Menge M das Element a bzw. x enthält, erfolgt über `in`:

```
'a' in M
```

```
'x' in M
```

Liste die Elemente einer Menge M auf und bestimme die Anzahl der Elemente.

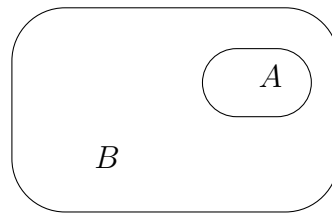
```
M = {'a', 'b', 'c', 'd'}
n = 0
for i in M:
    print("Die Menge M hat das Element " + str(i))
    n = n + 1

print("Anzahl der Elemente in der Menge M: " + str(n))
```

Die Variable i durchläuft durch die Schleifenkonstruktion `for i in M: ...` nacheinander alle Elemente der Menge M . Um die Anzahl der Elemente festzustellen, wird dabei die Variable n hochgezählt. n muß zunächst initialisiert, d.h. auf den Wert 0 gesetzt werden. Die Anweisung $n = n + 1$ ist nicht im Sinne einer mathematischen Gleichung zu verstehen, sondern bewirkt, daß der Variablen n als neuer Wert das Ergebnis der Operation $n + 1$ zugewiesen wird.

D Sind alle Elemente a einer Menge A gleichzeitig auch in einer Menge B enthalten, so nennt man A **Teilmenge** von B : $A \subseteq B$. B wird als **Obermenge** von A bezeichnet: $B \supseteq A$. Besitzt B Elemente, die nicht in A enthalten sind, so wird A als **echte Teilmenge** von B bezeichnet: $A \subset B$.

Symbolisch lassen sich Zusammenhänge zwischen Mengen durch sogenannte **Venn-Diagramme** darstellen:



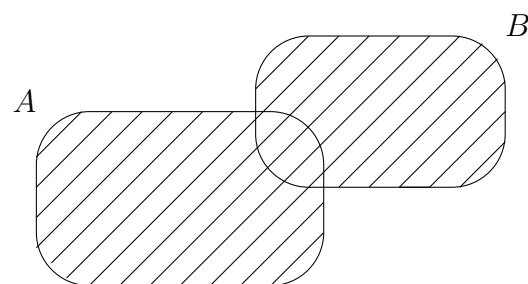
$$A \subseteq B$$

A ist Teilmenge von B

Beispiel: $A = \{a, b, c\}$, $B = \{a, b, c, d\}$: $A \subseteq B$ bzw. $B \supseteq A$

D Die **Vereinigungsmenge** M zweier Mengen A und B ist die Menge aller Elemente, die zur Menge A oder zur Menge B gehören.

$$M = A \cup B = \{x | x \in A \vee x \in B\}$$



$$A \cup B$$

A vereinigt mit B

Beispiel: $A = \{a, b, c, d\}$, $B = \{c, d, e, f\}$: $A \cup B = \{a, b, c, d, e, f\}$

CAS-Beispiel



Bilden der Vereinigungsmenge zweier Mengen mittels des Operators union:

```
M = {'a', 'b', 'c', 'd'}
N = {'a', 'b', 'c', 'd'}
P = {'c', 'd', 'e', 'f'}
Q = {'x', 'x', 'z'}
```

```
M.union(N)
```

```
M.union(P)
```

```
M.union(Q)
```

Kürzer geht es folgendermaßen:

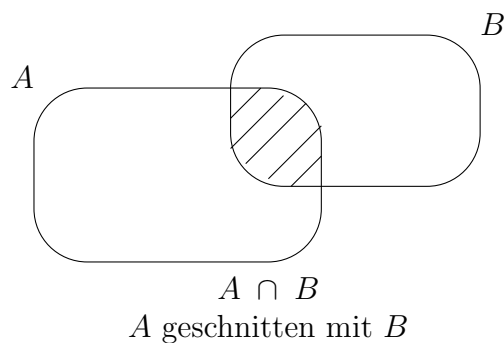
```
M | N
```

```
M | P
```

```
M | Q
```

D Die **Schnittmenge** M zweier Mengen A und B ist die Menge aller Elemente, die sowohl zur Menge A als auch zur Menge B gehören.

$$M = A \cap B = \{x | x \in A \wedge x \in B\}$$



Beispiel: $A = \{a, b, c, d\}$, $B = \{c, d, e, f\}$: $A \cap B = \{c, d\}$

CAS-Beispiel



Bilden der Schnittmenge zweier Mengen mittels des Operators `intersection`:

```
M = {'a', 'b', 'c', 'd'}
N = {'a', 'b', 'c', 'd'}
P = {'c', 'd', 'e', 'f'}
Q = {'x', 'x', 'z'}
```

```
M.intersection(N)
```

```
M.intersection(P)
```

```
M.intersection(Q)
```

Kürzer geht es folgendermaßen:

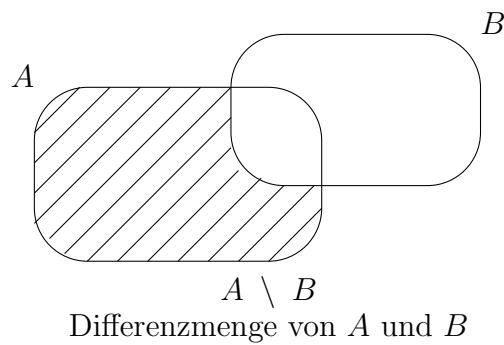
M & N

M & P

M & Q

D Die **Differenzmenge** M zweier Mengen A und B ist die Menge aller Elemente, die zur Menge A aber nicht zur Menge B gehören.

$$M = A \setminus B = \{x \mid x \in A \wedge x \notin B\}$$



Beispiel: $A = \{a, b, c, d\}$, $B = \{c, f, g\}$: $A \setminus B = \{a, b, d\}$

CAS-Beispiel



Bilden der Differenzmenge zweier Mengen mittels des Operators `difference`:

```
M = {'a', 'b', 'c', 'd'}
N = {'a', 'b', 'c', 'd'}
P = {'c', 'd', 'e', 'f'}
Q = {'x', 'x', 'z'}
```

```
M.difference(N)
```

```
M.difference(P)
```

```
M.difference(Q)
```

Kürzer geht es folgendermaßen:

```
M - N
```

```
M - P
```

```
M - Q
```

Mit `difference` läßt sich feststellen, ob A eine Teilmenge von B ist, da dann $A \setminus B$ die leere Menge ergeben muß.

```
A = {'a', 'b', 'c'}
B = {'a', 'b', 'c', 'd'}
```

```
A.difference(B)
```

```
B.difference(A)
```

Im obigen Beispiel gilt offensichtlich $A \subset B$, aber $B \not\subset A$, d.h. B ist nicht Teilmenge von A . Dies kann man alternativ auch mit dem Operator `issubset` bestimmen:

```
A.issubset(B)
```

```
B.issubset(A)
```

S Rechenregeln

Kommutativ- bzw. Vertauschungsgesetze

$$A \cup B = B \cup A$$

$$A \cap B = B \cap A$$

Assoziativ- bzw. Verknüpfungsgesetze

$$A \cup (B \cup C) = (A \cup B) \cup C$$

$$A \cap (B \cap C) = (A \cap B) \cap C$$

Distributiv- bzw. Verteilungsgesetze

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

CAS-Beispiel



Überprüfung der Rechenregeln:

$$A = \{'a', 'b', 'c', 'd'\}$$

$$B = \{'c', 'd', 'e', 'f'\}$$

$$C = \{'e', 'f', 'g', 'h', 'i'\}$$

$$A \mid B$$

$$B \mid A$$

$$A \& B$$

$$B \& A$$

$$(A \mid B) \mid C$$

$$A \mid (B \mid C)$$

$$(A \& B) \& C$$

$$A \& (B \& C)$$

$$A \mid (B \& C)$$

$$(A \mid B) \& (A \mid C)$$

$$A \& (B \mid C)$$

$$(A \& B) \mid (A \& C)$$

D Sind a und b beliebige Elemente, so bezeichnet (a,b) ein **geordnetes Paar** oder **Dupel**. Entsprechend wird (a,b,c) als **Tripel** und (x_1,x_2,x_3,\dots,x_n) als **n-Tupel** bezeichnet.

D Das **kartesische Produkt** M zweier Mengen A und B ist die Menge aller geordneten Paare (a,b) , die sich aus den Elementen $a \in A$ und $b \in B$ bilden lassen.

$$M = A \times B = \{(a,b) \mid a \in A, b \in B\}$$

entsprechend:

$$A \times B \times C = \{(a,b,c) \mid a \in A, b \in B, c \in C\}$$

$$A \times A = A^2 = \{(a_1, a_2) \mid a_1 \in A, a_2 \in A\}$$

CAS-Beispiel

Bilden des kartesischen Produktes zweier Mengen mittels verschachtelter for-Schleifen:

```
A = {'a', 'b', 'c', 'd'}
B = {'c', 'd'}

cartesian_product = set()

for i in A:
    for j in B:
        cartesian_product.add((i, j))

cartesian_product
```

Alternativ bietet das Modul `itertools` eine Funktion zur Bildung des kartesischen Produktes:

```
from itertools import product

set(product(A, B))
```

1.3 Verknüpfungen und Gruppen

D Eine **Verknüpfung** – repräsentiert durch das Symbol \circ – ist eine eindeutige Vorschrift, die zwei Elementen a und b einer Menge M ein drittes Element c zuordnet.

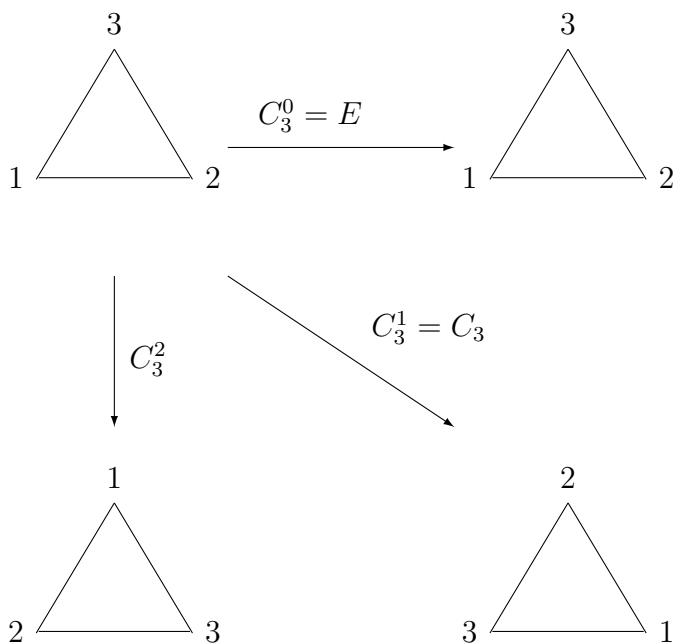
$$c = a \circ b$$

Die Menge M heißt bzgl. der Verknüpfung \circ **abgeschlossen**, falls $c \in M$ ist; d.h. es gilt:

$$c = a \circ b \in M \quad \forall a, b \in M$$

! Die Reihenfolge einer Verknüpfung zweier Elemente ist nicht notwendigerweise beliebig, d.h. $a \circ b$ muß nicht gleich $b \circ a$ sein !

Beispiel: $M = \{C_3^n | \text{Drehungen}^\ddagger \text{ um } \frac{n}{3} \cdot 360^\circ (n = 0, 1, 2)\}$



Die Verknüpfung \circ zweier Elemente von M läßt sich hier als Hintereinanderausführung zweier Drehungen interpretieren. Das Ergebnis aller möglichen Verknüpfungen, die sich dabei ergeben wird in der **Verknüpfungstafel** zusammengefaßt:

[‡]Drehungen sind immer im mathematischen Sinn zu verstehen; d.h. bei positivem Drehwinkel gegen den Uhrzeigersinn

		b			
		$E \quad C_3 \quad C_3^2$			
\circ		E	C_3	C_3^2	
a	$\left\{ \begin{array}{l} E \\ C_3 \\ C_3^2 \end{array} \right.$	E	C_3	C_3^2	$\left. \vphantom{\left\{ \begin{array}{l} E \\ C_3 \\ C_3^2 \end{array} \right.}} \right\} a \circ b$
		C_3	C_3^2	E	
		C_3^2	E	C_3	

$a \circ b$ impliziert daß zuerst Rotation b und dann Rotation a auszuführen ist.

D Ist auf einer Menge G eine Verknüpfung \circ definiert, so wird G als **Gruppe** bezüglich \circ bezeichnet, falls folgende **Gruppenaxiome** erfüllt sind:

- a) die Menge G ist bzgl. \circ abgeschlossen:
 $a \circ b = c \in G \quad \forall a, b \in G$
- b) es gilt das Assoziativgesetz:
 $a \circ (b \circ c) = (a \circ b) \circ c \quad \forall a, b, c \in G$
- c) es existiert ein **linksneutrales Element** e , so daß gilt:
 $e \circ a = a \quad \forall a \in G$
- d) es existiert zu jedem Element $a \in G$ ein **linksinverses Element** $a^{-1} \in G$, so daß gilt:
 $a^{-1} \circ a = e \quad \forall a \in G$

D Eine Gruppe G wird **abelsch** oder **kommutativ** genannt, falls die Verknüpfung \circ kommutativ ist, d.h. es gilt:

$$a \circ b = b \circ a \quad \forall a, b \in G$$

D Gruppen mit endlich vielen Elementen nennt man **endlich**. Die **Ordnung** einer

Gruppe gibt die Anzahl der Elemente an. Statt Verknüpfungstafel spricht man bei einer Gruppe von einer **Gruppentafel**.

D Ist $U \subseteq G$, wobei G eine Gruppe bzgl. \circ ist, und ist U ebenfalls eine Gruppe bzgl. \circ , so nennt man U **Untergruppe** von G .

S Kürzungsregel 1

G sei eine Gruppe bzgl. \circ , dann gilt:

$$a \circ b = a \circ c \implies b = c \quad \forall a, b, c \in G \quad (1.1)$$

B

$$\begin{aligned} a \circ b &= a \circ c \\ \underbrace{a^{-1} \circ a}_{e} \circ b &= a^{-1} \circ a \circ c \\ e \circ b &= c \end{aligned}$$

S Jedes linksneutrale Element einer Gruppe G ist gleichzeitig auch rechtsneutrales Element (einfach: **neutrales Element**).

B

Es gilt

$$\begin{aligned} a &= e \circ a \\ a^{-1} \circ a &= a^{-1} \circ (e \circ a) \\ e &= a^{-1} \circ (e \circ a) \end{aligned} \quad (1.2)$$

Weiterhin gilt auch

$$\begin{aligned} e &= e \circ e \\ e &= (a^{-1} \circ a) \circ e \\ e &= a^{-1} \circ (a \circ e) \end{aligned} \quad (1.3)$$

Gleichsetzen von (1.2) und (1.3) führt auf $a^{-1} \circ (e \circ a) = a^{-1} \circ (a \circ e)$. Anwendung der Kürzungsregel 1 führt schliesslich auf $e \circ a = a \circ e$, d.h. e ist links- und rechtsneutrales Element. Zusätzlich ist noch strenggenommen die Eindeutigkeit zu zeigen.

S

Für alle Elemente a einer Gruppe G gilt:

$$(a^{-1})^{-1} = a$$

B Sei $b \in G$. Dann ist wegen d) $b^{-1} \circ b = e$. Setze $b = a^{-1}$. Dann folgt:

$$\begin{aligned} (a^{-1})^{-1} \circ a^{-1} &= e \\ ((a^{-1})^{-1} \circ a^{-1}) \circ a &= e \circ a \\ (a^{-1})^{-1} \circ (a^{-1} \circ a) &= a \\ (a^{-1})^{-1} \circ e &= a \\ (a^{-1})^{-1} &= a \end{aligned}$$

S Jedes linksinverse Element einer Gruppe G ist gleichzeitig auch rechtsinverses Element (einfach: **inverses Element**).

B Sei $b \in G$. Dann ist $b^{-1} \circ b = e$. Setze $b = a^{-1}$:

$$\begin{aligned} (a^{-1})^{-1} \circ a^{-1} &= e \\ a \circ a^{-1} &= e \end{aligned}$$

D.h. a^{-1} ist links- und rechtsinverses Element von a .

S **Kürzungsregel 2**

G sei eine Gruppe bzgl. \circ , dann gilt:

$$b \circ a = c \circ a \implies b = c \quad \forall a, b, c \in G \quad (1.4)$$

B siehe Kürzungsregel 1

S **Eindeutige Lösung einer Gleichung**

G sei eine Gruppe bzgl. \circ , dann gilt:

$$\left. \begin{aligned} 1) \quad a \circ x = b &\implies x = a^{-1} \circ b \\ 2) \quad x \circ a = b &\implies x = b \circ a^{-1} \end{aligned} \right\} \forall a, b \in G$$

B

- 1) $a \circ x = b$
 $a^{-1} \circ a \circ x = a^{-1} \circ b$
 $x = a^{-1} \circ b$
- 2) entsprechend

1.4 Funktionen

D Seien A und B zwei beliebige Mengen, so versteht man unter einer **Funktion** oder **Abbildung f von A nach B**

$$f: A \longrightarrow B$$

eine eindeutige Vorschrift, die einem $x \in A$ genau ein $y \in B$ zuordnet:

$$f: x \longmapsto y = f(x)$$

y heißt **Funktionswert** oder **Bildpunkt** von x .

Die Menge aller $x \in A$, denen durch f ein $y \in B$ zugeordnet wird ist der **Definitionsbereich** $D(f)$ der Funktion.

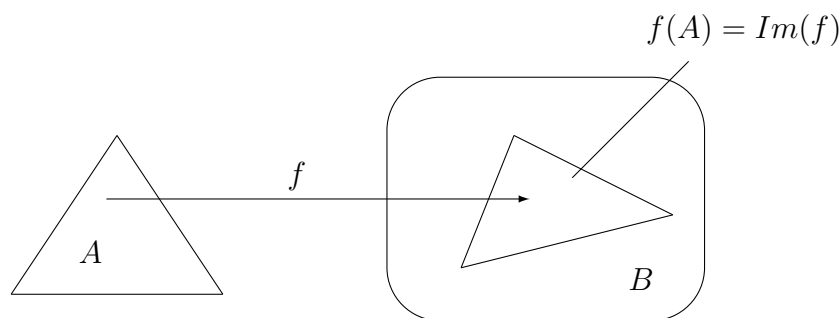
Die Menge B ist der **Wertevorrat** $W(f)$ der Funktion.

Die Menge aller Bildpunkte der Elemente einer Teilmenge U von A heißt **Bildmenge von U** .

$$f(U) = \{f(x) | x \in U \subseteq A\}$$

Offensichtlich gilt: $f(U) \subseteq B$.

Die Bildmenge von A wird als die **Bildmenge der Funktion f** bzw. **Wertebereich der Funktion f** , kurz $Im(f)$, bezeichnet.



Fallen B und $f(A) = Im(f)$ zusammen, so spricht man von einer **Abbildung von A auf B** . Die Funktion $f: A \longrightarrow B$ wird dann als **surjektiv** bezeichnet.

Gilt

$$f(x_1) = f(x_2) \implies x_1 = x_2 \quad \forall x_1, x_2 \in A$$

d.h. zwei verschiedene Elemente x_1, x_2 besitzen immer verschiedene Bildpunkte, dann nennt man die Funktion **injektiv** oder **eindeutig**.

Ist $f: A \longrightarrow B$ sowohl surjektiv als auch injektiv, so nennt man f **bijektiv**.

Zu jeder bijektiven Abbildung $f : A \longrightarrow B$ gibt es eine **Umkehrfunktion** f^{-1} mit:

$$\begin{aligned} f^{-1} : B &\longrightarrow A \\ y &\longmapsto x \in A \quad (\text{mit } f(x) = y) \end{aligned}$$

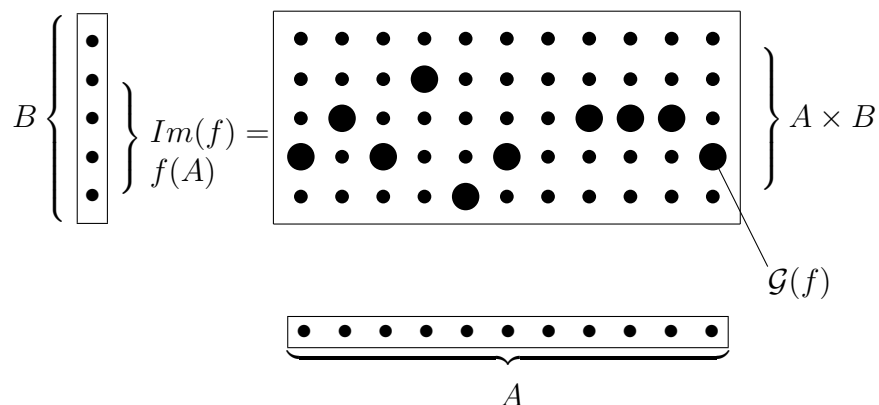
Zu jeder Funktion $f : A \longrightarrow B$ gehört ihr **Graph** $\mathcal{G}(f)$, der als Teilmenge des direkten Produktes von A und B definiert ist:

$$\begin{aligned} \mathcal{G}(f) &= \{(x, y) \mid x \in A \wedge y = f(x) \in B\} \\ \mathcal{G}(f) &\subset A \times B \end{aligned}$$

Anmerkung: Zu jedem $x \in A$ gibt es genau ein $y \in B$ mit $(x, y) \in \mathcal{G}$. Dieser Sachverhalt bietet eine alternative Möglichkeit den Funktionsbegriff einzuführen ohne den Ausdruck ‘‘Vorschrift’’ zu verwenden.

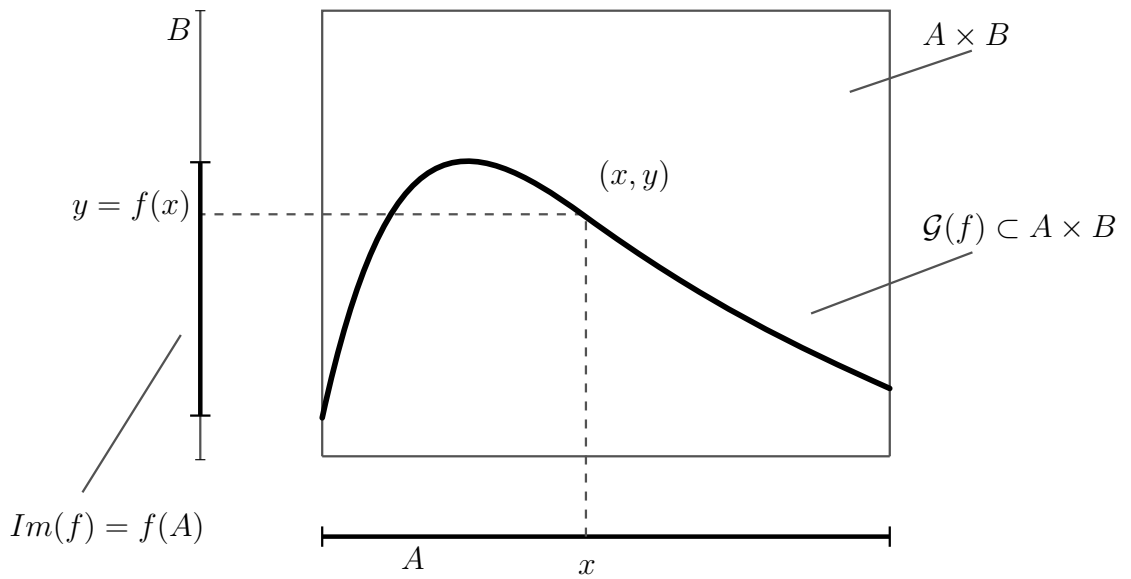
Beispiele:

- A und B sind endliche Mengen



- $A, B \subset \mathbb{R}$

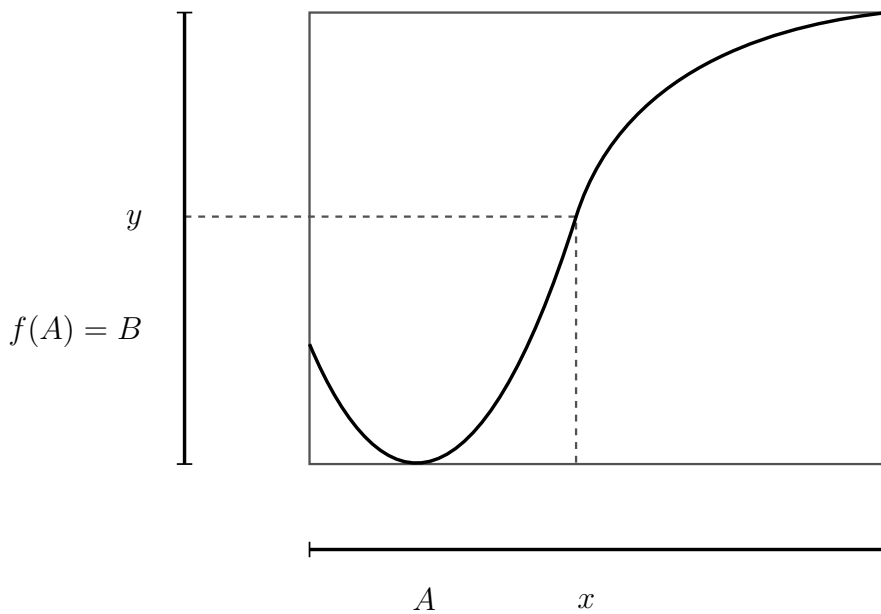
\mathbb{R} ist die Menge aller reellen Zahlen (s.u.)



• surjektive Funktion

$A, B \subset \mathbb{R}, f(A) = B$

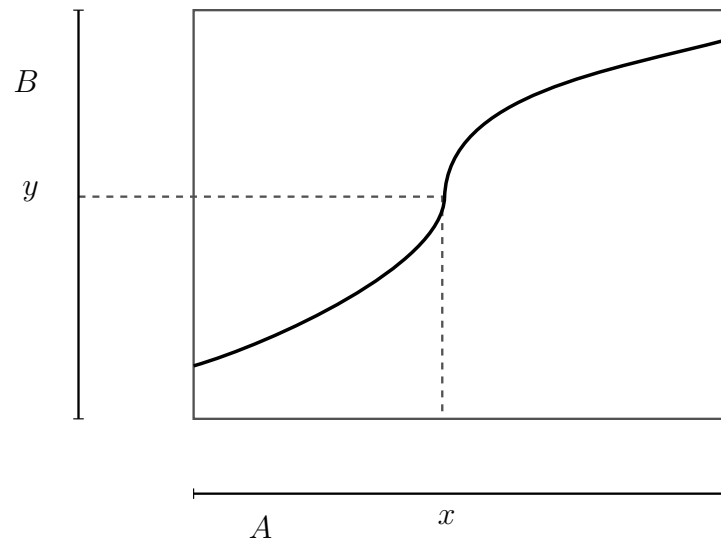
Jedes $y \in B$ wird mindestens einmal als Funktionswert angenommen.



• injektive Funktion

$A, B \subset \mathbb{R}, f(x_1) = f(x_2) \implies x_1 = x_2$

Jedes $y \in B$ wird höchstens einmal als Funktionswert angenommen.



- bijektive Funktion mit Umkehrfunktion
 $A, B \subset \mathbb{R}$, f und f^{-1} sind surjektiv und injektiv d.h. bijektiv
 jedes $y \in B$ wird genau einmal als Funktionswert angenommen.

CAS-Beispiel



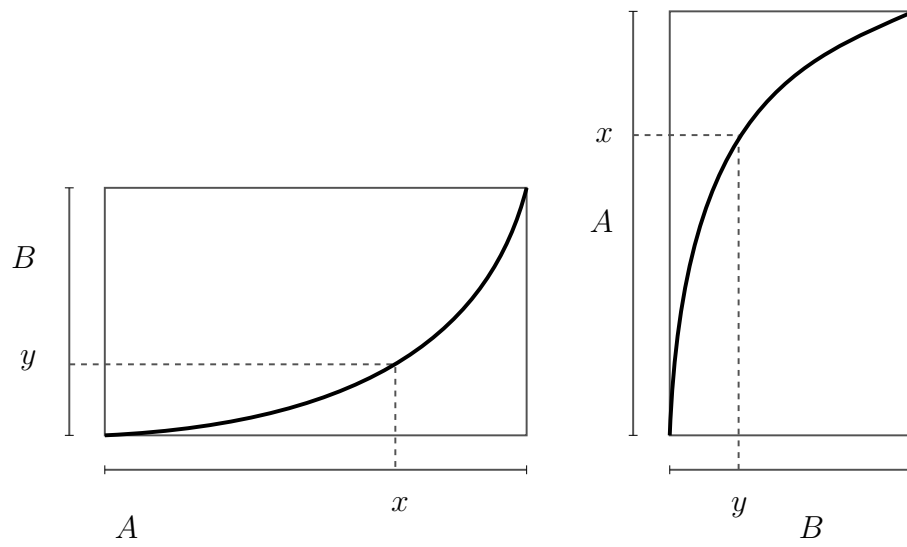
Die meisten numerischen Funktionen können dem Modul `NumPy` entnommen werden. Zur graphischen Darstellung wird das Modul `Matplotlib` verwendet (beide müssen Sie ggf. zunächst installieren). Im Folgenden sei das für x bzw. y dargestellte Intervall der Definitions- bzw. Wertebereich einer Funktion $f(x)$. Rufen Sie die folgenden Graphikbefehle auf und entscheiden Sie, ob die jeweilige Funktion surjektiv, injektiv oder bijektiv ist. Ändern Sie die Aufrufe ab und finden Sie je ein eigenes Beispiel für eine surjektive, eine injektive und eine bijektive Funktion.

```
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(-5, 5)
plt.axis([-5, 5, -3, 30])
plt.plot(x, x**2)
plt.show()
```

```
x = np.linspace(0, 3)
plt.axis([0, 3, 0, 20])
plt.plot(x, np.exp(x))
plt.show()
```

```
x = np.linspace(-2*np.pi, 2*np.pi)
plt.axis([-2*np.pi, 2*np.pi, -1, 1])
plt.plot(x, np.sin(x))
plt.show()
```

1.5 Zahlen

1.5.1 Natürliche Zahlen, Prinzip der vollständigen Induktion

D Der “natürliche Zählprozeß”, der ausgehend von der “0” die **natürlichen Zahlen** ($\mathbb{N} = \{0, 1, 2, 3, \dots\}$) erzeugt, läßt sich durch die **Peano-Axiome** formalisieren:

- a) Die natürlichen Zahlen bilden eine Menge \mathbb{N} mit dem ausgezeichneten Element “0”.
- b) Auf \mathbb{N} ist eine Abbildung $\nu : \mathbb{N} \mapsto \mathbb{N} \setminus \{0\}$ erklärt, die zu $n \in \mathbb{N}$ den Nachfolger $\nu(n)$ angibt.
- c) $n_1 \neq n_2 \implies \nu(n_1) \neq \nu(n_2)$
- d) Enthält eine Teilmenge $A \subseteq \mathbb{N}$ die Zahl 0 und ist mit jedem $n \in A$ auch $\nu(n) \in A$, so ist $A = \mathbb{N}$ (Prinzip der vollständigen Induktion).

Statt $0, \nu(0), \nu(\nu(0)), \nu(\nu(\nu(0))), \dots$ schreibt man $0, 1, 2, 3, \dots$

! Alle Eigenschaften der natürlichen Zahlen lassen sich “rein logisch” aus den Peano-Axiomen ableiten. Insbesondere ist festzuhalten:

- Auf der Menge der natürlichen Zahlen ist eine **Ordnung** (kleiner, gleich, größer) gegeben, mit der Eigenschaft: für beliebige $x, y \in \mathbb{N}$ gilt entweder $x < y$ oder $x = y$ oder $x > y$.
- Es lassen sich die bekannten Rechenoperationen einführen:

–Addition	}	kommutativ
–Multiplikation		
–Subtraktion		
–Division		

Alternative Formulierung für das **Prinzip der vollständigen Induktion**:

$\mathcal{A}(n)$ sei eine Aussage über die Zahl n ($n \in \mathbb{N}$). Ist die Aussage für $n = n_0$ richtig und folgt aus der Richtigkeit von $\mathcal{A}(k)$ für ein beliebiges $k \in \mathbb{N}, k \geq n_0$, diejenige von $\mathcal{A}(k+1)$,

so ist $\mathcal{A}(n)$ für alle $n \in \mathbb{N}, n \geq n_0$ erfüllt.



Die ‘‘Aussagenkette’’ muß nicht unbedingt bei $n_0 = 0$ gestartet werden.

Beispiel:



Für jedes $n \geq 1$ gilt:

$$\sum_{i=1}^n i = 1 + 2 + \dots + n = \frac{n(n+1)}{2}$$

Induktionsanfang: Die Aussage ist richtig für $n = n_0 = 1$.

Induktionsannahme: Die Aussage ist richtig für ein $k \geq 1$.

Induktionsschritt:

$$\begin{aligned} \sum_{i=1}^{k+1} i &= \left(\sum_{i=1}^k i \right) + (k+1) \\ &= \frac{k(k+1)}{2} + (k+1) \\ &= \frac{k(k+1) + 2(k+1)}{2} \\ &= \frac{(k+1)(k+2)}{2} \\ &= \frac{(k+1)[(k+1)+1]}{2} \end{aligned}$$

Dies ist genau die Beziehung die für $(k+1)$ erwartet wird! Damit folgt aus der Korrektheit der Aussage für k die für $(k+1)$. Da die Aussage für $n = n_0 = 1$ korrekt ist, ist sie somit für alle $n \geq 1$ korrekt.

CAS-Beispiel



Berechne die Summe $s = \sum_{k=1}^{100} k$ aller natürlichen Zahlen k von 1 bis 100

```
s = 0
for i in range(101):
    s = s + i

s
```

Durch die Schleifenkonstruktion `for i in range(101): ...` werden alle natürlichen Zahlen i von 1 bis 100 durchlaufen. Die Zahlen i werden dabei aufsummiert, wobei die Zwischensumme in der Variable s gespeichert wird.

Kürzer geht es mit der Python-internen Funktion `sum` oder der Funktion `Sum` des Moduls `SymPy`. Die Auswertung des Ausdrucks kann mit `.doit()` forciert werden:

```
sum(i for i in range(101))
```

```
from sympy.abc import i
from sympy import Sum

Sum(i, (i, 1, 100)).doit()
```

Für eine unbestimmte obere Summationsgrenze n schreibt man:

```
from sympy.abc import n

Sum(i, (i, 1, n)).doit()
```

Vollkommen entsprechend lassen sich kompliziertere Summen wie $\sum_{k=m}^n k$, $\sum_{l=0}^{(n-1)} (2l+1)$,

$\sum_{k=1}^n k^2$ oder $\sum_{k=1}^n k^3$ auswerten:

```
from sympy.abc import k, l, m, n

Sum(k, (k, m, n)).doit()
```

```
Sum(2*l+1, (l, 0, n-1)).doit()
```

```
Sum(k**2, (k, 1, n)).doit()
```

```
Sum(k**3, (k, 1, n)).doit()
```

1.5.2 Ganze, rationale und reelle Zahlen

\mathbb{N} bildet bzgl. der Addition keine Gruppe.

D Die Menge \mathbb{Z} der ganzen Zahlen entsteht durch Erweiterung der Menge \mathbb{N} durch

Ergänzung mit den inversen Elementen bzgl. der Addition.

Es gilt: $\mathbb{N} \subset \mathbb{Z}$

S \mathbb{Z} bildet bzgl. der Addition eine abelsche Gruppe.

Die Ordnung von \mathbb{N} wird auf \mathbb{Z} übertragen.

$\mathbb{Z}^* = \mathbb{Z} \setminus \{0\}$ bildet bzgl. der Multiplikation keine Gruppe.

D Die Menge \mathbb{Q} der rationalen Zahlen ist die Menge aller Zahlen, die sich durch p/q mit $p \in \mathbb{Z}$, $q \in \mathbb{N}^* = \mathbb{N} \setminus \{0\}$ darstellen lassen.

$$\mathbb{Q} = \{x \mid x = p/q, p \in \mathbb{Z}, q \in \mathbb{N}^*\}$$

Es gilt: $\mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q}$

D Auf einer Menge K sei eine Addition $+$ und eine Multiplikation \cdot definiert. K wird Körper genannt, falls

- K eine abelsche Gruppe bzgl. der Addition bildet mit dem neutralen Element "0"
- $K^* = K \setminus \{0\}$ eine abelsche Gruppe bzgl. der Multiplikation bildet
- das Distributivgesetz

$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \forall a, b, c \in K$$

gilt.

S \mathbb{Q} ist ein geordneter Körper. Die Ordnung von \mathbb{Z} überträgt sich auf \mathbb{Q} durch

$$\frac{p}{q} < \frac{p'}{q'} \iff p \cdot q' < p' \cdot q$$

S Jede rationale Zahl läßt sich als endlicher oder periodischer Dezimalbruch darstellen

und umgekehrt.

Beispiel:

$$\begin{aligned}\frac{1}{6} &= 0,166666\dots = 0,1\bar{6} \\ 0,1\bar{6} &= \frac{1}{10}(1 + 0,\bar{6}) \\ &= \frac{1}{10}\left(1 + \frac{1}{10}(6 + 0,\bar{6})\right)\end{aligned}$$

$$\begin{aligned}\text{Nebenrechnung: } 0,\bar{6} &= \frac{1}{10}(6 + 0,\bar{6}) \\ 9 \cdot 0,\bar{6} &= 6 \\ 0,\bar{6} &= \frac{2}{3}\end{aligned}$$

$$\begin{aligned}\text{Damit: } 0,1\bar{6} &= \frac{1}{10}\left(1 + \frac{2}{3}\right) \\ &= \frac{1}{10} \cdot \frac{5}{3} \\ &= \frac{1}{6}\end{aligned}$$

S Die Gleichung $x^2 = 2$ besitzt keine Lösung in \mathbb{Q} .

$$\nexists x(x^2 = 2, x = p/q \in \mathbb{Q})$$

B Annahme: Behauptung doch erfüllt (*):

$$(p/q)^2 = 2 \quad \text{mit} \quad p \in \mathbb{Z}, q \in \mathbb{N}^* .$$

Es gilt: p und q sind nicht beide gleichzeitig gerade (**) – ansonsten wird solange gekürzt bis dies erfüllt ist. Nun ist aber nach Annahme:

$$\begin{aligned}p^2 &= 2q^2, \\ &\Downarrow\end{aligned}$$

d.h. p ist gerade.[§] Setze $p = 2m$

[§] Dies gilt, da für $p = 2n$ gerade $\implies p^2 = 4n^2$ gerade und $p = (2n+1)$ ungerade $\implies p^2 = 4n^2 + 4n + 1$ ungerade.

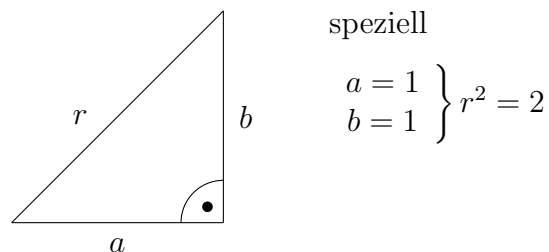
$$\begin{aligned}
 (2m)^2 &= 2q^2 \\
 4m^2 &= 2q^2 \\
 2m^2 &= q^2 \\
 &\downarrow
 \end{aligned}$$

q ist gerade

Dies stellt einen Widerspruch zu obiger Aussage (**) dar. Damit muß die ursprüngliche Annahme (*) falsch sein; d.h. die Aussage des Satzes ist richtig!

Dennoch muß es Zahlen r mit der Eigenschaft $r^2 = 2$ geben.

Satz des Pythagoras $r^2 = a^2 + b^2$



r nennt man **irrationale** Zahl.

Jede irrationale Zahl läßt sich durch einen unendlichen nichtperiodischen Dezimalbruch darstellen.

D Die Menge aller rationalen und irrationalen Zahlen bildet die **Menge der reellen Zahlen** \mathbb{R} .

Es gilt: $\mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R}$

\mathbb{R} ist ein **geordneter Körper**

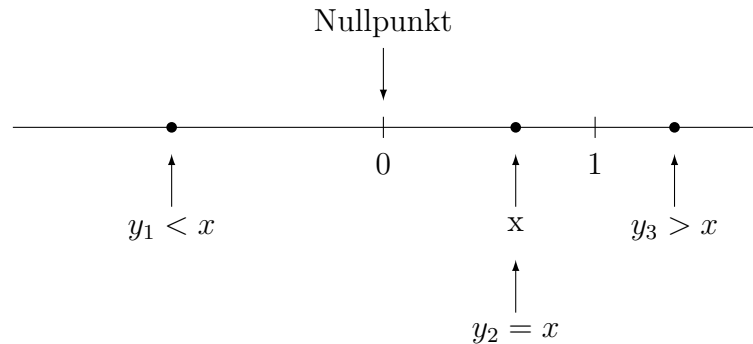
d.h. u.a.

$$x > y \implies x + z > y + z$$

und

$$x > 0, y > 0 \implies x \cdot y > 0$$

Graphische Darstellung mittels der **Zahlengeraden**:



Intervalle (Teilmengen) von \mathbb{R}

$[a, b]$	$= \{x \mid a \leq x \leq b, x \in \mathbb{R}\}$	abgeschlossen	
$]a, b]$	$= \{x \mid a < x \leq b, x \in \mathbb{R}\}$	halboffen	
$[a, b[$	$= \{x \mid a \leq x < b, x \in \mathbb{R}\}$	halboffen	
$]a, b[$	$= \{x \mid a < x < b, x \in \mathbb{R}\}$	offen	

CAS-Beispiel



Zur Lösung einer Gleichung kann in Python das Modul **SymPy** verwendet werden (dieses müssen Sie ggf. noch installieren). Dazu muß zunächst die Gleichung so umgestellt werden, daß auf der rechten Seite der Gleichung 0 steht. Beachten Sie zudem, daß x , genau wie später a , b und c durch `from sympy.abc import ...` als Variablen deklariert werden. Löse die Gleichung $x^2 = 2 \Rightarrow x^2 - 2 = 0$:

```
from sympy.abc import x
from sympy.solvers import solve

solve(x**2 - 2, x)
```

Auch **SymPy** liefert keine rationale Zahl als Ergebnis, sondern die beiden Lösungen $\sqrt{2}$ und $-\sqrt{2}$, die zu einer Lösungsmenge zusammengefaßt sind.

Natürlich läßt sich mit der Funktion `solve` auch die sogenannte *Mitternachtsformel* erhalten, d.h. die Lösungen der Gleichung $ax^2 + bx + c = 0$ erhält man mit:

```
from sympy.abc import a, b, c

solve(a*x**2 + b*x + c, x)
```

Die einzelnen Lösungen können in einer Lösungsmenge A abgelegt werden.


```

A = solve(a*x**2 + b*x + c, x)

a0 = 2
b0 = 3
c0 = -1

for l in range(len(A)):
    A[l] = A[l].subs([(a, a0), (b, b0), (c, c0)])
    print("Loesung " + str(l+1) + ": " + str(A[l]))
    x = A[l]
    TEST = a0*x**2 + b0*x + c0
    print("Einsetzen liefert (symbolisch):" + str(TEST)),
    print("Einsetzen liefert (numerisch) :" + str(float(TEST)))

```

In der vorausgegangenen Auflistung der Lösungen wurden die einzelnen Lösungen in der Liste `A` abgespeichert. Auf die Lösung l kann nun über `A[l]` direkt zugegriffen werden. Wie man sieht, muß die Gleichung nicht neu gelöst werden, sondern es wird bei einer nachträglichen Festlegung der Koeffizienten a , b und c auf die zuvor erfolgte allgemeine Lösung zurückgegriffen. Dies erfolgt durch den Aufruf `subs`, bei dem die symbolischen Ausdrücke a , b und c durch die numerischen Werte von a_0 , b_0 und c_0 ersetzt werden. Durch `float` kann der symbolische Ausdruck in einen numerischen umgewandelt werden.

Für die getroffene Wahl der Koeffizienten a , b und c ist das Ganze gut gegangen, d.h. die Lösungen der Gleichung sind reelle Zahlen. Hätten wir für $c = +2$ gewählt, so würde dies nicht mehr zutreffen. Dies gilt bereits für den Spezialfall $a = 1$, $b = 0$ und $c = 1$, der im Folgenden behandelt wird.

1.5.3 Komplexe Zahlen

S Die Gleichung $r^2 = -1$ besitzt keine Lösung in \mathbb{R} .

$$\nexists r (r^2 = -1, r \in \mathbb{R})$$

B

$$\left. \begin{array}{l} x > 0, y > 0 \implies x \cdot y > 0 \\ x < 0, y < 0 \implies x \cdot y > 0 \end{array} \right\} x \neq 0 \in \mathbb{R} \implies x^2 > 0$$

damit gilt für $r \in \mathbb{R}$

$$\begin{array}{l} r = 0 \implies r^2 = 0 \\ r \neq 0 \implies r^2 > 0 \end{array}$$

d.h. der Fall $r^2 < 0$ tritt niemals auf für $r \in \mathbb{R}$.

D Die **imaginäre Zahl** i ist definiert als Lösung der Gleichung $x^2 = -1$, durch $i^2 = -1$.

D Die Menge der **komplexen Zahlen** \mathbb{C} ist die Menge aller Zahlen der Form

$$z = x + iy \quad \text{mit} \quad x, y \in \mathbb{R}$$

x wird **Realteil** von z genannt; $x = \operatorname{Re}(z)$

y wird **Imaginärteil** von z genannt; $y = \operatorname{Im}(z)$

N Alternative Schreibweise: $z = (x, y) \in \mathbb{R}^2$

D Die **Addition** auf \mathbb{C} ist definiert durch:

$$(x + iy) + (u + iv) = (x + u) + i(y + v) \in \mathbb{C}$$

↑

Abgeschlossenheit

Beispiel:

$$(5 + i3) + (7 - i2) = 12 + i$$

D Die **Multiplikation** auf \mathbb{C} ist definiert durch:

$$\begin{aligned} (x + iy) \cdot (u + iv) &= x \cdot u + x \cdot iv + iy \cdot u + iy \cdot iv \\ &= (x \cdot u + i^2 y \cdot v) + i(x \cdot v + y \cdot u) \\ &= \underbrace{(x \cdot u - y \cdot v)}_{\in \mathbb{R}} + i \underbrace{(x \cdot v + y \cdot u)}_{\in \mathbb{R}} \in \mathbb{C} \end{aligned}$$

↑

Abgeschlossenheit

Beispiel:

$$\begin{aligned} (5 + i3) \cdot (7 - i2) &= 35 - (-6) + i(-10 + 21) \\ &= 41 + i11 \end{aligned}$$

S \mathbb{C} ist ein Körper bzgl. der Addition und Multiplikation.

Es gilt: $\mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R} \subset \mathbb{C}$

\mathbb{C} ist nicht geordnet; d.h. $z_1 < z_2$ ergibt keinen Sinn

CAS-Beispiel



Löse die Gleichung $x^2 = -1$ nach x auf:

```
from sympy.abc import x
from sympy.solvers import solve

solve(x**2 + 1, x)
```

Die imaginäre Zahl i wird also von SymPy durch `I` dargestellt. `I` ist daher ein reservierter Variablenname und sollte vom Nutzer nicht verwendet werden. Von Python wird die imaginäre Zahl i standardmäßig als `j` dargestellt. In dieser Schreibweise werden komplexe Zahlen als $a + bj$ ausgedrückt. Beachten Sie, daß die Angabe des Imaginärteils in dieser Schreibweise zwingend erforderlich ist. Ein Vorteil ist, daß die Variable `j` vom Benutzer ohne Probleme verwendet werden kann (siehe unten).

Das Rechnen mit komplexen Zahlen erfolgt nun in der Weise wie man es vom Umgang mit reellen Zahlen gewohnt ist:

```
from sympy import I

(5+3*I) + (7-2*I)
```

```
(5+3j) + (7-2j)
```

```
j = 5
(5+3j) + (7-2j)
```

```
j = "Text"
(5+3j) + (7-2j)
```

```
(5+3j) * (7-2j)
```

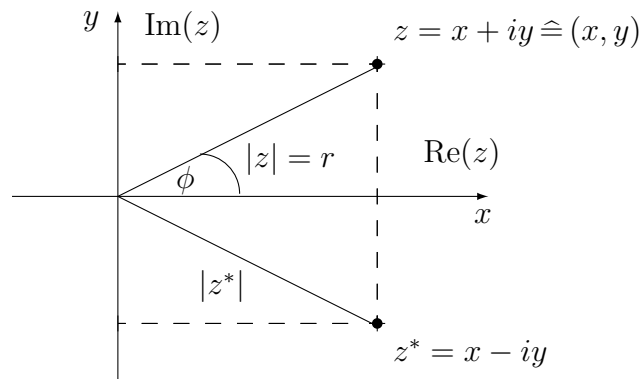
Einige kompliziertere Beispiele sind:

```
(5+3j) / (7-2j)
```

```
(1+1j)**4
```

```
(5+3j)**2 / (1+1j)
```

Darstellung komplexer Zahlen in der **Gaußschen Zahlenebene**



D $z^* = x - iy$ wird die zu $z = x + iy$ **komplex konjugierte Zahl** genannt.

D Der Ausdruck $|z| = \sqrt{z z^*}$ wird **Absolutbetrag** der komplexen Zahl z genannt.

$$\sqrt{z z^*} = \sqrt{(x + iy)(x - iy)} = \sqrt{x^2 + y^2}$$

CAS-Beispiel



Für das Rechnen mit komplexen Zahlen gibt es einige spezielle Funktionen: `conjugate` liefert zu z das komplex konjugierte z^* , den Real- und Imaginärteil von z erhält man mit `real` bzw. `imag` und den Absolutbetrag schließlich mit `abs`:

```
z = 5 + 12j
```

```
z.conjugate()
```

```
z.real
```

```
z.imag
```

```
abs(z)
```

```
from math import sqrt
sqrt(z.real**2 + z.imag**2)
```

Damit kann man sich leicht von der Korrektheit der Relationen $(z^*)^* = z$, $\operatorname{Re}(z) = \frac{1}{2}(z + z^*)$ sowie $\operatorname{Im}(z) = \frac{1}{2i}(z - z^*)$ überzeugen:

```
z.conjugate().conjugate()
```

```
(z + z.conjugate()) / 2
```

```
(z - z.conjugate()) / 2j
```

Darstellung in **Polarkoordinaten** (r, ϕ) :

Aus der **Eulerschen Formel** (Ableitung erfolgt später)

$$e^{i\phi} = \cos \phi + i \sin \phi$$

ergibt sich die Darstellung (für $z \neq 0$):

$$\begin{aligned} z &= x + iy & \phi &= \operatorname{sign}(y) \arccos \frac{x}{|z|}, & -\pi < \phi \leq \pi \\ &= |z| \cos \phi + i |z| \sin \phi \\ &= |z| (\cos \phi + i \sin \phi) \\ &= |z| e^{i\phi} \\ &= r e^{i\phi} \end{aligned}$$

r ist der **Absolutbetrag** und ϕ die **Phase** der komplexen Zahl z . Die Vorzeichenfunktion ist definiert als

$$\operatorname{sign}(x) = \begin{cases} -1 & \text{für } x < 0 \\ 0 & \text{für } x = 0 \\ +1 & \text{für } x > 0 \end{cases}$$

CAS-Beispiel



Für die Darstellung einer komplexen Zahl in kartesischer bzw. polarer Form stellt das Modul `mpmath` die Funktionen `rect` bzw. `polar` zur Verfügung. Die Funktion `polar` verwendet als Argument eine komplexe Zahl der Form `a + bj` (Python-Standard) bzw. `a + b*I` (SymPy) und gibt als Ergebnis ein Dupel (r, ϕ) aus.

Führen Sie die folgenden Kommandos aus und überlegen Sie sich weitere Beispiele:

```
from mpmath import mp, rect, polar
mp.pretty = True

a = -3 + 4j
b = -1
c = 1 + 1j
```

```
polar(a)
```

```
polar(b)
```

```
polar(c)
```

```
a*c
polar(a*c)
```

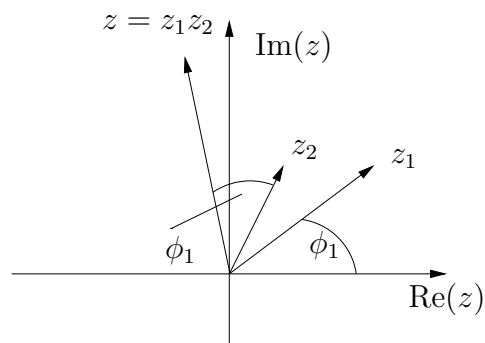
Die Funktion `rect` verwendet als Argumente die Variablen r und ϕ und gibt als Ergebnis eine komplexe Zahl der Form $a + bi$ aus.

```
rect(3, 0.5)
```

In der Polardarstellung lassen sich die Multiplikation, Division und das Potenzieren komplexer Zahlen besonders einfach ausführen. Mit $z_1 = r_1 e^{i\phi_1}$ und $z_2 = r_2 e^{i\phi_2}$ gilt:

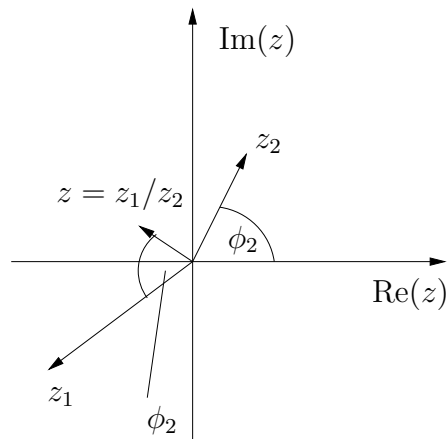
Multiplikation:

$$\begin{aligned} z &= z_1 \cdot z_2 \\ &= r_1 e^{i\phi_1} \cdot r_2 e^{i\phi_2} \\ &= r_1 r_2 e^{i\phi_1} e^{i\phi_2} \\ &= r_1 r_2 e^{i(\phi_1 + \phi_2)} \end{aligned}$$



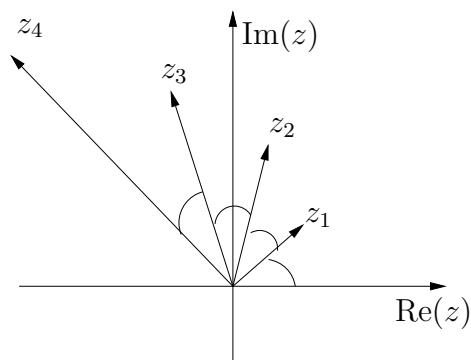
Division:

$$\begin{aligned} z &= z_1 / z_2 \\ &= \frac{r_1}{r_2} e^{i(\phi_1 - \phi_2)} \end{aligned}$$



Potenzieren:

$$\begin{aligned} z^n &= z_1^n \\ &= r_1^n e^{i\phi n} \end{aligned}$$



CAS-Beispiel



Die Prozedur `plot_komplex` gibt komplexe Zahlen in kartesischer und polarer Form aus und bildet diese in der Gaußschen Zahlenebene ab. Dabei werden die komplexen Zahlen in die internen Variablen z_1 , z_2 etc. überführt. Führen Sie die folgenden Aufrufe aus und überzeugen Sie sich, daß alles zusammenpaßt.

Zuvor muß das Modul `mathchem` zur Verfügung gestellt werden. Eine Anleitung zur Installation des Moduls sowie das Modul selbst finden Sie am Ende dieses Skriptes.

```
from mathchem import plot_komplex

a = 3 + 5j
b = -2 - 3j
```

```
plot_komplex([a, b, a+b])
```

```
plot_komplex([a, b, a-b])
```

```
plot_komplex([a, b, a*b])
```

```
plot_komplex([a, b, a/b])
```

```
plot_komplex([a, b, a**b])
```

Ändern Sie die Werte für a und b ab und gehen Sie die verschiedenen Aufrufe durch.

Kapitel 2

Analytische Geometrie und Lineare Algebra

2.1 Vektoralgebra

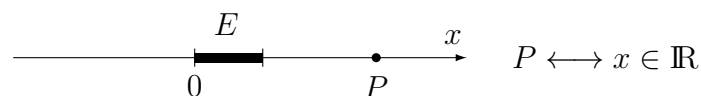
Motivation:

- geometrisch anschauliche Zusammenhänge mathematisch auszudrücken
- Möglichkeit geometrische Objekte Rechenoperationen zu unterwerfen
- Einführung von Begriffen, die in vielen anderen Bereichen Anwendung finden

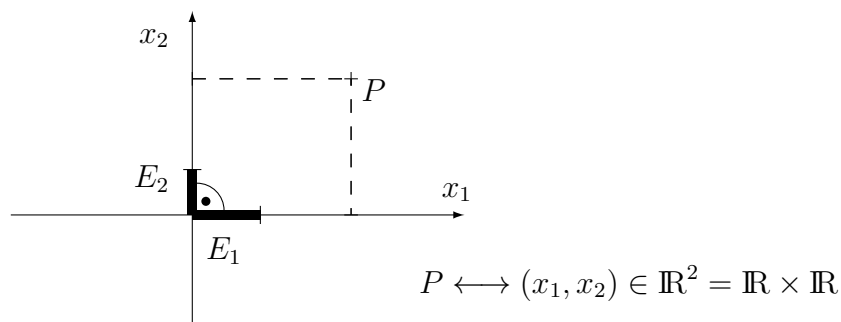
2.1.1 Einführung des Vektorbegriffs in Anlehnung an die Euklidische Geometrie

Repräsentation von Punkten im Ortsraum durch n -Tupel

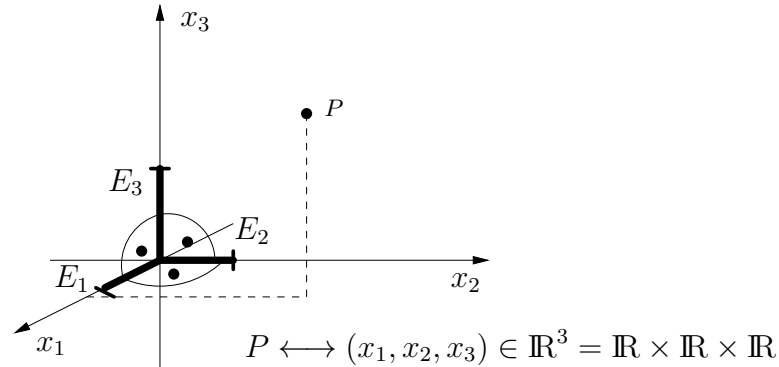
- Punkte auf einer Geraden



- Punkte in einer Ebene



- Punkte im 3-dimensionalen Ortsraum



Jeder Punkt im Ortsraum lässt sich eindeutig einem Punkt eines abstrakten Raumes zuordnen.

! Bei der Zuordnung von Punkten des Ortsraumes zu Zahlentripeln hängen die Koordinaten von der Wahl des Koordinatensystems ab.

! Es wird im folgenden immer ein (rechtwinkliges) **kartesisches Koordinatensystem** vorausgesetzt

D Unter dem **n-dimensionalen Raum** \mathbb{R}^n versteht man die Menge aller geordneten n -Tupel (x_1, x_2, \dots, x_n) mit $x_i \in \mathbb{R}$. Jedes n -Tupel repräsentiert einen Punkt im Raum \mathbb{R}^n . Die Größen x_i sind seine **Koordinaten**.

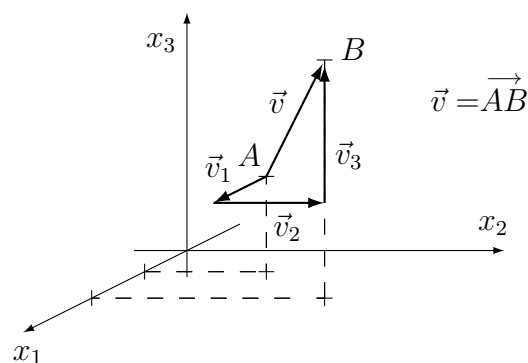
D Unter einem **Vektor** \vec{v} des \mathbb{R}^n versteht man das n -Tupel, das sich aus der Differenz der Koordinaten zweier Punkte des \mathbb{R}^n ergibt.

$$\vec{v} = \begin{pmatrix} x_{1,B} - x_{1,A} \\ x_{2,B} - x_{2,A} \\ \vdots \\ x_{n,B} - x_{n,A} \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}$$

N Schreibweise in der Regel als **Spaltenvektor**, gelegentlich als **Zeilenvektor** (v_1, v_2, \dots, v_n)

Symbol: \vec{v} , \underline{v} , \mathbf{v} u.s.w.

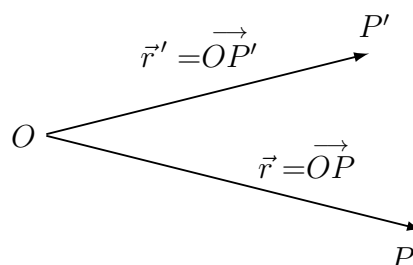
Geometrische Deutung:



- Ein Vektor ist eine gerichtete Strecke
- Der Aufpunkt (A) ist nicht in Raum festgelegt

! Die **Koordinaten** v_1, v_2, \dots, v_n bzw. **Komponenten** $\vec{v}_1, \vec{v}_2, \dots, \vec{v}_n$ eines Vektors hängen von der Wahl des Koordinatensystems ab.

D **Ortsvektoren** sind ortsfeste Vektoren mit dem Ursprung als Aufpunkt.

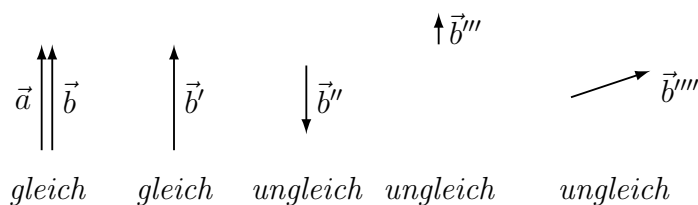


2.1.2 Gleichheit, Addition, Subtraktion und Multiplikation mit einem Skalar

D Zwei Vektoren \vec{a} und \vec{b} aus \mathbb{R}^n sind **gleich** wenn ihre Koordinaten gleich sind.

$$a_i = b_i \quad i = 1, \dots, n$$

Beispiel:

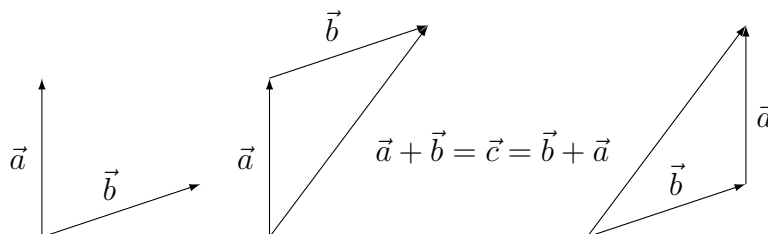


D Zwei Vektoren \vec{a} und \vec{b} aus \mathbb{R}^n werden zu einem Vektor \vec{c} **addiert** indem man die Koordinaten addiert.

$$\vec{c} = \vec{a} + \vec{b} \quad \text{mit} \quad c_i = a_i + b_i \quad i = 1, \dots, n$$

$$\text{bzw.} \quad \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} a_1 + b_1 \\ \vdots \\ a_n + b_n \end{pmatrix}$$

Beispiel:



S Die Addition von Vektoren ist kommutativ.

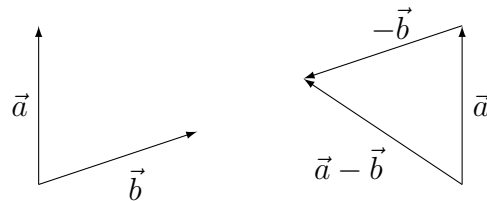
$$\vec{a} + \vec{b} = \vec{b} + \vec{a}$$

B

$$\vec{a} + \vec{b} = \begin{pmatrix} a_1 + b_1 \\ \vdots \\ a_n + b_n \end{pmatrix} = \begin{pmatrix} b_1 + a_1 \\ \vdots \\ b_n + a_n \end{pmatrix} = \vec{b} + \vec{a}$$

D Zwei Vektoren werden voneinander **subtrahiert** indem ihre Koordinaten voneinander subtrahiert werden:

$$\vec{c} = \vec{a} - \vec{b} = \begin{pmatrix} a_1 - b_1 \\ \vdots \\ a_n - b_n \end{pmatrix} = \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix}$$

Beispiel:

D Der **Nullvektor** ist ein Vektor dessen Koordinaten alle gleich Null sind.

$$\vec{0} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

S Die Menge aller Vektoren aus \mathbb{R}^n bildet bzgl. der Vektoraddition eine **kommutative Gruppe**, da die Gruppenaxiome

$$G1: \text{ Abgeschlossenheit } \vec{a}, \vec{b} \in \mathbb{R}^n \implies \vec{a} + \vec{b} = \vec{c} \in \mathbb{R}^n$$

- G2: Assoziativgesetz $\vec{a} + (\vec{b} + \vec{c}) = (\vec{a} + \vec{b}) + \vec{c}$
 G3: Existenz eines neutralen Elements $\vec{0} + \vec{a} = \vec{a}$, $\vec{0} \in \mathbb{R}^n$
 G4: Existenz eines inversen Elements $\vec{a}^{-1} + \vec{a} = \vec{0}$, mit $\vec{a}^{-1} = -\vec{a}$
 sowie das Kommutativgesetz $\vec{a} + \vec{b} = \vec{b} + \vec{a}$

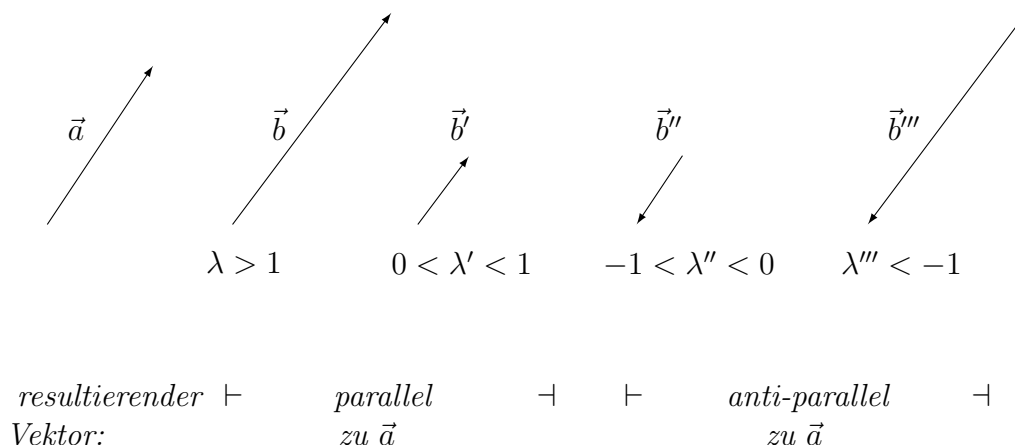
erfüllt sind.

D Multiplikation eines Vektors mit einem Skalar

Ein Vektor $\vec{a} \in \mathbb{R}^n$ wird mit einem Skalar $\lambda \in \mathbb{R}$ multipliziert indem jede Koordinate des Vektors mit λ multipliziert wird.

$$\vec{b} = \lambda \vec{a} \quad \text{mit} \quad b_i = \lambda a_i \quad i = 1, \dots, n$$

Beispiel:



CAS-Beispiel



Wenn Sie das Tool Jupyter Notebook verwenden, führen Sie diesen Befehl aus, um interaktive Plots zu erhalten. Ansonsten können Sie den Befehl ignorieren.

```
%matplotlib notebook
```

Vektoren können in Python als Arrays erzeugt werden. Das Modul `mathchem` stellt die Funktion `plot_2Dvektor` zur Verfügung, mit der zweidimensionale Vektoren graphisch dargestellt werden können. Die eingegebenen Vektoren werden dabei unabhängig von ihrer ursprünglichen Benennung in die internen Variablen \vec{a} , \vec{b} , \vec{c} etc. überführt.

Führen Sie die folgenden Befehle aus und überzeugen Sie sich von der Korrektheit der Ergebnisse.

```
from numpy import array
a = array([[3, 1]])
b = array([[-1, 4]])
```

Ggf. muß das Modul `mathchem` zur Verfügung gestellt werden:

```
from mathchem import plot_2Dvektor
```

```
plot_2Dvektor([a, b, a+b])
```

```
plot_2Dvektor([a, b, a-b])
```

```
plot_2Dvektor([a, b, 3*a+2*b])
```

```
plot_2Dvektor([a, 2*a])
```

```
plot_2Dvektor([a, -1*a])
```

```
plot_2Dvektor([a, -a])
```

Ändern Sie die Vektoren \vec{a} und \vec{b} ab und stellen Sie diese neuen Vektoren ebenfalls graphisch dar:

```
a = array([[1, 0]])
b = array([[0, 1]])
```

```
plot_2Dvektor([a, b, a+b])
```

```
plot_2Dvektor([a, b, a-b])
```

```
plot_2Dvektor([a, b, 3*a+2*b])
```

```
plot_2Dvektor([a, 2*a])
```

```
plot_2Dvektor([a, -1*a])
```

```
plot_2Dvektor([a, -a])
```

S Rechenregeln ($\mu, \lambda \in \mathbb{R}; \vec{a}, \vec{b} \in \mathbb{R}^n$)

$$a) \lambda \vec{a} = \vec{a} \lambda$$

$$b) \lambda(\mu \vec{a}) = (\lambda \mu) \vec{a}$$

$$c) \lambda(\vec{a} + \vec{b}) = \lambda \vec{a} + \lambda \vec{b}$$

$$d) (\lambda + \mu) \vec{a} = \lambda \vec{a} + \mu \vec{a}$$

CAS-Beispiel



Verwenden Sie die Funktion `plot_2Dvektor` des `mathchem`-Moduls und überzeugen Sie sich von der Korrektheit der Rechenregeln.

```
from numpy import array
a = array([[25.4, 41.2]])
b = array([[-22.3, 15.7]])
```

Ggf. muß das `mathchem`-Modul noch geladen werden:

```
from mathchem import plot_2Dvektor
```

```
plot_2Dvektor([a, 0.4*a])
```

```
plot_2Dvektor([a, a*0.4])
```

```
plot_2Dvektor([a, 0.4*(2*a)])
```

```
plot_2Dvektor([a, (0.4*2)*a])
```

```
plot_2Dvektor([a, 0.5*(a+b)])
```

```
plot_2Dvektor([a, 0.5*a+0.5*b])
```

```
plot_2Dvektor([a, (0.6-0.2)*a])
```

```
plot_2Dvektor([a, 0.6*a-0.2*a])
```


Die Operationen

$$\vec{c} = \lambda \vec{a} \quad \text{mit} \quad \lambda = 0$$

$$\text{bzw.} \quad \vec{c} = \vec{a} - \vec{b} \quad \text{mit} \quad \vec{a} = \vec{b}$$

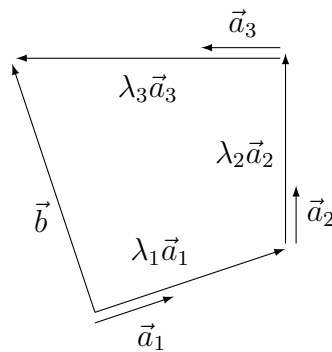
führen auf den Nullvektor $\vec{c} = \vec{0}$ mit $c_i = 0, i = 1, \dots, n$

D Die Addition von k Vektoren \vec{a}_i ($i = 1, \dots, k$), die mit Vorfaktoren λ_i versehen sind, nennt man **Linearkombination**.

$$\vec{b} = \sum_{i=1}^k \lambda_i \vec{a}_i = \lambda_1 \vec{a}_1 + \dots + \lambda_k \vec{a}_k$$

$$= \begin{pmatrix} \lambda_1 a_{1,1} + \lambda_2 a_{1,2} + \dots + \lambda_k a_{1,k} \\ \vdots \\ \lambda_1 a_{n,1} + \lambda_2 a_{n,2} + \dots + \lambda_k a_{n,k} \end{pmatrix}$$

Beispiel:



Schwerpunkt:

$$\vec{r}_{\text{Sp}} = \frac{\sum_i m_i \vec{r}_i}{\sum_i m_i}$$

elektrisches Dipolmoment:

$$\vec{d} = \sum_i q_i \vec{r}_i$$

2.1.3 Das skalare Produkt zweier Vektoren

D Das **innere oder skalare Produkt** (\vec{a}, \vec{b}) zweier Vektoren \vec{a} und $\vec{b} \in \mathbb{R}^n$ ist gegeben durch die reelle Zahl:

$$\begin{aligned}(\vec{a}, \vec{b}) &= \sum_{i=1}^n a_i b_i \\ &= a_1 b_1 + a_2 b_2 + \dots + a_n b_n\end{aligned}$$

! Unabhängig von der Wahl des Koordinatensystems (Beweis folgt später).

Allgemein definiert man für komplexwertige Koordinaten:

$$\begin{aligned}(\vec{a}, \vec{b}) &= \sum_{i=1}^n a_i^* b_i \quad \text{mit } \vec{a}, \vec{b} \in \mathbb{C}^n \\ &= \left(\sum_{i=1}^n a_i b_i^* \right)^* = (\vec{b}, \vec{a})^*\end{aligned}$$

S **Rechenregeln** ($\lambda \in \mathbb{R}$; $\vec{a}, \vec{b}, \vec{c} \in \mathbb{R}^n$)

- a) $(\vec{a}, \vec{b}) = (\vec{b}, \vec{a})$ *Kommutativgesetz*
- b) $(\vec{a}, \vec{b} + \vec{c}) = (\vec{a}, \vec{b}) + (\vec{a}, \vec{c})$ *Distributivgesetz*
- c) $\lambda(\vec{a}, \vec{b}) = (\lambda\vec{a}, \vec{b}) = (\vec{a}, \lambda\vec{b})$

CAS-Beispiel



Das Skalarprodukt zweier Vektoren \vec{a} und \vec{b} lässt sich in Python auf viele Arten erhalten. Testen Sie die verschiedenen Methoden aus:

```
from numpy import array
a = array([[1, 2, 5]])
b = array([[4, -1, 1]])
```

Manuelle Berechnung des Skalarprodukts:

```
a[0, 0]*b[0, 0] + a[0, 1]*b[0, 1] + a[0, 2]*b[0, 2]
```

Alternative Konstruktion in Form einer for-Schleife...

```
n = 3
s = 0

for i in range(n):
    s = s + a[0, i]*b[0, i]

s
```

... und als Summe:

```
sum(a[0, i]*b[0, i] for i in range(3))
```

Deutlich einfacher geht es mit folgenden Ausdrücken:

```
from numpy import inner

inner(a, b)
inner(a, b)[0, 0]
```

Überzeugen Sie sich anhand der folgenden Beispiele von der Korrektheit der Rechenregeln für Skalarprodukte:

```
a = array([[1, 2, 4]])
b = array([[0, -2, 7]])
c = array([[5, 7, -3]])
```

```
inner(a, b)[0, 0]
```

```
inner(b, a)[0, 0]
```

```
inner(a, b+c)[0, 0]
```

```
inner(a, b)[0, 0] + inner(a, c)[0, 0]
```

```
2*inner(a, b)[0, 0]
```

```
inner(2*a, b)[0, 0]
```

```
inner(a, 2*b)[0, 0]
```

B

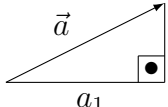
zu a)
$$(\vec{a}, \vec{b}) = \sum_{i=1}^n a_i b_i = \sum_{i=1}^n b_i a_i = (\vec{b}, \vec{a})$$

D Die **Länge eines Vektors** $\vec{a} \in \mathbb{R}^n$ ist definiert durch die reelle Zahl a mit:

$$a = \sqrt{(\vec{a}, \vec{a})} = \left(\sum_{i=1}^n a_i^2 \right)^{1/2} \quad \text{bzw.} \quad \left(\sum_{i=1}^n a_i^* a_i \right)^{1/2} \quad \text{für} \quad \vec{a} \in \mathbb{C}^n$$

N $|\vec{a}| = a$ “ a -Betrag”

Beispiel:

$n = 2$: Satz des Pythagoras:  $|\vec{a}| = \sqrt{a_1^2 + a_2^2}$

D Unter **Normierung** eines Vektors versteht man die Multiplikation mit einem Skalar, so daß der neue Vektor \vec{a}' eine vorgegebene Länge λ besitzt:

$$\vec{a}' = \frac{\lambda}{|\vec{a}|} \vec{a}$$

Speziell: $\lambda = 1$ führt \vec{a} in einen sogenannten **Einheitsvektor** über.

$$\vec{a}' = \vec{e} = \frac{1}{|\vec{a}|} \vec{a} \quad \text{mit} \quad |\vec{e}| = 1$$

Beispiel:

$$\vec{a} = \begin{pmatrix} 1 \\ -1 \\ 2 \end{pmatrix} \implies |\vec{a}| = \sqrt{6} \quad \text{und} \quad \vec{e} = \frac{1}{\sqrt{6}} \vec{a} = \frac{1}{\sqrt{6}} \begin{pmatrix} 1 \\ -1 \\ 2 \end{pmatrix} = \begin{pmatrix} 1/\sqrt{6} \\ -1/\sqrt{6} \\ 2/\sqrt{6} \end{pmatrix}$$

N $\vec{e}_{\vec{a}}, \hat{e}_{\vec{a}}$ oder \hat{a} .

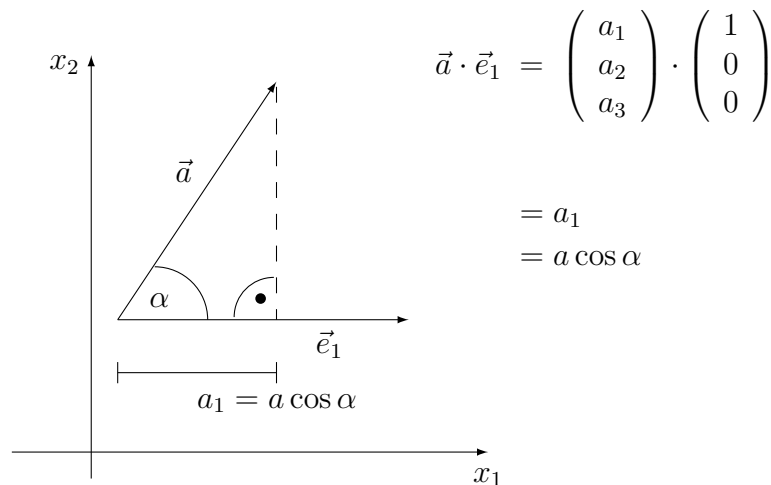
Speziell: Einheitsvektoren längs der Koordinatenachsen $\vec{e}_x, \vec{e}_y, \vec{e}_z$ oder $\hat{x}, \hat{y}, \hat{z}$

D Zwei Vektoren \vec{a} und \vec{b} sind **orthogonal** zueinander falls $(\vec{a}, \vec{b}) = 0$ ist.

Bedeutung des skalaren Produkts im Ortsraum

N In diesem Zusammenhang schreibt man üblicherweise $\vec{a} \cdot \vec{b}$ statt (\vec{a}, \vec{b})

Speziell: \vec{b} ist Einheitsvektor parallel zu einer der Koordinatenachsen, z.B. $\vec{b} = \vec{e}_1$.



Da das Skalarprodukt unabhängig vom Koordinatensystem ist, gilt allgemein:

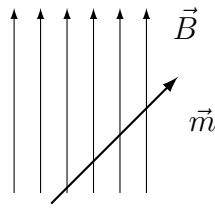
$$\begin{aligned} \vec{a} \cdot \vec{b} &= ab \cos \sphericalangle(\vec{a}, \vec{b}) \\ &= ab \cos \alpha \end{aligned}$$

für orthogonale Vektoren gilt $\vec{a} \cdot \vec{b} = 0$ d.h.

$$\cos \alpha = 0 \implies \alpha = \pm 90^\circ \quad \text{bzw.} \quad \pm \frac{\pi}{2}$$

Anwendungen:

Magnetisches Moment \vec{m} im Magnetfeld \vec{B}



Zeeman-Energie $E = -\vec{m} \cdot \vec{B}$

Arbeit W gegen eine konstante Kraft \vec{F} (Erdbziehung) längs eines Weges \vec{s}

$$\begin{aligned} W &= \vec{F} \cdot \vec{s} \\ \vec{F} &= -\vec{F}_{\text{grav}} \\ \vec{F}_{\text{grav}} &= m\vec{g} = mg \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \\ \vec{s} &= a \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \\ W &= mga(0 \cdot 1 + 0 \cdot 2 + 1 \cdot 3) \\ &= 3mga \end{aligned}$$

2.1.4 Das Kreuzprodukt und Spatprodukt

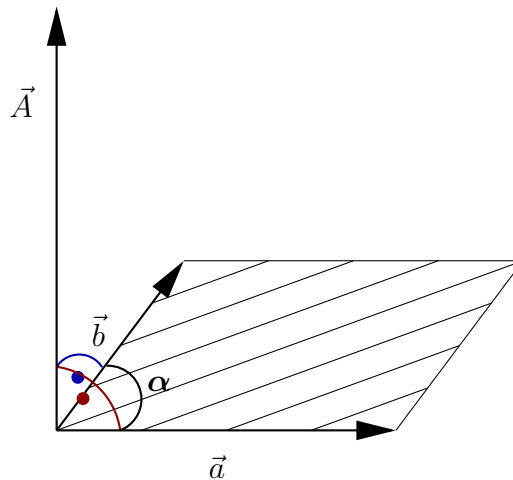
D Das **Kreuzprodukt** $\vec{a} \times \vec{b}$ zweier Vektoren \vec{a} und $\vec{b} \in \mathbb{R}^3$ ist definiert durch:

$$\vec{c} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{pmatrix} = \vec{a} \times \vec{b}$$

Merkregel

$$\begin{aligned} \vec{c} &= \begin{array}{c} + \quad + \quad + \quad - \quad - \\ \left| \begin{array}{ccc} \vec{e}_1 & \vec{e}_2 & \vec{e}_3 \\ \times & \times & \times \\ a_1 & a_2 & a_3 \\ \times & \times & \times \\ b_1 & b_2 & b_3 \end{array} \right| \\ = \vec{e}_1 a_2 b_3 + \vec{e}_2 a_3 b_1 + \vec{e}_3 a_1 b_2 \\ - \vec{e}_1 a_3 b_2 - \vec{e}_2 a_1 b_3 - \vec{e}_3 a_2 b_1 \end{array} \end{aligned}$$

Bedeutung des Kreuzprodukts im Ortsraum

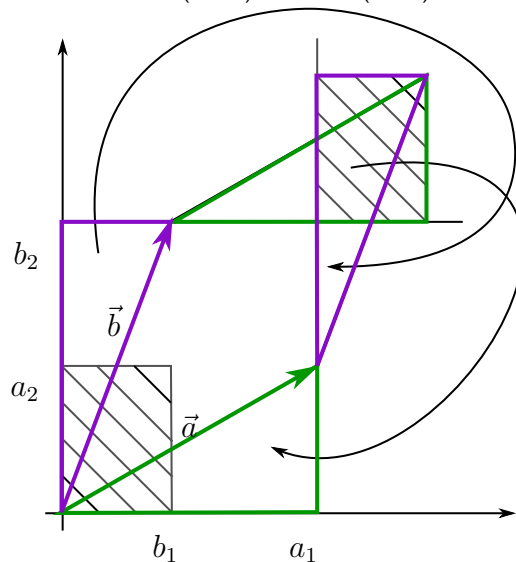


$\vec{A} = \vec{a} \times \vec{b}$ gibt die Fläche des von \vec{a} und \vec{b} aufgespannten Parallelogramms an:

$$|\vec{A}| = |\vec{a} \times \vec{b}| = |\vec{a}| |\vec{b}| \sin \alpha$$

wobei \hat{A} die Orientierung der Fläche angibt. Die Vektoren \vec{a} , \vec{b} und \vec{A} bilden ein **Rechtssystem**.

Für den Spezialfall $\vec{a} = \begin{pmatrix} a_1 \\ a_2 \\ 0 \end{pmatrix}$, $\vec{b} = \begin{pmatrix} b_1 \\ b_2 \\ 0 \end{pmatrix}$ ist dieser Zusammenhang leicht zu zeigen:



$$\vec{A} = \vec{a} \times \vec{b} \quad \text{mit} \quad A_3 = (a_1 b_2 - a_2 b_1)$$

d.h. die Fläche des Parallelogramms, das von \vec{a} und \vec{b} aufgespannt wird, ist gleich der Differenz der Rechteckflächen $a_1 b_2$ und $a_2 b_1$.

S Rechenregeln

$$a) \vec{a} \times \vec{b} = -\vec{b} \times \vec{a}$$

$$b) \lambda(\vec{a} \times \vec{b}) = (\lambda\vec{a}) \times \vec{b} = \vec{a} \times (\lambda\vec{b})$$

$$c) \vec{a} \times (\vec{b} + \vec{c}) = \vec{a} \times \vec{b} + \vec{a} \times \vec{c}$$

$$d) \vec{a} \times (\vec{b} \times \vec{c}) \neq (\vec{a} \times \vec{b}) \times \vec{c}$$

e) *Zyklische Vertauschung:*

$$(\vec{a} \times \vec{b}) \cdot \vec{c} = (\vec{b} \times \vec{c}) \cdot \vec{a} = (\vec{c} \times \vec{a}) \cdot \vec{b}$$

f) *Grassmannscher Entwicklungssatz:*

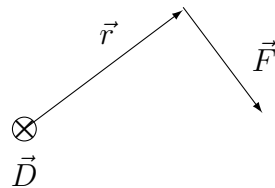
$$\vec{a} \times (\vec{b} \times \vec{c}) = \vec{b}(\vec{a} \cdot \vec{c}) - \vec{c}(\vec{a} \cdot \vec{b})$$

g) *Lagrangesche Identität:*

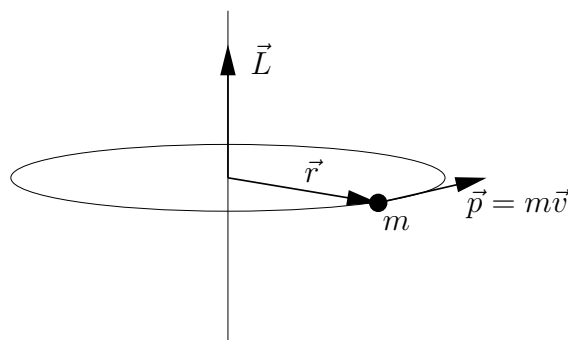
$$(\vec{a} \times \vec{b}) \cdot (\vec{c} \times \vec{d}) = (\vec{a} \cdot \vec{c})(\vec{b} \cdot \vec{d}) - (\vec{a} \cdot \vec{d})(\vec{b} \cdot \vec{c})$$

Anwendungen:

$$\text{Drehmoment} \quad \vec{D} = \vec{r} \times \vec{F}$$



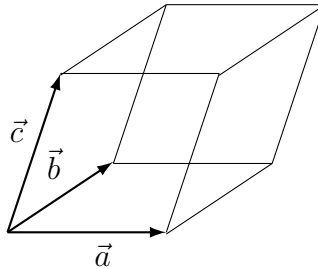
$$\text{Drehimpuls} \quad \vec{L} = \vec{r} \times \vec{p}$$



$$\text{Lorentzkraft} \quad \vec{F} = q\vec{v} \times \vec{B}$$

D Das **Spatprodukt** dreier Vektoren $\vec{a}, \vec{b}, \vec{c} \in \mathbb{R}^3$ ist definiert durch:

$$(\vec{a} \times \vec{b}) \cdot \vec{c}$$



S Das **Spatprodukt** $(\vec{a} \times \vec{b}) \cdot \vec{c}$ gibt das Volumen des durch die Vektoren \vec{a} , \vec{b} , \vec{c} aufgespannten Parallelepipeds an:

$$V = (\vec{a} \times \vec{b}) \cdot \vec{c}$$

! $V \geq 0$ falls \vec{a} , \vec{b} , \vec{c} ein Rechtssystem bilden; sonst gilt $V < 0$.

2.1.5 Geraden- und Ebenengleichung

Im folgenden wird i.allg. von $\vec{a}, \vec{b}, \dots \in \mathbb{R}^3$ ausgegangen.

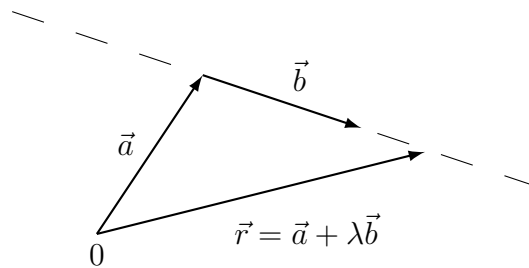
D Der **Abstand** zweier Punkte A und B ist die Länge des Vektors, der sich als Differenz ihrer Ortsvektoren \vec{r}_A und \vec{r}_B ergibt:

$$d_{AB} = d_{BA} = |\vec{r}_A - \vec{r}_B|$$

S Alle Punkte deren Ortsvektoren durch

$$\vec{r} = \vec{a} + \lambda \vec{b} \quad \lambda \in \mathbb{R}; \vec{b} \neq \vec{0}$$

gegeben sind, liegen auf einer **Geraden**.

**CAS-Beispiel**

Nutzen Sie die Prozedur `plot_gerade` des Moduls `mathchem` und stellen Sie folgende Geraden in drei Dimensionen dar:

```
from mathchem import plot_gerade
from numpy import array

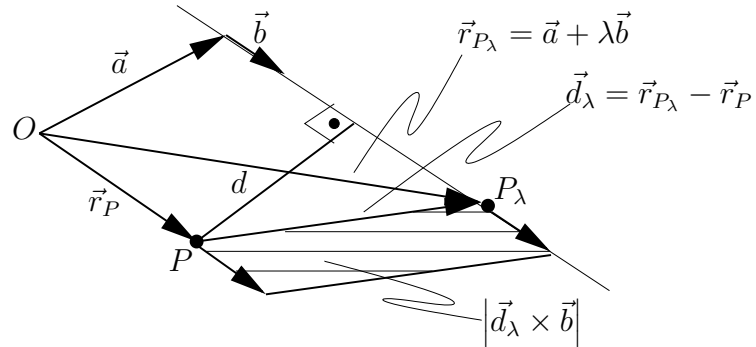
a = array([[1, 2, 3]])
b = array([[-2, 0, 7]])
```

```
plot_gerade(a, b, 3.5)
```

```
plot_gerade(a, b, -2.2)
```

```
plot_gerade(a, b, 0)
```

Abstand eines Punktes zu einer Geraden:



Es gilt:

$$d |\vec{b}| = |\vec{d}_\lambda \times \vec{b}| \quad \text{mit} \quad \vec{d}_\lambda = \vec{r}_{P_\lambda} - \vec{r}_P$$

wobei λ willkürlich gewählt werden kann

$$\begin{aligned} d &= \frac{|\vec{d}_\lambda \times \vec{b}|}{|\vec{b}|} \\ &= |\vec{d}_\lambda \times \hat{b}| \end{aligned}$$

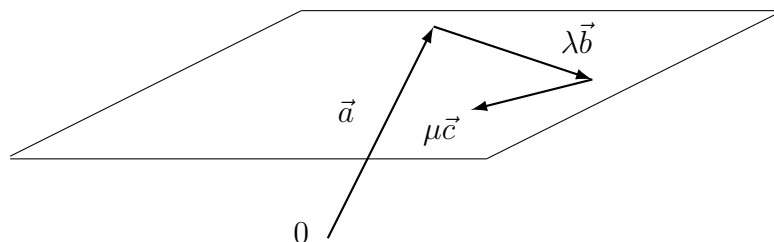
Speziell: $\lambda = 0$

$$\begin{aligned} d &= |(\vec{a} - \vec{r}_P) \times \hat{b}| \\ &= |\vec{a} \times \hat{b} - \vec{r}_P \times \hat{b}| \end{aligned}$$

S Alle Punkte deren Ortsvektoren durch

$$\vec{r} = \vec{a} + \lambda \vec{b} + \mu \vec{c} \quad \lambda, \mu \in \mathbb{R}; \vec{b} \times \vec{c} \neq \vec{0}$$

gegeben sind, liegen auf einer **Ebene**.

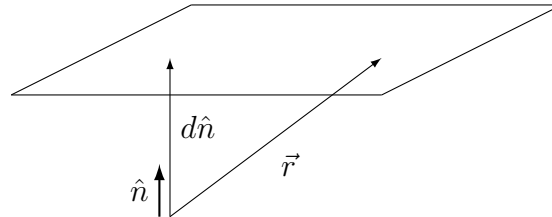


S Hessesche Normalform

Alle Punkte deren Ortsvektoren die Gleichung

$$\hat{n} \cdot \vec{r} = d \quad d \in \mathbb{R}; \hat{n} \neq \vec{0}$$

erfüllen, liegen auf einer Ebene senkrecht zum Einheitsvektor \hat{n} im Abstand d zum Ursprung.



2.1.6 Lineare Vektorräume

D Sei V eine abelsche Gruppe bzgl. $+$, K ein Körper und α eine Abbildung mit der Eigenschaft:

$$\begin{aligned} \alpha &: K \times V \longrightarrow V \\ (\lambda, \vec{a}) &\longmapsto \alpha(\lambda, \vec{a}) = \lambda \vec{a} \end{aligned}$$

dann heißt V **linearer Vektorraum** falls die Eigenschaften

$$\left. \begin{array}{l} \text{Abgeschlossenheit} \\ a) \lambda \vec{a} + \mu \vec{b} \in V \\ \text{Assoziativgesetz} \\ b) \lambda(\mu \vec{a}) = (\lambda\mu)\vec{a} \\ \text{Distributivgesetze} \\ c) (\lambda + \mu)\vec{a} = \lambda \vec{a} + \mu \vec{a} \\ d) \lambda(\vec{a} + \vec{b}) = \lambda \vec{a} + \lambda \vec{b} \end{array} \right\} \begin{array}{l} \forall \lambda, \mu \in K \\ \forall \vec{a}, \vec{b} \in V \end{array}$$

erfüllt sind.

$$\begin{array}{ll} \vec{a} \in V & \text{heißt Vektor} \\ \lambda \in K & \text{heißt Skalar} \end{array}$$

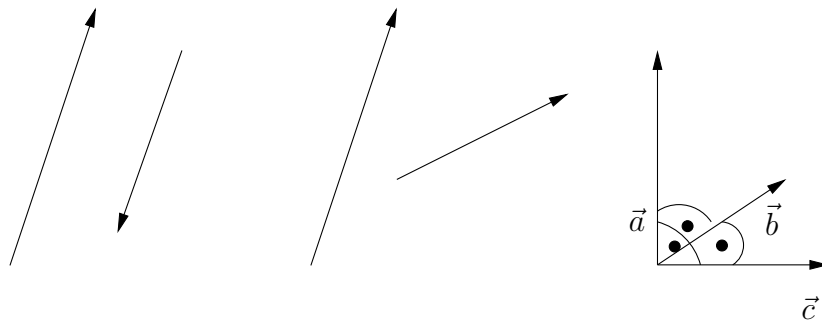
! Im allgemeinen wird im folgenden $V = \mathbb{R}^n$ und $K = \mathbb{R}$ vorausgesetzt; das Adjektiv

“linear” entfällt.

D p Vektoren $\vec{a}_1, \dots, \vec{a}_p$ eines Vektorraumes V heißen **linear abhängig** wenn es $\lambda_1, \dots, \lambda_p \in \mathbb{R}$ gibt, die nicht alle gleichzeitig 0 sind, so daß

$$\lambda_1 \vec{a}_1 + \dots + \lambda_p \vec{a}_p = \vec{0}$$

Nicht linear abhängige Vektoren heißen **linear unabhängig**.



linear abhängig

linear unabhängig

D Gibt es in einem Vektorraum eine maximale Zahl k von linear unabhängigen Vektoren, so heißt k die **Dimension** von V , sonst heißt V **unendlich dimensional**.

N

$$\dim V = k \text{ oder } V_k \text{ statt } V$$

D Ist V_k ein Vektorraum der Dimension k , so heißen je k linear unabhängige Vektoren eine **Basis** von V_k .

N

Die Basisvektoren $\vec{a}_1, \dots, \vec{a}_k$ spannen den Vektorraum V_k auf.

S

Ist V_k ein Vektorraum der Dimension k und bilden $\vec{a}_1, \dots, \vec{a}_k$ eine Basis, so läßt sich

jeder Vektor \vec{b} aus V auf **eindeutige** Weise als Linearkombination

$$\vec{b} = \sum_{i=1}^k \lambda_i \vec{a}_i$$

darstellen.

B

$\vec{a}_1, \dots, \vec{a}_k$ sind linear unabhängig
 $\vec{b}, \vec{a}_1, \dots, \vec{a}_k$ sind damit linear abhängig

$$\implies \lambda_b \vec{b} + \lambda_1 \vec{a}_1 + \dots + \lambda_k \vec{a}_k = \vec{0}$$

$$\vec{b} = -\frac{\lambda_1}{\lambda_b} \vec{a}_1 - \dots - \frac{\lambda_k}{\lambda_b} \vec{a}_k$$

$$\text{oder } \vec{b} = \lambda'_1 \vec{a}_1 + \dots + \lambda'_k \vec{a}_k$$

damit ist \vec{b} als Linearkombination darstellbar.

Eindeutigkeit:

$$\begin{aligned} \text{Annahme es gelte weiter: } \vec{b} &= \mu_1 \vec{a}_1 + \dots + \mu_k \vec{a}_k \\ \text{Differenz bilden } \implies \vec{0} &= (\lambda'_1 - \mu_1) \vec{a}_1 + \dots + (\lambda'_k - \mu_k) \vec{a}_k \\ &= \nu_1 \vec{a}_1 + \dots + \nu_k \vec{a}_k \end{aligned}$$

Es gilt $\nu_i = 0 \forall i$ wegen der linearen Unabhängigkeit der \vec{a}_i .

Damit $\lambda'_i = \mu_i \forall i$ und damit ist die Eindeutigkeit bewiesen.

S

Alternative Formulierung: Jedem Vektor \vec{b} eines linearen Vektorraumes V_k der Dimension k wird bezüglich der festen Basis $\vec{a}_1, \dots, \vec{a}_k$ durch die Vektorgleichung

$$\vec{b} = \lambda_1 \vec{a}_1 + \dots + \lambda_k \vec{a}_k$$

in umkehrbar eindeutiger Weise ein geordnetes k -Tupel $(\lambda_1, \dots, \lambda_k)$ zugeordnet

$$\vec{b} \longleftrightarrow (\lambda_1, \dots, \lambda_k).$$

N

Die λ_i heißen **Komponenten** von \vec{b} bzgl. der Basis $\vec{a}_1, \dots, \vec{a}_k$

Alternativ:

$$\begin{aligned}\lambda_i &: \text{Koordinaten} \\ \lambda_i \vec{a}_i &: \text{Komponenten}\end{aligned}$$

Berechnung der Komponenten

$$\lambda_1 \vec{a}_1 + \dots + \lambda_k \vec{a}_k = \vec{b}$$

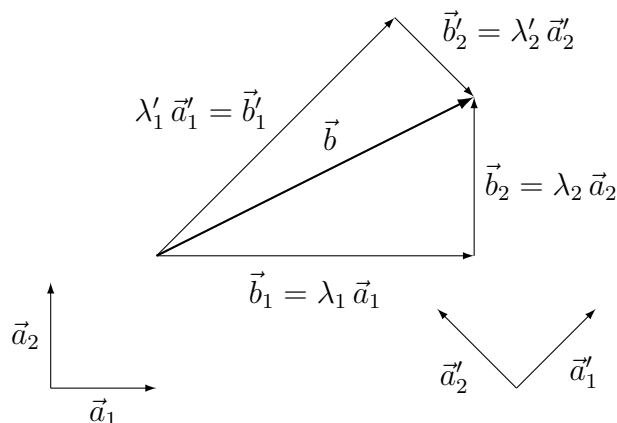
$$\vec{a}_i = \begin{pmatrix} a_{1,i} \\ \vdots \\ a_{k,i} \end{pmatrix}$$

führt auf

$$\begin{pmatrix} \lambda_1 a_{1,1} + \lambda_2 a_{1,2} + \dots + \lambda_k a_{1,k} \\ \vdots \\ \lambda_1 a_{i,1} + \lambda_2 a_{i,2} + \dots + \lambda_k a_{i,k} \\ \vdots \\ \lambda_1 a_{k,1} + \lambda_2 a_{k,2} + \dots + \lambda_k a_{k,k} \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_i \\ \vdots \\ b_k \end{pmatrix}$$

Dies stellt ein System von k linearen Gleichungen in den Unbekannten $\lambda_1, \dots, \lambda_k$ dar. Der Lösungsweg wird später behandelt.

D Den Übergang bei der Festlegung der Komponenten eines Vektors bzgl. einer Basis $\vec{a}_1, \dots, \vec{a}_k$ zu einer Festlegung bzgl. einer Basis $\vec{a}'_1, \dots, \vec{a}'_k$ nennt man **Basistransformation**.



Schmidtsches Orthogonalisierungsverfahren

Basisvektoren müssen nicht notwendigerweise orthogonal zueinander sein.

Ein Verfahren einen Satz von orthogonalen Basisvektoren zu erhalten ist das Schmidtsche Orthogonalisierungsverfahren.

Sei \vec{a}_i $i = 1, \dots, k$ ein Satz von Basisvektoren. Setze:

$$\begin{aligned}\vec{b}_1 &= \vec{a}_1 \\ \vec{b}_2 &= \vec{a}_2 - (\vec{a}_2 \cdot \vec{b}_1) \frac{\vec{b}_1}{|\vec{b}_1|^2} = \vec{a}_2 - (\vec{a}_2 \cdot \vec{e}_{\vec{b}_1}) \vec{e}_{\vec{b}_1} \\ \vec{b}_3 &= \vec{a}_3 - (\vec{a}_3 \cdot \vec{b}_1) \frac{\vec{b}_1}{b_1^2} - (\vec{a}_3 \cdot \vec{b}_2) \frac{\vec{b}_2}{b_2^2} = \vec{a}_3 - (\vec{a}_3 \cdot \vec{e}_{\vec{b}_1}) \vec{e}_{\vec{b}_1} - (\vec{a}_3 \cdot \vec{e}_{\vec{b}_2}) \vec{e}_{\vec{b}_2}\end{aligned}$$

allg.

$$\vec{b}_i = \vec{a}_i - \sum_{j=1}^{i-1} \frac{\vec{a}_i \cdot \vec{b}_j}{b_j^2} \vec{b}_j = \vec{a}_i - \sum_{j=1}^{i-1} (\vec{a}_i \cdot \vec{e}_{\vec{b}_j}) \vec{e}_{\vec{b}_j}$$

Beispiel:

$$\vec{a}_1 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \quad \vec{a}_2 = \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} \quad \vec{a}_3 = \begin{pmatrix} 3 \\ 1 \\ 0 \end{pmatrix}$$

$$\vec{b}_1 = \vec{a}_1 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}$$

$$\begin{aligned}\vec{b}_2 &= \vec{a}_2 - (\vec{a}_2 \cdot \vec{b}_1) \frac{\vec{b}_1}{|\vec{b}_1|^2} \\ &= \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} - 4 \cdot \frac{1}{6} \cdot \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \\ &= \begin{pmatrix} 4/3 \\ -2/3 \\ -1/3 \end{pmatrix}\end{aligned}$$

$$\begin{aligned}\vec{b}_3 &= \vec{a}_3 - (\vec{a}_3 \cdot \vec{b}_1) \frac{\vec{b}_1}{b_1^2} - (\vec{a}_3 \cdot \vec{b}_2) \frac{\vec{b}_2}{b_2^2} \\ &= \begin{pmatrix} 3 \\ 1 \\ 0 \end{pmatrix} - 4 \cdot \frac{1}{6} \cdot \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} - \frac{10}{3} \cdot \frac{3}{7} \cdot \begin{pmatrix} 4/3 \\ -2/3 \\ -1/3 \end{pmatrix} \\ &= \begin{pmatrix} 3/7 \\ 9/7 \\ -6/7 \end{pmatrix}\end{aligned}$$

Normierung (auf 1) führt zu einem Satz von **orthonormalen** Basisvektoren \vec{e}_i (orthogonal **und** normiert):

$$\begin{aligned}\vec{e}_1 &= \frac{\vec{a}_1}{a_1} \\ \vec{e}_i &= \frac{\vec{a}_i - \sum_{j=1}^{i-1} (\vec{a}_i \cdot \vec{e}_j) \vec{e}_j}{\left| \vec{a}_i - \sum_{j=1}^{i-1} (\vec{a}_i \cdot \vec{e}_j) \vec{e}_j \right|}\end{aligned}$$

2.2 Matrizen

Motivation

- Lösung linearer Gleichungssysteme
- Beschreibung von Basistransformationen

u.s.w.

2.2.1 Definitionen

D Ein Schema

$$A = (A_{ij})_{\substack{i=1\dots m \\ j=1\dots n}} = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mn} \end{pmatrix}$$

von $m \cdot n$ Elementen $A_{ij} \in \mathbb{R}$ heißt eine **Matrix** mit **m Zeilen** und **n Spalten**, kurz **$m \times n$ -Matrix**.

N

$$m = n$$

die Matrix heißt **quadratisch**

$$A_{i1} \quad A_{i2} \quad \dots \quad A_{in}$$

i -te Zeile

$$A_{1j}$$

$$A_{2j}$$

$$\vdots$$

$$A_{nj}$$

j -te Spalte

$$\left(A_{i1} \quad A_{i2} \quad \dots \quad A_{in} \right)$$

i -ter Zeilenvektor

$$\begin{pmatrix} A_{1j} \\ A_{2j} \\ \vdots \\ A_{nj} \end{pmatrix}$$

j -ter Spaltenvektor

Diagonalelemente

alle A_{ij} mit $i = j$

Nicht – oder Außer –

diagonalelemente

alle A_{ij} mit $i \neq j$

Diagonalmatrix

$$\begin{pmatrix} A_{11} & & & & \\ & A_{22} & & & \\ & & \ddots & & \\ & & & \ddots & \\ & 0 & & & A_{nn} \end{pmatrix}$$

d.h. $A_{ij} = 0$ für $i \neq j$

Einheitsmatrix

$$\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & \ddots & \\ 0 & & & & 1 \end{pmatrix}$$

d.h. $A_{ij} = 0$ für $i \neq j$
 $A_{ij} = 1$ für $i = j$

Nullmatrix

$$\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

d.h. $A_{ij} = 0$ für alle i, j

Obere Dreiecksmatrix

$$\begin{pmatrix} \text{hatched} & & & & \\ & \text{hatched} & & & \\ & & \text{hatched} & & \\ & & & \text{hatched} & \\ 0 & & & & \end{pmatrix}$$

d.h. $A_{ij} = 0$ $j < i$

Untere Dreiecksmatrix

$$\begin{pmatrix} \text{hatched} & & & & \\ \text{hatched} & & & & \\ \text{hatched} & & & & \\ \text{hatched} & & & & \\ \text{hatched} & & & & \\ & & & 0 & \\ & & & & \text{hatched} \end{pmatrix}$$

d.h. $A_{ij} = 0$ $j > i$

Bandmatrix

$$\begin{pmatrix} \text{hatched} & & & & \\ & \text{hatched} & & & \\ & & \text{hatched} & & \\ & & & \text{hatched} & \\ & & & & \text{hatched} \\ & & & & & 0 \\ & & & & & & \text{hatched} \\ & & & & & & & \text{hatched} \\ & & & & & & & & \text{hatched} \\ & & & & & & & & & 0 \end{pmatrix}$$

d.h. $A_{ij} = 0$ für $|i - j| > l$

Symmetrische Matrix

$A_{ij} = A_{ji}$

Antisymm. Matrix

$$A_{ij} = -A_{ji}$$

Transponierte Matrix
zur Matrix $A = (A_{ij})$

$$\begin{pmatrix} A_{11} & A_{21} & A_{31} & \dots & A_{m1} \\ & A_{22} & & & \\ \vdots & & A_{33} & & \vdots \\ & & & \ddots & \\ A_{1n} & \dots & \dots & \dots & A_{mn} \end{pmatrix} \quad \begin{array}{l} A = (A_{ij})_{\substack{i=1\dots m \\ j=1\dots n}} \\ A^T = (A_{ji})_{\substack{j=1\dots n \\ i=1\dots m}} \end{array}$$

2.2.2 Gleichheit, Addition, Subtraktion und Multiplikation mit einem Skalar

D Zwei $m \times n$ -Matrizen $A = (A_{ij})$ und $B = (B_{ij})$ ($A_{ij}, B_{ij} \in \mathbb{R}$) heißen **gleich** wenn für alle Elemente

$$A_{ij} = B_{ij} \quad i = 1, \dots, m; j = 1, \dots, n$$

gilt.

D Zwei $m \times n$ -Matrizen A und B werden **addiert/subtrahiert** indem ihre Elemente addiert/subtrahiert werden.

$$C = A \pm B$$

bzw. $(C_{ij}) = (A_{ij}) \pm (B_{ij})$ mit $C_{ij} = A_{ij} \pm B_{ij}$

S Die Addition von Matrizen ist kommutativ.

S Die Menge aller $m \times n$ -Matrizen $A = (A_{ij})$ mit $A_{ij} \in \mathbb{R}$ bildet bzgl. der Matrixaddition eine abelsche Gruppe.

D Eine Matrix A wird mit einem **Skalar** λ **multipliziert** indem jedes Element der Matrix mit dem Skalar multipliziert wird.

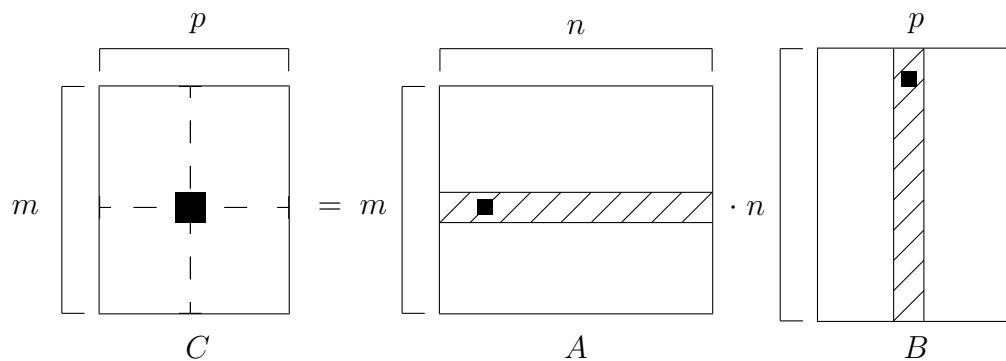
$$\lambda A = (\lambda A_{ij})$$

$$\text{d. h.} \quad \lambda \begin{pmatrix} A_{11} & \dots & A_{1n} \\ \vdots & & \vdots \\ A_{m1} & \dots & A_{mn} \end{pmatrix} = \begin{pmatrix} \lambda A_{11} & \dots & \lambda A_{1n} \\ \vdots & & \vdots \\ \lambda A_{m1} & \dots & \lambda A_{mn} \end{pmatrix}$$

2.2.3 Matrixmultiplikation

D Das **Produkt** einer $m \times n$ -Matrix A mit einer $n \times p$ -Matrix B ist definiert durch die $m \times p$ -Matrix C mit dem Elementen

$$C_{ij} = \sum_{k=1}^n A_{ik} B_{kj} \quad (i = 1, \dots, m; j = 1, \dots, p)$$



! Die Matrixmultiplikation ist im allg. nicht kommutativ

$$\text{d.h. i.allg. :} \quad AB \neq BA$$

Beispiel:

$$\begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 4 & 5 \\ 2 & 5 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 \\ 2 & 2 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix} = \begin{pmatrix} 3 & 1 \\ 8 & 6 \end{pmatrix}$$

D Eine Matrix B heißt **Links-** bzw. **Rechtsinverse** einer Matrix A falls gilt

$$BA = E \quad \text{bzw.} \quad AB = E$$

D Eine Matrix A heißt **regulär**, falls es eine Matrix B gibt, die gleichzeitig Links- und Rechtsinverse von A ist; sonst **singulär**.

! A und B müssen notwendigerweise quadratisch sein.

N

$$B = A^{-1}$$

Darstellung von Vektoren als Matrizen:

$$\vec{a} = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} = \underline{\underline{a}} \quad \vec{b} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} = \underline{\underline{b}}$$

Entsprechende Berechnung des Skalarprodukts:

$$\begin{aligned} \vec{a} \cdot \vec{b} &= \sum_{i=1}^n a_i b_i = \begin{pmatrix} a_1 & a_2 & \dots & a_n \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ a_n \end{pmatrix} \\ &= \underline{\underline{a}}^T \underline{\underline{b}} \end{aligned}$$

2.2.4 Elementare Umformungen

D Unter einer **elementaren Umformung** einer Matrix versteht man die Manipulation einer einzelnen Zeile oder Spalte dieser Matrix und die einfache Kombination solcher

Manipulationen.

! Im folgenden nur Zeilenumformungen – Spaltenumformungen laufen analog!

1. Multiplikation aller Elemente einer Zeile mit einem Skalar λ :

$$A = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1j} & \dots & A_{1m} \\ A_{21} & \dots & \dots & A_{2j} & \dots & A_{2m} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{i1} & A_{i2} & \dots & A_{ij} & \dots & A_{im} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{n1} & A_{n2} & \dots & A_{nj} & \dots & A_{nm} \end{pmatrix} \rightarrow A' = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1j} & \dots & A_{1m} \\ A_{21} & \dots & \dots & A_{2j} & \dots & A_{2m} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \lambda A_{i1} & \lambda A_{i2} & \dots & \lambda A_{ij} & \dots & \lambda A_{im} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{n1} & A_{n2} & \dots & A_{nj} & \dots & A_{nm} \end{pmatrix}$$

kompakter durch Zeilenvektoren \vec{a}_i ausgedrückt:

$$A = \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix} \rightarrow A' = \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \lambda \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix}$$

Dies läßt sich ausdrücken durch eine Matrixmultiplikation:

$$A' = U_1 A \quad \text{mit} \quad U_1 = \begin{pmatrix} 1 & 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & \dots & 0 & 0 \\ \vdots & & \ddots & \vdots & & & \vdots \\ 0 & 0 & \dots & \lambda & \dots & 0 & 0 \\ \vdots & & & \vdots & \ddots & & \vdots \\ 0 & 0 & \dots & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & 0 & \dots & 0 & 1 \end{pmatrix} \leftarrow i$$

CAS-Beispiel



Es wird zunächst eine beliebige 4×4 -Matrix A sowie eine 4×4 -Umformungsmatrix U_1 , die die dritte Zeile von A mit $\frac{1}{3}$ multiplizieren soll, erstellt (achten Sie darauf, daß die Indizes von Matrizen genau wie bei Listen von 0 beginnen). Die Multiplikation $U_1 A$ liefert das gewünschte Ergebnis:

```
import numpy as np

A = np.matrix([[1, 2, 2, 3],
               [2,-3, 5, 6],
               [3, 0, 2, 1],
```

```

                [2,-3, 4, 1]])

U1 = np.identity(4)
U1[2, 2] = 1/3
U1

```

```
np.dot(U1, A)
```

Alternative (hier müssen die Einträge der Matrix A als Gleitpunktkommazahlen definiert werden):

```

A = np.matrix([[1, 2, 2, 3],
               [2,-3, 5, 6],
               [3, 0, 2, 1],
               [2,-3, 4, 1]], dtype = float)

A[2,:] = A[2,:] * 1/3
A

```

Beachten Sie jedoch, daß dieser Aufruf im Gegensatz zum obigen Aufruf die ursprüngliche Matrix A überschreibt.

2. Addition der j -ten Zeile zur i -ten Zeile:

$$A = \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_i \\ \vec{a}_j \\ \vdots \\ \vec{a}_n \end{pmatrix} \rightarrow A' = \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_i + \vec{a}_j \\ \vec{a}_j \\ \vdots \\ \vec{a}_n \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1m} \\ \dots & \dots & \dots & \dots \\ A_{i1} + A_{j1} & A_{i2} + A_{j2} & \dots & A_{im} + A_{jm} \\ A_{j1} & A_{j2} & \dots & A_{jm} \\ \dots & \dots & \dots & \dots \\ A_{n1} & A_{n2} & \dots & A_{nm} \end{pmatrix}$$

Ausdrücken durch Matrixmultiplikation:

$$A' = U_2 A \quad \text{mit} \quad U_2 = \begin{pmatrix} 1 & 0 & & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & & & & & & \\ & & \ddots & \vdots & \dots & \vdots & & \\ 0 & 0 & & 1 & \dots & 1 & & \\ & & & & \ddots & & & \\ 0 & 0 & & & & 1 & & \\ \vdots & \vdots & & & & & & \\ 0 & 0 & & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & & 0 & \dots & 0 & 0 & 1 \end{pmatrix} \leftarrow i$$

\uparrow
 j

CAS-Beispiel



Die folgende 4×4 -Umformungsmatrix U_2 soll Zeile 1 zu Zeile 3 addieren. Die Multiplikation $U_2 A$ liefert das Ergebnis:

```
import numpy as np

A = np.matrix([[1, 2, 2, 3],
               [2,-3, 5, 6],
               [3, 0, 2, 1],
               [2,-3, 4, 1]])

U2 = np.identity(4)
U2[2, 0] = 1
U2
```

```
np.dot(U2, A)
```

Alternative:

```
A[2,:] = A[2,:] + A[0,:]
A
```

3. Addition einer mit einem Skalar $\lambda \in \mathbb{R} \setminus \{0\}$ multiplizierten Zeile zu einer anderen Zeile:

$$\begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_i \\ \vec{a}_j \\ \vdots \\ \vec{a}_n \end{pmatrix} \xrightarrow{1} \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \lambda \vec{a}_i \\ \vec{a}_j \\ \vdots \\ \vec{a}_n \end{pmatrix} \xrightarrow{2} \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \lambda \vec{a}_i \\ \vec{a}_j + \lambda \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix} \xrightarrow{1} \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \frac{1}{\lambda} \lambda \vec{a}_i \\ \vec{a}_j + \lambda \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix} = \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_i \\ \vec{a}_j + \lambda \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix}$$

$$A' = U_3 A \quad \text{mit} \quad U_3 = U_1' U_2 U_1$$

alternative Möglichkeit den i -ten zum j -ten Zeilenvektor zu addieren:

$$\frac{1}{\lambda} \vec{a}_j \rightarrow \frac{1}{\lambda} \vec{a}_j + \vec{a}_i \rightarrow \lambda \left(\frac{1}{\lambda} \vec{a}_j + \vec{a}_i \right) = \vec{a}_j + \lambda \vec{a}_i$$

CAS-Beispiel



Die folgende 4×4 -Umformungsmatrix U_3 soll das -2 -fache der Zeile 1 zu Zeile 4 addieren. Die Multiplikation $U_3 A$ liefert das Ergebnis:

```
import numpy as np

A = np.matrix([[1, 2, 2, 3],
               [2,-3, 5, 6],
               [3, 0, 2, 1],
               [2,-3, 4, 1]])

U3 = np.identity(4)
U3[3, 0] = -2
U3
```

```
np.dot(U3, A)
```

Alternative:

```
A[3,:] = A[3,:] + (-2*A[0,:])
A
```

4. Vertauschen zweier Zeilen:

$$\begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_i \\ \vec{a}_j \\ \vdots \\ \vec{a}_n \end{pmatrix} \xrightarrow{2} \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_i \\ \vec{a}_j + \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix} \xrightarrow[\lambda=-1]{1} \begin{pmatrix} \vec{a}_1 \\ \vdots \\ -\vec{a}_i \\ \vec{a}_j + \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix} \xrightarrow{2} \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_j \\ \vec{a}_j + \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix} \xrightarrow[\lambda=-1]{3} \begin{pmatrix} \vec{a}_1 \\ \vdots \\ \vec{a}_j \\ \vec{a}_i \\ \vdots \\ \vec{a}_n \end{pmatrix}$$

Als Kombination von elementaren Umformungen ausgedrückt:

$$A' = U_4 A \quad \text{mit} \quad U_4 = U_3 U_2' U_1 U_2$$

CAS-Beispiel



Die folgende 4×4 -Umformungsmatrix U_4 soll die Zeilen 2 und 4 vertauschen. Die Multiplikation $U_4 A$ liefert das Ergebnis:

```
import numpy as np

A = np.matrix([[1, 2, 2, 3],
               [2,-3, 5, 6],
               [3, 0, 2, 1],
               [2,-3, 4, 1]])

U4 = np.identity(4)
U4[3, 3] = 0
U4[1, 1] = 0
U4[1, 3] = 1
U4[3, 1] = 1
U4
```

```
np.dot(U4, A)
```

Alternative:

```
A[[1, 3]] = A[[3, 1]]
A
```

Im Folgenden wird gezeigt, daß sich U_4 aus den anderen U-Matrizen erhalten läßt. Da zwei U_2 -Matrizen auftauchen, wird zusätzlich die Matrix U_{2a} erstellt.

```
A = np.matrix([[1, 2, 2, 3],
               [2,-3, 5, 6],
               [3, 0, 2, 1],
               [2,-3, 4, 1]])

U2a = np.identity(4)
U2a[3, 1] = 1
U1 = np.identity(4)
U1[1, 1] = -1
U2 = np.identity(4)
U2[1, 3] = 1
U3 = np.identity(4)
U3[3, 1] = -1
```

```
U3
```

```
U2
```

```
U1
```

```
U2a
```

```
np.dot(U3, np.dot(U2, np.dot(U1, np.dot(U2a, A))))
```

Das Ergebnis der Multiplikation $U_3 U_2 U_1 U_{2a}$ stimmt mit U_4 überein.

S Mittels elementarer Umformungen läßt sich jede quadratische Matrix auf die Form

$$\begin{pmatrix} 1 & 0 & 0 & & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & & & & & & \\ 0 & 0 & 1 & & & & & & \\ & & & \ddots & & & & & \\ 0 & & & & 1 & & & & 0 \\ 0 & & & & & 1 & & & 0 \\ 0 & & & & & & 0 & & 0 \\ 0 & & & & & & & 0 & 0 \\ 0 & & & & & & & & \ddots & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 & 0 & & 0 \end{pmatrix}$$

bringen.

B

- 1) A_{11} durch Vertauschen von Zeilen (Spalten) zu einem Element $\neq 0$ überführen.
- 2) Durch Multiplikation der 1. Zeile mit $\frac{1}{A_{11}}$ erreicht man, daß $A_{11} = 1$ wird.
- 3) 1. Spalte und 1. Zeile bis auf A_{11} auf 0 bringen.
- 4) Behandeln der Restmatrix wie zuvor.
- 5) Vollständige Induktion.

2.2.5 Rang einer Matrix

D

Die maximale Anzahl linear unabhängiger Zeilen-/Spaltenvektoren einer Matrix heißt der **Zeilen-/Spaltenrang** dieser Matrix.

S

Eine elementare Umformung ändert den Zeilen-/Spaltenrang einer Matrix nicht.

B

zu Umformung vom Typ 1: Zeilenrang r , \vec{a}_i Zeilenvektoren, $r \leq n$

vorher:

$$\sum_{i=1}^r \lambda_i \vec{a}_i \stackrel{!}{=} 0 \Rightarrow \lambda_i = 0 \quad \forall i = 1, \dots, r$$

nachher:

$$\vec{a}_i' = \begin{cases} \lambda \vec{a}_j & \text{für } i = j \\ \vec{a}_i & \text{sonst} \end{cases}$$

$$\sum_{i=1}^r \lambda_i' \vec{a}_i' \stackrel{!}{=} 0 \Rightarrow \sum_{i=1}^r \lambda_i'' \vec{a}_i = 0$$

$$\lambda_i'' = \lambda_i' \quad \forall i \neq j$$

$$\lambda_j'' = \lambda_j' \lambda \quad \text{für } i = j \Rightarrow \lambda_i'' = 0 \quad \forall i = 1, \dots, r$$

$$\Rightarrow \lambda_i' = 0 \quad \forall i = 1, \dots, r$$

S Der Zeilenrang einer Matrix ist gleich ihrem Spaltenrang und wird kurz mit **Rang** bezeichnet.

B Der letzte Satz zusammen mit der Tatsache, daß jede Matrix auf die Form

$$\begin{array}{c} \text{Zeilenrang} \\ \left[\begin{array}{cccccccc} 1 & 0 & 0 & & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & & & & & & \\ 0 & 0 & 1 & & & & & & \\ & & & \ddots & & & & & \\ 0 & & & & 1 & & & & 0 \\ 0 & & & & & 1 & & & 0 \\ 0 & & & & & & 0 & & 0 \\ 0 & & & & & & & 0 & 0 \\ 0 & & & & & & & & \ddots & 0 \\ 0 & 0 & 0 & & 0 & 0 & 0 & 0 & & 0 \end{array} \right] \end{array}$$

Spaltenrang

gebracht werden kann.

CAS-Beispiel



Im Folgenden wird durch mehrere elementare Umformungen die 3×3 -Matrix A auf die Form einer Einheitsmatrix gebracht. Die Prozeduren $UMAT1 \dots UMAT4$, die von `mathchem` zur Verfügung gestellt werden, entsprechen dabei den Umformungsmatrizen $U_1 \dots U_4$ der vorherigen Beispiele mit den angegebenen Parametern.

```
import numpy as np

A = np.matrix([[1, 2, 3],
               [2, -3, 6],
               [3, 0, 1]])
```

`_` steht für das letzte Ergebnis.

```
from mathchem import UMAT1, UMAT3

np.dot(UMAT3(3, 3, 1, -3), A)
```

```
np.dot(UMAT3(3, 2, 1, -2), _)
```

```
np.dot(UMAT1(3, 2, -1/7) , _)
```

```
np.dot(UMAT3(3, 3, 2, 6) , _)
```

```
np.dot(UMAT1(3, 3, -1/8) , _)
```

```
np.dot(UMAT3(3, 1, 2, -2), _)
```

```
np.dot(UMAT3(3, 1, 3, -3), _)
```

Man kann auch die einzelne Operationen in einem Schritt rechnen lassen:

```
A = np.matrix([[1, 2, 3],
               [2,-3, 6],
               [3, 0, 1]])

umat = np.dot(UMAT3(3, 1, 3, -3),
              np.dot(UMAT3(3, 1, 2, -2),
                    np.dot(UMAT1(3, 3, -1/8 ),
                          np.dot(UMAT3(3, 3, 2, 6),
                                np.dot(UMAT1(3, 2, -1/7 ),
                                      np.dot(UMAT3(3, 2, 1, -2),
                                            UMAT3(3, 3, 1, -3)))))))

umat*A
```

Man kommt in diesem Beispiel nur mit Zeilenumformungen aus. Warum?

Versuchen Sie, die Matrix B umzuformen:

```
B = np.matrix([[3 , 4, 5],
               [1/8, 6, 12],
               [0 , 2, 4]], dtype = float)
```

! Die Zeilen-/Spaltenvektoren einer Matrix spannen einen Vektorraum der Dimension r auf.

D (alternativ) Eine quadratische $n \times n$ -Matrix deren Rang $r = n$ ist, heißt **regulär**, sonst **singulär**.

2.2.6 Inverse von quadratischen Matrizen

S Ist der Rang r einer $n \times n$ -Matrix A gleich der Dimension n dieser Matrix, so existiert ihre **inverse Matrix** A^{-1} , die gleichzeitig Links- und Rechtsinverse ist, mit der Eigenschaft:

$$AA^{-1} = E = A^{-1}A$$

B Invertierbarkeit von Matrizen.

Ist $r = n$, so gibt es eine Reihe von k elementaren Umformungen U_i ($i = 1, \dots, k$) mit

$$\left(\prod_{i=1}^k U_i \right) A = E \quad \Rightarrow \quad \prod_{i=1}^k U_i = A^{-1}$$

Die rechtsinverse Matrix A_r^{-1} ist gleich der linksinversen Matrix A_l^{-1} :

$$\begin{aligned} \text{Es gilt: } A_l^{-1} A &= E \quad \text{und} \quad A A_r^{-1} = E \\ A A_r^{-1} A &= E A = A \\ \underbrace{A_l^{-1} A}_{E} A_r^{-1} A &= \underbrace{A_l^{-1} A}_{E} \\ E A_r^{-1} A &= E \\ A_r^{-1} A &= E \Rightarrow A_r^{-1} = A_l^{-1} \end{aligned}$$

Das Gauß-Jordan Verfahren zur Matrixinversion:

Forme eine $n \times n$ -Matrix A , deren Rang gleich ihrer Dimension ist, mittels elementarer Umformungen auf die Einheitsmatrix E um. Wende die gleichen Umformungen auf die Einheitsmatrix E an: es entsteht dabei die inverse Matrix A^{-1} .

CAS-Beispiel



Im Folgenden wird die Matrix A schrittweise in eine Einheitsmatrix überführt. Die selben Umformungen werden auf die Einheitsmatrix E angewendet.

```
import numpy as np

A = np.matrix([[1, 2, 3],
               [2,-3, 6],
               [3, 0, 1]])

E = np.identity(3)
```

```
from mathchem import UMAT1, UMAT3

np.dot(UMAT3(3, 3, 1,-3), A)
```

```
np.dot(UMAT3(3, 2, 1,-2), _)
```

```
np.dot(UMAT1(3, 2, -1/7), _)
```

```
np.dot(UMAT3(3, 3, 2, 6), _)
```

```
np.dot(UMAT1(3, 3, -1/8), _)
```

```
np.dot(UMAT3(3, 1, 2,-2), _)
```

```
np.dot(UMAT3(3, 1, 3,-3), _)
```

```
np.dot(UMAT3(3, 3, 1,-3), E)
```

```
np.dot(UMAT3(3, 2, 1,-2), _)
```

```
np.dot(UMAT1(3, 2, -1/7), _)
```

```
np.dot(UMAT3(3, 3, 2, 6), _)
```

```
np.dot(UMAT1(3, 3, -1/8), _)
```

```
np.dot(UMAT3(3, 1, 2,-2), _)
```



```
np.dot(UMAT3(3, 1, 3,-3), _)
```

Das Ergebnis sollte am Ende der inversen Matrix A^{-1} entsprechen.
Überprüfen Sie dies.

```
A.getI()
```

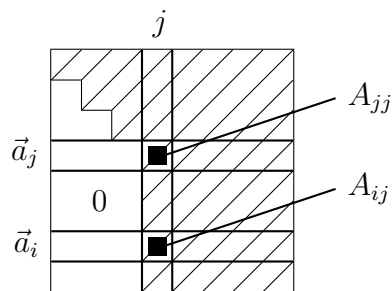
Das Gauß Verfahren:

Ausgangspunkt: $AA^{-1} = E$

1) Bringe A durch elementare Umformungen auf obere Dreiecksform A' :

$$\underbrace{\prod_i U_i A}_{A'} A^{-1} = \underbrace{\prod_i U_i E}_{E'} A^{-1}$$

$$A' A^{-1} = E'$$



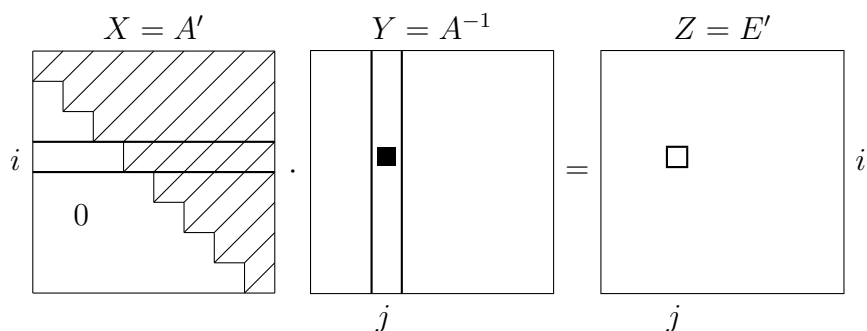
— Für jede Spalte $j = 1, \dots, n$
 — Für jede Zeile $i = (j + 1), \dots, n$

$$\vec{a}_i \rightarrow \vec{a}_i - \vec{a}_j A_{ij} / A_{jj}$$

$$\vec{e}_i \rightarrow \vec{e}_i - \vec{e}_j E_{ij} / E_{jj}$$

$\vec{a}_i, \vec{a}_j, \vec{e}_i, \vec{e}_j$: Zeilenvektoren

2) Löse das Gleichungssystem $A' A^{-1} = E'$ für A^{-1} :



Für jede Spalte $j = 1, \dots, n$

Für jede Zeile $i = n, \dots, 1$

$$\sum_{k=1}^n X_{ik} Y_{kj} = Z_{ij}$$

$$X_{ii} Y_{ij} + \sum_{k=i+1}^n X_{ik} Y_{kj} = Z_{ij}$$

$$Y_{ij} = (Z_{ij} - \sum_{k=i+1}^n X_{ik} Y_{kj}) / X_{ii}$$

Beispiel:

- 1) Forme A in eine obere Dreiecksmatrix um. Wende dieselben elementaren Umformungen auf E an.

$$\begin{array}{cc} \overbrace{\begin{pmatrix} 1 & 2 & 0 \\ 2 & 0 & 2 \\ 1 & 1 & 1 \end{pmatrix}}^A & \overbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}^E \\ \begin{pmatrix} 1 & 2 & 0 \\ 0 & -4 & 2 \\ 1 & 1 & 1 \end{pmatrix} & \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ \begin{pmatrix} 1 & 2 & 0 \\ 0 & -4 & 2 \\ 0 & -1 & 1 \end{pmatrix} & \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \\ \underbrace{\begin{pmatrix} 1 & 2 & 0 \\ 0 & -4 & 2 \\ 0 & 0 & \frac{1}{2} \end{pmatrix}}_{A'} & \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -\frac{1}{2} & -\frac{1}{4} & 1 \end{pmatrix}}_{E'} \end{array}$$

- 2) Löse die Gleichung

$$\begin{pmatrix} 1 & 2 & 0 \\ 0 & -4 & 2 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -\frac{1}{2} & -\frac{1}{4} & 1 \end{pmatrix}$$

Berechne die Elemente der 1. Spalte von A^{-1} :

- a) $0a + 0d + \frac{1}{2}g = -\frac{1}{2} \Rightarrow g = -1$
 b) $0a - 4d - 2 = -2 \Rightarrow d = 0$
 c) $1a + 0 - 1 = 1 \Rightarrow a = 1$

entsprechend für 2. und 3. Spalte!

$$\begin{aligned} \Rightarrow & \begin{pmatrix} 1 & 2 & 0 \\ 0 & -4 & 2 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} 1 & 1 & -2 \\ 0 & -\frac{1}{2} & 1 \\ -1 & -\frac{1}{2} & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -\frac{1}{2} & -\frac{1}{4} & 1 \end{pmatrix} \\ \Rightarrow & \begin{pmatrix} 1 & 2 & 0 \\ 2 & 0 & 2 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & -2 \\ 0 & -\frac{1}{2} & 1 \\ -1 & -\frac{1}{2} & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

S Rechenregeln

$$\begin{aligned} (AB)^{-1} &= B^{-1}A^{-1} \\ (AB)^T &= B^T A^T \\ (A^T)^{-1} &= (A^{-1})^T \end{aligned}$$

D Eine reelle $n \times n$ -Matrix A heißt **orthogonal**, wenn sie der Bedingung $A^T = A^{-1}$ genügt.

Für orthogonale Matrizen gilt:

$$\begin{aligned} A^T A &= \underbrace{A^{-1} A}_{= E} \\ \sum_{k=1}^n A_{ik}^{-1} A_{kj} &= E_{ij} \\ \sum_{k=1}^n A_{ki} A_{kj} &= \delta_{ij} \quad \leftarrow \text{Kroneckersymbol} \end{aligned}$$

D.h. Spalten- und Zeilenvektoren orthogonaler Matrizen sind orthogonal und normiert, d.h. **orthonormiert**.

CAS-Beispiel



A ist eine orthogonale Matrix:

```
import numpy as np

A = np.matrix([[1/3,           ,           2/3,           2/3],
               [2/np.sqrt(5), ,           0, -1/np.sqrt(5)],
               [2/np.sqrt(45), -5/np.sqrt(45), 4/np.sqrt(45)],
               dtype = float])
```

Zur Überprüfung der Orthogonalität werden zunächst die drei Spaltenvektoren aus A nach a_1 , a_2 und a_3 kopiert, wobei die Prozedur $A[:, n]$ verwendet wird:

```
a1 = A[:,0]
a2 = A[:,1]
a3 = A[:,2]
```

Durch Bilden des Skalarprodukts wird nun überprüft, ob die Vektoren normiert sind. Begründen Sie die Vorgehensweise und kommentieren Sie das Ergebnis:

```
a1.transpose()*a1
```

```
a2.transpose()*a2
```

```
a3.transpose()*a3
```

Als nächstes wird überprüft, ob die Vektoren gegenseitig orthogonal sind:

```
a1.transpose()*a2
```

```
a1.transpose()*a3
```

```
a2.transpose()*a3
```

Überprüfen Sie, ob die Zeilenvektoren b_1 , b_2 und b_3 , die sie mit $A[n, :]$ erhalten, ebenfalls orthonormiert sind. $B = A^T$ ist die Transponierte von A . Wenn A orthogonal ist, muß auch gelten: $B = A^{-1}$.

Was erwarten Sie dann für das Produkt BA ?

```
B = A.transpose()
np.dot(B, A)
```

```
B-A.getI()
```

Versuchen Sie, andere orthogonale Matrizen zu erstellen.

D Gilt für eine komplexe Matrix $A^{-1} = A^\dagger$, so wird sie als **unitär** bezeichnet. Hierbei ist $A^\dagger = A^{T*}$ die komplex transponierte Matrix zur komplexen Matrix A .

2.3 Determinanten

2.3.1 Definitionen

D Unter einer **Permutation** versteht man eine beliebige Veränderung der Anordnung der Elemente eines n -Tupels.

Beispiel:

$$P = \begin{pmatrix} a & b & c \\ b & c & a \end{pmatrix}, \quad P' = \begin{pmatrix} \alpha & \beta & \gamma & \delta \\ \delta & \gamma & \alpha & \beta \end{pmatrix}$$

D Eine Permutation, bei der nur zwei Elemente vertauscht werden nennt man **Transposition**.

Beispiel:

$$T_{23} = \begin{pmatrix} a & b & c \\ a & c & b \end{pmatrix}$$

S Jede Permutation P lässt sich in eine endliche Anzahl von Transpositionen zerlegen.

Beispiel:

$$P = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix} \quad 1. T_{13} = \begin{pmatrix} a & b & c \\ c & b & a \end{pmatrix} \quad 2. T_{23} = \begin{pmatrix} a & b & c \\ a & c & b \end{pmatrix}$$

$$\Rightarrow P = T_{23} \circ T_{13}$$

! Es kommt auf die Reihenfolge an, d.h. die Verknüpfung ist nicht kommutativ!

S Jede Permutation P lässt sich entweder in eine gerade oder ungerade Zahl von Transpositionen zerlegen. Die Funktion **Signum**(P) gibt an, ob eine gerade oder ungerade

Anzahl von Transpositionen vorliegt:

$$\text{Signum}(P) = \begin{cases} +1 \\ -1 \end{cases} \text{ für } \begin{cases} \text{gerade} \\ \text{ungerade} \end{cases} \text{ Anzahl von Transpositionen}$$

D Jeder $n \times n$ -Matrix A wird durch die Vorschrift

$$D = \sum_P \text{Signum}(P) A_{1p_1} A_{2p_2} A_{3p_3} \dots A_{np_n}$$

in eindeutiger Weise eine Zahl D zugeordnet, die als **Determinante n -ter Ordnung** bezeichnet wird. Hierbei wird über alle $n!$ Permutationen $P = \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ p_1 & p_2 & p_3 & \dots & p_n \end{pmatrix}$ der Spaltenindizes i summiert.

N

$$D = \det A = |A| = \begin{vmatrix} A_{11} & \dots & A_{1n} \\ \vdots & & \vdots \\ A_{n1} & \dots & A_{nn} \end{vmatrix}$$

Beispiel:

$n=2$:

$$\det A = \begin{vmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{vmatrix} \begin{matrix} 1 & 2 \\ \text{Signum}(P) \end{matrix}$$

$$\det A = \begin{matrix} +1(A_{11}A_{22}) & 1 & 2 & + \\ -1(A_{12}A_{21}) & 2 & 1 & - \end{matrix}$$

$n=3$:

$$\det A = \begin{vmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{vmatrix} \begin{matrix} 1 & 2 & 3 \\ \text{Signum}(P) \end{matrix}$$

$$\det A = \begin{matrix} +1(A_{11}A_{22}A_{33}) & 1 & 2 & 3 & + \\ -1(A_{11}A_{23}A_{32}) & 1 & 3 & 2 & - \\ -1(A_{12}A_{21}A_{33}) & 2 & 1 & 3 & - \\ +1(A_{12}A_{23}A_{31}) & 2 & 3 & 1 & + \\ -1(A_{13}A_{22}A_{31}) & 3 & 2 & 1 & - \\ +1(A_{13}A_{21}A_{32}) & 3 & 1 & 2 & + \end{matrix}$$

Regel von Sarrus:

Die Determinante einer 3×3 -Matrix lässt sich mit Hilfe der Regel von Sarrus leicht berechnen. Man schreibt den 1. und 2. Spaltenvektor rechts neben die Matrix. Die Produkte aller Elemente der Diagonalen von links oben nach rechts unten werden addiert, diejenigen Produkte der Elemente der Diagonalen von rechts oben nach links unten subtrahiert.

$$\begin{array}{ccccccc}
 & & + & & + & & + & & - & & - & & - \\
 & & \diagdown & & \diagdown & & \diagdown & & \diagup & & \diagup & & \diagup \\
 & & A_{11} & A_{12} & A_{13} & | & A_{11} & A_{12} & & & & & \\
 & & \diagdown & \diagup & \diagdown & \diagup & \diagdown & \diagup & & & & & \\
 & & A_{21} & A_{22} & A_{23} & | & A_{21} & A_{22} & & & & & \\
 & & \diagdown & \diagup & \diagdown & \diagup & \diagdown & \diagup & & & & & \\
 & & A_{31} & A_{32} & A_{33} & | & A_{31} & A_{32} & & & & & \\
 \\
 & & = & A_{11}A_{22}A_{33} & + & A_{12}A_{23}A_{31} & + & A_{13}A_{21}A_{32} \\
 & & - & A_{13}A_{22}A_{31} & - & A_{11}A_{23}A_{32} & - & A_{12}A_{21}A_{33}
 \end{array}$$

CAS-Beispiel

Die Determinante einer Matrix erhält man mittels `numpy.linalg.det`:

```
import numpy as np

A = np.matrix([[1, 2, 3],
               [2,-3, 6],
               [3, 0, 1]])

np.linalg.det(A)
```

```
B = np.matrix([[1, 2, 3],
               [1,-3, 6],
               [2, 4, 6]])

np.linalg.det(B)
```

```
C = np.matrix([[1, 2, 2, 3],
               [2,-3, 4, 6],
               [3, 0, 2, 1],
               [2,-3, 4, 1]])

np.linalg.det(C)
```

D Als **Unterdeterminante** D_{ij} ($n-1$)-ter Ordnung bezeichnet man eine Determinante, die durch Streichen der i -ten Zeile und der j -ten Spalte einer $n \times n$ -Matrix entsteht.

$K_{ij} = (-1)^{i+j} D_{ij}$ wird als **algebraisches Komplement** bezeichnet.

Beispiel:

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} D_{23} = \begin{vmatrix} A_{11} & A_{12} \\ A_{31} & A_{32} \end{vmatrix} = A_{11}A_{32} - A_{31}A_{12}$$

$$K_{23} = (-1)^{2+3} D_{23} = -A_{11}A_{32} + A_{31}A_{12}$$

2.3.2 Eigenschaften von Determinanten

S Die Determinante einer oberen/unteren Dreiecksmatrix A ist:

$$D = \prod_{i=1}^n A_{ii}$$

Spezialfälle:

Matrizen, die elementare Umformungen repräsentieren:

- Einheitsmatrix E : $\det E = 1$
- U_1 : $\det U_1 = \lambda$
- U_2 : $\det U_2 = 1$
- U_3 : $\det U_3 = 1$

S Für Umformungsmatrizen U_4 vom Typ 4 gilt:

$$\det U_4 = -1$$

CAS-Beispiel



Überzeugen Sie sich von den Aussagen über die Determinanten von Umformungsmatrizen.


```
import numpy as np

U1 = np.identity(4)
U1[2, 2] = 1/3
U1
```

```
np.linalg.det(U1)
```

```
U2 = np.identity(4)
U2[2, 0] = 1
U2
```

```
np.linalg.det(U2)
```

```
U3 = np.identity(4)
U3[3, 0] = -2
U3
```

```
np.linalg.det(U3)
```

```
U4 = np.identity(4)
U4[3, 3] = 0
U4[1, 1] = 0
U4[1, 3] = 1
U4[3, 1] = 1
U4
```

```
np.linalg.det(U4)
```

S Die Determinante eines Produktes von Matrizen M_1 und M_2 ist gleich dem Produkt der Determinanten von M_1 und M_2 .

$$\det(M_1 M_2) = \det(M_1) \det(M_2)$$

CAS-Beispiel



Es werden zunächst zwei beliebige Matrizen A und B erzeugt:

```
import numpy as np

A = np.matrix([[1, 2, 3],
               [2, -3, 1],
```

```

        [3, 0, 1]])
B = np.matrix([[1, 0, 2],
               [4,-1, 2],
               [0, 2, 1]])

```

Im nächsten Schritt werden die Determinanten von A und B berechnet und multipliziert:

```

detA = np.linalg.det(A)
detB = np.linalg.det(B)
detA*detB

```

Die Determinante des Produkts AB liefert folgendes Ergebnis:

```

np.linalg.det(np.dot(A, B))

```

S *Multipliziert man eine Zeile/Spalte einer Matrix A mit einem Skalar λ wobei die Matrix A' entsteht, so wird $\det A$ mit λ multipliziert; d.h. $\det A' = \lambda \det A$.*

B

$$\begin{aligned}
 A' &= U_1 A \\
 \det A' &= \det(U_1 A) = \det(U_1) \det(A) = \lambda \det A
 \end{aligned}$$

S *Es gilt:*

$$\det(\lambda M) = \lambda^n \det M$$

B siehe oben

S *Addiert man zu einer Zeile/Spalte einer Matrix A eine Linearkombination der übrigen, wobei die Matrix A' entsteht, so ändert sich die Determinante nicht, d.h. es gilt*

$\det A' = \det A$.

B

$$\begin{aligned} A' &= \prod_i U_3^i A \\ \det A' &= \det \left(\prod_i U_3^i \right) \det A \\ &= \left(\prod_i (\det U_3^i) \right) \det A \\ &= \left(\prod_i 1 \right) \det A = \det A \end{aligned}$$

S Vertauscht man in einer Matrix A zwei Zeilen/Spalten, wobei die Matrix A' entsteht, so ändert sich das Vorzeichen der Determinante, d.h. es gilt $\det A' = -\det A$.

B

$$A' = U_4 A$$

U_4 enthält U_1 mit $\lambda = -1 \Rightarrow$

$$\det U_4 = -1 \quad \text{und} \quad \det A' = -\det A$$

S Es gilt:

$$\det A = \det A^T$$

B Folgt direkt aus der Definition.

Beispiel:

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \rightarrow A^T = \begin{pmatrix} A_{11} & A_{21} \\ A_{12} & A_{22} \end{pmatrix}$$

$$\det A = A_{11}A_{22} - A_{12}A_{21} \stackrel{!}{=} A_{11}A_{22} - A_{21}A_{12} = \det A^T$$

S Es gilt:

$$\det(A^{-1}) = (\det A)^{-1}$$

B

$$A A^{-1} = E$$

$$\Rightarrow \det(A A^{-1}) = \det E$$

$$\det(A) \det(A^{-1}) = 1$$

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

CAS-Beispiel



Es wird zunächst eine Matrix A aufgestellt und dann ihre Determinante berechnet. Im letzten Schritt wird die Determinante von A^{-1} , der Inversen von A , berechnet.

Was fällt Ihnen auf?

```
import numpy as np

A = np.matrix([[1, 2, 3],
               [2, -3, 1],
               [3, 0, 1]])
```

```
np.linalg.det(A)
```

```
AINV = A.getI()
np.linalg.det(AINV)
```

```
1/np.linalg.det(A)
```

S Für orthogonale Matrizen gilt:

$$\det A = \pm 1$$

B

$$\begin{aligned} E &= A A^{-1} \\ \det E = 1 &= \det(A A^{-1}) \\ &= \det(A) \det A^{-1} \\ &= \det(A) \det A^T \\ &= (\det A)^2 \\ \Rightarrow \det A &= \pm 1 \end{aligned}$$

CAS-Beispiel



A ist eine orthogonale Matrix (Überzeugen Sie sich davon).
Welches Ergebnis erhält man für die Determinante ?

```
import numpy as np

A = np.matrix([[1/3      ,      2/3,      2/3],
               [2/np.sqrt(5) ,      0, -1/np.sqrt(5)],
               [2/np.sqrt(45), -5/np.sqrt(45), 4/np.sqrt(45)]],
              dtype = float)
```

```
np.linalg.det(A)
```

S Sind die Zeilen-/Spaltenvektoren einer Matrix A linear abhängig, so gilt $\det A = 0$. (Linear abhängig heißt, daß mindestens ein Zeilen-/Spaltenvektor durch die anderen darstellbar ist.) In diesem Fall ist der Rang r der Matrix kleiner als ihre Dimension n : $r < n$.

B Durch k elementare Umformungen bringt man die Matrix auf folgende Form:

$$A' = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\Rightarrow \det A' = \underbrace{\lambda_k \lambda_{k-1} \dots \lambda_1}_{\substack{\text{Vorfaktoren der} \\ k \text{ elementaren} \\ \text{Umformungen}}} \det A = 0$$

$$\Rightarrow \det A' = 0$$

Anmerkung: Bei linear unabhängigen Zeilen-/Spaltenvektoren stünden in der gesamten Diagonalen Einsen.

Also gilt:

$$\begin{aligned} \det A = 0 &\Leftrightarrow \text{Rang}(A) = r < n \Leftrightarrow A \text{ ist singular} \Leftrightarrow A^{-1} \text{ existiert nicht} \\ \det A \neq 0 &\Leftrightarrow \text{Rang}(A) = r = n \Leftrightarrow A \text{ ist regulär} \Leftrightarrow A^{-1} \text{ existiert} \end{aligned}$$

CAS-Beispiel



A ist eine 3×3 -Matrix, deren Determinante und inverse Matrix A^{-1} berechnet werden. Was fällt Ihnen auf?

```
import numpy as np

A = np.matrix([[1, 2, 3],
               [2, -3, 1],
               [3, 0, 1]])

np.linalg.det(A)
```

```
A.getI()
```

B ist eine 3×3 -Matrix, deren Determinante und inverse Matrix B^{-1} berechnet werden. Was fällt Ihnen auf?

```
B = np.matrix([[1, 1, 2],
               [4,-1, 2],
               [5,-1, 4]])

np.linalg.det(B)
```

```
B.getI()
```

2.3.3 Berechnung einer Determinanten

- entsprechend der Definition
- für 3×3 -Matrizen: Regel von Sarrus
- Umformung in eine Dreiecksmatrix

$$M \rightarrow M' \quad (M' \text{ ist obere Dreiecksmatrix})$$

$$M' = U_1 \dots U_n M$$

U_i : elementare Umformung

$$\Rightarrow \det M' = \prod_{i=1}^n M'_{ii}$$

$$\Rightarrow \det M = (-1)^k \det M'$$

k : Zahl der Zeilenvertauschungen

- Laplacescher Entwicklungssatz

S **Laplacescher Entwicklungssatz:** Ist A eine $n \times n$ -Matrix, D_{ij} eine Unterdeterminante von A und $K_{ij} = (-1)^{i+j} D_{ij}$ das zugehörige algebraische Komplement, so gilt

$$\begin{aligned} \det A &= \sum_{k=1}^n A_{ik} K_{ik} \\ &= \sum_{k=1}^n (-1)^{i+k} A_{ik} D_{ik} \end{aligned}$$

(Entwicklung nach der i -ten Zeile), bzw.

$$\begin{aligned}\det A &= \sum_{k=1}^n A_{kj} K_{kj} \\ &= \sum_{k=1}^n (-1)^{j+k} A_{kj} D_{kj}\end{aligned}$$

(Entwicklung nach der j -ten Spalte)

Beispiel:

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix}$$

Entwicklung nach der ersten Zeile: ($i = 1$)

$$\begin{aligned}D = \det A &= \sum_{k=1}^3 (-1)^{1+k} A_{1k} D_{1k} \\ &= (-1)^{1+1}(A_{11}D_{11}) + (-1)^{1+2}(A_{12}D_{12}) + (-1)^{1+3}(A_{13}D_{13}) \\ &= +1A_{11}(A_{22}A_{33} - A_{23}A_{32}) \\ &\quad -1A_{12}(A_{21}A_{33} - A_{23}A_{31}) \\ &\quad +1A_{13}(A_{21}A_{32} - A_{22}A_{31})\end{aligned}$$

2.4 Lineare Gleichungssysteme

2.4.1 Definitionen

D Für gegebene $A_{ik}, b_i \in \mathbb{R}$ mit $i = 1, \dots, n; k = 1, \dots, m$ wird das System von Gleichungen:

$$\begin{aligned} A_{11}x_1 + A_{12}x_2 + \dots + A_{1m}x_m &= b_1 \\ A_{21}x_1 + A_{22}x_2 + \dots + A_{2m}x_m &= b_2 \\ &\vdots \\ A_{n1}x_1 + A_{n2}x_2 + \dots + A_{nm}x_m &= b_n \end{aligned}$$

als **lineares Gleichungssystem (LGS)** für das unbekannte Lösungstupel (x_1, \dots, x_m) bezeichnet.

! Im folgenden gelte $m = n$; also genausoviel Unbekannte x_i wie Gleichungen.

N

$$A\vec{x} = \vec{b}$$

A : $n \times n$ -**Koeffizientenmatrix**, mit $\vec{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$, $\vec{b} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$

Der Rang von A heißt der **Rang des LGS**.

D Für $\vec{b} = \vec{0}$ spricht man von einem **homogenen LGS**, das stets die **triviale Lösung** $\vec{x} = \vec{0}$ hat (und eventuell weitere Lösungen). Die Lösung eines **inhomogenen LGS** mit $\vec{b} \neq \vec{0}$, das nicht das zugehörige homogene LGS löst, heißt **partikulär**.

2.4.2 Lösbarkeit eines LGS's

S Die Lösungen eines homogenen LGS's $A\vec{x} = \vec{0}$ bilden einen linearen Vektorraum der Dimension $h = n - r$, wenn r der Rang von A ist und n die Dimension von A .

B

$$A\vec{x} = \vec{0}$$

Elementare Umformungen führen zu

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \vec{x}' = 0$$

\vec{x}' gegenüber \vec{x} gegebenenfalls zeilenvertauscht.

Damit ist $x_i = 0$ für $i = 1, \dots, r$ und x_i frei wählbar für $i = r + 1, \dots, n$.

Somit können $h = n - r$ linear unabhängige Vektoren \vec{x} bestimmt werden.

S **Abgeschlossenheit** (des aufgespannten linearen Vektorraumes):

Seien \vec{x} und \vec{y} Lösungen, dann ist auch $\lambda\vec{x} + \mu\vec{y}$ Lösung.

B

$$A(\lambda\vec{x} + \mu\vec{y}) = \lambda A\vec{x} + \mu A\vec{y} = \lambda\vec{0} + \mu\vec{0} = \vec{0}$$

S Damit ein homogenes LGS nicht nur die triviale Lösung besitzt, muß $\det A = 0$ sein,

d.h. $r < n$. Gilt $n = r$, so besitzt das LGS nur die triviale Lösung $\vec{x} = \vec{0}$.

B Folgt unmittelbar aus obigem.

S Ein inhomogenes LGS $A\vec{x} = \vec{b}$ ist genau dann lösbar, wenn der Rang der Koeffizientenmatrix A gleich dem Rang der erweiterten $n \times (n + 1)$ -Matrix

$$\tilde{A} = \begin{pmatrix} A_{11} & \dots & A_{1n} & b_1 \\ \vdots & \cdot & \cdot & \vdots \\ A_{n1} & \dots & A_{nn} & b_n \end{pmatrix}$$

ist.

B Mit $A = (\vec{a}_1, \vec{a}_2, \dots, \vec{a}_n)$ (Spaltenvektoren) läßt sich schreiben:

$$\vec{a}_1 x_1 + \vec{a}_2 x_2 + \dots + \vec{a}_n x_n = \vec{b}$$

D.h. \vec{b} muß eine Linearkombination der \vec{a}_i sein. Dies ist nur möglich, wenn \vec{b} in dem von der Basis $\{\vec{a}_1, \vec{a}_2, \dots, \vec{a}_n\}$ aufgespannten Vektorraum liegt.

S Jede Lösung \vec{x} eines inhomogenen LGS's ist die Summe einer speziellen (partiikulären) Lösung \vec{x}_1 und einer beliebigen Lösung \vec{x}_0 des zugehörigen homogenen LGS's:

$$\begin{aligned} A\vec{x}_1 &= \vec{b}_1 \\ A\vec{x}_0 &= \vec{0} \\ \Rightarrow A(\vec{x}_1 + \vec{x}_0) &= A\vec{x}_1 + A\vec{x}_0 = \vec{b} + \vec{0} = \vec{b} \end{aligned}$$

S Ein inhomogenes LGS ist genau dann eindeutig lösbar, wenn das dazugehörige homogene LGS nur die triviale Lösung besitzt.

B In diesem Fall ist $\text{Rang } A = n$. Damit existiert A^{-1} :

$$\vec{x} = A^{-1}\vec{b}$$

ist die gesuchte Lösung!

Da $\vec{b} = \sum_i \vec{a}_i x_i$ (mit \vec{a}_i : Spaltenvektoren von A) eindeutig ist, bezüglich der Entwicklungskoeffizienten von x_i , ist auch \vec{x} als Lösung von $A\vec{x} = \vec{b}$ eindeutig.

CAS-Beispiel



A und \vec{b} legen ein LGS $A\vec{x} = \vec{b}$ fest.

```
import numpy as np

A = np.matrix([[1, 2, 2],
               [2,-2, 1],
               [2, 0, 1]])

b = np.array([[2, -1, 2]])
```

Die Determinante von A ist ungleich 0. Was folgt daraus?

```
np.linalg.det(A)
```

Die inverse Matrix zu A erhält man mit:

```
AINV = A.getI()
```

Die gesuchte Lösung \vec{x} ergibt sich aus:

```
x = AINV*b.transpose()
x
```

Überprüfen Sie das Ergebnis auf Korrektheit.
Gibt es weitere Lösungen?

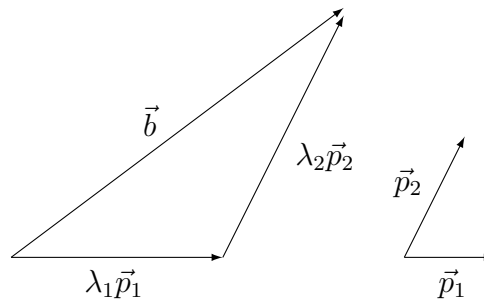
Alternative Berechnung von \vec{x} : Cramersche Regel (siehe z.B. Bronstein)

2.5 Basistransformation und Symmetrieoperationen

2.5.1 Koordinaten eines Vektors bezüglich einer festen Basis

Bezogen auf eine feste Basis \vec{p}_i mit $i = 1, \dots, n$, die einen Vektorraum aufspannt, läßt sich jeder Vektor dieses Raumes als Linearkombination darstellen (vgl. 2.1.6).

$$\vec{b} = \sum_i \lambda_i \vec{p}_i$$



Bestimmung der Koordinaten λ_i :

$$\begin{aligned} \vec{p}_1 \cdot \vec{b} &= \vec{p}_1 \cdot \vec{p}_1 \lambda_1 + \vec{p}_1 \cdot \vec{p}_2 \lambda_2 + \vec{p}_1 \cdot \vec{p}_3 \lambda_3 + \dots \\ &\vdots \\ \vec{p}_n \cdot \vec{b} &= \vec{p}_n \cdot \vec{p}_1 \lambda_1 + \vec{p}_n \cdot \vec{p}_2 \lambda_2 + \vec{p}_n \cdot \vec{p}_3 \lambda_3 + \dots \\ \vec{c} &= A \cdot \vec{x} \end{aligned}$$

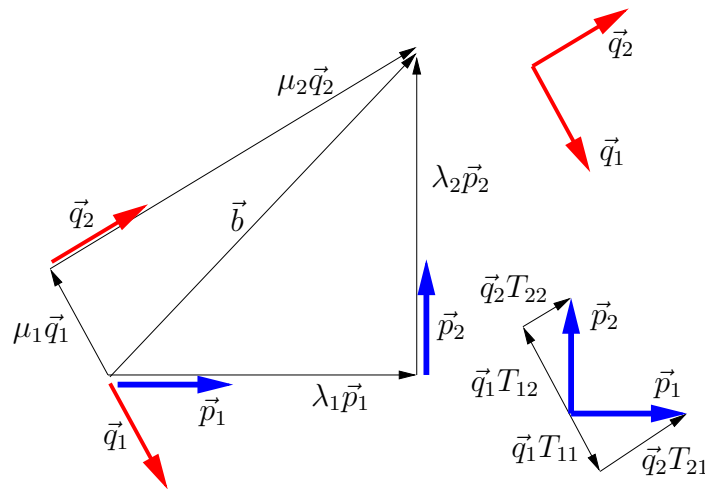
$$\text{mit } \vec{c} = \begin{pmatrix} \vec{p}_1 \cdot \vec{b} \\ \vdots \\ \vec{p}_n \cdot \vec{b} \end{pmatrix}, A = \begin{pmatrix} \vec{p}_1 \cdot \vec{p}_1 & \dots & \vec{p}_1 \cdot \vec{p}_n \\ \vdots & \ddots & \vdots \\ \vec{p}_n \cdot \vec{p}_1 & \dots & \vec{p}_n \cdot \vec{p}_n \end{pmatrix}, \vec{x} = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix}.$$

Lösung des inhomogenen LGS liefert \vec{x} und damit die gesuchten Koordinaten λ_i : $\vec{x} = A^{-1} \cdot \vec{c}$

2.5.2 Basistransformation

Problemstellung:

Wie sind die Koordinaten eines Vektors bezüglich einer Basis $\vec{p}_1, \dots, \vec{p}_n$ verknüpft mit den Koordinaten bzgl. einer zweiten Basis $\vec{q}_1, \dots, \vec{q}_n$?



$$\vec{b} = \lambda_1 \vec{p}_1 + \lambda_2 \vec{p}_2 = \sum_{i=1}^n \vec{p}_i \lambda_i \quad \vec{b}_{\vec{p}} = \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix}$$

$$\vec{b} = \mu_1 \vec{q}_1 + \mu_2 \vec{q}_2 = \sum_{i=1}^n \vec{q}_i \mu_i \quad \vec{b}_{\vec{q}} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$$

Ziel ist es nun, herauszufinden, welcher Zusammenhang zwischen $\vec{b}_{\vec{p}}$ und $\vec{b}_{\vec{q}}$ besteht, wenn die Basisvektoren über

$$\vec{p}_i = \sum_{j=1}^n \vec{q}_j T_{ji}$$

miteinander verknüpft sind (sind die Vektoren $\{\vec{q}_i\}$ orthonormiert dann ist $T_{ji} = (\vec{q}_j, \vec{p}_i)$).

Der Vektor \vec{b} muß unabhängig vom Koordinatensystem sein:

$$\begin{aligned} \vec{b} &= \lambda_1 \vec{p}_1 + \lambda_2 \vec{p}_2 + \dots + \lambda_n \vec{p}_n \\ &= (\vec{q}_1 T_{11} + \vec{q}_2 T_{21} + \dots + \vec{q}_n T_{n1}) \lambda_1 \\ &\quad + (\vec{q}_1 T_{12} + \vec{q}_2 T_{22} + \dots + \vec{q}_n T_{n2}) \lambda_2 \\ &\quad + \dots \\ &\quad + (\vec{q}_1 T_{1n} + \vec{q}_2 T_{2n} + \dots + \vec{q}_n T_{nn}) \lambda_n \\ &= (T_{11} \lambda_1 + T_{12} \lambda_2 + \dots + T_{1n} \lambda_n) \vec{q}_1 \\ &\quad + (T_{21} \lambda_1 + T_{22} \lambda_2 + \dots + T_{2n} \lambda_n) \vec{q}_2 \\ &\quad + \dots \\ &\quad + (T_{n1} \lambda_1 + T_{n2} \lambda_2 + \dots + T_{nn} \lambda_n) \vec{q}_n \\ &= \mu_1 \vec{q}_1 + \mu_2 \vec{q}_2 + \dots + \mu_n \vec{q}_n \\ &= \sum_{i=1}^n \mu_i \vec{q}_i \end{aligned}$$

Also sind die neuen Koordinaten gegeben durch

$$\mu_i = \sum_{j=1}^n T_{ij} \lambda_j$$

$$\begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix} = \vec{b}_{\vec{q}} = T \cdot \vec{b}_{\vec{p}} = T \cdot \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix}$$

Eigenschaften der **Transformationsmatrix T** :

Sind \vec{p}_i und \vec{q}_i orthonormiert, so gilt:

- Die Matrixelemente in der j -ten Spalte sind die Komponenten des alten Basisvektors \vec{p}_j bezüglich der neuen Basis \vec{q}_i mit $i = 1, \dots, n$.
- T ist regulär
- T ist orthogonal $T^{-1} = T^T$
- $\det T = \pm 1$
- Skalarprodukte werden durch eine Basistransformation nicht verändert:

$$\begin{aligned} \vec{x}' \cdot \vec{y}' &= (T \vec{x}) \cdot (T \vec{y}) \\ &= (T \vec{x})^T (T \vec{y}) \\ &= \vec{x}^T T^T T \vec{y} \\ &= \vec{x}^T T^{-1} T \vec{y} \\ &= \vec{x}^T \vec{y} \\ &= \vec{x} \cdot \vec{y} \end{aligned}$$

Insbesondere bleibt die Länge des Vektors erhalten!

- Sind \vec{q}_i und \vec{p}_i Basisvektoren im Ortsraum, so spricht man von **Koordinatentransformation**.

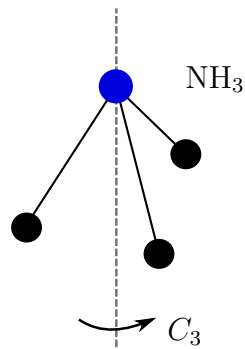
2.5.3 Symmetrieoperationen

D **Symmetrieoperationen** wie Drehung, Spiegelung etc., sind Operationen, die ein Objekt in sich selbst überführen.

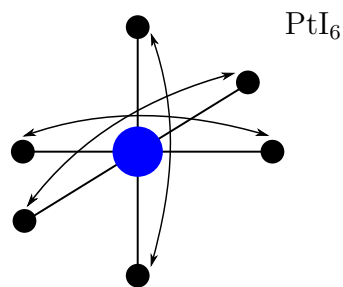
- **Identität E** :
Triviale Symmetrieoperation (1-Element)

- n -fache **Drehung** C_n :

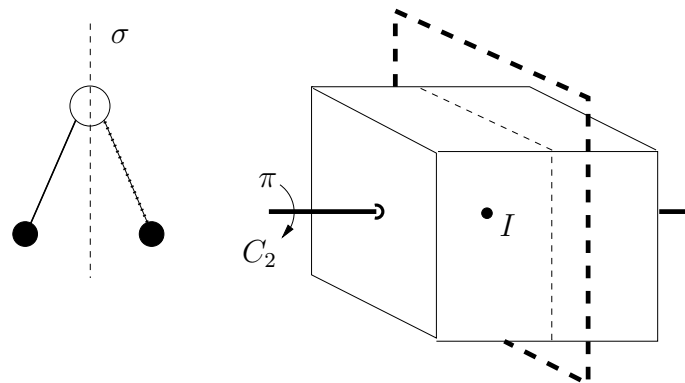
Drehung um Vielfaches von $2\pi/n$ um eine feste Achse



- **Inversion** I : Punktspiegelung am Symmetriezentrum

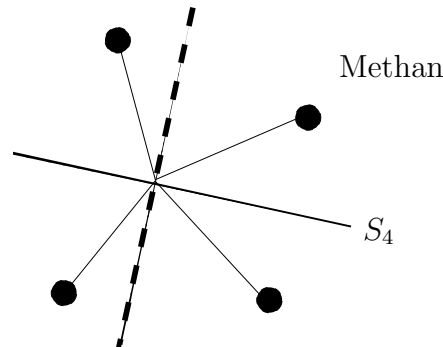


- **Spiegelung** σ : Spiegelung an einer Ebene – lässt sich zusammensetzen aus einer Drehung C_2 und der Inversion I , wobei die Drehachse orthogonal zur Spiegelebene steht und das Inversionszentrum mit dem Schnittpunkt von Drehachse und Ebene zusammenfällt.



- **Drehspiegelung S_n :**

Drehung um $2\pi/n$ mit anschließender Spiegelung an einer Ebene senkrecht zur Drehachse.



Symmetrieeoperationen lassen sich durch ihre Auswirkung auf einen sinnvoll gewählten Satz von Basisvektoren (-funktionen) mittels Matrizen darstellen. Für die Ortsvektoren bestehen diese Darstellungen aus 3×3 -Matrizen, wobei gilt: $\vec{r}' = M_s \cdot \vec{r}$



Aktive Interpretation der Operation,

d.h. die Ortsvektoren werden geändert, das Koordinatensystem bleibt im Raum fest. Es gilt:

$$M_{\text{aktiv}} = M_{\text{passiv}}^{-1}$$

wobei M_{passiv} eine Koordinatentransformation repräsentiert.

Klassifizierung der obigen Symmetrieeoperationen:

- alle Operationen sind Punktoperationen, d.h. es gibt mindestens einen Fixpunkt \vec{r} mit $\vec{r} = \vec{r}'$.
- $\det M_s = +1$ für E, C_n : eigentliche Drehungen
 $\det M_s = -1$ für I, σ, S_n : uneigentliche Drehungen

Bei Festkörpern tritt weiterhin die **Translation** als Symmetrieeoperation auf:

$$\vec{r}' = \vec{r} + \vec{a} \quad \text{mit} \quad \vec{a} = n_1 \vec{a}_1 + n_2 \vec{a}_2 + n_3 \vec{a}_3 \quad (\text{drei Raumrichtungen})$$

Kombination von Rotation und Translation:

$$\vec{r}' = \{R | \vec{a}\} \vec{r} = R\vec{r} + \vec{a}$$

In Matrixnotation zusammengefaßt:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} R_{11} & R_{12} & R_{13} & a_x \\ R_{21} & R_{22} & R_{23} & a_y \\ R_{31} & R_{32} & R_{33} & a_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

2.6 Eigenwertprobleme

2.6.1 Definitionen

D Die Gleichung $A\vec{x} = \lambda\vec{x}$ bzw. $(A - \lambda E)\vec{x} = 0$ zu einer gegebenen $n \times n$ -Matrix A stellt ein spezielles **Eigenwertproblem (EWP)** dar. Der gesuchte Vektor \vec{x} wird als **Eigenvektor**, der gesuchte Skalar λ wird als **Eigenwert** bezeichnet.

$(A - \lambda E)$ heißt **charakteristische Matrix**, das Polynom $\det(A - \lambda E)$ heißt **charakteristisches Polynom**. Die Gleichung in λ , die sich aus $(A - \lambda E) = 0$ ergibt heißt **charakteristische Gleichung**. Diese Gleichung n -ten Grades in λ besitzt n Lösungen λ_i mit $i = 1, \dots, n$. Sind m der λ_i gleich, so heißt dieser Eigenwert m -fach **entartet**.

Zu gegebenen Matrizen A und B nennt man $A\vec{x} = \lambda B\vec{x}$ ein **verallgemeinertes EWP**.

2.6.2 Lösung eines Eigenwertproblems

1. Schritt:

Lösen der charakteristischen Gleichung $\det(A - \lambda E) = 0$, liefert die Eigenwerte λ_i für $i = 1, \dots, n$.

2. Schritt:

Lösen des homogenen LGS's $(A - \lambda_i E)\vec{x}_i = \vec{0}$ (bzw. $A_i\vec{x}_i = \vec{0}$ mit $A_i = A - \lambda_i E$) liefert zu jedem λ_i ein \vec{x}_i ($i = 1, \dots, n$).

3. Schritt

Normieren der n Eigenvektoren: $\vec{x}_i \rightarrow \frac{1}{|\vec{x}_i|}\vec{x}_i$.

Beispiel:

$$A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$$

1. Schritt:

charakteristische Gleichung:

$$\begin{aligned} 0 &= \det \begin{pmatrix} 1 - \lambda & 2 \\ 2 & 1 - \lambda \end{pmatrix} = (1 - \lambda)^2 - 4 \\ &\Leftrightarrow (1 - \lambda)^2 = 4 \Rightarrow \lambda_{1,2} = 1 \pm 2 \Rightarrow \lambda_1 = -1 \quad \text{und} \quad \lambda_2 = 3 \end{aligned}$$

2. Schritt:

Einsetzen in das homogene LGS, um die Eigenvektoren \vec{x}_i herauszufinden:

Für λ_1 :

$$\begin{aligned} \left[\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} - \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \right] \vec{x}_1 &= \vec{0} \\ \begin{pmatrix} 1+1 & 2 \\ 2 & 1+1 \end{pmatrix} \vec{x}_1 &= \vec{0} \end{aligned}$$

Mit $\vec{x}_1 = \begin{pmatrix} a \\ b \end{pmatrix}$:

$$\begin{aligned} \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \Rightarrow +a + b &= 0 \end{aligned}$$

also ist $\vec{x}_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ ein möglicher Eigenvektor zu λ_1 .

Für λ_2 :

$$\begin{aligned} \left[\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} - \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \right] \vec{x}_2 &= \vec{0} \\ \begin{pmatrix} 1-3 & 2 \\ 2 & 1-3 \end{pmatrix} \vec{x}_2 &= \vec{0} \end{aligned}$$

Mit $\vec{x}_2 = \begin{pmatrix} a \\ b \end{pmatrix}$:

$$\begin{aligned} \begin{pmatrix} -2 & 2 \\ 2 & -2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \Rightarrow -a + b &= 0 \end{aligned}$$

also ist $\vec{x}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ ein möglicher Eigenvektor zu λ_2 .

3. Schritt

Normieren der Eigenvektoren:

$$\vec{x}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \vec{x}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Allgemeine Lösung für obiges Beispiel: $H = \begin{pmatrix} \alpha & \beta \\ \beta & \alpha \end{pmatrix}$

Das zugehörige EWP tritt auf der in quantenmechanischen Behandlung des H_2^+ -Moleküls im Rahmen der Hückel-Theorie.

1. Schritt

$$\begin{aligned} \begin{pmatrix} \alpha - \lambda & \beta \\ \beta & \alpha - \lambda \end{pmatrix} \vec{x} = \vec{0} &\Rightarrow (\alpha - \lambda) = \pm\beta \Rightarrow \lambda_{1,2} = \alpha \pm \beta \\ &\Rightarrow \lambda_1 = \alpha - \beta \quad \lambda_2 = \alpha + \beta \end{aligned}$$

2. Schritt und 3. Schritt

Für λ_1 :

$$\begin{aligned} \left[\begin{pmatrix} \alpha & \beta \\ \beta & \alpha \end{pmatrix} - \begin{pmatrix} \alpha - \beta & 0 \\ 0 & \alpha - \beta \end{pmatrix} \right] \vec{x}_1 &= \vec{0} \\ \begin{pmatrix} \beta & \beta \\ \beta & \beta \end{pmatrix} \vec{x}_1 &= \vec{0} \\ \Rightarrow \vec{x}_1 &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \end{aligned}$$

Für λ_2 :

$$\begin{aligned} \left[\begin{pmatrix} \alpha & \beta \\ \beta & \alpha \end{pmatrix} - \begin{pmatrix} \alpha + \beta & 0 \\ 0 & \alpha + \beta \end{pmatrix} \right] \vec{x}_2 &= \vec{0} \\ \begin{pmatrix} -\beta & \beta \\ \beta & -\beta \end{pmatrix} \vec{x}_2 &= \vec{0} \\ \Rightarrow \vec{x}_2 &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \end{aligned}$$

CAS-Beispiel



Die Matrix H ist die sogenannte Hückelmatrix, die sich bei der Behandlung des π -Elektronensystems von Benzol ergibt. Sollen Variablen in der Matrix verwendet werden, so wird das Modul SymPy benötigt:

```
from sympy import Matrix
from sympy.abc import a, b

H = Matrix([[a, b, 0, 0, 0, b],
            [b, a, b, 0, 0, 0],
            [0, b, a, b, 0, 0],
            [0, 0, b, a, b, 0],
            [0, 0, 0, b, a, b],
            [b, 0, 0, 0, 0, b]])
```

Die Energiewerte für elektronische Zustände erhält man als Eigenwerte von H mittels `eigenvals`:

```
H.eigenvals()
```

Die Eigenvektoren lassen sich mittels `eigenvects` ermitteln:

Anmerkung: Die folgenden Berechnungen können je nach Computer auch einige Minuten in Anspruch nehmen!

```
H.eigenvects()
```

Diese Prozedur liefert gleichzeitig die Eigenwerte und deren Entartung. Sind die Eigenwerte des Beispiels entartet?

Nochmals dasselbe für Cyclobutadien:

```
H = Matrix([[a, b, 0, b],
            [b, a, b, 0],
            [0, b, a, b],
            [b, 0, b, a]])

T = H.eigenvects()

U = Matrix([(eigenvect/eigenvect.norm())[:],
            for eigenvect in [T[0][2][0], T[0][2][1],
                             T[1][2][0], T[2][2][0]]])

G = U*H*U.transpose()
G
```

```
H.eigenvals()
```

```
G.eigenvals()
```

Alternativ können die Eigenwerte und Eigenvektoren auch direkt mittels `diagonalize` erhalten werden:

```
T, G = H.diagonalize(sort = True, normalize = True)
```

```
T
```

```
G
```

```
H.eigenvals()
```

```
G.eigenvals()
```

2.6.3 Eigenschaften eines Eigenwertproblems

S Sind die Eigenwerte λ_i zu $(A - \lambda E)\vec{x} = 0$ nicht entartet, so gilt:

- a) zu jedem λ_i gibt es genau ein \vec{x}_i
- b) die n Eigenvektoren \vec{x}_i sind linear unabhängig.

B zu b)

Annahme: lineare Abhängigkeit, dann gilt:

$$\begin{aligned}\sum_{i=1}^n \mu_i \vec{x}_i &= 0 \\ A \sum_{i=1}^n \mu_i \vec{x}_i &= 0 \\ \sum_{i=1}^n \mu_i A \vec{x}_i &= 0 \\ \sum_{i=1}^n \mu_i \lambda_i \vec{x}_i &= 0 \\ \sum_{i=1}^n \nu_i \vec{x}_i &= 0\end{aligned}$$

$\Rightarrow \mu_i = \text{const} \cdot \nu_i = \text{const} \cdot \mu_i \lambda_i$ das hieße, daß $\lambda_i = \text{const}$ sein muß, was aber ein Widerspruch zur Voraussetzung ist!

S Ist ein Eigenwert λ_i m_i -fach entartet, so gibt es hierzu s_i linear unabhängige Eigenvektoren mit $1 \leq s_i \leq m_i$. Bei insgesamt k verschiedenen Eigenwerten λ_i ($i = 1, \dots, k$) mit Entartung m_i gilt daher: $k \leq s \leq n$ mit $s = \sum_{i=1}^k s_i$ und $n = \sum_{i=1}^k m_i$.

S Das Eigenwertproblem $(A - \lambda E)\vec{x} = 0$ mit $A = A^T$ (d.h. reelle symmetrische Matrix) besitzt:

- n reelle Eigenwerte
- n linear unabhängige Eigenvektoren
- sind die Eigenwerte nicht entartet, so sind die Eigenvektoren orthogonal
- ist ein Eigenwert m -fach entartet, so sind die Eigenvektoren linear unabhängig und lassen sich orthonormieren.

N M^\dagger ist die komplex transponierte Matrix von M , d.h. $(M^\dagger)_{ij} = M_{ji}^*$.

B zu a) Erweitere die Aussage auf hermitesche Matrizen mit $A = A^\dagger$

$$\begin{aligned}
 A\vec{x} &= \lambda\vec{x} \\
 Ax &= \lambda x \\
 x^\dagger Ax &= \lambda x^\dagger x \\
 (A^\dagger x)^\dagger x &= \lambda |x|^2 \\
 (Ax)^\dagger x &= \lambda |x|^2 \\
 (\lambda x)^\dagger x &= \lambda |x|^2 \\
 \lambda^* x^\dagger x &= \lambda |x|^2 \\
 \lambda^* &= \lambda \Rightarrow \lambda \text{ ist reell!}
 \end{aligned}$$

Dies ist eine wichtige Eigenschaft der Hamiltonmatrix H in der Quantenmechanik, wo das Eigenwertproblem $(H - \lambda E)\vec{x} = 0$ zu lösen ist.

S Die Eigenwerte λ_i zu einer Matrix A und zu einer Matrix $A' = TAT^{-1}$ sind gleich.

B

$$A\vec{x} = \lambda\vec{x}$$

hierfür gilt die charakteristische Gleichung $\det(A - \lambda E) = 0$

$$\begin{aligned}
 TAT^{-1}T\vec{x} &= T\lambda\vec{x} \quad \text{mit} \quad TAT^{-1} = A' \quad \text{und} \quad T\vec{x} = \vec{x}' \\
 \Rightarrow A'\vec{x}' &= \lambda\vec{x}'
 \end{aligned}$$

die charakteristische Gleichung hierfür lautet:

$$\begin{aligned}
 \det(A' - \lambda E) &= 0 \\
 \det(TAT^{-1} - \lambda E) &= 0 \\
 \det(T(A - \lambda T^{-1}ET)T^{-1}) &= 0 \\
 \det(T) \det(T^{-1}) \det(A - \lambda E) &= 0 \\
 \det(A - \lambda E) &= 0
 \end{aligned}$$

dies ist nun die gleiche charakteristische Gleichung wie zu $A\vec{x} = \lambda\vec{x}$. Damit sind auch die Eigenwerte gleich!

! Speziell gilt dies für Basistransformationen bei denen $T^{-1} = T^T$ gilt.

Beispiel:

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad T^{-1} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$\det(A - \lambda E) = \lambda^2 - 1 = 0 \Rightarrow \lambda_{1,2} = \pm 1$$

$$\begin{aligned} TAT^{-1} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \\ &= \frac{1}{2} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \\ &= \frac{1}{2} \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \end{aligned}$$

$$\det(TAT^{-1} - \lambda E) = 0$$

$$\det \begin{pmatrix} 1 - \lambda & 0 \\ 0 & -1 - \lambda \end{pmatrix} = 0$$

$$(1 - \lambda)(-1 - \lambda) = 0$$

$$-1 + \lambda^2 = 0$$

$$\lambda_{1,2} = \pm 1$$

CAS-Beispiel



H ist eine 3×3 -Matrix, deren Eigenwerte bestimmt werden sollen. Die Anweisung `shape(H)` findet heraus, welche Dimension H besitzt.

```
import numpy as np

H = np.matrix([[2, 1, 1],
               [1, 2, 1],
               [1, 1, 2]])

np.shape(H)
```

T ist eine beliebige orthogonale Transformationsmatrix.

```
T = np.matrix([[1/3,                2/3,                2/3],
               [2/np.sqrt(5),      0, 1/np.sqrt(5)],
               [2/np.sqrt(45), -5/np.sqrt(45), 4/np.sqrt(45)]]

TINV = T.getI()
np.dot(T, TINV)
```

Im Folgenden werden die Matrix H und ihre Transformierte $G = THT^{-1}$ aufgelistet. Zudem werden die jeweiligen Eigenwerte berechnet.

Was fällt Ihnen auf?

```
from numpy.linalg import eig, eigvals
```

```
H
```

```
G = np.dot(T, np.dot(H, TINV))
```

```
eigvals(H)
```

```
eigvals(G)
```

Berechnen Sie die Eigenvektoren von H und G .

Was fällt auf?

```
eig(H)
```

```
eig(G)
```

Im nächsten Beispiel wird die Matrix T aus den zuvor berechneten Eigenvektoren der Matrix H erstellt. Was fällt Ihnen auf, wenn Sie die errechneten Eigenvektoren mit solchen vergleichen, wie sie z.B. vom [SHMO-Tool](#) erhalten werden?

Anmerkung: Die folgenden Berechnungen können je nach Computer auch einige Minuten in Anspruch nehmen!

```
from sympy import Matrix
from sympy.abc import a, b

H = Matrix([[a, b, 0, 0, 0, b],
            [b, a, b, 0, 0, 0],
            [0, b, a, b, 0, 0],
            [0, 0, b, a, b, 0],
            [0, 0, 0, b, a, b],
            [b, 0, 0, 0, b, a]])

H.rank()
T, G = H.diagonalize(sort = True, normalize = True)
```

```
T
```

```
G
```

Was fällt bei der folgenden Auflistung von H und G auf?

```
H.eigenvals()
```

```
G.eigenvals()
```

Die Rechnung liefert eine diagonale Matrix G mit den Eigenwerten auf der Diagonalen – daher oft die Bezeichnung **Diagonalisierung** statt Eigenwertbestimmung.

S Für eine Dreiecksmatrix stimmen die Eigenwerte mit den Diagonalelementen überein.

B

$$\begin{aligned}\det(D - \lambda E) &= \det D' \\ &= \prod_{i=1}^n (D_{ii} - \lambda) = 0\end{aligned}$$

Die Nullstellen sind die Eigenwerte $\lambda_i = D_{ii}$ ($i = 1, \dots, n$).

N Statt Eigenwertbestimmung spricht man daher oft von **Diagonalisierung**.

S Zu jeder reellen symmetrischen Matrix M läßt sich eine Transformation T angeben, die M in eine Diagonalmatrix M' , überführt. Die Diagonalelemente von M' sind die gesuchten Eigenwerte von M . Die Transformation nennt man **Hauptachsentransformation**.

B

$$\begin{aligned}(M - \lambda E)\vec{x} &= 0 \\ (T M T^{-1} T - T \lambda E)\vec{x} &= 0 \\ (T M T^{-1} - \lambda E) T \vec{x} &= 0\end{aligned}$$

wähle T so, daß die Zeilenvektoren mit den Eigenvektoren \vec{x}_i von M übereinstimmen.

$$\begin{aligned}\left[\begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_n \end{pmatrix} M(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n) - \lambda E \right] \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_n \end{pmatrix} \vec{x} &= 0 \\ M' \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_n \end{pmatrix} \vec{x} &= \lambda E \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_n \end{pmatrix} \vec{x}\end{aligned}$$

Speziell für $\vec{x} = \vec{x}_j$:

$$M' \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} = \lambda_j E \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \leftarrow j$$

$$\begin{aligned} \Rightarrow M'_{ij} &= 0 \quad i \neq j \\ M'_{ij} &= \lambda_i \quad \text{für alle } j \end{aligned}$$

Kapitel 3

Funktionen einer Variablen

3.1 Grundbegriffe

3.1.1 Definitionen

D Seien A und B zwei beliebige Mengen, so versteht man unter einer **Funktion** oder **Abbildung** f von A nach B

$$f : A \longrightarrow B$$

eine eindeutige Vorschrift, die einem $x \in A$ genau ein $y \in B$ zuordnet:

$$f : x \longmapsto y = f(x) = y(x)$$

Definitionsbereich	}	siehe Abschnitt 1.4
Wertebereich		
Funktionswert		
surjektiv		
injektiv/eineindeutig		
bijektiv		
Umkehrabbildung		
Graph		
u.s.w.		

Im folgenden gehen wir von $A, B \subset \mathbb{R}$ – in einigen Fällen von $A, B \subset \mathbb{C}$ – aus. Damit sind die Größen x und y , die wir als **unabhängige** und **abhängige Variablen** bzw. **Veränderliche** bezeichnen, einfach Zahlen. Wir können daher die "Vorschrift" f in der Regel durch einfache Rechenoperationen wie Addition, Multiplikation u.s.w. ausdrücken.

3.1.2 Darstellung einer Funktion

a) Angabe des Definitions- und Wertebereichs, sowie der Abbildungsvorschrift.

Beispiel:

$$\begin{aligned} f: \mathbb{R} &\longrightarrow \mathbb{R} \\ x &\longmapsto y = 25x + 10x^3 + 2x^5 \\ g: \mathbb{R}^+ &\longrightarrow \mathbb{R} \\ x &\longmapsto y = \sqrt{x} \end{aligned}$$

Dabei ist es möglich die Funktion abschnittsweise zu definieren.

Stufenfunktion

$$y = h(x) = \begin{cases} 0 & \text{für } x < 0 \\ 1 & \text{für } x \geq 0 \end{cases}$$

Signumfunktion

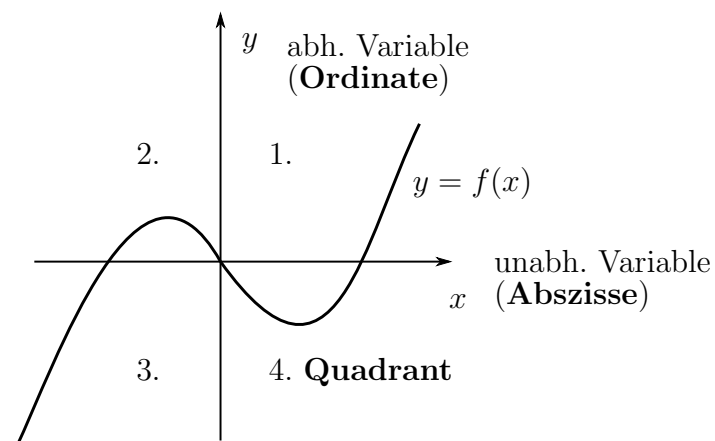
$$y = \text{sign}(x) = \begin{cases} -1 & \text{für } x < 0 \\ 0 & \text{für } x = 0 \\ +1 & \text{für } x > 0 \end{cases}$$

b) **Wertetabelle**

unabh. Variab. x	...	-3	-2	-1	0	1	2	3	...
abh. Variab. y	...	9	4	1	0	1	4	9	...

- falls kein einfacher Zusammenhang zwischen x und y besteht
- zum Festhalten von einzelnen Meßergebnissen

c) Graphische Darstellung mittels eines **kartesischen Koordinatensystems**:

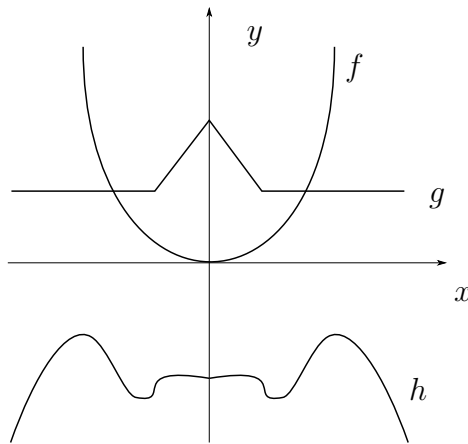


d) Charakterisierung einer Funktion

D Gilt für eine Funktion

$$f(x) = f(-x),$$

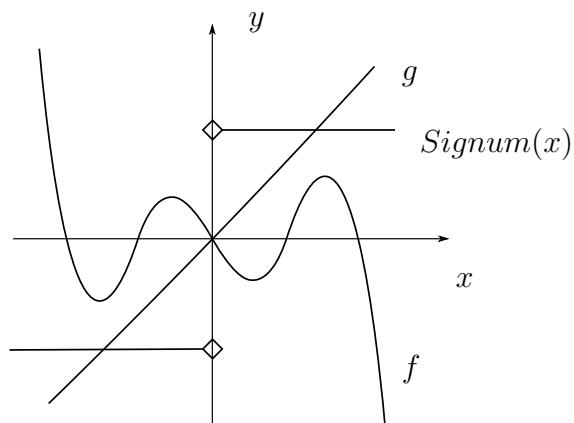
so wird sie als **gerade** oder **symmetrisch** bezeichnet.



D Gilt für eine Funktion

$$f(x) = -f(-x),$$

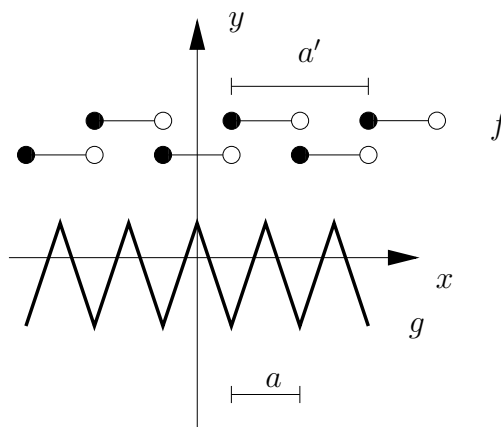
so wird sie als **ungerade** oder **punktsymmetrisch** oder **antisymmetrisch** bezeichnet.



D Gilt für eine Funktion

$$f(x) = f(x + na)$$

für alle $n \in \mathbb{Z}$, so wird die Funktion als **periodisch** bezeichnet und a die **Periode** der Funktion.



D Existiert ein $G \in \mathbb{R}$, so daß für alle $x \in I = [a, b] \subset D(f)$ gilt:

$$|f(x)| \leq G$$

so heißt die Funktion auf I beschränkt. Gilt:

$$f(x) \leq G' \quad \text{bzw.} \quad f(x) \geq G''$$

so heißt $f(x)$

nach oben bzw. **nach unten**

beschränkt. Liegt diese Eigenschaft vor, so wird die

kleinste obere bzw. **größte untere**

Schranke G von $f(x)$ als

$$\begin{array}{ll} \text{Supremum} & \text{bzw.} & \text{Infimum} \\ \sup_{x \in I} f(x) = G & & \inf_{x \in I} f(x) = G \end{array}$$

bezeichnet. Gilt weiterhin $G \in \{f(x) | x \in I\}$, so wird G als

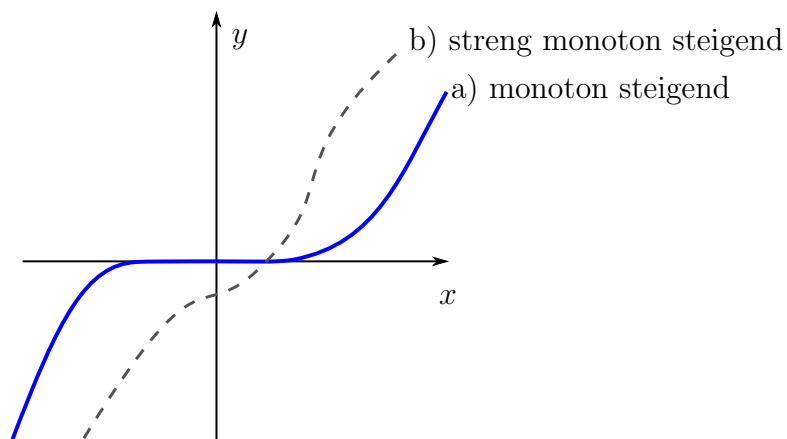
$$\begin{array}{ccc} \text{Maximum} & \text{bzw.} & \text{Minimum} \\ \max_{x \in I} f(x) = G & & \min_{x \in I} f(x) = G \end{array}$$

bezeichnet. Ist $I = D(f)$, so spricht man von einem **absoluten** bzw. **globalen** Min./Max. – ansonsten von einem **lokalen** Min./Max.

D Gilt für eine Funktion $f(x)$ für alle $x_1, x_2 \in D(f)$ mit $x_1 < x_2$:

- a) $f(x_1) \leq f(x_2)$ so heißt f **monoton steigend**
- b) $f(x_1) < f(x_2)$ so heißt f **streng monoton steigend**
- c) $f(x_1) \geq f(x_2)$ so heißt f **monoton fallend**
- d) $f(x_1) > f(x_2)$ so heißt f **streng monoton fallend**

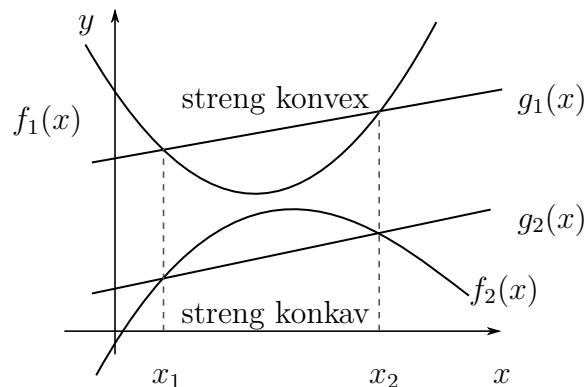
(oder wachsend statt steigend)



D Gilt für eine Funktion $f(x)$ für beliebiges $x_1, x, x_2 \in D(f)$ mit $x_1 < x < x_2$ und der Hilfsfunktion

$$g(x) = \frac{f(x_2) - f(x_1)}{x_2 - x_1} \cdot (x - x_1) + f(x_1)$$

- a) $g(x) \geq f(x)$ so heißt $f(x)$ **konvex**
- b) $g(x) > f(x)$ so heißt $f(x)$ **streng konvex**
- c) $g(x) \leq f(x)$ so heißt $f(x)$ **konkav**
- d) $g(x) < f(x)$ so heißt $f(x)$ **streng konkav**



CAS-Beispiel



Wenn Sie das Tool Jupyter Notebook verwenden, führen Sie diesen Befehl aus, um interaktive Plots zu erhalten. Ansonsten können Sie den Befehl ignorieren.

```
%matplotlib notebook
```

Charakterisieren Sie die folgenden Funktionen bzgl. Symmetrie, Beschränktheit, Monotonie und Krümmung, wobei der dargestellte Bereich gleich dem Definitionsbereich sei.

```
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, np.sin(x), label = r'\sin(x)')
plt.plot(x, np.cos(x), label = r'\cos(x)')
plt.plot(x, np.sin(x)*np.cos(x), label = r'\sin(x) \cos(x)')
plt.legend()
plt.show()
```

```
x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, x**2, label = r'$x^2$')
plt.plot(x, x*np.cos(x), label = r'$x \cos(x)$')
plt.plot(x, x**2*np.cos(x), label = r'$(x^2) \cos(x)$')
plt.legend()
plt.show()
```

```
x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, x*np.sin(x), label = r'$x \sin(x)$')
plt.plot(x, x**2*np.sin(x), label = r'$(x^2) \sin(x)$')
plt.legend()
plt.show()
```

```
x = np.linspace(-1, 1)
```

```
plt.axis([-1, 1, -20, 20])
plt.grid()
plt.plot(x, 1/x)
plt.show()
```

Wie Sie sehen, produziert diese Anweisung noch nicht das gewünschte Ergebnis. Beachten Sie, daß die Funktion `numpy.linspace` standardmäßig 50 Punkte erzeugt. Um die Funktion bei $x = 0$ auszuwerten, muß in diesem Beispiel 0 als Wert von x manuell eingefügt werden. Durch folgende Anweisung kann die Polstelle bei $x = 0$ sichtbar gemacht werden:

```
x = np.linspace(-1, 1)
x = np.append(x, 0)
x.sort()

plt.axis([-1, 1, -20, 20])
plt.grid()
plt.plot(x, 1/x, label = r'\frac{1}{x}$')
plt.legend()
plt.show()
```

```
x = np.linspace(-1, 1)
x = np.append(x, 0)
x.sort()

plt.axis([-1, 1, -10, 50])
plt.grid()
plt.plot(x, 1/x**2, label = r'\frac{1}{x^2}$')
plt.legend()
plt.show()
```

Das Produkt zweier Funktionen, die beide symmetrisch oder beide antisymmetrisch sind, führt auf eine symmetrische Funktion. Das Produkt einer symmetrischen mit einer antisymmetrischen Funktion ist hingegen immer antisymmetrisch.

3.1.3 Umkehrfunktion

Jede bijektive Funktion $f(x)$ besitzt eine **Umkehrfunktion** oder **inverse Funktion** $f^{-1}(y)$ (siehe 1.4).

Offensichtlich erhält man $f^{-1}(y)$ indem man $f(x)$ nach x auflöst

- dies ist nur möglich falls $f(x)$ injektiv ist

- weiterhin ist $f^{-1}(y)$ nur dann als Funktion zu bezeichnen, wenn zu jedem $y \in B$ ein $x = f^{-1}(y)$ vorliegt, d.h. $f(x)$ surjektiv ist.

D.h. $f(x)$ muß bijektiv sein.



In der Regel vertauscht man nach Auflösen von $y = f(x)$ nach x die Variablennamen

x und y um wieder x als unabhängige Variable zu haben (in Gegensatz zur Darstellung in Abschnitt 1.4).

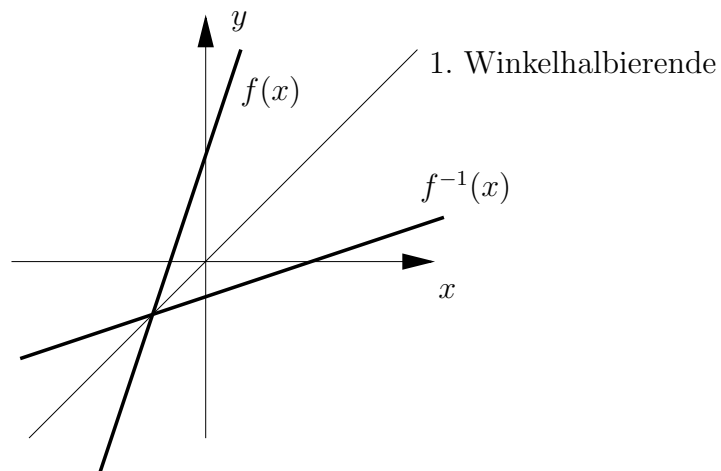
$$y = f(x) = 5 + 7x$$

$$1) \downarrow$$

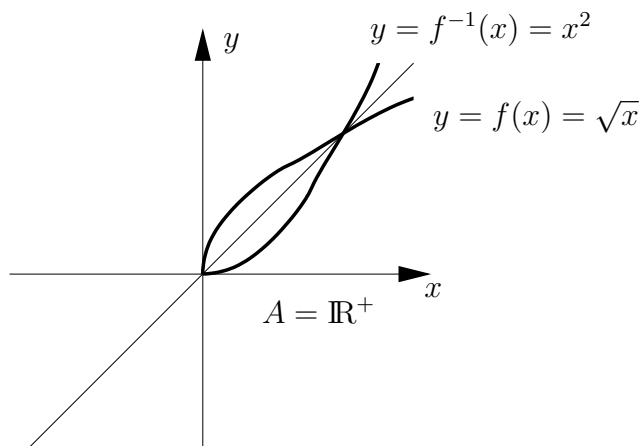
$$x = \frac{1}{7}(y - 5)$$

$$2) \downarrow$$

$$y = \frac{1}{7}(x - 5)$$



Offensichtlich geht f^{-1} durch Spiegelung der Funktion f an der 1. Winkelhalbierende hervor.



S Jede streng monoton steigende (fallende) Funktion f ist umkehrbar, sofern der Wertebereich ggf. so eingeschränkt wird, daß die Funktion surjektiv wird.

B Für eine streng monoton steigende (fallende) Funktion $f(x)$ und beliebige x_1, x_2

mit $x_1 \neq x_2$ gilt

$$\begin{array}{l} x_1 < x_2 \text{ damit } f(x_1) < f(x_2) \quad (f(x_1) > f(x_2)) \text{ oder} \\ x_2 < x_1 \text{ damit } f(x_2) < f(x_1) \quad (f(x_2) > f(x_1)) \end{array}$$

damit gilt immer $f(x_1) \neq f(x_2) \forall x_1, x_2, x_1 \neq x_2$, d.h. $f(x)$ ist injektiv.

S Ist eine Funktion f streng monoton steigend/fallend, so ist auch ihre Umkehrfunktion streng monoton steigend/fallend.

B Annahme: f sei streng monoton steigend.

$$\text{Mit } \begin{pmatrix} x_1 < x_2 \\ x_1 = x_2 \\ x_1 > x_2 \end{pmatrix} \text{ gilt daher } \begin{pmatrix} f(x_1) < f(x_2) \\ f(x_1) = f(x_2) \\ f(x_1) > f(x_2) \end{pmatrix}$$

D.h. $x_1 < x_2 \Leftrightarrow f(x_1) < f(x_2)$.

Mit $y_i = f(x_i)$ und $x_i = f^{-1}(y_i)$, $i = 1, 2$:

$$f^{-1}(y_1) < f^{-1}(y_2) \Leftrightarrow y_1 < y_2$$

D.h. $y_1 < y_2 \Rightarrow f^{-1}(y_1) < f^{-1}(y_2)$.

Damit ist auch $f^{-1}(y)$ streng monoton steigend!

Ist f monoton fallend, so läuft der Beweis analog.

3.1.4 Verknüpfung von Funktionen

Durch die Verknüpfung zweier oder mehrerer Funktionen läßt sich eine neue Funktion erhalten, deren Definitionsbereich gegenüber denen der ursprünglichen Funktionen eingeschränkt sein kann.

D Die **Summe**, die **Differenz**, das **Produkt** und der **Quotient** zweier Funktionen sind wie folgt definiert:

$$\begin{array}{l} h(x) = (f \pm g)(x) = f(x) \pm g(x) \quad \text{mit } D(f \pm g) = D(f) \cap D(g) \\ h(x) = (f \cdot g)(x) = f(x) \cdot g(x) \quad \text{mit } D(f \cdot g) = D(f) \cap D(g) \\ h(x) = (f/g)(x) = f(x)/g(x) \quad \text{mit } D(f/g) = D(f) \cap D(g) \setminus \{x | g(x) = 0\} \end{array}$$

Beispiel:

$$f : x \mapsto x^2$$

$$\begin{aligned}
 g : x &\mapsto \sin x \\
 (f \pm g) : x &\mapsto x^2 \pm \sin x \\
 (f \cdot g) : x &\mapsto x^2 \cdot \sin x \\
 (f/g) : x &\mapsto x^2 / \sin x \quad \text{für } x \neq n\pi, n \in \mathbb{Z}
 \end{aligned}$$

S Es gelten die Assoziativ- und Kommutativgesetze der Addition und Multiplikation, sowie das Distributivgesetz:

$$\begin{aligned}
 f + g &= g + f \\
 f \cdot g &= g \cdot f \\
 f + (g + h) &= (f + g) + h \\
 f \cdot (g \cdot h) &= (f \cdot g) \cdot h \\
 (f + g) \cdot h &= f \cdot h + g \cdot h
 \end{aligned}$$

Das Argument x der Funktionen wurde weggelassen.

D Die **Verkettung** oder **Komposition** $f \circ g$ zweier Funktionen f und g ist definiert durch:

$$(f \circ g)(x) = f(g(x)) \quad \text{mit} \quad D(f \circ g) = \{x \mid x \in D(g) \wedge g(x) \in D(f)\}$$

D.h. zuerst wird die Funktion g ausgewertet und dann das Ergebnis als Argument in die Funktion f eingesetzt.

Beispiel:

$$\begin{aligned}
 g : x &\mapsto 5 + 7x \\
 f : x &\mapsto \frac{1}{7}x - \frac{5}{7} \\
 (f \circ g)(x) &= \frac{1}{7}[5 + 7x] - \frac{5}{7} \\
 &= \frac{5}{7} + x - \frac{5}{7} \\
 &= x \\
 &= I(x) \quad \text{identische Abbildung}
 \end{aligned}$$

In diesem Beispiel ist die Funktion g gerade die Umkehrfunktion von f , d.h. es gilt $g = f^{-1}$. Offensichtlich gilt ganz allgemein

$$(f \circ f^{-1})(x) = I(x)$$

S Für die Verkettung von Funktionen gilt das Assoziativgesetz, nicht jedoch das Kommutativgesetz:

$$f \circ (g \circ h) = (f \circ g) \circ h$$

in der Regel gilt: $f \circ g \neq g \circ f$

Beispiel:

$$\begin{aligned} f : x &\mapsto x^2 \\ g : x &\mapsto \sin x \\ f \circ g : x &\mapsto [\sin(x)]^2 = \sin^2 x \\ g \circ f : x &\mapsto \sin x^2 \end{aligned}$$

CAS-Beispiel



Die Rechengesetze für Funktionen lassen sich graphisch leicht demonstrieren. Dazu werden die beiden Funktionen $f(x)$ und $g(x)$ sowie deren Verknüpfungen $h1(x), \dots, h6(x)$ definiert:

```
import numpy as np
import matplotlib.pyplot as plt

def f(x):
    return x + 2

def g(x):
    return np.sin(x)

def h1(x):
    return f(x) + g(x)

def h2(x):
    return f(x) - g(x)

def h3(x):
    return f(x) * g(x)

def h4(x):
    return f(x) / g(x)

def h5(x):
    return f(g(x))

def h6(x):
    return g(f(x))
```



```
x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, f(x), label = r'$f(x)$')
plt.plot(x, g(x), label = r'$g(x)$')
plt.plot(x, h1(x), label = r'$f(x) + g(x)$')
plt.legend()
plt.show()
```

```
x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, f(x), label = r'$f(x)$')
plt.plot(x, g(x), label = r'$g(x)$')
plt.plot(x, h2(x), label = r'$f(x) - g(x)$')
plt.legend()
plt.show()
```

```
x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, f(x), label = r'$f(x)$')
plt.plot(x, g(x), label = r'$g(x)$')
plt.plot(x, h3(x), label = r'$f(x) g(x)$')
plt.legend()
plt.show()
```

```
x = np.linspace(-5, 5)
y = h4(x)
y[np.abs(y) > 20] = np.nan
plt.axis([-5, 5, -5, 5])
plt.grid()
plt.plot(x, f(x), label = r'$f(x)$')
plt.plot(x, g(x), label = r'$g(x)$')
plt.plot(x, y, label = r'$f(x) / g(x)$')
plt.legend()
plt.show()
```

```
x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, f(x), label = r'$f(x)$')
plt.plot(x, g(x), label = r'$g(x)$')
plt.plot(x, h5(x), label = r'$f(g(x))$')
plt.legend()
plt.show()
```

```
x = np.linspace(-5, 5)
plt.grid()
plt.plot(x, f(x), label = r'$f(x)$')
plt.plot(x, g(x), label = r'$g(x)$')
plt.plot(x, h6(x), label = r'$g(f(x))$')
plt.legend()
plt.show()
```

3.2 Grenzwerte

3.2.1 Zahlenfolgen

D **Zahlenfolgen** oder kurz **Folgen** sind spezielle Funktionen a mit dem Definitionsbereich $D(a) = \mathbb{N}$ (oder einer Teilmenge von \mathbb{N}):

$$\begin{aligned} a : \mathbb{N} &\longrightarrow \mathbb{R} \\ n &\longmapsto a(n) = a_n \end{aligned}$$

Statt $a(n)$ schreibt man oft a_n .

a_n wird als n -tes Glied der Zahlenfolge a_0, a_1, a_2, \dots bezeichnet.

Das Symbol $\{a_n\}$ repräsentiert die gesamte Zahlenfolge.

Gilt für alle Glieder a_n $|a_n| \leq B$, so heißt die Folge **beschränkt**.

Gibt es zu jeder noch so kleinen Zahl $\epsilon > 0$ eine natürliche Zahl $n_0 \in \mathbb{N}$, so daß für alle $n \geq n_0$ $|a_n - G| < \epsilon$ gilt, so heißt G der **Grenzwert der Zahlenfolge** und man sagt $\{a_n\}$ **konvergiert gegen G** .

Symbolisch: $\lim_{n \rightarrow \infty} a_n = G$.

Speziell für $G = 0$ spricht man von **Nullfolge**.

Existiert kein Grenzwert, so sagt man die Folge **divergiert**.

Monotonie: vgl. Abschnitt 3.1.

! $\{a_n\}$ ist nicht mit der Menge $\{a_n | n \in \mathbb{N}\}$ zu verwechseln, da hier die Reihenfolge von Bedeutung ist.

Grenzwertuntersuchung: n_0 hängt in der Regel von ϵ ab weshalb oft $n_0(\epsilon)$ geschrieben wird.

Beispiel:

$$a_n = \frac{1}{n+1} \quad \text{gefordert: } |a_n - 0| < \epsilon \quad \text{mit Grenzwert } G = 0$$

Setze $n_0 = \frac{1}{\epsilon} - 1$.

Beispiele für Zahlenfolgen:

		Beschränkt	Monoton	Konvergent	Grenzwert
1	$a_n = 5 + n$	–	+	–	–
2	$a_n = (-1)^n$	+	–	–	–
3	$a_n = (-1)^n \cdot n$	–	–	–	–
4	$a_n = \sqrt{n}$	–	+	–	–
5	$a_n = \frac{1}{n+1}$	+	+	+	0
6	$a_n = \left(1 + \frac{1}{n}\right)^n, n > 0$	+	+	+	$e = 2,7182\dots$ Eulersche Zahl

Berechnung von Grenzwerten:

$$\begin{aligned}
 a_n &= \frac{7n^2 + 3n + 2}{5n^2 + 3} \\
 &= \frac{7 + 3\frac{1}{n} + 2\frac{1}{n^2}}{5 + 3\frac{1}{n^2}} \\
 \lim_{n \rightarrow \infty} \frac{7 + 3/n + 2/n^2}{5 + 3/n^2} &= \frac{7}{5}
 \end{aligned}$$

D Sei $\{n_k\}$ eine streng monoton wachsende Folge mit $n_k \in \mathbb{N}$, dann nennt man $\{a_{n_k}\}$ eine **Teilfolge** von $\{a_n\}$.

Beispiel:

$$\begin{aligned}
 \{n_k\} &= 2, 4, 6, 8, \dots \\
 \{a_n\} &= 1, 4, 9, 16, 25, 36, \dots \\
 \{a_{n_k}\} &= 4, 16, 36, \dots
 \end{aligned}$$

CAS-Beispiel



Mittels der Prozedur **SERIES** des **mathchem**-Moduls lassen sich Folgen bezüglich ihres Grenzwertes untersuchen und graphisch darstellen.

```
from mathchem import SERIES
from sympy.abc import n
```

```
def a(n):
    return (7*n**2+3*n+2)/(5*n**2+3)
```

```
SERIES(a(n), n, 1, 100)
```

```
SERIES(a(n), n, 20, 600)
```

```
SERIES((1+1/n)**n, n, 1, 200)
```

```
SERIES((-1)**n, n, 1, 100)
```

3.2.2 Grenzwert einer Funktion

D Eine Funktion $f(x)$ ($x \in A = D(f) \subseteq \mathbb{R}$) besitzt an einer Stelle x_0 den **Grenzwert** G , falls es zu jedem $\epsilon > 0$ ein $\delta(x_0, \epsilon) > 0$ gibt, so daß für alle x mit $|x - x_0| < \delta(x_0, \epsilon)$ gilt:

$$|f(x) - G| < \epsilon$$

symbolisch:

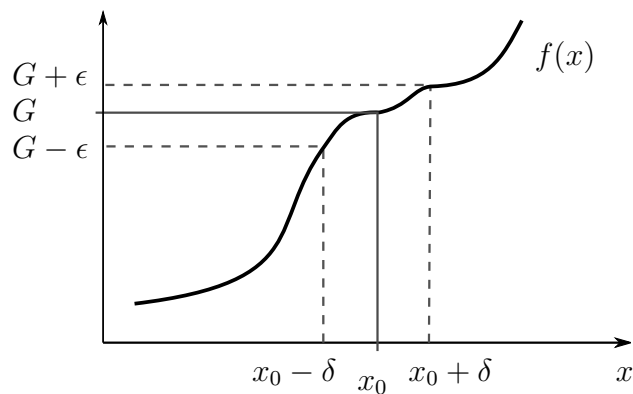
$$\lim_{x \rightarrow x_0} f(x) = G$$

alternativ läßt sich schreiben:

$$\lim_{\delta \rightarrow 0} f(x_0 \pm \delta) = G$$

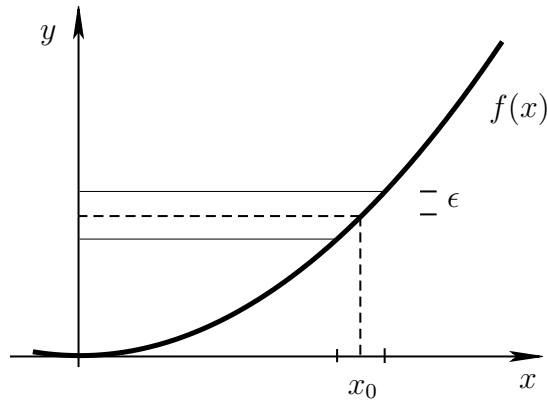
! Die Funktion $f(x)$ muß für $x = x_0$ nicht notwendigerweise definiert sein.

Der “maximale Abstand” δ , den x von x_0 haben darf damit $|f(x) - G| < \epsilon$, hängt in der Regel von ϵ und x_0 ab – daher $\delta(x_0, \epsilon)$.



Beispiel:

$$f(x) = x^2$$



$$\begin{aligned} |f(x_0 + \delta) - G| &< \epsilon \\ |(x_0 + \delta)^2 - x_0^2| &< \epsilon \end{aligned}$$

f ist überall definiert somit $G = f(x_0) = x_0^2$

Annahme: $x_0 > 0$: Wähle $\delta > 0$

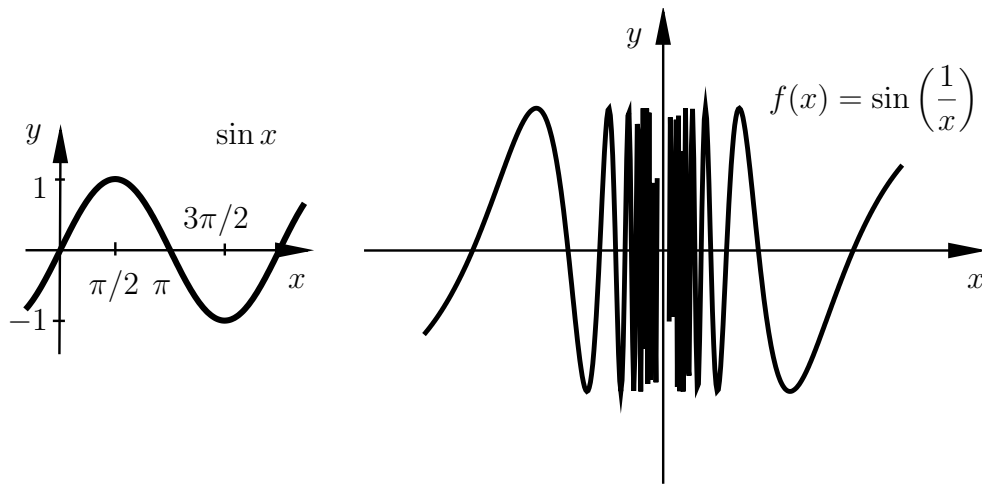
$$\begin{aligned} |(x_0 \pm \delta)^2 - x_0^2| &\stackrel{!}{\leq} (x_0 + \delta)^2 - x_0^2 < \epsilon \\ (x_0 + \delta)^2 &< x_0^2 + \epsilon \\ \delta &< -x_0 + \sqrt{x_0^2 + \epsilon} \\ \delta(x_0, \epsilon) &< x_0 \left(\sqrt{1 + \frac{\epsilon}{x_0^2}} - 1 \right) \end{aligned}$$

d.h. zu jedem ϵ und x_0 ist ein δ angebar, so daß die Grenzwertbedingung erfüllbar ist.

S Eine Funktion $f(x)$ besitzt an der Stelle x_0 genau dann den Grenzwert G , wenn für alle Zahlenfolgen $\{x_n\}$ mit $\lim_{n \rightarrow \infty} x_n = x_0$ gilt $\lim_{n \rightarrow \infty} f(x_n) = G$.

Beispiel:

$$\lim_{x \rightarrow 0} \sin\left(\frac{1}{x}\right) = ?$$



Zahlenfolge:

$$\{x_n\} = \frac{1}{\pi} \quad \frac{1}{2\pi} \quad \frac{1}{3\pi} \quad \frac{1}{4\pi} \quad \dots \quad \lim_{n \rightarrow \infty} x_n = 0 \quad \lim_{n \rightarrow \infty} f(x_n) = 0$$

$$\{x_n\} = \frac{2}{\pi} \quad \frac{2}{5\pi} \quad \frac{2}{9\pi} \quad \frac{2}{13\pi} \quad \dots \quad \lim_{n \rightarrow \infty} x_n = 0 \quad \lim_{n \rightarrow \infty} f(x_n) = +1$$

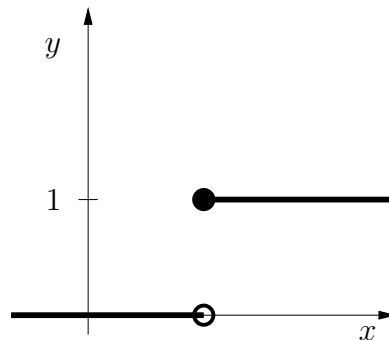
$$\{x_n\} = \frac{2}{3\pi} \quad \frac{2}{7\pi} \quad \frac{2}{11\pi} \quad \frac{2}{15\pi} \quad \dots \quad \lim_{n \rightarrow \infty} x_n = 0 \quad \lim_{n \rightarrow \infty} f(x_n) = -1$$

Verschiedene Zahlenfolgen führen zu unterschiedlichen Grenzwerten! Damit kein Grenzwert für die Funktion selbst.

Beispiel:

Stufenfunktion

$$h(x) = \begin{cases} 1 & x \geq a \\ 0 & x < a \end{cases}$$



$$\{x_n\} = a + \frac{1}{n^3} \quad \lim_{n \rightarrow \infty} x_n = a \quad \lim_{n \rightarrow \infty} h(x_n) = +1$$

$$\{x_n\} = a - \frac{1}{n^3} \quad \lim_{n \rightarrow \infty} x_n = a \quad \lim_{n \rightarrow \infty} h(x_n) = 0$$

Es existiert kein Grenzwert.

D Existiert zu jedem $\epsilon > 0$ ein $\delta(x_0, \epsilon) > 0$ so daß für alle x mit $x_0 < x < x_0 + \delta$ bzw. $x_0 - \delta < x < x_0$ gilt:

$$|f(x) - G| < \epsilon$$

so bezeichnet man G als

$$\begin{array}{ccc} \text{rechts-} & \text{bzw.} & \text{linksseitigen Grenzwert} \\ \lim_{x \rightarrow x_0+0} f(x) = G & & \lim_{x \rightarrow x_0-0} f(x) = G \end{array}$$

alternativ

$$\begin{array}{ccc} \lim_{\delta \rightarrow 0^+} f(x_0 + \delta) = G & & \lim_{\delta \rightarrow 0^+} f(x_0 - \delta) = G \\ \text{oder} & & \lim_{\delta \rightarrow 0^-} f(x_0 + \delta) = G \quad \delta \text{ negativ!} \end{array}$$

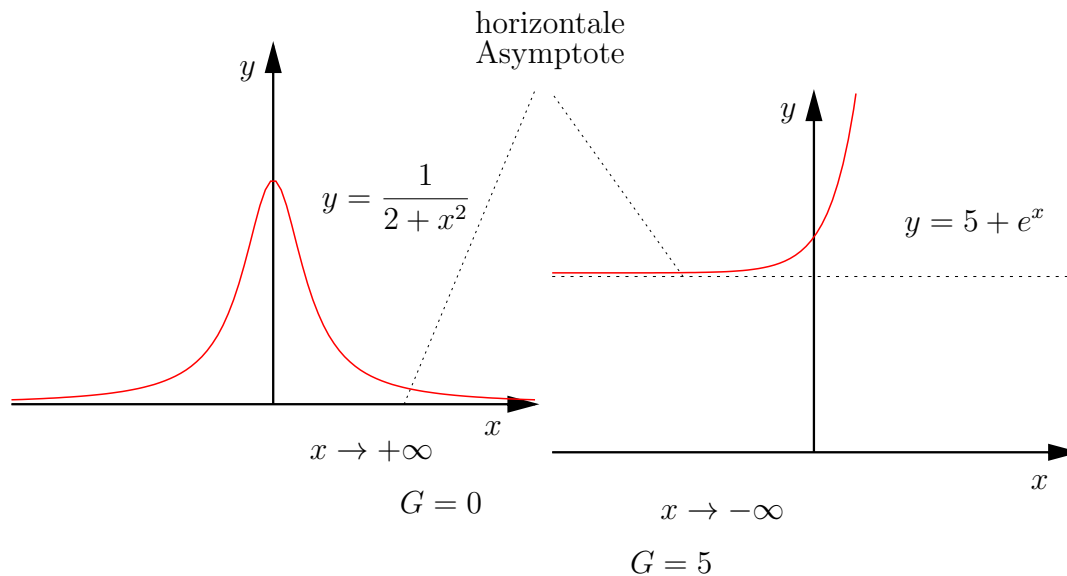
! Damit ein Grenzwert im allgemeinen Sinn existiert muß demnach der links- und rechtsseitige Grenzwert existieren und beide gleich sein.

D Existiert zu jedem $\epsilon > 0$ ein $x_1(\epsilon)$, so daß für alle $x \geq x_1$ bzw. $x \leq x_1$ gilt

$$|f(x) - G| < \epsilon$$

so besitzt f den **Grenzwert G** .

$$\lim_{x \rightarrow \infty} f(x) = G \quad \text{bzw.} \quad \lim_{x \rightarrow -\infty} f(x) = G$$

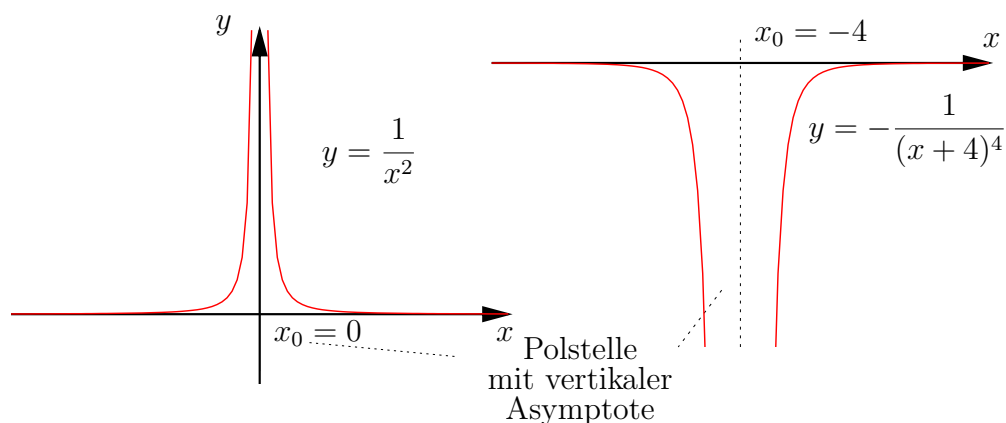


D Existiert zu jedem beliebigen A ein $\delta(A) > 0$, so daß für alle x mit $0 < |x - x_0| < \delta$ gilt

$$f(x) \geq A \quad \text{bzw.} \quad f(x) < A$$

so besitzt $f(x)$ an der Stelle x_0 den **uneigentlichen Grenzwert** $+\infty$ bzw. $-\infty$.

$$\lim_{x \rightarrow x_0} f(x) = \pm\infty$$



D Analoge Definitionen gelten weiterhin für die **uneigentlichen Grenzwerte**

$$\lim_{x \rightarrow \pm\infty} f(x) = \pm\infty \quad \text{bzw.} \quad \lim_{x \rightarrow x_0 \pm 0} f(x) = \pm\infty$$

3.2.3 Methoden der Grenzwertbestimmung

a) Ausnutzen des folgenden Satzes:

S Seien f und g zwei Funktionen mit den Grenzwerten F und G an der Stelle $x = x_0$, so gilt:

$$\begin{aligned} a) \quad \lim_{x \rightarrow x_0} (f \pm g)(x) &= F \pm G \\ b) \quad \lim_{x \rightarrow x_0} (f \cdot g)(x) &= F \cdot G \\ c) \quad \lim_{x \rightarrow x_0} (f/g)(x) &= F/G \text{ falls } G \neq 0 \end{aligned}$$

Beispiel:

Aus $\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$ (siehe unten) und $\lim_{x \rightarrow 0} (x - 2)^2 = 4$ folgt:

$$\begin{aligned} \lim_{x \rightarrow 0} \frac{\sin x}{x(x-2)^2} &= \lim_{x \rightarrow 0} \frac{\sin x}{x} / \lim_{x \rightarrow 0} (x-2)^2 \\ &= 1/4 \end{aligned}$$

b) Umformen der Funktionsterme, d.h. kürzen oder erweitern

$$\begin{aligned} \lim_{x \rightarrow 0} \frac{\sqrt{x+1} - 1}{x} &= \lim_{x \rightarrow 0} \frac{(\sqrt{x+1} - 1)(\sqrt{x+1} + 1)}{x(\sqrt{x+1} + 1)} \\ &= \lim_{x \rightarrow 0} \frac{x + 1 - 1}{x(\sqrt{x+1} + 1)} \\ &= \lim_{x \rightarrow 0} \frac{1}{\sqrt{x+1} + 1} \\ &= \frac{1}{2} \end{aligned}$$

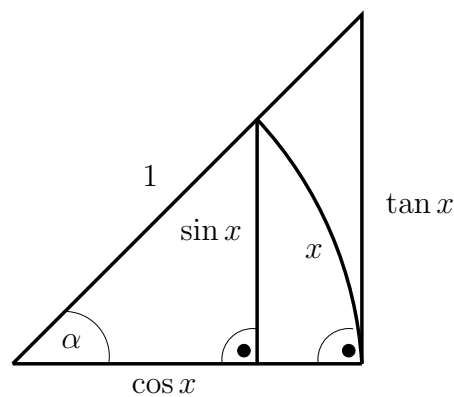
$$\begin{aligned} \lim_{x \rightarrow 0} \frac{\cos x - 1}{x} &= \lim_{x \rightarrow 0} \frac{(\cos x - 1)(\cos x + 1)}{x(\cos x + 1)} \\ &= \lim_{x \rightarrow 0} \frac{-\sin^2 x}{x(\cos x + 1)} \\ &= - \left(\lim_{x \rightarrow 0} \frac{\sin x}{x} \right)^2 \cdot \lim_{x \rightarrow 0} \frac{1}{\cos x + 1} \cdot \lim_{x \rightarrow 0} x \\ &= -1 \cdot \frac{1}{2} \cdot 0 \\ &= 0 \end{aligned}$$

c) Ausnutzen des Satzes:

S Besitzen zwei Funktionen $g(x)$ und $h(x)$ bei x_0 den Grenzwert G und gibt es ein $\delta > 0$ so daß für alle x mit $|x - x_0| < \delta$ $g(x) \leq f(x) \leq h(x)$ gilt, so besitzt auch $f(x)$ den Grenzwert G bei x_0 .

Beispiel:

$$\lim_{x \rightarrow 0} \frac{\sin x}{x}$$



Vergleich der Flächeninhalte führt auf die Relation:

$$\frac{1}{2} \sin x \cos x \leq \pi 1^2 \frac{x}{2\pi} \leq \frac{1}{2} \cdot 1 \cdot \tan x = \frac{1}{2} \frac{\sin x}{\cos x}$$

$$\cos x \leq \frac{x}{\sin x} \leq \frac{1}{\cos x}$$

$$\lim_{x \rightarrow 0} \cos x = 1$$

$$\lim_{x \rightarrow 0} \frac{1}{\cos x} = 1$$

$$\Rightarrow \lim_{x \rightarrow 0} \frac{x}{\sin x} = 1$$

$$\Rightarrow \lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

d) Regel von l'Hospital (s.u.)

e) Reihenentwicklung (s.u.)

Die Sympy-Funktion `limit` erlaubt sowohl die Bestimmung des Grenzwertes von Zahlenfolgen als auch von Funktionen. Im letzten Fall kann insbesondere zusätzlich der links- und rechtsseitige Grenzwert bestimmt werden. Ohne explizite Angabe wird automatisch der rechtsseitige Grenzwert bestimmt (außer für $x \rightarrow +\infty$).

```
from sympy import *
x = symbols('x')

limit(sin(x)/x, x, 0)
```

```
limit(sin(x)/abs(x), x, 0)
```

```
limit(sin(x)/abs(x), x, 0, '+')
```

```
limit(sin(x)/abs(x), x, 0, '-')
```

Man sieht am letzten Beispiel, daß der links- und rechtsseitige Grenzwert durchaus existieren können, auch wenn der eigentliche Grenzwert $\lim_{x \rightarrow 0} \frac{\sin(x)}{|x|}$ nicht existiert.

```
def fn(x):
    return (x**2-1)/(x-1)

x0 = 1

fn(x0)
```

```
limit(fn(x), x, x0, '-')
```

```
limit(fn(x), x, x0, '+')
```

Auch die Grenzwerte $\lim_{x \rightarrow \pm\infty} f(x)$ und Polstellen lassen sich mittels `limit` untersuchen, was allerdings nicht immer zum Erfolg führen muß.

```
limit(1/x, x, 0)
```

```
limit(1/x, x, 0, '-')
```

```
limit(1/x, x, 0, '+')
```

```
limit((1+1/x)**x, x, oo)
```

```
limit((2*x**2+9*x+2)/(5*x**2+3), x, oo)
```

```
limit(1/x, x, oo)
```

```
limit(sin(x)/x, x, oo)
```

3.3 Stetigkeit

3.3.1 Definitionen

D Existiert zu jedem $\epsilon > 0$ ein $\delta(x_0, \epsilon) > 0$, so daß für alle x mit $|x - x_0| < \delta(x_0, \epsilon)$ gilt:

$$|f(x) - f(x_0)| < \epsilon$$

so ist die Funktion $f(x)$ an der Stelle x_0 **stetig**.

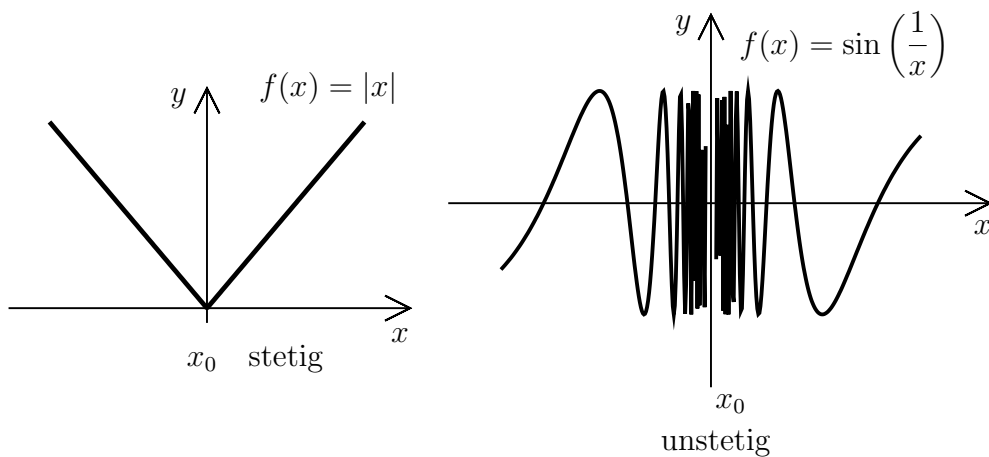
Ist eine Funktion stetig für alle $x \in I = [a, b] \subset D(f)$, so heißt $f(x)$ **stetig auf dem Intervall I** .

Bedeutung: kleine Änderungen in der Variablen x dürfen nur kleine Änderungen der Variablen y bewirken.

! Offensichtlich muß gelten

$$\lim_{x \rightarrow x_0} f(x) = f(x_0) \quad \text{d.h.}$$

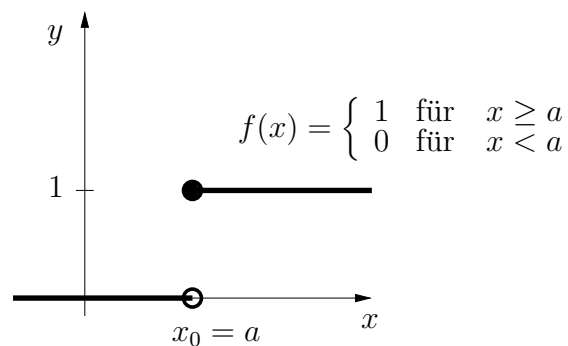
- a) die Funktion muß für x_0 definiert sein
- b) der Grenzwert an der Stelle x_0 muß existieren
- c) Grenz- und Funktionswert bei x_0 müssen übereinstimmen



D Ist für ein x_0 eine Funktion nicht stetig oder **unstetig**, so wird x_0 als **Unstetigkeitsstelle** bezeichnet. Besitzt eine Funktion endlich viele Unstetigkeitsstellen in einem Intervall I , so wird die Funktion als **stückweise stetig** im Intervall I bezeichnet.

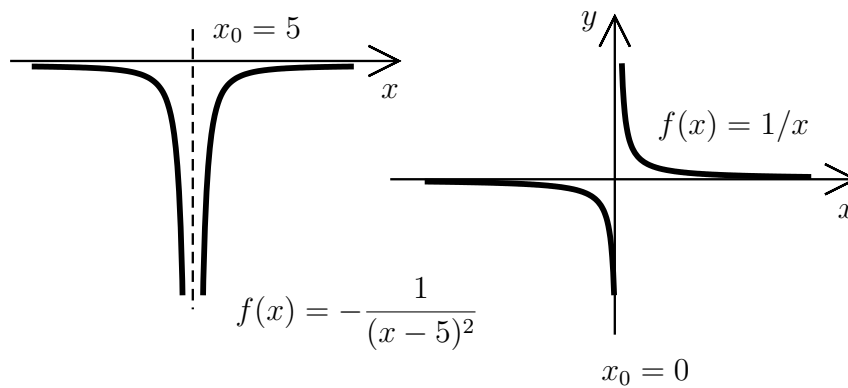
D Existiert ein endlicher links- und rechtsseitiger Grenzwert G_l bzw. G_r mit $G_l \neq G_r$ an einer Unstetigkeitsstelle x_0 , so wird x_0 als **Sprungstelle** bezeichnet, mit einem **Sprung** $G_l - G_r$.

Beispiel:



Die Stufenfunktion ist stückweise stetig mit einem Sprung 1 bei der Sprungstelle $x_0 = a$.

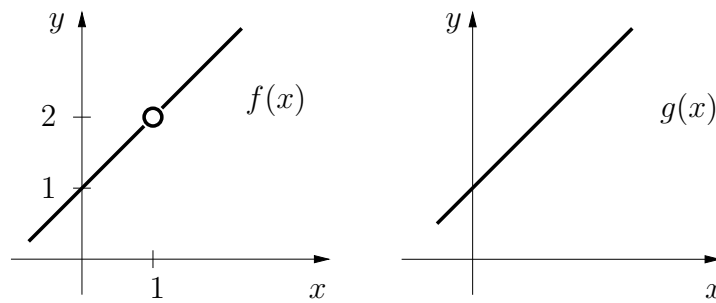
D Besitzt $f(x)$ bei x_0 den Grenzwert $\pm\infty$, so wird x_0 als **Polstelle** bezeichnet. Ändert sich das Vorzeichen der Funktion von $+\infty \rightarrow -\infty$ bzw. umgekehrt, so handelt es sich um einen **Pol mit Vorzeichenwechsel**.



D Existiert an einer Unstetigkeitsstelle x_0 einer Funktion $f(x)$ ein endlicher Grenzwert G , so lässt sich durch Erweiterung der Definition von $f(x)$ durch $f(x_0) = G$ die Unstetigkeit beseitigen. x_0 heißt **hebbare Unstetigkeit**.

Beispiel:

$$\begin{aligned}
 f(x) &= \frac{x^2 - 1}{x - 1} \\
 \lim_{x \rightarrow 1} f(x) &= \lim_{\delta \rightarrow 0} \frac{1 \pm 2\delta + \delta^2 - 1}{1 \pm \delta - 1} \\
 &= \lim_{\delta \rightarrow 0} 2 \pm \delta \\
 &= 2 \\
 &\Downarrow \\
 g(x) &= \begin{cases} 2 & \text{für } x = 1 \\ \frac{x^2 - 1}{x - 1} & \text{sonst} \end{cases}
 \end{aligned}$$



3.3.2 Eigenschaften von stetigen Funktionen

S Addition, Subtraktion, Multiplikation, Division und Verkettung von stetigen Funktionen führt wieder zu stetigen Funktionen:

$$g, f \text{ stetig} \Rightarrow h(x) = (g + f)(x) = g(x) + f(x) \text{ stetig} \\ \text{u. s. w.}$$

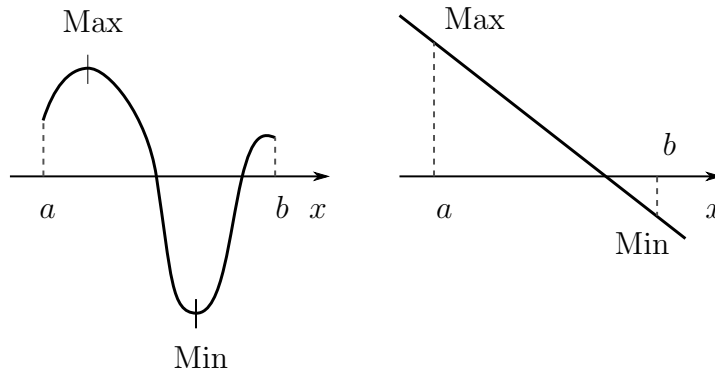
Für die Division f/g muß jedoch gelten daß $g(x) \neq 0$.

D Gibt es für eine (nicht notwendigerweise stetige) Funktion $f(x)$ ein $\delta > 0$ so daß $f(x) - f(x_0) \underset{(\geq)}{\underset{(\leq)}{>}} 0$ (*) für $x \in I = [x_0 - \delta, x_0 + \delta]$, so besitzt $f(x)$ an der Stelle x_0 ein **lokales Minimum** bzw. **Maximum**. Gilt (*) für $I = D(f)$, so spricht man von einem **absoluten Minimum/Maximum**.

$$\text{Symbolisch: } \min_{x \in I} f(x) = f(x_0)$$

S **Satz von Weierstrass**

Für jede in einem abgeschlossenen Intervall $I = [a, b]$ stetige Funktion $f(x)$ existieren ihr Maximum $\max_{x \in I} f(x)$ und ihr Minimum $\min_{x \in I} f(x)$.

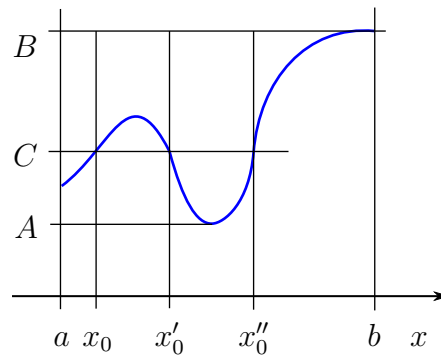


S Zwischenwertsatz

Für jede in einem abgeschlossenen Intervall $I = [a, b]$ stetige Funktion $f(x)$ mit

$$A = \min_{x \in I} f(x) \quad \text{und} \quad B = \max_{x \in I} f(x)$$

existiert zu jedem C mit $A \leq C \leq B$ wenigstens ein x_0 mit $f(x_0) = C$.



3.4 Elementare Funktionen

D **Elementare Funktionen** sind Funktionen die sich durch einen endlichen analytischen Ausdruck darstellen lassen.

algebraisch: $\sum_{i=1}^k a_i x^n y^m = 0$

transzendent: keine algebraische Darstellung möglich

nicht elementar: z.B. Stufenfunktion

3.4.1 Ganze rationale Funktionen

D **Ganze rationale Funktionen** sind Funktionen vom Typ

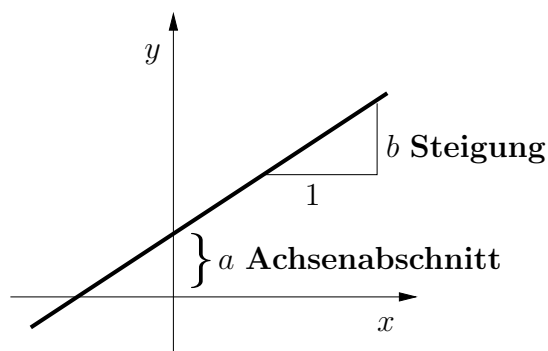
$$y = f(x) = \sum_{i=0}^n a_i x^i \quad a_i \in \mathbb{R}.$$

Die a_i 's heißen **Koeffizienten**, n ist der **Grad der Funktion** bzw. des **Polynoms**

$$\sum_{i=0}^n a_i x^i.$$

Lineare Gleichung:

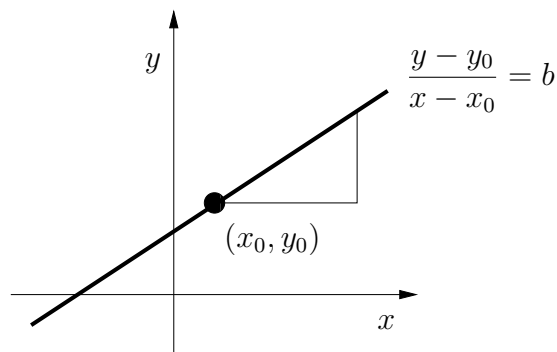
$$y = f(x) = a + bx$$



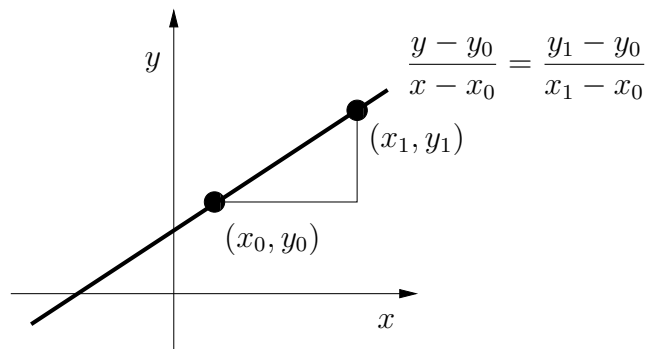
Nullstelle:

$$\begin{aligned} 0 = f(x) &= a + bx \\ \Rightarrow x &= -\frac{a}{b} \quad \text{falls } b \neq 0! \end{aligned}$$

Punktsteigungsform:



Zwei-Punkteform:



CAS-Beispiel



Die Untersuchung einer Geradengleichung auf Nullstelle läßt sich mit Python einfach durchführen (Zur Erinnerung: Es wird vorausgesetzt, daß auf der rechten Seite der Gleichung Null steht):

```
from sympy.solvers import solve
from sympy.abc import a, b, x

solve(a+b*x, x)
```

```
solve(5+2*x, x)
```

```
solve(3-5*x, x)
```

Daß die Punktsteigungsform einer Geraden sich tatsächlich in die übliche Form einer Geradengleichung überführen läßt, kann man zeigen, indem man die Punktsteigungsform nach y auflöst.

```
from sympy import symbols

x, x0, y, y0, b = symbols('x, x0, y, y0, b')

solve((y-y0)/(x-x0) - b, y)
```

Ganz analog läßt sich die Zweipunkteform behandeln:

```

from sympy import collect
x, x0, x1, y, y0, y1 = symbols('x, x0, x1, y, y0, y1')

pl = solve((y-y0)/(x-x0) - (y1-y0)/(x1-x0), y)
h = pl[0].expand()
collect(h, x)

```

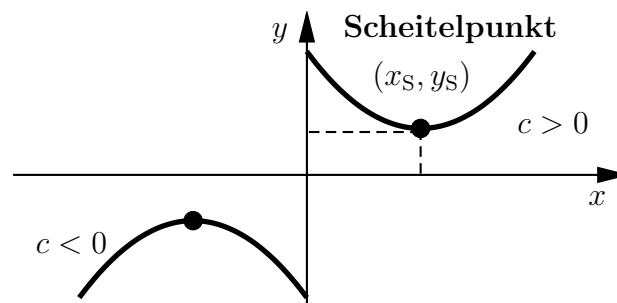
Zunächst wird die Gleichung

$$\frac{y - y_0}{x - x_0} = \frac{y_1 - y_0}{x_1 - x_0}$$

nach y aufgelöst und das Ergebnis in einer Liste der Länge 1 namens `pl` abgespeichert. Die Terme der Lösung werden mittels `expand` so weit wie möglich ausmultipliziert, anschließend wird durch die Anweisung `collect(h, x)` die Variable x ausgeklammert.

Parabel:

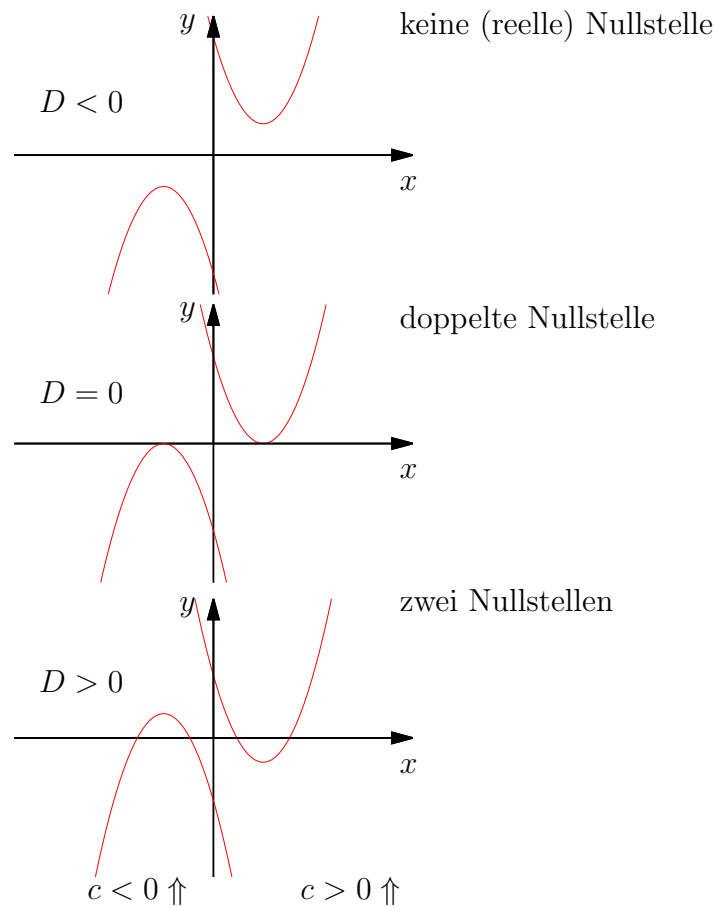
$$\begin{aligned}
y &= a + bx + cx^2 \\
&= c \left[\frac{a}{c} + \frac{b}{c}x + x^2 \right] \\
&= c \left[\frac{a}{c} - \left(\frac{b}{2c} \right)^2 + \left(\frac{b}{2c} \right)^2 + 2 \frac{b}{2c}x + x^2 \right] \\
&= a - \frac{b^2}{4c} + c \left(\frac{b}{2c} + x \right)^2 \\
x_S &= -\frac{b}{2c} \\
y_S &= a - \frac{b^2}{4c}
\end{aligned}$$



Nullstelle:

$$\begin{aligned}
0 = f(x) &= a + bx + cx^2 \\
x_{1,2} &= \frac{-b \pm \sqrt{b^2 - 4ca}}{2c}
\end{aligned}$$

$$\text{Diskriminante } D = b^2 - 4ca$$



S Fundamentalsatz der Algebra:

Jede ganze rationale Funktion n -ten Grades mit reellen Koeffizienten a_i lässt sich auf die Form bringen:

$$\sum_{i=0}^n a_i x^i = c(x - x_1)(x - x_2) \dots (x - x_k)(x^2 + p_1x + q_1) \dots (x^2 + p_lx + q_l)$$

wobei gilt: $k + 2l = n$ und $c, x_i, q_i, p_i \in \mathbb{R}$.

z. B.

$$a_0 + a_1x + a_2x^2 = a_2 \left[x - \frac{-a_1 + \sqrt{a_1^2 - 4a_0a_2}}{2a_2} \right] \left[x - \frac{-a_1 - \sqrt{a_1^2 - 4a_0a_2}}{2a_2} \right]$$

d. h.

$$c = a_2$$

$$x_{1,2} = \frac{-a_1 \pm \sqrt{a_1^2 - 4a_0a_2}}{2a_2}$$

D Die Faktoren $(x - x_i)$ werden als **Linearfaktoren** bezeichnet – sie legen die **Nullstellen** x_i der Funktion fest. Tritt ein x_i mehrfach (m -fach) auf, so spricht man von **m -facher Nullstelle**.

CAS-Beispiel



Die Nullstellensuche bei einer Parabel kann ebenfalls mit der Funktion `solve` vorgenommen werden:

```
from sympy.solvers import solve
from sympy.abc import a, b, c, x

solve(a+b*x+c*x**2, x)
```

Rationale Funktionen sind durch ein Polynom in x festgelegt. Da Polynome in vielen Bereichen eine wichtige Rolle spielen, stellt Python eine Reihe von Funktionen zur Verfügung, mit denen sich viele Standardaufgaben lösen lassen (siehe unten).

Mit der Funktion `expand` lassen sich beliebige Produkte und Summen in ein Polynom der Form $\sum_{i=0}^n a_i x^i$ umformen:

```
def p(x):
    return a*(x-3)*c - x**3*11 + x**2*(2-x)*(4+x)

p(x).expand()
```

Mittels `degree` läßt sich der Grad des Polynoms abfragen.

```
from sympy import degree

degree(p(x))
```

Die Koeffizienten aller Potenzen in x bekommt man mittels `all_coeffs` aufgelistet.

```
from sympy import Poly

p = Poly(p(x), x)
p.all_coeffs()
```

Um ein Polynom für ein bestimmtes x auszuwerten, setzt man einfach in das definierte Polynom ein:

```
p(2)
```

Entsprechend erklärt sich folgendes:

```
import numpy as np
import matplotlib.pyplot as plt
```

```
def p(x):
    return x**4 + 3*x**2 + 5

u = np.linspace(-4, 4)
plt.plot(u, p(u))
plt.show()
```

3.4.2 Gebrochen rationale Funktionen

D Funktionen vom Typ

$$f(x) = \frac{\sum_{i=0}^m a_i x^i}{\sum_{i=0}^n b_i x^i}$$

heißen **gebrochen rationale Funktionen**.

Falls $m \geq n$ **unechte** gebrochen rationale Funktion
bzw. $m < n$ **echte** gebrochen rationale Funktion

Alle Nullstellen der Funktion im Nenner, die nicht gleichzeitig Nullstellen der Funktion im Zähler sind, führen zu Singularitäten d.h. Polen. Handelt es sich um eine k -fache Nullstelle, so spricht man von **Pol k -ter Ordnung**.

Beispiel:

$$\begin{aligned} f(x) &= \frac{3 + 5x}{25 - 10x + x^2} \\ &= \frac{3 + 5x}{(5 - x)(5 - x)} \\ &= \frac{3 + 5x}{(5 - x)^2} \end{aligned}$$

$x_0 = 5$ ist 2-fache Polstelle.

CAS-Beispiel



Vollkommen entsprechend zur Untersuchung der Nullstellen einer ganzen rationalen Funktion lassen sich die Null- und Polstellen einer gebrochen rationalen Funktion untersuchen. Die Nullstellen lassen sich direkt über die Funktion `solve` bestimmen:

```

from sympy.solvers import solve
from sympy.abc import x

def f(x):
    return (4+5*x)/(4+5*x-7*x**2)

solve(f(x), x)

```

Zur Bestimmung der Polstellen wird der Nenner mit `fraction` gleich Null gesetzt und nach x aufgelöst. Im folgenden Beispiel wird noch bestimmt, ob es sich um einen Pol mit oder ohne Vorzeichenwechsel oder um eine hebbare Unstetigkeitsstelle wie bei $f(x) = \frac{x-1}{x^2-1}$ handelt.

```

from sympy import fraction
from sympy import limit

zaehler, nenner = fraction(f(x))

pol = solve(nenner, x)

for i in pol:
    ll = limit(f(x), x, i, '-')
    ul = limit(f(x), x, i, '+')
    if (ll != ul):
        if ((ll > 0 and ul > 0) or (ll < 0 and ul < 0)):
            print("Pol ohne Vorzeichenwechsel bei x = " + str(i))
        else:
            print("Pol mit Vorzeichenwechsel bei x = " + str(i))
    else:
        print("Hebbare Unstetigkeit bei x = " + str(i))

```

Diese Prozedur kann auch direkt aus dem `mathchem`-Modul aufgerufen werden (dieses muß ggf. zur Verfügung gestellt werden):

```

def g(t):
    return (t-1)/(t**2-1)

```

```

from mathchem import POLE
from sympy.abc import t

POLE(g(t), t)

```

3.4.3 Potenz- und Wurzelfunktionen

S *Regeln zum Potenzrechnen*

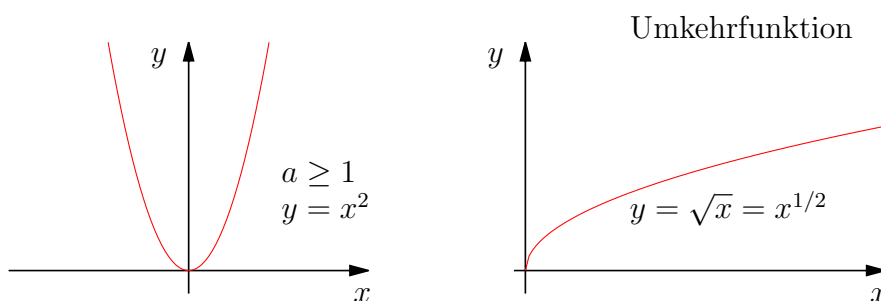
$$x^u \cdot x^v = x^{u+v}$$

$$\begin{aligned}
 (x^u)^v &= x^{u \cdot v} \\
 x^u \cdot y^u &= (x \cdot y)^u \\
 \text{mit } x^n &= 1 \cdot \underbrace{x \cdot x \cdot \dots \cdot x}_{n\text{-mal}} \quad \text{für } n \in \mathbb{N} \\
 x^0 &= 1 \\
 x^{-u} &= 1/x^u \\
 x^{1/n} &= \sqrt[n]{x}
 \end{aligned}$$

D Funktionen der Form

$$f(x) = x^a \quad \text{mit } a \in \mathbb{R}$$

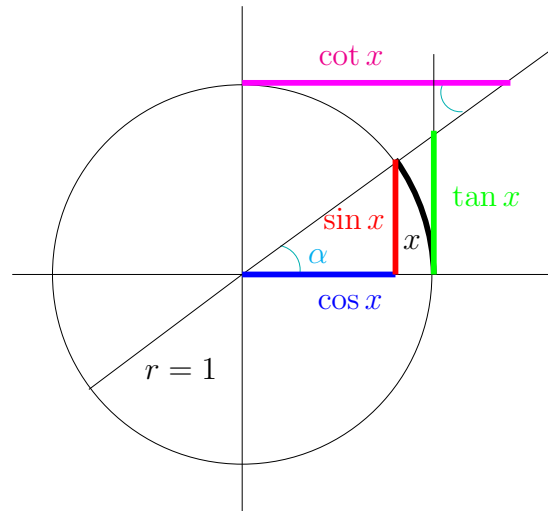
heißen **Potenzfunktionen**, a heißt **Exponent**.



D Die Umkehrfunktion der Potenzfunktion heißt **Wurzelfunktion**.

3.4.4 Trigonometrische Funktionen

D Die **trigonometrischen bzw. Kreisfunktionen** lassen sich anhand des **Einheitskreises** wie folgt definieren, wobei das Argument x der Bogenlänge auf dem Einheitskreis entspricht.



Einheitskreis:

$$\begin{aligned} r &= 1 \\ u &= 2\pi r \\ &= 2\pi \end{aligned}$$

Bogenlänge x : Maß für Öffnungswinkel α eines Kreissegmentes.

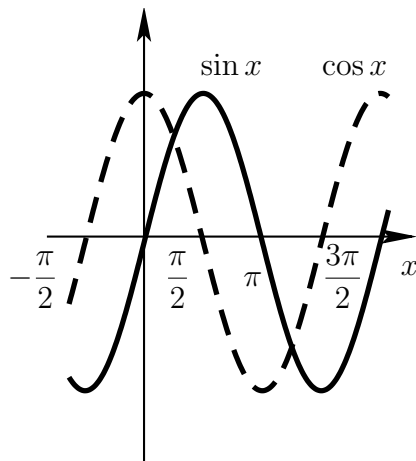
Aus der Einheitskreiskonstruktion lassen sich folgende Eigenschaften ablesen:

$$\cos^2 x + \sin^2 x = 1 \quad (\text{Satz des Pythagoras})$$

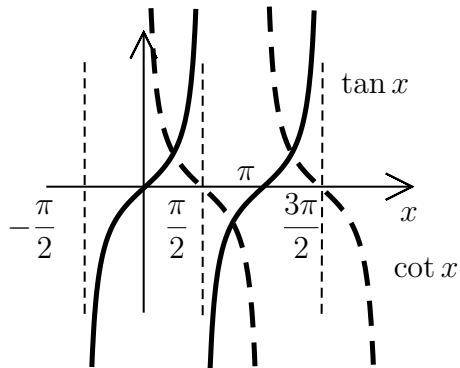
$$\tan x = \frac{\sin x}{\cos x} \quad (\text{ähnliche Dreiecke})$$

$$\cot x = \frac{\cos x}{\sin x} = \frac{1}{\tan x}$$

Die obige Definition gilt zunächst nur für $0 \leq x \leq \pi/2$. Durch Berücksichtigung der Ausrichtung der Strecken mit Länge $\sin x$, ... läßt sich der Definitionsbereich auf $0 \leq x \leq 2\pi$ ausdehnen. Die entsprechende Erweiterung auf $-\infty < x < +\infty$ erfolgt in analoger Weise.



$$\begin{aligned} \sin(x) &= \cos\left(x - \frac{\pi}{2}\right) = -\cos\left(x + \frac{\pi}{2}\right) \\ \cos(x) &= \sin\left(x + \frac{\pi}{2}\right) = -\sin\left(x - \frac{\pi}{2}\right) \end{aligned}$$



$$\tan(x) = -\cot\left(x + \frac{\pi}{2}\right)$$

$$\cot(x) = -\tan\left(x - \frac{\pi}{2}\right)$$

	sin	cos	tan	cot
Symmetrie	u	g	u	u
Periode	2π	2π	π	π
Nullstellen	$n \cdot \pi$	$(n + \frac{1}{2}) \cdot \pi$	$n \cdot \pi$	$(n + \frac{1}{2}) \cdot \pi$
Polstellen	—	—	$(n + \frac{1}{2}) \cdot \pi$	$n \cdot \pi$
Beschränktheit	+	+	—	—
Maxima	$\pi/2 + n \cdot 2\pi$	$n \cdot 2\pi$	—	—
Minima	$-\pi/2 + n \cdot 2\pi$	$\pi + n \cdot 2\pi$	—	—

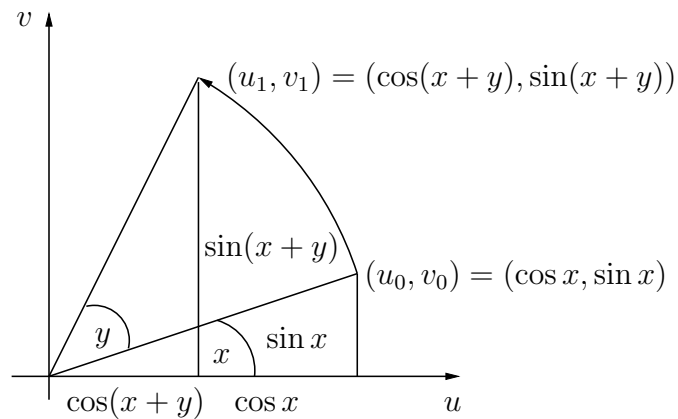
S Additionstheorem

Für alle $x, y \in \mathbb{R}$ gilt:

$$\cos(x + y) = \cos x \cos y - \sin x \sin y$$

$$\sin(x + y) = \sin x \cos y + \cos x \sin y$$

B



Darstellung in polaren Koordinaten

$$\begin{aligned} \begin{pmatrix} u_1 \\ v_1 \end{pmatrix} &= \begin{pmatrix} \cos y & -\sin y \\ \sin y & \cos y \end{pmatrix} \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} \\ &= \begin{pmatrix} u_0 \cos y - v_0 \sin y \\ u_0 \sin y + v_0 \cos y \end{pmatrix} \\ \begin{pmatrix} \cos(x+y) \\ \sin(x+y) \end{pmatrix} &= \begin{pmatrix} \cos x \cos y - \sin x \sin y \\ \cos x \sin y + \sin x \cos y \end{pmatrix} \end{aligned}$$

Mittels des Additionstheorems lassen sich viele weitere nützliche Beziehungen ableiten (\rightarrow Formelsammlung).

Beispiel:

$$\begin{aligned} \cos 2x &= \cos x \cos x - \sin x \sin x \\ &= \cos^2 x - \sin^2 x \\ &= \cos^2 x - 1 + \cos^2 x \\ &= 2 \cos^2 x - 1 \end{aligned}$$

bzw.

$$\cos^2 x = \frac{1}{2} (\cos 2x + 1)$$

CAS-Beispiel



Umformungen von Ausdrücken, die \cos und/oder \sin enthalten, lassen sich wie gewohnt durchführen.

```
from sympy import sin, cos, simplify
from sympy.abc import x, y

simplify(cos(x)*cos(y) - sin(x)*sin(y))
```

```
simplify(sin(x)*cos(y) + cos(x)*sin(y))
```

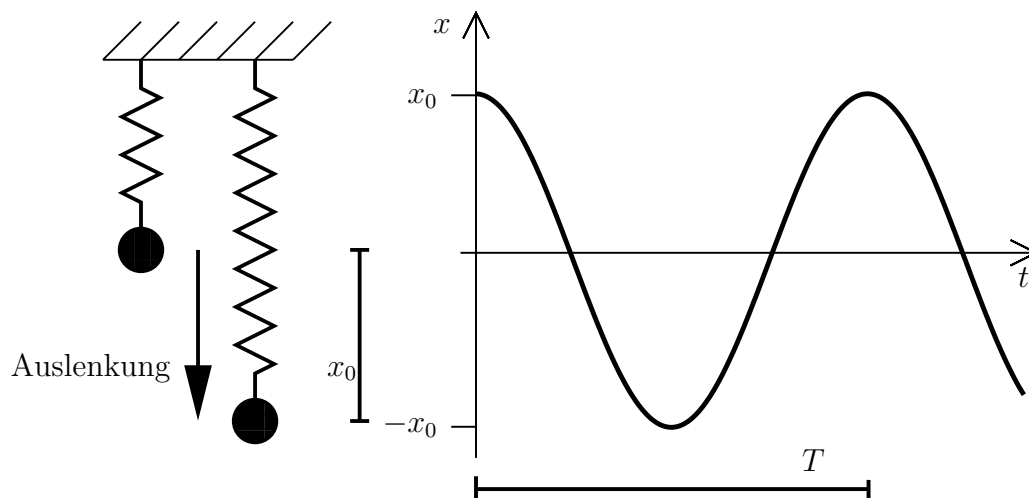
Von der Korrektheit der Beziehung $\cos^2(x) = \frac{1}{2}(\cos(2x) + 1)$ kann man sich leicht graphisch überzeugen:

```
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(-1, 6.5)
plt.plot(x, np.cos(x), label = r"$\cos(x)$")
plt.plot(x, np.cos(2*x), label = r"$\cos(2x)$")
plt.plot(x, np.cos(2*x)+1, label = r"$\cos(2x)+1$")
plt.plot(x, 0.5*(np.cos(2*x)+1), label = r"$\frac{1}{2}(\cos(2x)+1)$")
plt.plot(x, np.cos(x)**2, label = r"$\cos^2(x)$")
plt.legend()
plt.show()
```

Anwendungen:

Viele räumlich oder zeitlich periodische Erscheinungen lassen sich durch die sin- bzw. cos-Funktion darstellen.

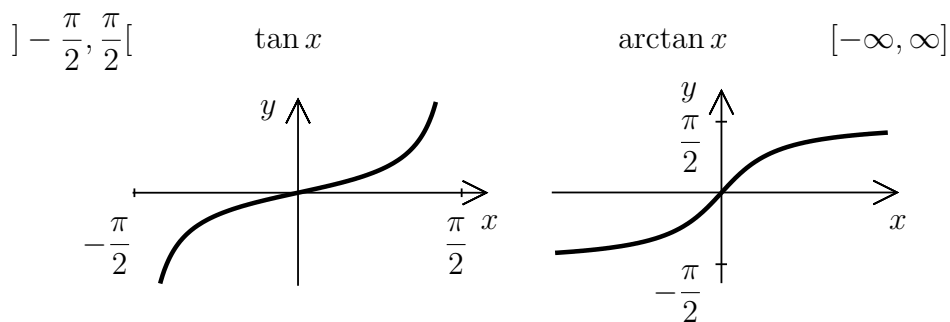
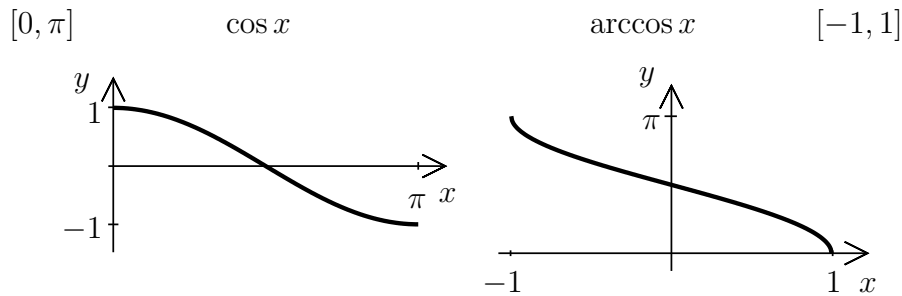
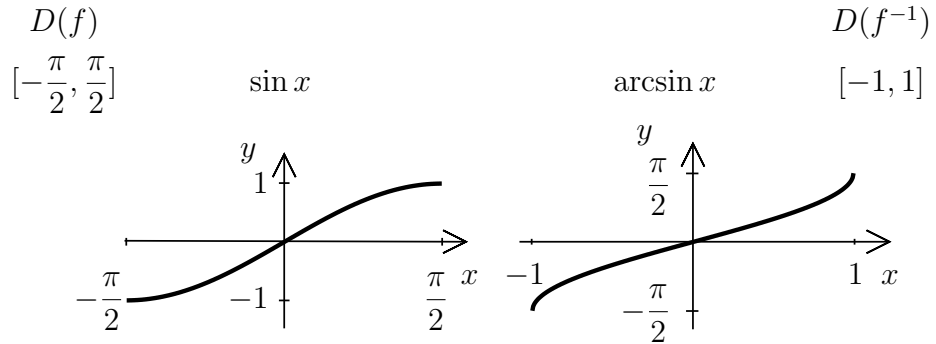


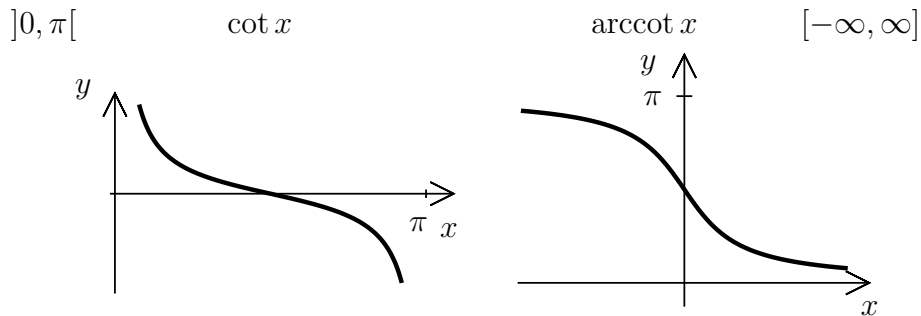
Schwingungsdauer	T
Frequenz	$f = 1/T$
Kreisfrequenz oder Winkelgeschwindigkeit	$\omega = 2\pi f$
Phase	Φ

$$\begin{aligned}
 x(t) &= x_0 \sin\left(2\pi \frac{t}{T} + \Phi\right) \\
 &= x_0 \sin(2\pi f t + \Phi) \\
 &= x_0 \sin(\omega t + \Phi)
 \end{aligned}$$

Umkehrfunktionen:

Die Umkehrfunktionen der trigonometrischen Funktionen heißen **zyklometrische** oder **Arcusfunktionen**. Sie lassen sich nur einführen falls der Definitionsbereich der trigonometrischen Funktionen eingeschränkt wird.



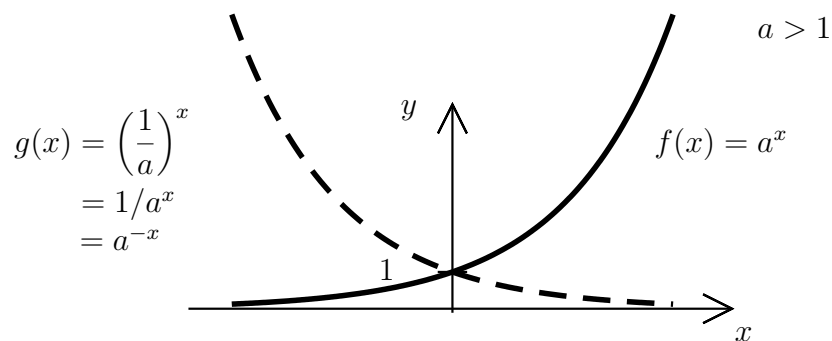


3.4.5 Exponential- und Logarithmusfunktion

D Eine Funktion der Form

$$y = a^x \quad a \in \mathbb{R}^+$$

heißt **Exponentialfunktion zur Basis a** . Falls die Basis nicht explizit angegeben wird, geht man stillschweigend von $a = e (=2.718\dots \text{Eulersche Zahl})$ aus.

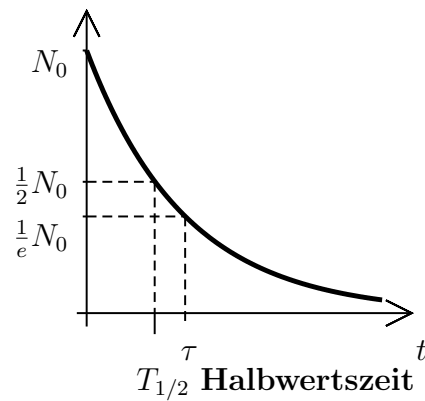


Anwendungen:

- Radioaktiver Zerfall

$$N(t) = N_0 e^{-t/\tau}$$

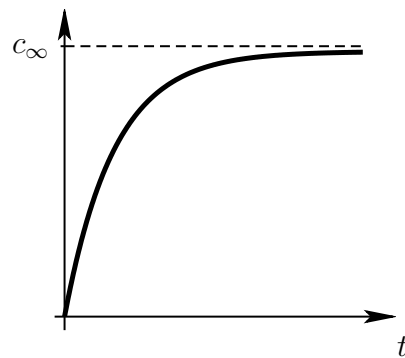
τ : Relaxationszeit (Zerfallszeit)



- chemische Reaktion 1. Ordnung (bzw. Relaxation)

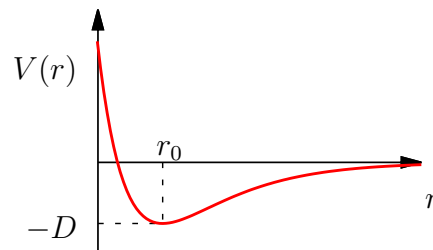
$$c(t) = c_{\infty}(1 - e^{-kt})$$

k : Geschwindigkeitskonstante



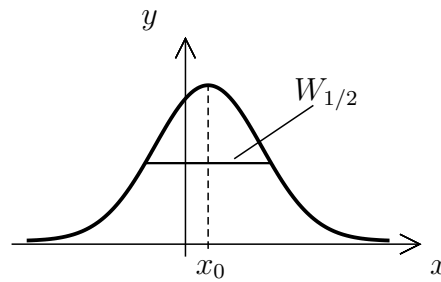
- Morse-Potential

$$V(r) = D[1 - e^{-a(r-r_0)}]^2 - D$$



- Gauß-Funktion ("Glockenkurve")
 - Statistik
 - Quantenmechanik

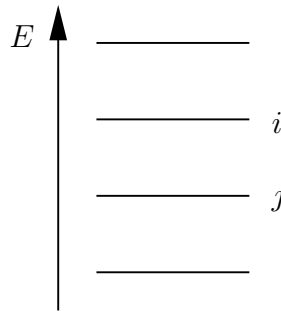
$$y = e^{-\left(\frac{x-x_0}{a}\right)^2}$$



Halbwertsbreite: $W_{1/2} = 2a \sqrt{\ln 2}$

• Boltzmann Verteilung

$$\frac{p_i}{p_j} = \frac{g_i}{g_j} e^{-\frac{E_i - E_j}{kT}}$$



CAS-Beispiel



Plotten Sie die Funktionen:

$$c(t) = c_\infty (1 - e^{-kt})$$

$$V(r) = D [1 - e^{-a(r-r_0)}]^2$$

$$y(x) = e^{-\left(\frac{x-x_0}{a}\right)^2} \text{ und}$$

$$p(T) = e^{-\frac{\Delta E}{kT}}$$

für verschiedene Parameter c_∞ , k , D u.s.w. Ändern Sie dazu die folgenden Beispiele entsprechend ab.

```
import numpy as np
import matplotlib.pyplot as plt

def c(t):
    return 10*(1-np.exp(-1/2*t))

t = np.linspace(0, 20)
plt.plot(t, c(t))
plt.show()
```

```
def V(r):
    return 2*(1-np.exp(-0.9*(r-1)))**2-2
```



```
r = np.linspace(0, 10)
plt.plot(r, V(r))
plt.show()
```

```
def y(x):
    return np.exp(-((x-2)/0.5)**2)

x = np.linspace(0, 4)
plt.plot(x, y(x))
plt.show()
```

```
def p(T):
    return np.exp(-1.0e-21/(1.38e-23*T))

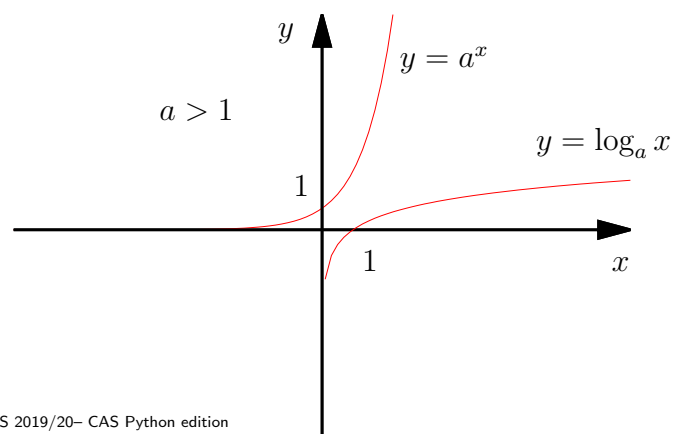
T = np.linspace(1, 600, 100)
plt.plot(T, p(T))
plt.show()
```

Umkehrfunktion:

D Die Umkehrfunktion zur Exponentialfunktion $y = f(x) = a^x$ heißt **Logarithmusfunktion** $y = \log_a x$ zur **Basis a** .

Für $a = 10$: $\log_{10} x = \lg x$ **dekadischer Logarithmus**
 Briggscher Logarithmus
 Für $a = e$: $\log_e x = \ln x$ **natürlicher Logarithmus**
logarithmus naturalis

$u(v) = v^w$ Potenzfunktion
 $v(u) = u^{1/w}$ Wurzelfunktion
 $u(w) = v^w$ Exponentialfunktion
 $w(u) = \log_v u$ Logarithmusfunktion



S Rechenregeln

$$\log_a(x \cdot y) = \log_a x + \log_a y$$

$$\log_a(x/y) = \log_a x - \log_a y$$

$$\log_a x^y = y \log_a x$$

Anwendungen:

- Rechenschieber:

Abbildung von Multiplikation und Division auf Addition und Subtraktion.

- pH-Wert:

Negativer dekadischer Logarithmus der Hydroniumionenkonzentration $[\text{H}_3\text{O}^+]$ in $\left[\frac{\text{mol}}{\text{l}}\right]$.

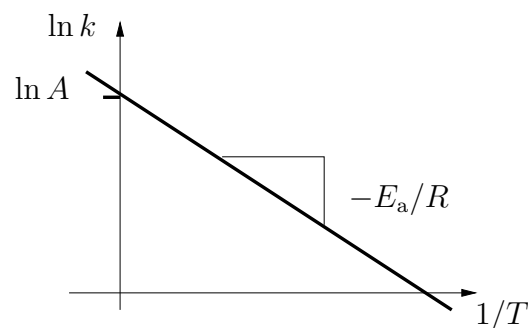
- Arrhenius-Plot:

Die Reaktionsrate vieler chemischen Reaktionen hängt über $k = A \cdot e^{-E_a/RT}$ von der Temperatur ab. Logarithmische Auftragung liefert Vorfaktor A und Aktivierungsenergie E_a .

$$\ln k = \ln A - E_a/RT$$

$$\ln k \left(\frac{1}{T}\right) = \ln A - E_a/R \cdot \left(\frac{1}{T}\right)$$

$$y(x) = y_0 + m \cdot x$$



Arrhenius-Plot

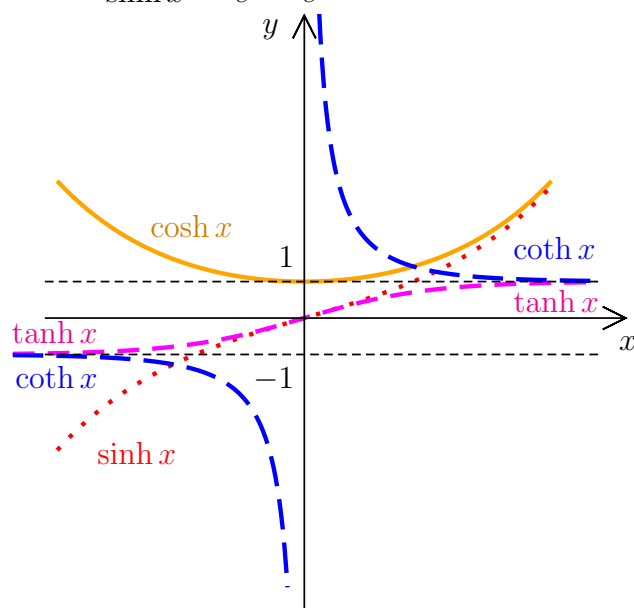
Zusammengesetzte Funktionen:

$$f(x) = \sinh x = \frac{1}{2}(e^x - e^{-x}) \quad \text{Sinushyperbolicus}$$

$$f(x) = \cosh x = \frac{1}{2}(e^x + e^{-x}) \quad \text{Cosinushyperbolicus}$$

$$f(x) = \tanh x = \frac{\sinh x}{\cosh x} = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad \text{Tangenshyperbolicus}$$

$$f(x) = \coth x = \frac{\cosh x}{\sinh x} = \frac{e^x + e^{-x}}{e^x - e^{-x}} \quad \text{Cotangenshyperbolicus}$$



$\cosh x \rightarrow$ Kettenlinie

$\tanh x \rightarrow$ Magnetismus, NMR

Umkehrfunktionen: Areafunktionen

$\operatorname{arsinh} x$

$\operatorname{arcosh} x$

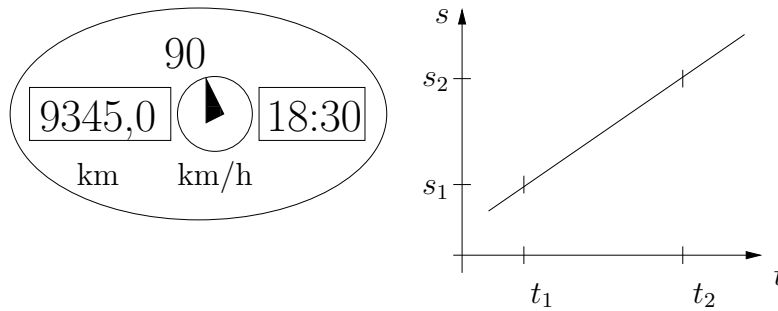
$\operatorname{artanh} x$

$\operatorname{arcoth} x$

3.5 Differentialrechnung

3.5.1 Definitionen und geometrische Deutung

Geschwindigkeit bei gleichförmiger Bewegung:



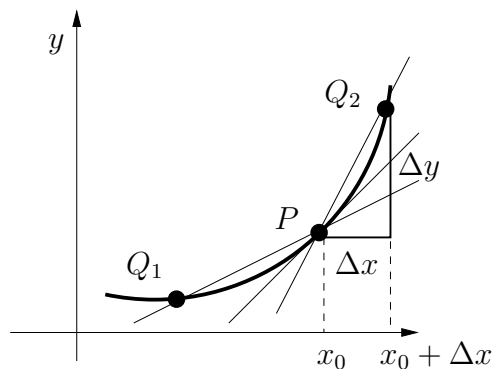
$$\begin{array}{rcl} s_2 = 9345 \text{ km} & t_2 = 18 : 30 & \\ s_1 = 9315 \text{ km} & t_1 = 18 : 10 & \\ \hline \Delta s = 30 \text{ km} & \Delta t = 20 \text{ min} & \end{array}$$

$$\begin{aligned} \frac{\Delta s}{\Delta t} &= \frac{30 \text{ km}}{20 \text{ min}} = 90 \text{ km/h} = v \quad \text{Geschwindigkeit} \\ s(t) &= s_1 + v(t - t_1) \\ &= (s_1 - v \cdot t_1) + v \cdot t \end{aligned}$$

Tangentenproblem (momentane Geschwindigkeit):

Steigung der Sekante

$$\underbrace{\frac{f(x_0 + \Delta x) - f(x_0)}{(x_0 + \Delta x) - x_0}}_{\text{Differenzenquotient}} = \frac{\Delta y}{\Delta x}$$



Nähern sich Q_1 und Q_2 immer mehr an P an, so sollten beide Sekanten am Ende mit der Tangente zusammenfallen.

D Existiert für eine Funktion $f(x)$ in $x_0 \in D(f)$ der Grenzwert des **Differenzenquotient** $\frac{\Delta f}{\Delta x}$

$$\lim_{x \rightarrow x_0} \frac{f(x) - f(x_0)}{x - x_0} = \lim_{\Delta x \rightarrow 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}$$

so sagt man: $f(x)$ ist in x_0 differenzierbar.

Der Grenzwert heißt **1. Ableitung von $f(x)$ in x_0** :

$$\frac{df(x_0)}{dx} = \left. \frac{df}{dx} \right|_{x=x_0} = f'(x_0) = \lim_{x \rightarrow x_0} \frac{f(x) - f(x_0)}{x - x_0}$$

$\frac{df}{dx}$ wird als **Differentialquotient** bezeichnet.

Falls $x = t$, d.h. Zeit, so schreibt man oft \dot{f} statt f' .

Ist $f(x)$ differenzierbar für alle $x \in I \subset D(f)$, so heißt $f(x)$ **im Intervall I differenzierbar**.

Existiert für eine Funktion $f(x)$ in $x_0 \in D(f)$ der Grenzwert

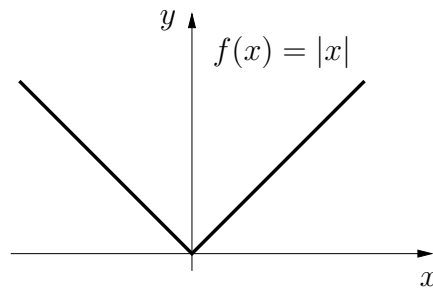
$$f'_r(x_0) = \lim_{x \rightarrow x_0+0} \frac{f(x) - f(x_0)}{x - x_0} \quad \text{bzw.} \quad f'_l(x_0) = \lim_{x \rightarrow x_0-0} \frac{f(x) - f(x_0)}{x - x_0}$$

so sagt man: $f(x)$ ist in x_0

	rechts-	bzw.	linksseitig differenzierbar
	$f'_r(x_0)$	bzw.	$f'_l(x_0)$
heißen	rechts-	bzw.	linksseitige Ableitung

! Damit eine Funktion in x_0 differenzierbar ist, müssen offensichtlich rechts- und linksseitige Ableitungen existieren und gleich sein.

Beispiel:



$$f'_r(x_0 = 0) = +1$$

$$f'_l(x_0 = 0) = -1$$

$f'(x_0)$ existiert nicht!

3.5.2 Ableitungen elementarer Funktionen

- Ableitung der Potenzfunktion $f(x) = x^n$ $n \in \mathbb{N} \setminus \{0\}$

$$\begin{aligned} f'(x) &= \lim_{\Delta x \rightarrow 0} \frac{f(x_0 + \Delta x) - f(x)}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \frac{(x + \Delta x)^n - x^n}{\Delta x} \end{aligned}$$

Binomische Formel

$$(x + \Delta x)^n = \sum_{i=0}^n \binom{n}{i} x^{n-i} \Delta x^i$$

mit $\binom{n}{i} = \frac{n!}{i!(n-i)!}$ **Binominalkoeffizient**

$$\begin{aligned} (x + \Delta x)^n &= x^n + \frac{n!}{(n-1)!} x^{n-1} \Delta x + \frac{n!}{2!(n-2)!} x^{n-2} \Delta x^2 + \dots + \Delta x^n \\ &= x^n + nx^{n-1} \Delta x + \frac{n(n-1)}{2} x^{n-2} \Delta x^2 + \dots + \Delta x^n \end{aligned}$$

$$\begin{aligned} f'(x) &= \lim_{\Delta x \rightarrow 0} \left(nx^{n-1} + \frac{n(n-1)}{2} x^{n-2} \Delta x + \dots + \Delta x^{n-1} \right) \\ &= \lim_{\Delta x \rightarrow 0} nx^{n-1} \end{aligned}$$

$$\frac{d}{dx} x^n = nx^{n-1}$$

- Ableitung der Sinusfunktion $\sin x$

$$\begin{aligned} f'(x) &= \lim_{\Delta x \rightarrow 0} \frac{\sin(x + \Delta x) - \sin x}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \frac{\sin x \cos \Delta x + \cos x \sin \Delta x - \sin x}{\Delta x} \end{aligned}$$

$$\begin{aligned}
 &= \sin x \lim_{\Delta x \rightarrow 0} \frac{\cos \Delta x - 1}{\Delta x} + \cos x \underbrace{\lim_{\Delta x \rightarrow 0} \frac{\sin \Delta x}{\Delta x}}_{=1} \\
 &= \sin x \lim_{\Delta x \rightarrow 0} \left[\frac{\cos \Delta x - 1}{\Delta x} \cdot \frac{\cos \Delta x + 1}{\cos \Delta x + 1} \right] + \cos x \\
 &= \sin x \lim_{\Delta x \rightarrow 0} \frac{\cos^2 \Delta x - 1}{\Delta x (\cos \Delta x + 1)} + \cos x \\
 &= \sin x \lim_{\Delta x \rightarrow 0} \left[-\frac{\sin^2 \Delta x}{\Delta x^2} \cdot \frac{\Delta x}{(\cos \Delta x + 1)} \right] + \cos x \\
 &= \sin x (-1) \cdot 1^2 \lim_{\Delta x \rightarrow 0} \frac{\Delta x}{(\cos \Delta x + 1)} + \cos x \\
 &= -\sin x \cdot \frac{\lim_{\Delta x \rightarrow 0} \Delta x}{\lim_{\Delta x \rightarrow 0} (\cos \Delta x + 1)} + \cos x \\
 &= -\sin x \cdot \frac{0}{2} + \cos x
 \end{aligned}$$

$$\frac{d}{dx} \sin x = \cos x$$

$f(x)$	$f'(x)$	$f(x)$	$f'(x)$
c	0	$\cot x$	$-\frac{1}{\sin^2 x}$
x^a	$ax^{a-1} \quad a \in \mathbb{R} \setminus \{0\}$	$\arcsin x$	$\frac{1}{\sqrt{1-x^2}}$
e^x	e^x	$\arccos x$	$-\frac{1}{\sqrt{1-x^2}}$
a^x	$\ln a \cdot a^x$	$\arctan x$	$\frac{1}{1+x^2}$
$\ln x$	$1/x$	$\operatorname{arccot} x$	$-\frac{1}{1+x^2}$
$\log_a x$	$\frac{1}{x} \log_a e = \frac{1}{x \ln a}$	$\sinh x$	$\cosh x$
$\sin x$	$\cos x$	$\cosh x$	$\sinh x$
$\cos x$	$-\sin x$	$\tanh x$	$\frac{1}{\cosh^2 x}$
$\tan x$	$\frac{1}{\cos^2 x}$	$\operatorname{coth} x$	$-\frac{1}{\sinh^2 x}$

CAS-Beispiel



Eine Möglichkeit, Funktionen abzuleiten, ist natürlich die Definition

$$\frac{df}{dx} = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

zu benutzen:

```
from sympy import limit, sin, cos, tan, exp
from sympy.abc import x, d, n

limit((sin(x+d)-sin(x))/d, d, 0)
```

```
limit((exp(x+d)-exp(x))/d, d, 0)
```

```
from sympy import simplify

simplify(limit(((x+d)**n-x**n)/d, d, 0))
```

```
simplify(limit((tan(x+d)-tan(x))/d, d, 0))
```

```
simplify(limit((sin(x+d)/cos(x+d)-sin(x)/cos(x))/d, d, 0))
```

Einfacher lassen sich Ableitungen mittels der internen Funktion `diff` erhalten.

```
from sympy import diff

diff(sin(x), x)
```

```
simplify(diff(x**n, x))
```

```
diff(exp(x), x)
```

```
simplify(diff(tan(x), x))
```

Überzeugen Sie sich davon, daß das Ergebnis für $\frac{d \tan(x)}{dx}$ mit dem Obigen übereinstimmt.

3.5.3 Differentiationsregeln

S Summenregel

Sind f und g differenzierbar bei x_0 , so ist auch

$$h(x) = a f(x) + b g(x) \quad a, b \in \mathbb{R}$$

differenzierbar mit

$$h'(x_0) = (a f + b g)'(x_0) = a f'(x_0) + b g'(x_0)$$

B

$$\begin{aligned} h'(x_0) &= \lim_{x \rightarrow x_0} \frac{[a f(x) + b g(x)] - [a f(x_0) + b g(x_0)]}{x - x_0} \\ &= a \lim_{x \rightarrow x_0} \frac{f(x) - f(x_0)}{x - x_0} + b \lim_{x \rightarrow x_0} \frac{g(x) - g(x_0)}{x - x_0} \\ &= a f'(x_0) + b g'(x_0) \end{aligned}$$

Beispiel:

$$\frac{d}{dx} \sum_{i=0}^n a_i x^i = \sum_{i=0}^n i a_i x^{i-1} = \sum_{i=1}^n i a_i x^{i-1} = \sum_{i=0}^{n-1} (i+1) a_{i+1} x^i$$

Der Grad der Funktion erniedrigt sich um 1.

S Produktregel

Sind f und g differenzierbar bei x_0 , so ist auch

$$h(x) = f(x) g(x)$$

differenzierbar mit

$$h'(x_0) = f'(x_0) g(x_0) + g'(x_0) f(x_0)$$

B

$$\begin{aligned}
h'(x_0) &= (f g)'(x_0) \\
&= \lim_{\Delta x \rightarrow 0} \frac{f(x_0 + \Delta x) g(x_0 + \Delta x) - f(x_0) g(x_0)}{\Delta x} \\
&= \lim_{\Delta x \rightarrow 0} \frac{f(x_0 + \Delta x) g(x_0 + \Delta x) - f(x_0) g(x_0 + \Delta x)}{\Delta x} \\
&\quad + \lim_{\Delta x \rightarrow 0} \frac{f(x_0) g(x_0 + \Delta x) - f(x_0) g(x_0)}{\Delta x} \\
&= \lim_{\Delta x \rightarrow 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} \lim_{\Delta x \rightarrow 0} g(x_0 + \Delta x) \\
&\quad + \lim_{\Delta x \rightarrow 0} \frac{g(x_0 + \Delta x) - g(x_0)}{\Delta x} f(x_0) \\
&= f'(x_0) g(x_0) + g'(x_0) f(x_0)
\end{aligned}$$

Beispiel:

$$\begin{aligned}
f(x) &= \sin x \sin x + \cos x \cos x \\
f'(x) &= \sin x \cos x - \cos x \sin x \\
&\quad + \cos x \sin x - \sin x \cos x \\
&= 0 \\
f'(x) &= \frac{d}{dx}(\sin^2 x + \cos^2 x) = \frac{d}{dx} 1 = 0
\end{aligned}$$

$$\begin{aligned}
f(x) &= \sin^5 x \\
f'(x) &= \frac{d}{dx}(\sin x \sin x \sin x \sin x \sin x) \\
&= \cos x \sin x \sin x \sin x \sin x \\
&\quad + \sin x \cos x \sin x \sin x \sin x \\
&\quad + \sin x \sin x \cos x \sin x \sin x \\
&\quad + \sin x \sin x \sin x \cos x \sin x \\
&\quad + \sin x \sin x \sin x \sin x \cos x \\
&= 5 \sin^4 x \cos x
\end{aligned}$$

S

Quotientenregel

Sind f und g differenzierbar bei x_0 , so ist auch

$$h(x) = f(x)/g(x)$$

für $g(x_0) \neq 0$ differenzierbar mit

$$h'(x_0) = \frac{f'(x_0) g(x_0) - g'(x_0) f(x_0)}{g^2(x_0)}$$

B

$$\begin{aligned} h'(x_0) &= (f/g)'(x_0) \\ &= \lim_{\Delta x \rightarrow 0} \frac{f(x_0 + \Delta x)/g(x_0 + \Delta x) - f(x_0)/g(x_0)}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \frac{\frac{f(x_0 + \Delta x)}{g(x_0 + \Delta x)} - \frac{f(x_0)}{g(x_0 + \Delta x)}}{\Delta x} + \lim_{\Delta x \rightarrow 0} \frac{\frac{f(x_0)}{g(x_0 + \Delta x)} - \frac{f(x_0)}{g(x_0)}}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} \lim_{\Delta x \rightarrow 0} \frac{1}{g(x_0 + \Delta x)} + \\ &\quad + f(x_0) \lim_{\Delta x \rightarrow 0} \left[\frac{g(x_0) - g(x_0 + \Delta x)}{g(x_0) g(x_0 + \Delta x)} / \Delta x \right] \\ &= f'(x_0) \frac{1}{g(x_0)} + f(x_0) \lim_{\Delta x \rightarrow 0} \frac{g(x_0) - g(x_0 + \Delta x)}{\Delta x} \lim_{\Delta x \rightarrow 0} \frac{1}{g(x_0) g(x_0 + \Delta x)} \\ &= f'(x_0) \frac{1}{g(x_0)} - f(x_0) g'(x_0) \frac{1}{g^2(x_0)} \\ &= \frac{f'(x_0) g(x_0) - f(x_0) g'(x_0)}{g^2(x_0)} \end{aligned}$$

Beispiel:

$$\begin{aligned} h(x) &= \frac{\sin x}{\cos x} = \tan x \\ h'(x) &= \frac{\cos x \cos x - \sin x (-\sin x)}{\cos^2 x} \\ &= \frac{1}{\cos^2 x} \end{aligned}$$

S

Kettenregel

Sind f und g differenzierbar, so ist auch die Verkettung

$$h(x) = f(g(x))$$

differenzierbar mit

$$h'(x) = \frac{df}{dg} \frac{dg}{dx}$$

B

$$\begin{aligned} \frac{dh}{dx} &= \lim_{\Delta x \rightarrow 0} \frac{f(g(x + \Delta x)) - f(g(x))}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \frac{f(g(x + \Delta x)) - f(g(x))}{g(x + \Delta x) - g(x)} \frac{g(x + \Delta x) - g(x)}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \frac{f(g(x + \Delta x)) - f(g(x))}{g(x + \Delta x) - g(x)} \frac{dg}{dx} \end{aligned}$$

mit $g(x + \Delta x) = g(x) + \Delta g(\Delta x)$

$$\begin{aligned} &= \lim_{\Delta g \rightarrow 0} \frac{f(g(x) + \Delta g) - f(g(x))}{g(x) + \Delta g - g(x)} \frac{dg}{dx} \\ &= \frac{df}{dg} \frac{dg}{dx} \end{aligned}$$

Beispiel:

$$y = \sin^5 x \quad f(g) = g^5, \quad g(x) = \sin x$$

$$\begin{aligned} y' &= 5g^4(x) \cos x \\ &= 5 \sin^4 x \cos x \end{aligned}$$

$$y = (\cos x + 2x^2 + 1)^{1/2}$$

$$y' = \frac{1}{2} (\cos x + 2x^2 + 1)^{-1/2} (-\sin x + 4x)$$

$$y = a^x = (e^{\ln a})^x = e^{x \ln a}$$

$$\begin{aligned} y' &= \ln a \, e^{x \ln a} \\ &= \ln a \, a^x \end{aligned}$$

S Die Funktion $f(x)$ sei in $I = [a, b]$ differenzierbar und besitze die Umkehrfunktion $f^{-1}(y)$. Dann ist $f^{-1}(y)$ differenzierbar mit der Ableitung

$$(f^{-1})'(y) = 1/f'(x)$$

mit $y = f(x)$.

B

$$\begin{aligned} I(x) &= f^{-1}(f(x)) = x \quad y = f(x) \\ \frac{d}{dx} I(x) &= \frac{df^{-1}}{dy} \frac{dy}{dx} = 1 \\ \frac{df^{-1}}{dy} &= \frac{1}{\frac{dy}{dx}} \end{aligned}$$

Beispiel:

$$f(x) = \sin x = y \quad f^{-1}(y) = \arcsin y = x$$

$$\begin{aligned} \frac{d}{dy} \arcsin y &= \frac{1}{\frac{dy}{dx}} \\ &= \frac{1}{\frac{d \sin x}{dx}} \\ &= \frac{1}{\cos x} \\ &= \frac{1}{\sqrt{1 - \sin^2 x}} \\ &= \frac{1}{\sqrt{1 - y^2}} \\ \arcsin' x &= \frac{1}{\sqrt{1 - x^2}} \end{aligned}$$

$$f(x) = e^x = y \quad f^{-1}(y) = \ln y = x$$

$$\begin{aligned} \frac{d}{dy} \ln y &= \frac{1}{\frac{dy}{dx}} \\ &= \frac{1}{\frac{de^x}{dx}} \\ &= \frac{1}{e^x} \\ &= \frac{1}{y} \\ \ln' x &= \frac{1}{x} \end{aligned}$$

CAS-Beispiel



Mit der Funktion `diff` kann man sich einfach von der Gültigkeit der Differentiationsregeln überzeugen. Durch den Aufruf `Function` wird den Funktionen f und g die jeweilige unabhängige Variable zugewiesen, wodurch man z.B. statt `diff(f(x), x)` auch einfach `diff(f, x)` schreiben kann.

```
from sympy import Function, diff
from sympy.abc import x

f = Function("f")(x)
g = Function("g")(x)
```

Summenregel:

```
diff(f+g, x)
```

Produktregel:

```
diff(f*g, x)
```

Quotientenregel:

```
from sympy import simplify

simplify(diff((f/g), x))
```

Kettenregel:

```
f = Function("f")(g)

diff(f, x)
```

Im letzten Aufruf wird durch `Function` die Abhängigkeit der Funktion f von der Variablen x gelöscht und anschließend die Abhängigkeit von der Funktion g deklariert.

3.5.4 Ableitungen höherer Ordnung

D Die Funktion $f(x)$ sei in der Umgebung von x differenzierbar. Faßt man $f'(x)$ wieder als Funktion von x auf, so führt die Differentiation bzw. Ableitung von $f'(x)$ zur **zweiten Ableitung von $f(x)$** .

$$(f')'(x) = f''(x) = f^{(2)}(x) = \frac{d^2}{dx^2}f(x)$$

Entsprechend führt n -faches Ableiten zur **n -ten Ableitung bzw. Ableitung n -ter Ordnung**

$$f^{(n)}(x) = \frac{d^n}{dx^n}f(x)$$

Existiert $f^{(n)}(x)$, so heißt $f(x)$ **n -mal differenzierbar**. Ist $f(x)$ n -mal differenzierbar in einem Intervall I und ist $f^{(n)}(x)$ stetig für $x \in I$, so heißt $f(x)$ **n -mal stetig differenzierbar**.

CAS-Beispiel



Höhere Ableitungen können durch weiteres Differenzieren erhalten werden:

```
from sympy import diff, sin
from sympy.abc import x

diff(sin(x), x)
```

```
diff(_, x)
```

```
diff(diff(sin(x), x), x)
```

Schneller geht es mit der folgenden Notation:

```
diff(sin(x), x, 2)
```

```
diff(sin(x), x, 3)
```

```
diff(sin(x), x, 4)
```

```
diff(sin(x), x, 5)
```

Beispiel:

geometrische Deutung

Funktion	Weg	$s = s(t)$	t : Zeit	Kurve
1. Ableitung	Geschwindigkeit	$v = \dot{s} = \frac{ds}{dt}$		Steigung
2. Ableitung	Beschleunigung	$a = \dot{v} = \ddot{s} = \frac{d^2s}{dt^2}$		Krümmung

gleichförmige Bewegung: $\dot{v} = 0$, d.h. $v = \text{const.}$

$$s = s_0 + v \cdot (t - t_0)$$

gleichmäßige Beschleunigung: $\dot{a} = 0$, d.h. $a = \text{const.}$

$$v = v_0 + a \cdot (t - t_0)$$

$$s = s_0 + v_0 \cdot (t - t_0) + \frac{1}{2}a \cdot (t - t_0)^2$$

$a = g$ Erdbeschleunigung \Rightarrow freier Fall.

Harmonischer Oszillator

$$\begin{aligned}
 x(t) &= x_0 \cos(\omega t) \\
 v(t) = \frac{dx(t)}{dt} &= -x_0 \omega \sin(\omega t) \\
 a(t) = \frac{d^2x(t)}{dt^2} &= -x_0 \omega^2 \cos(\omega t) \\
 &= -x_0 \frac{k}{m} \cos(\omega t) \\
 F = m \cdot a &= -x_0 k \cos(\omega t) \\
 &= -k \cdot x(t) \\
 \Rightarrow F &= -k \cdot x \quad \text{Hookesches Gesetz}
 \end{aligned}$$

3.5.5 Kurvendiskussion

Der Begriff der Ableitung erlaubt es Funktionen bzw. deren geometrischen Darstellung (Kurven) näher zu charakterisieren, d.h. ausgezeichnete Kurvenpunkte oder Kurvenabschnitte zu klassifizieren.

S Sei f eine in $I =]a, b[$ differenzierbare Funktion. Für ein lokales Minimum, bzw. Maximum bei x_0 gilt

$$f'(x_0) = 0$$

– die Umkehrung gilt nicht!

B z.B. lokales Minimum, d.h.

$$f(x + \Delta x) > f(x) \quad 0 < |\Delta x| < \delta$$

linksseitige Ableitung:

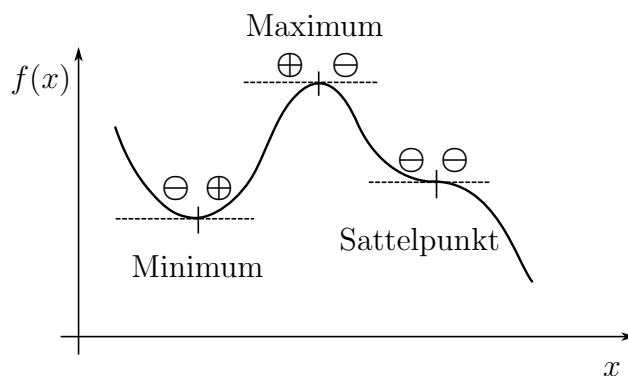
$$f'_l(x_0) = \lim_{\Delta x \rightarrow 0^-} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} \leq 0$$

rechtsseitige Ableitung:

$$f'_r(x_0) = \lim_{\Delta x \rightarrow 0^+} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} \geq 0$$

$$\begin{aligned} \text{Es gilt:} \quad & f'_l(x_0) = f'_r(x_0) = f'(x_0) \\ \Rightarrow \quad & f'(x_0) = 0 \end{aligned}$$

D Als **kritische Punkte** werden Kurvenpunkte mit einer horizontalen Tangente, d.h. mit $f'(x) = 0$, bezeichnet. Besitzt $f(x)$ an einem kritischen Punkt x_0 ein lokales Minimum oder Maximum, so spricht man von einer **Extremstelle**. Liegt an einem kritischen Punkt x_0 weder ein Maximum oder Minimum vor, so spricht man von einem **Terassenpunkt** oder einem **Sattelpunkt**.



Minimum	$x_0 = 2$	$f(x) = 5 + (x - 2)^2$	$f'(x) = 2(x - 2)$
Maximum	$x_0 = 4$	$f(x) = 2 - (x - 4)^8$	$f'(x) = -8(x - 4)^7$
Sattelpunkt	$x_0 = 0$	$f(x) = x^3$	$f'(x) = 3x^2$

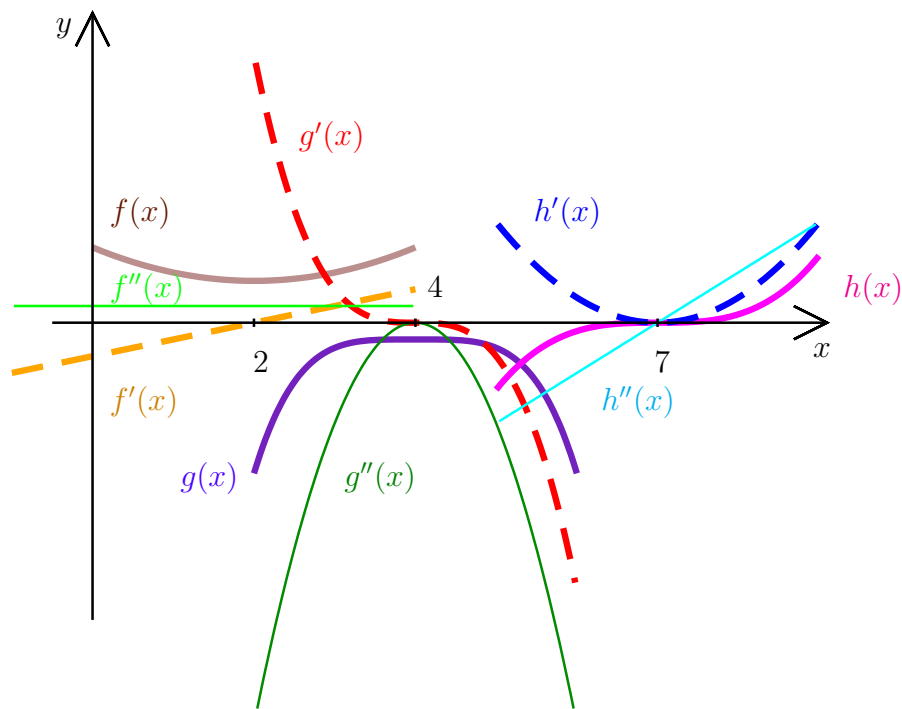
S Besitzt für eine in $I =]a, b[$ differenzierbare Funktion $f(x)$ ihre Ableitung $f'(x)$ ($x \in I$) bei x_0 eine Nullstelle, so ist x_0

- ein **Sattelpunkt** falls $f'(x)$ sein Vorzeichen beibehält
- eine **Extremstelle** falls $f'(x)$ bei x_0 sein Vorzeichen ändert und zwar ein
 - lokales Minimum falls $- \rightarrow +$
 - lokales Maximum falls $+ \rightarrow -$

S Ist $f(x)$ in $]a, b[$ zweimal stetig differenzierbar und gilt $f'(x_0) = 0$, so liegt bei x_0

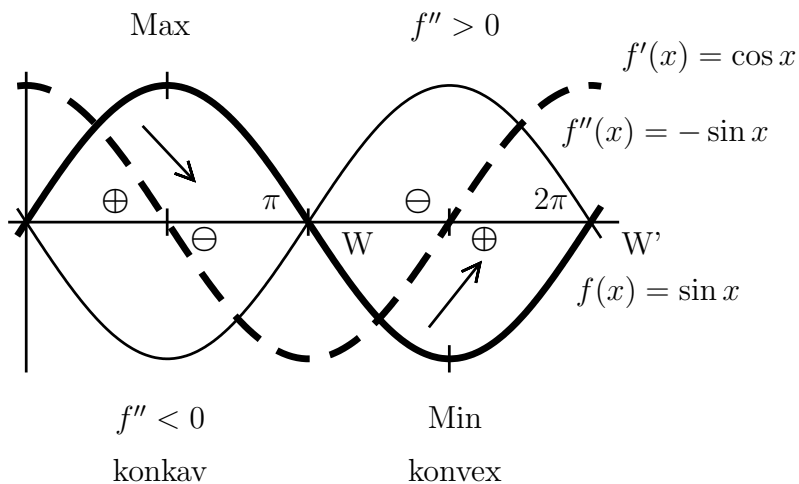
- ein lokales Minimum vor, falls $f''(x_0) > 0$

b) ein lokales Maximum vor, falls $f''(x_0) < 0$



Min.	$f(x) = 5 + (x - 2)^2$	$f'(x) = 2(x - 2)$	$f''(x) = 2 > 0$	$x_0 = 2$
Max.	$g(x) = -2 - (x - 4)^4$	$g'(x) = -4(x - 4)^3$	$g''(x) = -12(x - 4)^2 < 0$	$x_0 = 4$
Sattl.	$h(x) = (x - 7)^3$	$h'(x) = 3(x - 7)^2$	$h''(x) = 6(x - 7)$	$x_0 = 7$

D Besitzt die erste Ableitung einer in $]a, b[= I$ differenzierbaren Funktion an einer Stelle $x_0 \in I$ einen Extremwert, so wird die Stelle als **Wendepunkt** bezeichnet.



Bedeutung: Die Krümmung der Kurve ändert sich bei einem Wendepunkt.

S Gilt für eine zweimal differenzierbare Funktion $f''(x_0) = 0$ und

a) $f''(x)$ hat an der Stelle x_0 einen Vorzeichenwechsel
oder

b) $f'''(x_0)$ existiert mit $f'''(x_0) \neq 0$

so liegt bei x_0 ein Wendepunkt vor.

S Ist f eine in $]a, b[$ differenzierbare Funktion, dann ist $f(x)$ in $[a, b]$ genau dann monoton steigend/fallend falls gilt:

$$f'(x) \geq 0 \quad \text{bzw.} \quad f'(x) \leq 0 \quad \forall x \in]a, b[$$

S Eine in $]a, b[$ zweimal differenzierbare Funktion ist genau dann konvex/konkav falls gilt:

$$\begin{array}{ccc} f''(x) \geq 0 & \text{bzw.} & f''(x) \leq 0 \quad \forall x \in]a, b[\\ \text{konvex} & & \text{konkav} \end{array}$$

CAS-Beispiel



Kurvendiskussion von $f(x) = x^5 - 10x^3 + x$

```
from sympy import diff
from sympy.abc import x

def f(x):
    return x**5 - 10*x**3 + x

def f1(x):
    return diff(f(x), x)

def f2(x):
    return diff(f1(x), x)

def f3(x):
    return diff(f2(x), x)
```

Zunächst werden die Nullstellen durch Lösen der Gleichung $f(x) = 0$ gesucht. Überzeugen Sie sich durch die Graphik, daß das Ergebnis stimmt.

```
from sympy.solvers import solve
import numpy as np
import matplotlib.pyplot as plt

solve(f(x), x)
```

```
[float(i) for i in _]
```

```
plt.figure()
plt.subplot(211)
x0 = np.linspace(-5, 5)
plt.plot(x0, f(x0), label = r"$f(x)$")
plt.plot(x0, [f1(x).subs(x, i) for i in x0], label = r"$f'(x)$")
plt.legend()

plt.subplot(212)
x1 = np.linspace(-0.5, 0.5)
plt.plot(x1, f(x1), label = r"$f(x)$")
plt.plot(x1, [f1(x).subs(x, i) for i in x1], label = r"$f'(x)$")
plt.legend()
plt.show()
```

Notwendige Bedingung für das Vorliegen eines Extrempunktes ist: $f'(x) = 0$:

```
solve(f1(x), x)
```

```
[float(i) for i in _]
```

Im Folgenden werden alle Lösungen der Lösungsmenge `lcp` durchlaufen und geschaut, ob ein Vorzeichenwechsel vorliegt und welchen Wert $f''(x)$ annimmt.

Welcher Typ eines kritischen Punktes liegt vor?

```
lcp = solve(f1(x), x)

for i in range(len(lcp)):
    print("x" + str(i) + " = " + str(float(lcp[i])))
    print("f\'(x" + str(i) + "-epsilon): "
          + str(float(f1(x).subs(x, lcp[i]-1.0e-6))))
    print("f\'(x" + str(i) + "+epsilon): "
          + str(float(f1(x).subs(x, lcp[i]+1.0e-6))))
    print("f\'\'(x" + str(i) + " : "
          + str(float(f2(x).subs(x, lcp[i]))))
```

Schließlich wird nach Wendepunkten gesucht. Auch hier kann mittels zweier Kriterien entschieden werden, ob ein Wendepunkt vorliegt.

```
lip = solve(f2(x), x)

for i in range(len(lip)):
```

```

print("x" + str(i) + " = " + str(float(lip[i])))
print("f\''\'(x" + str(i) + "-epsilon): "
      + str(float(f2(x).subs(x, lip[i]-1.0e-6))))
print("f\''\'(x" + str(i) + "+epsilon): "
      + str(float(f2(x).subs(x, lip[i]+1.0e-6))))
print("f\''\''\'(x" + str(i) + ")          : "
      + str(float(f3(x).subs(x, lip[i]))))

```

Abschließend zwei konkrete Beispiele aus der physikalischen Chemie.

Zunächst soll das Maximum der Maxwell-Boltzmann- Geschwindigkeitsverteilung aufgesucht werden: m ist die Masse von H_2O in kg, k die Boltzmannkonstante und T die Temperatur in Kelvin.

```

k = 1.38066e-23
m = 2.98897e-26
T = 300

```

```

from sympy import pi, exp
from sympy.abc import v
import numpy as np
import matplotlib.pyplot as plt

def s(v):
    return (m/(2*pi*k*T))*v**2*exp((-m*v**2)/(2*k*T))

solve(diff(s(v), v), v)
[float(i) for i in _]

```

```

vrangle = np.linspace(0, 2000)
plt.plot(vrangle, [s(v).subs(v, i) for i in vrangle])
plt.show()

```

Die Funktion $U(l)$ gibt die Verteilungsfunktion eines Schwarzen Strahlers bei Temperatur T als Funktion der Wellenlänge l (λ) an. Zeigen Sie, daß das Wiensche Verschiebungsgesetz $\lambda_{\max}T = \text{const.}$ gilt:

```

h = 6.62607e-34
c = 2.99792e+8

def U(l):
    return (8*pi*h*c/l**5)*exp(-h*c/(l*k*T))/(1-exp(-h*c/(l*k*T)))

```

3.5.6 Eigenschaften differenzierbarer Funktionen

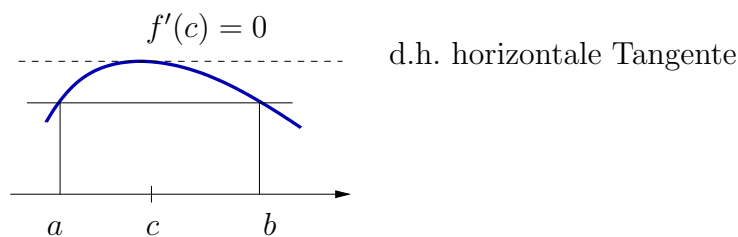
S Eine in x differenzierbare Funktion ist dort auch stetig – die Umkehrung gilt nicht.

B

$$\begin{aligned}
 \lim_{\Delta x \rightarrow 0} [f(x + \Delta x) - f(x)] &= \left[\lim_{\Delta x \rightarrow 0} f(x + \Delta x) \right] - f(x) \\
 &= \lim_{\Delta x \rightarrow 0} \left[\Delta x \frac{f(x + \Delta x) - f(x)}{\Delta x} \right] \\
 &= \lim_{\Delta x \rightarrow 0} \Delta x \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x} \\
 &= 0 \cdot f'(x) \\
 \Rightarrow \lim_{\Delta x \rightarrow 0} f(x + \Delta x) &= f(x)
 \end{aligned}$$

S Satz von Rolle

Zu einer in $[a, b]$ stetigen und in $]a, b[$ differenzierbaren Funktion $f(x)$ für die $f(a) = f(b)$ gilt, gibt es ein $c \in]a, b[$ mit $f'(c) = 0$.

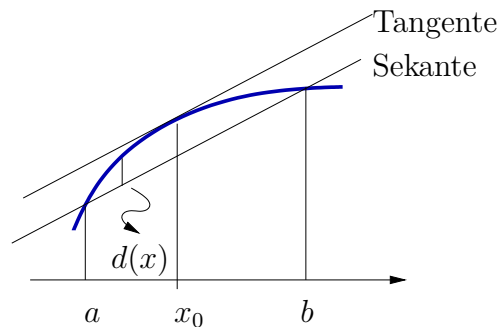


B Satz von Weierstrass: $f(x)$ besitzt in $[a, b]$ ein absolutes Maximum bzw. Minimum. Liegen diese bei a und b , so heißt dies daß $f(x) = f(a) \forall x \in [a, b]$ sonst gibt es eine Extremstelle und die hat eine horizontale Tangente, d.h. $f'(x) = 0$.

S Mittelwertsatz der Differentialrechnung

Zu jeder in $[a, b]$ stetigen und in $]a, b[$ differenzierbaren Funktion $f(x)$ gibt es ein $x_0 \in]a, b[$ mit

$$f'(x_0) = \frac{f(b) - f(a)}{b - a}$$



B Sekantengleichung:

$$s(x) = f(a) + m(x - a)$$

$$m = \frac{f(b) - f(a)}{b - a}$$

Differenz zwischen Sekante und Funktion $f(x)$:

$$d(x) = f(x) - s(x)$$

wobei gilt

$$d(a) = d(b) = 0$$

$d(x)$ erfüllt die Voraussetzungen des Satzes von Rolle \Rightarrow es gibt ein x_0 mit

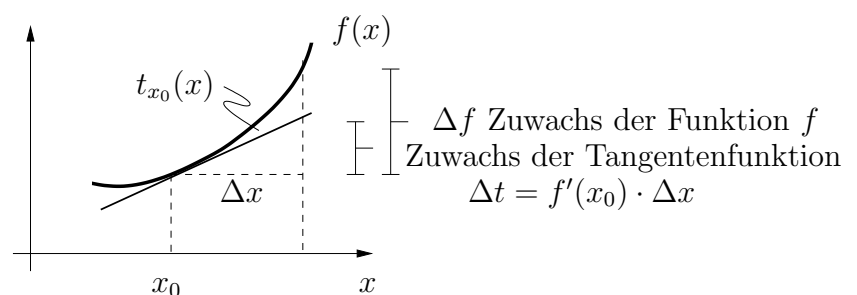
$$d'(x_0) = 0$$

$$= f'(x_0) - s'(x_0)$$

$$\Rightarrow f'(x_0) = s'(x_0) = m = \frac{f(b) - f(a)}{b - a}$$

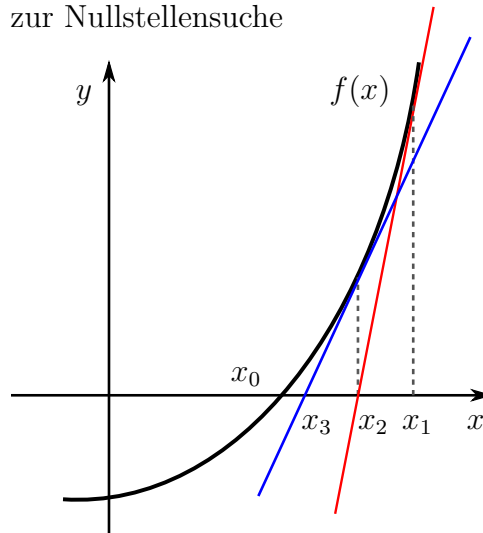
3.5.7 Anwendungen

- Lineare Näherung



$$f(x) \simeq f(x_0) + (x - x_0) \cdot f'(x_0) = t_{x_0}(x)$$

- Newtonsches Verfahren zur Nullstellensuche



1. Schritt rate $x_1 \simeq x_0$:

$$f(x) \simeq t_{x_1}(x) = f(x_1) + f'(x_1)(x - x_1)$$

2. Schritt

$$\text{Nullstelle } x_2 : t_{x_1}(x) = 0 \Rightarrow x_2 = x_1 - f(x_1)/f'(x_1)$$

3. Schritt

$$f(x) \simeq t_{x_2}(x) = f(x_2) + f'(x_2)(x - x_2)$$

$$\text{Nullstelle } x_3 : t_{x_2}(x) = 0 \Rightarrow x_3 = x_2 - f(x_2)/f'(x_2)$$

...

$$x_{i+1} = x_i - f(x_i)/f'(x_i)$$

Beispiel:

$$f(x) = e^x - 4$$

$$x_{i+1} = x_i - f(x_i)/f'(x_i)$$

$$= x_i - \frac{e^{x_i} - 4}{e^{x_i}}$$

$$x_{i+1} = x_i - 1 + 4 \cdot e^{-x_i}$$

alternativ

$$f(x) = e^x - 4 = 0$$

$$e^x = 4$$

$$x = \ln 4$$

1. Schritt

$$\begin{aligned}
 x_1 &= 1 \\
 x_2 &= 1,4715\dots \\
 x_3 &= 1,3898\dots \\
 x_4 &= 1,3863\dots \\
 x_5 &= 1,38629436 \\
 \ln 4 &= 1,38629436
 \end{aligned}$$

CAS-Beispiel



Der folgende Python-Code versucht bei vorgegebener Funktion f und Startwert x_0 numerisch die Nullstelle von f zu finden:

```

def newton(fexpr, var, x0):
    from sympy import diff
    from sympy.abc import x
    def f(x):
        return fexpr.subs(var, x)
    def df(x):
        return diff(f(x), x)
    i = 0
    while (abs(f(x0)) >= 5e-8):
        y = df(x).subs(x, x0)
        x0 -= f(x0)/y
        i += 1
        print("i = " + str(i) + ", x = " + str(x0)
              + ", f(x) = " + str(f(x0)))
    return x0

```

Im Folgenden wird *numerisch* die Nullstelle von $f(x) = \sin(x)$ gesucht, die in der Nähe des Startwerts $x_0 = 1$ liegt.

Können Sie das exakte Ergebnis angeben?

Gibt es weitere Nullstellen?

```

from sympy import sin
from sympy.abc import x

newton(sin(x), x, 1.0)

```

Diese Funktion können Sie auch im `mathchem`-Modul laden:

```

from mathchem import newton

newton(sin(x), x, 4.0)

```

- Regeln von l'Hospital

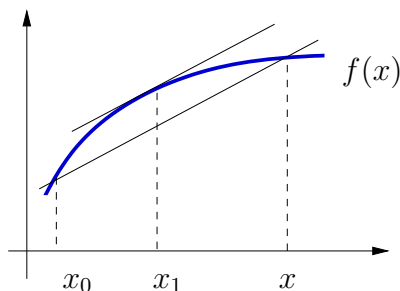
S Es seien f und g bei x_0 differenzierbare Funktionen (wobei $x = x_0$ ausgenommen werden kann) mit $f(x_0) = g(x_0) = 0$. Existiert der Grenzwert $\lim_{x \rightarrow x_0} \frac{f'(x)}{g'(x)}$, so gilt:

$$\lim_{x \rightarrow x_0} \frac{f(x)}{g(x)} = \lim_{x \rightarrow x_0} \frac{f'(x)}{g'(x)}$$

Die Aussage gilt auch für die uneigentlichen Grenzwerte $x_0 = \pm\infty$, bzw. einseitigen Grenzwerte $x \rightarrow x_0 \pm 0$.

B (1. Teil der Aussage)

Mittelwertsatz:



$$\begin{aligned} f'(x_1) &= \frac{f(x) - f(x_0)}{x - x_0} \\ \Rightarrow f(x) &= f(x_0) + f'(x_1) (x - x_0) \end{aligned}$$

entsprechend

$$g(x) = g(x_0) + g'(x_1^*) (x - x_0)$$

mit $x_1, x_1^* \in]x_0, x[$.

$$\begin{aligned} \lim_{x \rightarrow x_0} \frac{f(x)}{g(x)} &= \lim_{x \rightarrow x_0} \frac{f(x_0) + f'(x_1) (x - x_0)}{g(x_0) + g'(x_1^*) (x - x_0)} \\ &= \lim_{x \rightarrow x_0} \frac{f'(x_1)}{g'(x_1^*)} \\ &= \lim_{\substack{x_1 \rightarrow x_0 \\ x_1^* \rightarrow x_0}} \frac{f'(x_1)}{g'(x_1^*)} \end{aligned}$$

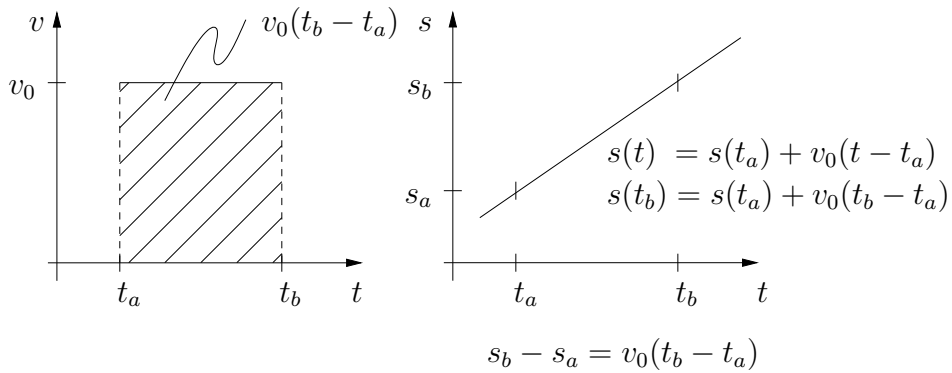
Beispiel:

$$\begin{aligned}\lim_{x \rightarrow 0} \frac{\sin x}{x} &= \lim_{x \rightarrow 0} \frac{\cos x}{1} = 1 \\ \lim_{x \rightarrow 0} \frac{e^x - 1}{x} &= \lim_{x \rightarrow 0} \frac{e^x}{1} = 1\end{aligned}$$

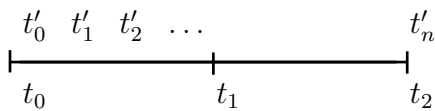
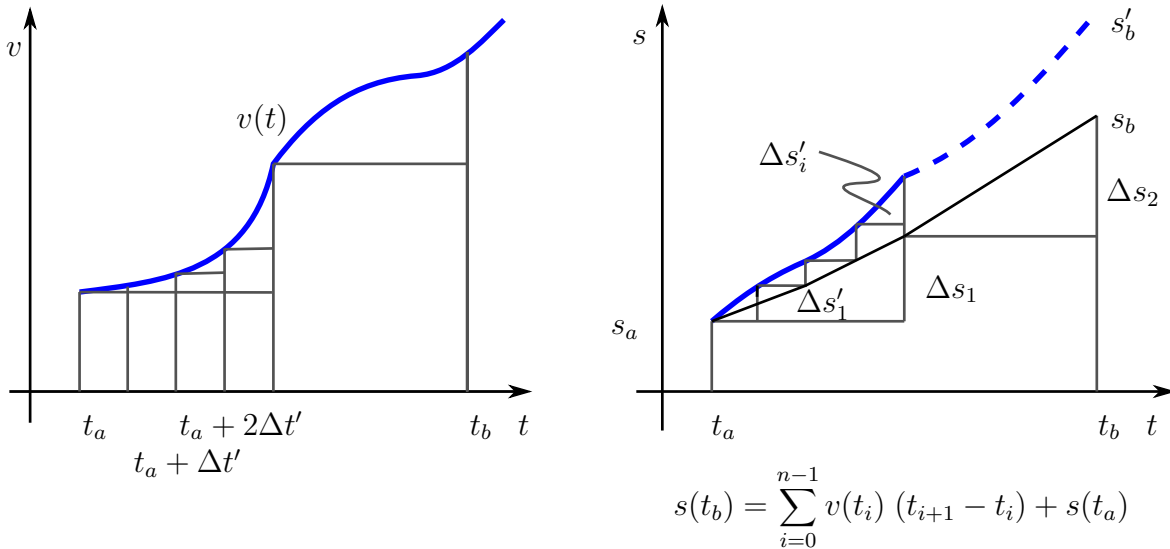
3.6 Integralrechnung

3.6.1 Geometrische Deutung und Definitionen

- gleichförmige Bewegung



- beliebige Geschwindigkeit



Zerlegung des Zeitintervalls:

$$t_i = t_a + i \Delta t \quad i = 0, \dots, n$$

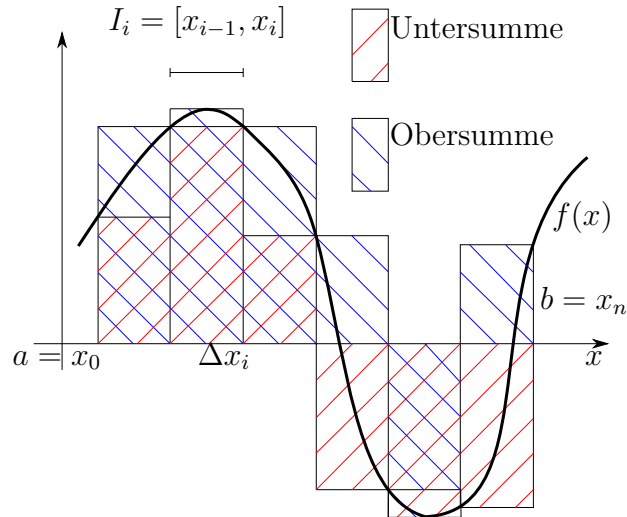
$$\Delta t = (t_b - t_a)/n \quad \text{\textbf{\textit{äquidistant}}}$$

Hängt das Ergebnis von der Zerlegung ab? Führt immer weitere Verfeinerung der Zerlegung zu einem eindeutigen Ergebnis (Grenzwert)?

D Durch Einfügen von Punkten x_i ($i = 0, \dots, n$) mit $a = x_0 < x_1 < x_2 \dots < x_n = b$ entsteht eine **Zerlegung Z** des Intervalls $[a, b]$. Die Länge des größten Teilintervalls

$I_i = [x_{i-1}, x_i]$ wird mit $\Delta(Z)$ bezeichnet:

$$\Delta(Z) = \max_{i=1 \dots n} (x_i - x_{i-1})$$



D

Untersumme:

$$U(Z) = \sum_{i=1}^n f_{\min,i} \Delta x_i$$

Obersumme:

$$O(Z) = \sum_{i=1}^n f_{\max,i} \Delta x_i$$

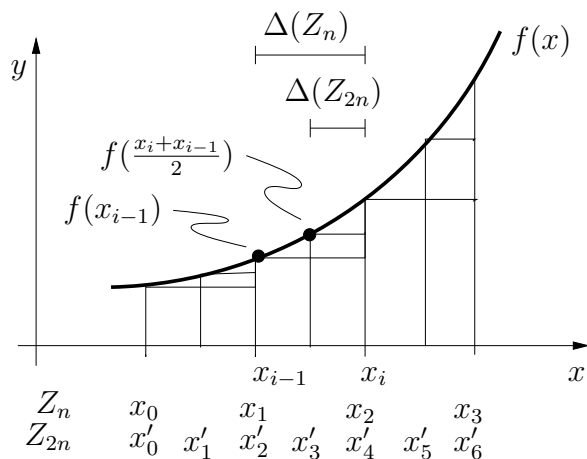
mit

$$\begin{aligned} \Delta x_i &= x_i - x_{i-1} \\ f_{\min,i} &= \min \{f(x) | x \in I_i\} \end{aligned}$$

! $f(x)$ sei stetig, sonst sup bzw. inf.

Vereinfachende Annahme:

$f(x)$ sei stetig und monoton steigend in $[a, b]$. Äquidistante Zerlegung Z_n in n Teilintervalle; Verfeinerung durch Halbierung der Teilintervalle.



$$\begin{aligned}
 U(Z_n) &= \sum_{i=1}^n f(x_{i-1}) \Delta(Z_n) \\
 U(Z_{2n}) &= \sum_{i=1}^{2n} f(x'_{i-1}) \Delta(Z_{2n}) \\
 U(Z_{2n}) - U(Z_n) &= \sum_{i=1}^n \left[f\left(\frac{x_i + x_{i-1}}{2}\right) - f(x_{i-1}) \right] \frac{1}{2} \Delta(Z_n) \\
 &> 0
 \end{aligned}$$

$U_n = U(Z_n)$ ist eine monoton steigende Zahlenfolge.

$U_n = U(Z_n)$ ist nach oben beschränkt (jedes O_m ist eine obere Schranke).

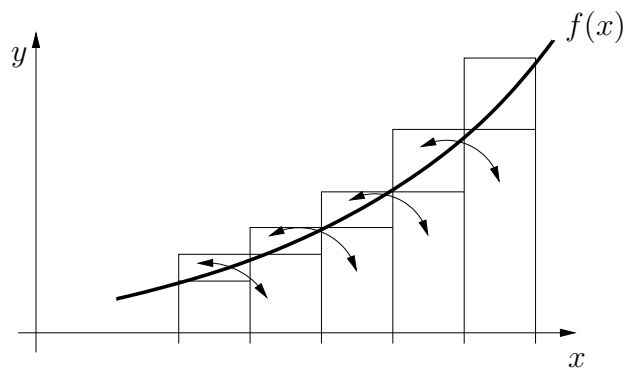
$\Rightarrow U_n$ besitzt einen Grenzwert U .

Entsprechende Überlegungen gelten für O_n :

O_n ist monoton fallend

O_n besitzt untere Schranke

\Rightarrow es existiert ein Grenzwert O .



$$O(Z_n) - U(Z_n) = \sum_{i=1}^n [f(x_i) - f(x_{i-1})] \Delta(Z_n)$$

$$\begin{aligned}
&= [f(x_n) - f(x_0)] \Delta(Z_n) \\
&= [f(b) - f(a)] \Delta(Z_n) \\
\Rightarrow \lim_{n \rightarrow \infty} (O(Z_n) - U(Z_n)) &= [f(b) - f(a)] \lim_{n \rightarrow \infty} \Delta(Z_n) \\
&= 0 \\
\Rightarrow \lim_{n \rightarrow \infty} O(Z_n) &= \lim_{n \rightarrow \infty} U(Z_n)
\end{aligned}$$

d.h. O_n und U_n besitzen denselben Grenzwert!

S Für jede in $[a, b]$ stückweise stetige Funktion $f(x)$ (endlich viele Unstetigkeitsstellen) gilt für jede beliebige Zerlegung Z des Intervalls $[a, b]$

$$U(Z) \leq I \leq O(Z)$$

d.h. U ist nach oben beschränkt mit der kleinsten oberen Schranke I

O ist nach unten beschränkt mit der größten unteren Schranke I

Zu jedem $\epsilon > 0$ gibt es eine Zerlegung Z so daß für jede weitere Verfeinerung der Zerlegung Z' gilt:

$$|U(Z') - I| < \epsilon, |O(Z') - I| < \epsilon$$

D Zu einer gegebenen Zerlegung Z wird die **Zwischensumme** $S(Z)$ definiert durch

$$S(Z) = \sum_{i=1}^n f(x_i^*) \Delta x_i$$

mit $x_{i-1} < x_i^* < x_i$ d.h. x_i^* ansonsten beliebig.

S Es gilt (sofern die Grenzwerte existieren):

$$\lim_{\Delta(Z) \rightarrow 0} U(Z) = \lim_{\Delta(Z) \rightarrow 0} O(Z) = \lim_{\Delta(Z) \rightarrow 0} S(Z) = I$$

D Eine Funktion $f(x)$ heißt **über das Intervall $[a, b]$ (im Riemannschen Sinne) integrierbar** wenn es zu jedem $\epsilon > 0$ ein $\delta(\epsilon) > 0$ und ein I gibt, so daß für jede Zerlegung Z von $[a, b]$ mit $\Delta(Z) < \delta$ gilt:

$$|S(Z) - I| < \epsilon$$

I wird als **bestimmtes Integral** der Funktion $f(x)$ über $[a, b]$ bezeichnet.

$$I = \int_a^b f(x)dx = \lim_{\Delta(Z) \rightarrow 0} \sum_{i=1}^n f(x_i^*) (x_i - x_{i-1})$$

a, b : **untere, obere Integrationsgrenze**

x : **Integrationsvariable**

Die obige Definition stellt noch keine offensichtlichen Anforderungen an eine Funktion damit sie integrabel ist.

S Riemannsches Integrabilitätskriterium

Eine in $[a, b]$ definierte und beschränkte Funktion $f(x)$ ist genau dann über $[a, b]$ integrierbar wenn es zu jedem $\epsilon > 0$ ein $\delta(\epsilon) > 0$ gibt, so daß für jede Zerlegung Z mit $\Delta(Z) < \delta$ gilt:

$$|O(Z) - U(Z)| < \epsilon$$

Im Riemannschen Sinne integrabel sind **u.a.** :

- a) jede auf $[a, b]$ stetige Funktion
- b) jede beschränkte, in $[a, b]$ stückweise stetige Funktion (endlich viele Unstetigkeitsstellen)
- c) jede auf $[a, b]$ beschränkte und monotone Funktion

3.6.2 Eigenschaften des bestimmten Integrals

S Seien $f(x)$ und $g(x)$ auf den Intervallen $[a, b]$, $[a, c]$ und $[c, b]$ integrierbare Funktionen, dann gilt:

$$a) \int_a^b f(x)dx = 0 \quad \text{für } a = b$$

$$b) \int_a^b f(x)dx = - \int_b^a f(x)dx$$

$$c) \int_a^b f(x)dx = \int_a^c f(x)dx + \int_c^b f(x)dx$$

$$d) \int_a^b [\alpha f(x) + \beta g(x)] dx = \alpha \int_a^b f(x) dx + \beta \int_a^b g(x) dx$$

$$e) f(x) \leq g(x) \forall x \in [a, b] \quad \text{mit} \quad a \leq b \quad \Rightarrow \quad \int_a^b f(x) dx \leq \int_a^b g(x) dx$$

$$f) \left| \int_a^b f(x) dx \right| \leq \int_a^b |f(x)| dx \quad \text{für} \quad a \leq b$$

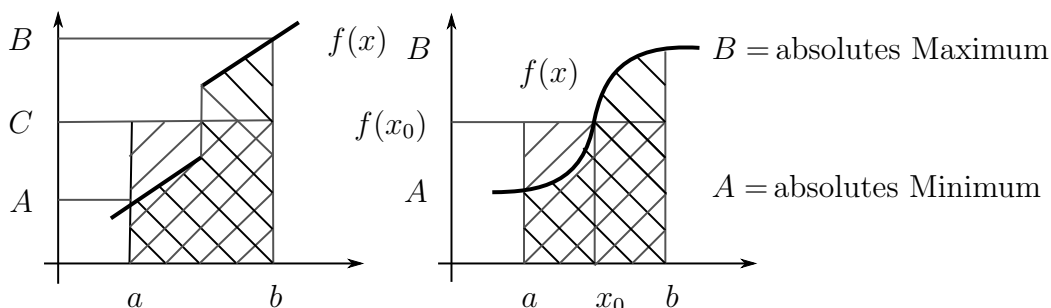
g) **erster Mittelwertsatz der Integralrechnung:**

Sei $f(x)$ auf $[a, b]$ mit $a \leq b$ integrierbar und gilt $A \leq f(x) \leq B \quad \forall x \in [a, b]$, dann existiert ein C mit $A \leq C \leq B$, so daß

$$\int_a^b f(x) dx = C (b - a)$$

Ist $f(x)$ stetig, so gibt es ein $x_0 \in [a, b]$ mit

$$\int_a^b f(x) dx = f(x_0) (b - a)$$



3.6.3 Stammfunktionen und unbestimmtes Integral

D Eine in $[a, b]$ differenzierbare Funktion $F(x)$ heißt **Stammfunktion** der Funktion $f(x)$ falls gilt

$$F'(x) = f(x) \quad \forall x \in [a, b]$$

Die Menge aller Stammfunktionen $F(x)$ wird als **unbestimmtes Integral** $\int f(x) dx$ bezeichnet.

Beispiel:

$$f(x) = \cos x \quad F(x) = \sin x$$

S Besitzt eine Funktion $f(x)$ zwei Stammfunktionen F und G , so unterscheiden sich diese nur um eine Konstante C , d.h.

$$\int f(x)dx = F(x) + C$$

B Annahme: F und G seien Stammfunktionen von f mit $F(x) = G(x) + h(x)$.

$$\begin{aligned} F'(x) &= f(x) \\ &= G'(x) + h'(x) \\ G'(x) &= f(x) \quad (G \text{ ist Stammfunktion}) \\ &= f(x) + h'(x) \\ \Rightarrow h'(x) &= 0 \\ \Rightarrow h(x) &= C \end{aligned}$$

Grundintegrale:

Funktion	Stammfunktion
$\int x^\alpha dx$	$= \frac{1}{\alpha+1} x^{\alpha+1} + C \quad \alpha \neq -1$
$\int \frac{1}{x} dx$	$= \ln x + C$
$\int e^{\alpha x} dx$	$= \frac{1}{\alpha} e^{\alpha x} + C \quad \alpha \neq 0$
$\int \sin(\alpha x) dx$	$= -\frac{1}{\alpha} \cos(\alpha x) + C \quad \alpha \neq 0$
$\int \cos(\alpha x) dx$	$= \frac{1}{\alpha} \sin(\alpha x) + C \quad \alpha \neq 0$

D Zu einer auf $[a, b]$ integrierbaren Funktion $f(x)$ ist die zugehörige **Integralfunktion** $I(x)$ durch

$$I : x \mapsto I(x) = \int_a^x f(s)ds \quad x \in [a, b]$$

definiert.

S Für die Integralfunktion $I(x)$ zu einer auf $[a, b]$ stetigen Funktion $f(x)$ gilt

$$I'(x) = f(x)$$

d. h. $\frac{d}{dx} \int_a^x f(s) ds = f(x)$

d. h. die Integralfunktion $I(x)$ ist eine Stammfunktion von $f(x)$.

B

$$\begin{aligned} \frac{d}{dx} I(x) &= \lim_{\Delta x \rightarrow 0} \frac{I(x + \Delta x) - I(x)}{\Delta x} = \lim_{\Delta x \rightarrow 0} \frac{\int_a^{x+\Delta x} f(s) ds - \int_a^x f(s) ds}{\Delta x} \\ &= \lim_{\Delta x \rightarrow 0} \frac{\int_x^{x+\Delta x} f(s) ds}{\Delta x} \end{aligned}$$

$$\begin{aligned} \underbrace{\lim_{\Delta x \rightarrow 0} \frac{\int_x^{x+\Delta x} f_{\min} ds}{\Delta x}}_* &\leq \lim_{\Delta x \rightarrow 0} \frac{\int_x^{x+\Delta x} f(s) ds}{\Delta x} \leq \underbrace{\lim_{\Delta x \rightarrow 0} \frac{\int_x^{x+\Delta x} f_{\max} ds}{\Delta x}}_{**} \\ \lim_{\Delta x \rightarrow 0} f_{\min} &\leq \lim_{\Delta x \rightarrow 0} \frac{\int_x^{x+\Delta x} f(s) ds}{\Delta x} \leq \lim_{\Delta x \rightarrow 0} f_{\max} \\ f(x) &\leq \lim_{\Delta x \rightarrow 0} \frac{\int_x^{x+\Delta x} f(s) ds}{\Delta x} \leq f(x) \\ \Rightarrow I'(x) &= \lim_{\Delta x \rightarrow 0} \frac{\int_x^{x+\Delta x} f(s) ds}{\Delta x} = f(x) \end{aligned}$$

* - monoton steigend, nach oben beschränkt

** - monoton fallend, nach unten beschränkt

S **Hauptsatz der Differential- und Integralrechnung**

Ist für eine in $[a, b]$ stetige Funktion $f(x)$ eine beliebige Stammfunktion $F(x)$ bekannt, so erhält man das bestimmte Integral $\int_a^b f(x) dx$ durch

$$\int_a^b f(x) dx = F(b) - F(a) = [F(x)]_a^b$$

Damit ist die Berechnung eines bestimmten Integrals auf die Bestimmung der Stammfunktion zurückgeführt.

B

$$\begin{aligned}
 \int_a^b f(s)ds &= \int_a^d f(s)ds + \int_d^b f(s)ds \\
 &= -\int_d^a f(s)ds + \int_d^b f(s)ds \\
 &= I(b) - I(a) \\
 &= [F(b) - C] - [F(a) - C] \\
 &= F(b) - F(a)
 \end{aligned}$$

denn die Stammfunktion $I(x)$ unterscheidet sich von $F(x)$ nur durch eine Konstante C (Hier: $I(x) = \int_a^x f(s)ds$).

Berechnung bestimmter Integrale:

$$\begin{aligned}
 \int_a^b x^\gamma dx &= \left[\frac{1}{\gamma+1} x^{\gamma+1} \right]_a^b \\
 &= \frac{1}{\gamma+1} [b^{\gamma+1} - a^{\gamma+1}] \\
 \int_a^b \sinh x dx &= \frac{1}{2} \int_a^b [e^x - e^{-x}] dx \\
 &= \frac{1}{2} [e^x + e^{-x}]_a^b \\
 &= [\cosh x]_a^b \\
 &= \cosh(b) - \cosh(a)
 \end{aligned}$$

CAS-Beispiel



Die Funktion `integrate` erlaubt sowohl bestimmte als auch unbestimmte Integrale zu berechnen. Hier einige Beispiele:

```

from sympy import integrate
from sympy.abc import x, a, b

integrate(x**3, x, a, b)

```

```
from sympy import sinh
integrate(sinh(x), (x, a, b))
```

Stimmt das Ergebnis? Abschließend einige unbestimmte Integrale. Der natürliche Logarithmus wird dabei von SymPy als $\log(x)$ dargestellt:

```
from sympy import tan, log, exp
integrate(tan(x), x)
```

```
integrate(log(x), x)
```

```
integrate(x*exp(x), x)
```

3.6.4 Integrationsregeln

Durch Ausnutzen der Differentiationsregeln erhält man eine Reihe von Regeln zum Auffinden einer Stammfunktion und damit zur Bestimmung eines bestimmten Integrals.

Im folgenden, seien $F(x)$ und $G(x)$ Stammfunktionen von $f(x)$ und $g(x)$.

S Summenregel

$$\int [\alpha f(x) + \beta g(x)] dx = \alpha F(x) + \beta G(x) + C$$

Aus

$$(F G)' = F'G + F G'$$

Produktregel folgt:

S

$$\int_a^b F'(x) G(x) dx = [F(x) G(x)]_a^b - \int_a^b F(x) G'(x) dx$$

partielle Integration oder

$$\int_a^b f(x) G(x) dx = [F(x) G(x)]_a^b - \int_a^b F(x) g(x) dx$$

Beispiel:

Anwendung auf ein bestimmtes Integral

$$\begin{aligned}
 \int_a^b x \cos x dx &= [x \sin x]_a^b - \int_a^b 1 \sin x dx \\
 &= [x \sin x]_a^b - [-\cos x]_a^b \\
 &= b \sin b - a \sin a + \cos b - \cos a
 \end{aligned}$$

Anwendung auf ein unbestimmtes Integral

$$\begin{aligned}
 \int x^2 e^x dx &= x^2 e^x - \int 2x e^x dx + C \\
 &= x^2 e^x - 2 \left[x e^x - \int e^x dx \right] + C \\
 &= x^2 e^x - 2x e^x + 2e^x + C \\
 &= (x^2 - 2x + 2) e^x + C = H(x)
 \end{aligned}$$

Kontrolle:

$$\begin{aligned}
 \text{Integrand } h(x) &= x^2 e^x \\
 \text{Stammfunktion } H(x) &= (x^2 - 2x + 2) e^x \\
 \frac{dH(x)}{dx} &= (x^2 - 2x + 2) e^x + (2x - 2) e^x \\
 &= x^2 e^x = h(x)
 \end{aligned}$$

Kettenregel \Rightarrow **Substitutionsregel:**

S Ist $g(u)$ auf $[a, b]$ stetig differenzierbar und $f(v)$ stetig für alle $v \in \{v | v = g(u), u \in [a, b]\}$ dann gilt:

$$\int_a^b f(g(u)) g'(u) du = \int_{g(a)}^{g(b)} f(v) dv$$

B Die Integralfunktion $I(x) = \int_{\alpha}^x f(v) dv$ (α beliebig) ist eine Stammfunktion von $f(v)$, d.h. $I'(x) = f(x)$. Die Verkettung $H = I \circ g$ von I und g besitzt die Ableitung

$$H'(u) = (I \circ g)'(u) = \frac{dI}{dg} \frac{dg}{du}$$

$$\begin{aligned}
 &= I'(g(u)) g'(u) \\
 &= f(g(u)) g'(u)
 \end{aligned}$$

d.h. $I \circ g$ ist Stammfunktion von $f g'$.

$$\begin{aligned}
 \int_a^b f(g(u))g'(u)du &= [(I \circ g)(u)]_a^b \\
 &= [I(g(u))]_a^b \\
 &= I(g(b)) - I(g(a)) \\
 &= [I(v)]_{g(a)}^{g(b)} \\
 &= \int_{g(a)}^{g(b)} f(v)dv
 \end{aligned}$$

S Ist $F(u)$ Stammfunktion von $f(u)$, so gilt

$$\int_{\alpha}^{\beta} f(ax + b)dx = \frac{1}{a} [F(ax + b)]_{\alpha}^{\beta}$$

B

$$\begin{aligned}
 \int_{\alpha}^{\beta} f(ax + b)dx &= \frac{1}{a} \int_{\alpha}^{\beta} f(ax + b) a dx \\
 & \quad g(x) = ax + b \\
 \int_{\alpha}^{\beta} f(ax + b)dx &= \frac{1}{a} [F(g(x))]_{g(\alpha)}^{g(\beta)} \\
 &= \frac{1}{a} [F(ax + b)]_{\alpha}^{\beta}
 \end{aligned}$$

S Ist $g(x)$ in $[a, b]$ differenzierbar mit $g(x) \neq 0$, so gilt

$$\begin{aligned}
 \int_a^b \frac{g'(x)}{g(x)} dx &= [\ln |g(x)|]_a^b \\
 &= [\ln |v|]_{g(a)}^{g(b)}
 \end{aligned}$$

$$\boxed{\mathbf{B}} \quad f(g) = \frac{1}{g}, \quad F(g) = \ln |g|$$

$$\begin{aligned} \int_a^b [g(x)]^{-1} g'(x) dx &= [F(v)]_{g(a)}^{g(b)} \\ &= [\ln |v|]_{g(a)}^{g(b)} \\ &= [\ln |g(x)|]_a^b \end{aligned}$$

Beispiel:

$$\begin{aligned} \int_a^b (x^2 + 1)^2 x dx &= \frac{1}{2} \int_a^b (x^2 + 1)^2 2x dx \\ &g(x) = x^2 + 1 \quad f(g) = g^2 \\ &= \frac{1}{2} \int_{g(a)}^{g(b)} f(v) dv \\ &= \frac{1}{2} \int_{g(a)}^{g(b)} v^2 dv \\ &= \frac{1}{2} \frac{1}{3} [v^3]_{g(a)}^{g(b)} \\ &= \frac{1}{6} [(x^2 + 1)^3]_a^b \end{aligned}$$

Formal läßt sich schreiben

$$\begin{aligned} \int_a^b f(g(x)) g'(x) dx &= \int_a^b f(g(x)) \frac{dg(x)}{dx} dx \\ &= \int_{g(a)}^{g(b)} f(g) dg \\ &= \int_{g(a)}^{g(b)} f(v) dv \end{aligned}$$

⇒ Achtet man auf die Integrationsgrenzen, so läßt sich mit den Differentialen nach den üblichen Rechenregeln verfahren.

$$T = \int_0^\pi \cos \theta e^{-\alpha \cos \theta} \sin \theta d\theta$$

$$u = \cos \theta$$

$$\begin{aligned}\pi &\rightarrow \cos \pi = -1 \\ 0 &\rightarrow \cos 0 = 1 \\ -\frac{du}{d\theta} &= \sin \theta \\ du &= -\sin \theta d\theta\end{aligned}$$

alternativ

$$\begin{aligned}v &= -\alpha \cos \theta \\ \frac{dv}{d\theta} &= \alpha \sin \theta \\ dv &= \alpha \sin \theta d\theta\end{aligned}$$

$$\begin{aligned}\Rightarrow T &= -\int_{u(\theta=0)}^{u(\theta=\pi)} u e^{-\alpha u} du \\ &= -\int_{+1}^{-1} u e^{-\alpha u} du \\ &= \int_{-1}^{+1} u e^{-\alpha u} du \\ &= \left[u \left(\frac{1}{-\alpha} e^{-\alpha u} \right) \right]_{-1}^{+1} - \int_{-1}^{+1} -\frac{1}{\alpha} e^{-\alpha u} du \\ &= -\frac{1}{\alpha} \left[+1 e^{-\alpha \cdot 1} - (-1) e^{-\alpha \cdot (-1)} \right] - \left[\left(\frac{1}{-\alpha} \right)^2 e^{-\alpha u} \right]_{-1}^{+1} \\ &= -\frac{1}{\alpha} \left[e^{-\alpha} + e^{\alpha} \right] - \left(\frac{1}{\alpha^2} \right) \left[e^{-\alpha} - e^{\alpha} \right] \\ &= -\frac{2}{\alpha} \left[\cosh \alpha - \frac{1}{\alpha} \sinh \alpha \right]\end{aligned}$$

$$\alpha = \frac{m B}{k T} :$$

$$\begin{aligned}\langle M \rangle &= m \langle \cos \theta \rangle \\ &= m \frac{\int \cos \theta e^{-\cos \theta \frac{m B}{k T}} \sin \theta d\theta}{\int e^{-\cos \theta \frac{m B}{k T}} \sin \theta d\theta} \\ &= m \frac{\left(-\frac{2}{\alpha} \right) \left[\cosh \alpha - \frac{1}{\alpha} \sinh \alpha \right]}{\left(-\frac{2}{\alpha} \right) \sinh \alpha} \\ \langle M \rangle &= m \left[\coth \alpha - \frac{1}{\alpha} \right]\end{aligned}$$

Langevin-Funktion:

$$\begin{aligned}L(\alpha) &= -\frac{1}{\alpha} + \coth \alpha \\ \alpha \ll 1 \\ L(\alpha) &\simeq \frac{\alpha}{3}\end{aligned}$$

$$\begin{aligned}\Rightarrow \langle M \rangle &\simeq \frac{m^2}{3k T} B \\ \Rightarrow \chi &= \frac{m^2}{3k T}\end{aligned}$$

3.6.5 Integration gebrochen rationaler Funktionen

Die Integration einer gebrochen rationalen Funktion

$$f(x) = \frac{\sum_{j=0}^m b_j x^j}{\sum_{i=0}^n a_i x^i}$$

erfolgt i. allg. in drei Schritten:

- a) Zerlegung der gebrochen rationalen Funktion in ein Polynom und eine echt gebrochen rationale Funktion – falls $m \geq n$.
- b) **Partialbruchzerlegung** der echt gebrochen rationalen Funktion

$$h(x) = \frac{\sum_{j=0}^{m'} c_j x^j}{\sum_{i=0}^n a_i x^i}$$

- c) Integration des Polynoms und der Partialbrüche unter Verwendung von Standardintegralen

Schritt a) "Kürzen" falls $m \geq n$, z.B.:

$$\begin{aligned}f(x) &= \frac{x^4 + 2x^2 + 3}{x^2 + 1} \\ &= x^2 + 1 + \frac{2}{x^2 + 1}\end{aligned}$$

Schritt b) Partialbruchzerlegung

S "Zur Erinnerung" *Fundamentalsatz der Algebra:*

Jede ganze rationale Funktion n -ten Grades mit reellen Koeffizienten a_i lässt sich auf die Form bringen:

$$\sum_{i=0}^n a_i x^i = c(x - x_1)(x - x_2) \dots (x - x_k)(x^2 + p_1 x + q_1) \dots (x^2 + p_l x + q_l)$$

wobei gilt: $k + 2l = n$, $c, x_i, q_i, p_i \in \mathbb{R}$.

Fallen u bzw. v der Nullstellen zusammen, so läßt sich schreiben

$$\sum_{i=0}^n a_i x^i = c(x - x_1)^{u_1} \dots (x - x_s)^{u_s} (x^2 + p_1 x + q_1)^{v_1} \dots (x^2 + p_t x + q_t)^{v_t}$$

mit

$$\underbrace{u_1 + \dots + u_s}_{s \text{ Summanden}} + 2 \underbrace{(v_1 + \dots + v_t)}_{t \text{ Summanden}} = n$$

d.h. jeweils verschiedene Nullstellen.

S Jede echt gebrochene rationale Funktion $f(x)$ läßt sich als Summe von Termen der Form

$$\frac{A_1}{x - x_i} + \frac{A_2}{(x - x_i)^2} + \dots + \frac{A_{u_i}}{(x - x_i)^{u_i}} \quad i = 1, \dots, s$$

und Termen der Form

$$\frac{B_1 x + C_1}{x^2 + p_i x + q_i} + \dots + \frac{B_{v_i} x + C_{v_i}}{(x^2 + p_i x + q_i)^{v_i}} \quad i = 1, \dots, t$$

darstellen. Die Konstanten A , B und C lassen sich durch Koeffizientenvergleich ermitteln.

Beispiel:

$$\begin{aligned} \frac{x+2}{x^2-1} &\stackrel{\text{a)}}{=} \frac{x+2}{(x-1)(x+1)} \\ &\stackrel{\text{b)}}{=} \frac{A_1}{x-1} + \frac{A_2}{x+1} \\ \Rightarrow 1 \cdot x + 2 &= A_1(x+1) + A_2(x-1) \\ &= (A_1 + A_2)x + (A_1 - A_2) \\ \Rightarrow A_1 + A_2 &= 1 \\ A_1 - A_2 &= 2 \\ A_1 &= 3/2 \\ A_2 &= -1/2 \end{aligned}$$

Schritt c) Integration der Partialbrüche

Nach erfolgter Partialbruchzerlegung treten nur 3 verschiedene Funktionstypen als Integrand auf.

$$\alpha) \quad \int \frac{A}{x - x_i} dx = A \ln |x - x_i| + \text{const.} \quad \text{für } n = 1$$

$$\beta) \quad \int \frac{A}{(x - x_i)^n} dx = \frac{A}{(1 - n)} (x - x_i)^{1-n} + \text{const.} \quad \text{für } n \geq 2$$

$$\gamma) \quad \int \frac{Bx + C}{(x^2 + px + q)^n} dx =$$

Substitution $u = x + p/2 \Rightarrow$

$$= \int \frac{Bu - Bp/2 + C}{(u^2 - pu + p^2/4 + pu - p^2/2 + q)^n} du$$

$$= \int \frac{Bu + C'}{(u^2 + k^2)^n} du$$

mit $C' = C - Bp/2$

$$k^2 = q - p^2/4 > 0$$

$$= \frac{1}{k^{2n}} \int \frac{Bku/k + C'}{\left(\left(\frac{u}{k}\right)^2 + 1\right)^n} d\left(\frac{u}{k}\right) k$$

$$= \underbrace{B'' \int \frac{v dv}{(1 + v^2)^n}}_{I_1} + \underbrace{C'' \int \frac{dv}{(1 + v^2)^n}}_{I_2}$$

mit $B'' = B k^{-2n+2}$

$$C'' = C' k^{-2n+1}$$

$$= (C - B p/2) k^{-2n+1}$$

$$v = \frac{u}{k}$$

$$I_1 = \frac{1}{2} \int \frac{2v dv}{(1 + v^2)^n}$$

$$= \frac{1}{2} \ln(1 + v^2) + \text{const.} \quad \text{für } n = 1$$

$$I_1' = \frac{1}{2} \int \frac{2v dv}{(1 + v^2)^n}$$

$$= \frac{1}{2} (1 - n)^{-1} (1 + v^2)^{1-n} + \text{const.} \quad \text{für } n \geq 2$$

$$I_2 = \int \frac{dv}{(1 + v^2)^n}$$

$$= \arctan v + \text{const.} \quad \text{für } n = 1$$

für $n \geq 2$: $I_2' = \int \frac{dv}{(1 + v^2)^n}$

$$\begin{aligned}
&= \int \frac{(1+v^2)dv}{(1+v^2)^{n+1}} \\
&= \int \frac{dv}{(1+v^2)^{n+1}} + \frac{1}{2} \int \frac{v \cdot 2v}{(1+v^2)^{n+1}} dv \\
\frac{1}{2} \int \frac{v \cdot 2v}{(1+v^2)^{n+1}} dv &= -\frac{1}{2n} \left[v \frac{1}{(1+v^2)^n} \right] + \frac{1}{2} \frac{1}{n} \int \frac{dv}{(1+v^2)^n}
\end{aligned}$$

$n \rightarrow n-1$

$$\begin{aligned}
\Rightarrow \int \frac{dv}{(1+v^2)^{n-1}} &= \int \frac{dv}{(1+v^2)^n} - \frac{1}{2(n-1)} \left[v \frac{1}{(1+v^2)^{n-1}} \right] \\
&+ \frac{1}{2} \frac{1}{n-1} \int \frac{dv}{(1+v^2)^{n-1}} \\
\Rightarrow I'_2 &= \frac{1}{2(n-1)} \left[v \frac{1}{(1+v^2)^{n-1}} \right] + \int \frac{dv}{(1+v^2)^{n-1}} \left[1 - \frac{1}{2} \frac{1}{n-1} \right] \\
I'_2 &= \frac{1}{2(n-1)} \frac{v}{(1+v^2)^{n-1}} + \frac{2n-3}{2n-2} \underbrace{\int \frac{dv}{(1+v^2)^{n-1}}}_{\text{vom Typ } I_2 \text{ oder } I'_2}
\end{aligned}$$

3.6.6 Uneigentliche Integrale

Bei der Definition des bestimmten Integrals $\int_a^b f(x)dx$ (siehe 3.6.1) wurde bislang vorausgesetzt, daß gilt:

- die Funktion $f(x)$ ist innerhalb $[a, b]$ beschränkt
- der Integrationsbereich $[a, b]$ ist endlich, d.h. $a, b \neq \pm\infty$

Beide Einschränkungen können bei geeigneter Erweiterung des Integrationsbegriffs entfallen.

Zu a):

D *Es sei $f(x)$ eine in $[a, b-\epsilon]$ ($0 < \epsilon < b-a$) beschränkte und integrierbare Funktion. Existiert der Grenzwert*

$$I = \lim_{\epsilon \rightarrow 0} \int_a^{b-\epsilon} f(x)dx$$

so heißt I (**konvergentes**) **uneigentliches Integral** von $f(x)$ auf $[a, b]$.

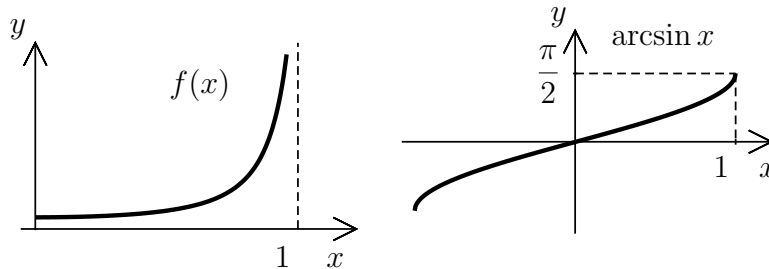
Symbolisch: $I = \int_a^b f(x)dx$

Existiert der Grenzwert nicht, so spricht man von einem (**divergentem**) **uneigentlichen Integral**.

Analog läßt sich die Definition ausdehnen für den Fall, daß $f(x)$ für $x = a$ oder für $x = a$ und $x = b$ nicht beschränkt ist.

Beispiel:

$$\begin{aligned} \int_0^1 \frac{1}{\sqrt{1-x^2}} dx &= \lim_{\epsilon \rightarrow 0} \int_0^{1-\epsilon} \frac{1}{\sqrt{1-x^2}} dx \\ &= \lim_{\epsilon \rightarrow 0} [\arcsin x]_0^{1-\epsilon} \\ &= \lim_{\epsilon \rightarrow 0} [\arcsin(1-\epsilon) - \arcsin 0] \\ &= \arcsin 1 \\ &= \frac{\pi}{2} \end{aligned}$$



Entsprechend werden $\int_{-1}^0 \frac{1}{\sqrt{1-x^2}}$ und $\int_{-1}^{+1} \frac{1}{\sqrt{1-x^2}}$ behandelt.

Zu b):

D Sei $f(x)$ eine beschränkte und integrable Funktion für $x \in [a, b]$ mit beliebigem $b > a$.
Existiert der Grenzwert

$$I = \lim_{b \rightarrow \infty} \int_a^b f(x) dx$$

so heißt I (**konvergentes**) **uneigentliches Integral** von $f(x)$ über $[a, \infty[$.

Symbolisch: $I = \int_a^\infty f(x) dx$

Existiert der Grenzwert nicht, so heißt $\int_a^\infty f(x) dx$ **divergent**.

Analog läßt sich die Definition erweitern für den Fall daß $a = -\infty$ bzw. $a = -\infty$ und $b = \infty$.

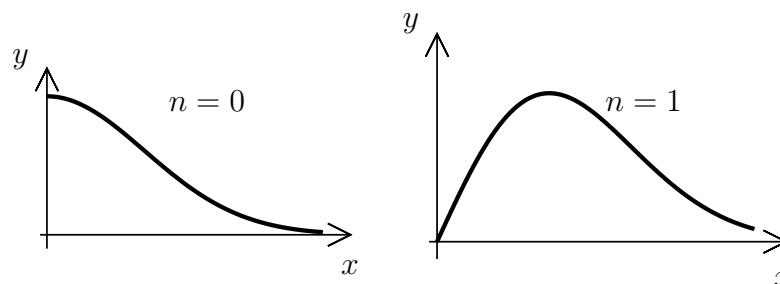
Beispiel:

$$\int_0^{\infty} x^n e^{-\beta x^2} dx$$

Es sei $0 \leq n \in \mathbb{N}$, $0 < \beta \in \mathbb{R}$.

Substitution $\sqrt{\beta} x = u$ führt auf die Form

$$\int_0^{\infty} u^n e^{-u^2} du$$



$$n = 0 : \quad \int_0^{\infty} e^{-u^2} du = \frac{\sqrt{\pi}}{2} \quad (\text{Beweis folgt später})$$

$$n = 1 : \quad \int_0^{\infty} u e^{-u^2} du = \frac{1}{2}$$

$$\text{mit } u^2 = s \quad \text{und} \quad \frac{ds}{du} = 2u$$

$$\begin{aligned} \int_0^{\infty} u e^{-u^2} du &= \int_0^{\infty} [e^{-s}] \frac{1}{2} ds \\ &= \frac{1}{2} [-e^{-s}]_0^{\infty} \\ &= -\frac{1}{2} [-0 - (-1)] \\ &= \frac{1}{2} \end{aligned}$$

$n > 1$: Partielle Integration

$$\begin{aligned} \int_0^{\infty} u^n e^{-u^2} du &= -\frac{1}{2} \int_0^{\infty} u^{n-1} (-2u e^{-u^2}) du \\ &= -\frac{1}{2} \left\{ [u^{n-1} e^{-u^2}]_0^{\infty} - \underbrace{\int_0^{\infty} (n-1) u^{n-2} e^{-u^2} du}_{\text{entweder gilt } n-2=0 \text{ oder } n-2=1} \right\} \end{aligned}$$

entweder gilt $n-2=0$ oder $n-2=1$
ansonsten weiter partielle Integration

CAS-Beispiel



Für die Berechnung uneigentlicher Integrale mittels Python sind keine Besonderheiten zu beachten. Diese kann mittels SciPy oder wahlweise mittels SymPy erfolgen.

$$\int_0^1 \frac{1}{\sqrt{1+x^2}} dx:$$

```
from scipy import integrate as int1, sqrt as sqrt1  
  
int1.quad(lambda x: 1/sqrt1(1-x**2), 0, 1)
```

```
from sympy import integrate as int2, sqrt as sqrt2  
from sympy.abc import x  
  
int2(1/sqrt2(1-x**2), (x, 0, 1))
```

$$\int_{-1}^1 \frac{1}{\sqrt{1+x^2}} dx:$$

```
from scipy import integrate, sqrt  
  
integrate.quad(lambda x: 1/sqrt(1-x**2), -1, 1)
```

$$\int_0^\infty e^{-x^2} dx:$$

```
from scipy import exp, inf  
  
integrate.quad(lambda x: exp(-x**2), 0, inf)
```

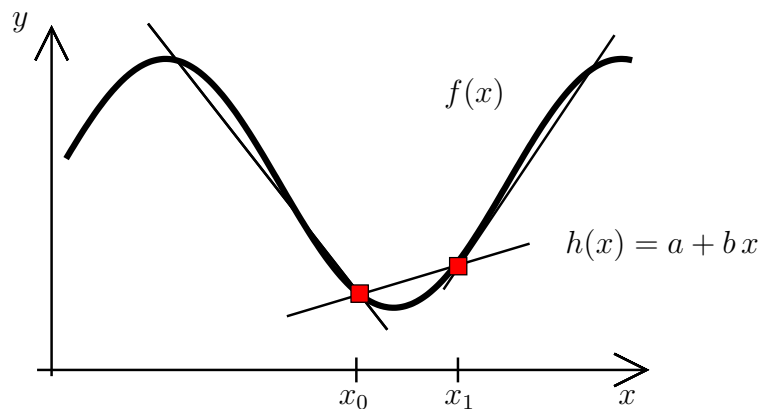

3.6.7 Interpolation und numerische Integration

a) Interpolation

In vielen Fällen ist die Auswertung einer betrachteten Funktion sehr aufwendig oder nicht zugänglich (Meßwerte). Die Interpolation bietet die Möglichkeit die Lücke zwischen bekannten Stützstellen zu "überbrücken".

Lineare Interpolation

Die Funktion $f(x)$ wird zwischen jeweils zwei Stützstellen (x_0 und x_1 mit $y_0 = f(x_0)$ und $y_1 = f(x_1)$) durch eine Gerade $h(x)$ ersetzt, so daß $h(x_i) = y_i, i = 0, 1$.



Bestimmung von a und b :

2-Punkteform

$$\frac{y - y_0}{x - x_0} = \frac{y_1 - y_0}{x_1 - x_0}$$

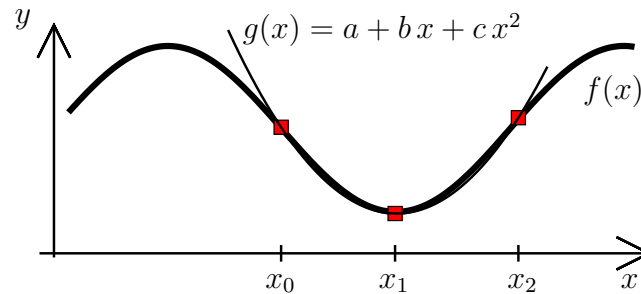
$$\Rightarrow h(x) = y = \underbrace{\left[y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0 \right]}_{= a} + \underbrace{\left[\frac{y_1 - y_0}{x_1 - x_0} \right]}_{= b} x$$

alternativ

$$\begin{aligned} h(x_0) &= a + b x_0 = y_0 \\ h(x_1) &= a + b x_1 = y_1 \\ \Rightarrow b(x_1 - x_0) &= y_1 - y_0 \\ b &= \frac{y_1 - y_0}{x_1 - x_0} \\ a &= y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0 \end{aligned}$$

Quadratische Interpolation

Die Funktion $f(x)$ wird im Bereich von drei Stützstellen (x_i mit $y_i = f(x_i), i = 0, 1, 2$) durch eine Parabel $g(x)$ ersetzt, so daß gilt $g(x_i) = y_i, i = 0, 1, 2$.



Bestimmung von a , b und c :

$$g(x_i) = a + bx_i + cx_i^2 = y_i \quad i = 0, 1, 2$$

d.h. man erhält ein lineares Gleichungssystem für die Koeffizienten a , b , c

$$\begin{pmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \end{pmatrix}$$

Speziell: Äquidistante Stützstellen mit Nullpunktverschiebung, d.h. $x_0 = -\Delta x$, $x_1 = 0$, $x_2 = +\Delta x \Rightarrow$

$$\begin{aligned} a - \Delta x b + \Delta x^2 c &= y_0 \\ a &= y_1 \\ a + \Delta x b + \Delta x^2 c &= y_2 \end{aligned}$$

$$\Rightarrow a = y_1 \quad b = \frac{y_2 - y_0}{2 \Delta x} \quad c = \frac{y_2 + y_0 - 2y_1}{2 \Delta x^2}$$

Interpolation mittels Polynomen höherer Ordnung:

Formel von Lagrange

$$I_n(x) = \sum_{i=0}^n y_i L_i^n(x)$$

mit den Polynomen

$$L_i^n(x) = \frac{(x - x_0)(x - x_1) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_n)}{(x_i - x_0)(x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)}$$

für $(n + 1)$ Stützstellen (x_i, y_i) $i = 0, \dots, n$.

b) Numerische Integration

Numerische (rechnerische) Integrationstechniken zielen darauf ab, einen Näherungswert für ein bestimmtes Integral $\int_a^b f(x)dx$ zu bestimmen.

Für eine gewählte Zerlegung Z des Intervalls $[a, b]$ liefern die zugehörigen Ober- und Untersummen einen Näherungswert für I .

Offensichtlich gilt für den Fehler ΔI , d.h. die Abweichung vom gesuchten Integral

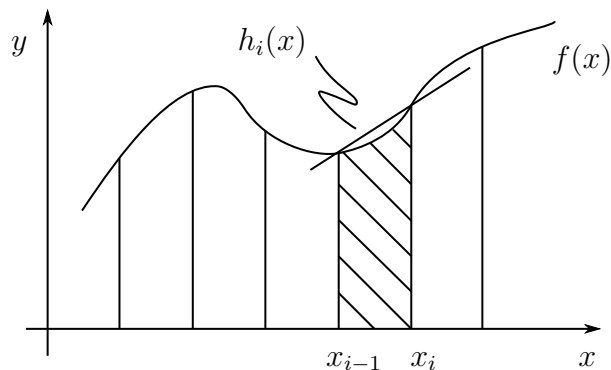
$$|\Delta I| < |O(Z) - U(Z)|$$



Eine numerische Integrationsformel sollte bei Verfeinerung der Zerlegung zu immer kleineren Fehlern führen und die Möglichkeit beinhalten den Fehler ΔI zu bestimmen.

Im folgenden wird – wie dies üblich ist – von einer äquidistanten Zerlegung Z ausgegangen.

Eine stückweise lineare Approximation des Integranden führt bei Zerlegung in n äquidistante Intervalle $[x_{i-1}, x_i = x_{i-1} + \Delta x]$ auf die **Trapezregel**.

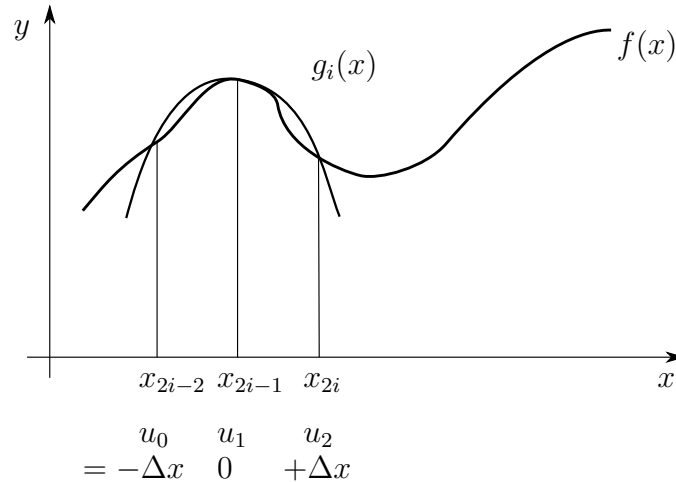


$$\begin{aligned} \int_a^b f(x)dx &\simeq \sum_{i=1}^n \int_{x_{i-1}}^{x_i} h_i(x)dx = \sum_{i=1}^n I_i \\ &= \Delta x \left(\frac{1}{2}f_0 + f_1 + \dots + f_{n-1} + \frac{1}{2}f_n \right) \end{aligned}$$

Segment i :

$$\begin{aligned} I_i &= \int_{x_{i-1}}^{x_i} h_i(x)dx \\ &= \Delta x (f_{i-1} + f_i) \frac{1}{2} \\ \text{mit } f_i &= f(x_i) \end{aligned}$$

Eine quadratische Approximation des Integranden führt bei Zerlegung in n äquidistante (n gerade!) Intervalle auf die **Simpsonregel**.



$$\begin{aligned} \int_a^b f(x) dx &\simeq \sum_{i=1}^{n/2} \int_{x_{2i-2}}^{x_{2i}} g_i(x) dx = \sum_{i=1}^n I_i \\ &= \frac{\Delta x}{3} (f_0 + 4f_1 + 2f_2 + 4f_3 + \dots + 2f_{n-2} + 4f_{n-1} + f_n) \end{aligned}$$

Segment i :

$$\begin{aligned} I_i &= \int_{x_{2i-2}}^{x_{2i}} g_i(x) dx \\ &= \int_{u_0}^{u_2} (a + bu + cu^2) du \\ &= \left[au + \frac{1}{2}bu^2 + \frac{1}{3}cu^3 \right]_{u_0}^{u_2} \\ &= a(u_2 - u_0) + \frac{1}{2}b(u_2^2 - u_0^2) + \frac{1}{3}c(u_2^3 - u_0^3) \\ &= 2a\Delta x + 0 + \frac{2}{3}c\Delta x^3 \\ &= 2y_1\Delta x + \frac{2}{3} \frac{y_2 + y_0 - 2y_1}{2\Delta x^2} \Delta x^3 \\ &= \frac{\Delta x}{3} (y_0 + 4y_1 + y_2) \end{aligned}$$

CAS-Beispiel



Im Folgenden wird das Integral der Funktion $f(x) = x \cos^2(x)$ im Intervall $[-1, 4]$ durch die Trapezregel und die Simpsonregel in immer kleiner werdenden Teilintervallen approximiert, bis der Fehler einen Wert von 10^{-5} unterschreitet. Dazu wird die Funktion

`NUMINT(fexpr, var, a, b, eps)` verwendet, die vom `mathchem`-Modul zur Verfügung gestellt wird. Hierbei wird die Anzahl der Teilintervalle schrittweise um 10 (Trapezregel) respektive 2 (Simpsonregel) erhöht:

Achtung: Je nach `eps`-Wert und Computer kann die nachfolgende Rechnung auch mehrere Minuten in Anspruch nehmen!

```
from sympy.abc import x
from sympy import cos

def f(x):
    return x*(cos(x))**2
```

```
from mathchem import NUMINT

NUMINT(f(x), x, -1, 4, 1.0e-5)
```

Wie man sieht, wird der Fehler für die Simpsonregel rasch klein – im Gegensatz zur einfachen Trapezregel.

3.7 Potenzreihenentwicklung von Funktionen

3.7.1 Taylorentwicklung

Polynome sind in der Anwendung, d.h. Funktionswertbestimmung, Differentiation und Integration besonders angenehm. Interpolation, bzw. Extrapolation, bieten die Möglichkeit, eine vorgegebene Funktion durch ein Näherungspolynom zu ersetzen. Das Polynom n -ten Grades wird dann durch $n + 1$ Stützstellen bestimmt. Alternativ läßt sich fordern, daß das Näherungspolynom in den n ersten Ableitungen (sofern diese existieren) an einem vorgegebenen Punkt mit denen der Funktion übereinstimmen.

Es soll nun eine Funktion $f(x)$, die n -fach differenzierbar ist, betrachtet werden und in der Nähe von $x_0 = 0$ durch ein Näherungspolynom $T_n(x)$ ersetzt werden.

Näherungspolynom:

$$T_n(x) = \sum_{i=0}^n a_i x^i$$

Bestimmung der Koeffizienten a_i :

$$\begin{aligned} T_n(x) &= \sum_{i=0}^n a_i x^i && \Rightarrow T_n(0) = a_0 \stackrel{!}{=} f(0) \\ T'_n(x) &= \sum_{i=0}^n i a_i x^{i-1} = \sum_{i=1}^n i a_i x^{i-1} && \Rightarrow T'_n(0) = 1 \cdot a_1 \stackrel{!}{=} f'(0) \\ T''_n(x) &= \sum_{i=1}^n i(i-1) a_i x^{i-2} = \sum_{i=2}^n i(i-1) a_i x^{i-2} && \Rightarrow T''_n(0) = 2 \cdot 1 \cdot a_2 \stackrel{!}{=} f''(0) \\ &\vdots \\ T_n^{(n)}(x) &= \sum_{i=n-1}^n i(i-1)\dots 1 a_i x^{i-n} \\ &= \sum_{i=n}^n i(i-1)\dots 1 a_i x^{i-n} && \Rightarrow T_n^{(n)}(0) = n \cdot \dots \cdot 2 \cdot 1 \cdot a_n \stackrel{!}{=} f^{(n)}(0) \end{aligned}$$

Damit

$$T_n(x) = \sum_{i=0}^n \frac{1}{i!} f^{(i)}(0) x^i$$

bzw. für allgemeine x_0 :

D Sei $f(x)$ bei x_0 mindestens n -mal differenzierbar, so bezeichnet man

$$T_{n,x_0}(x) = \sum_{i=0}^n \frac{1}{i!} f^{(i)}(x_0) (x - x_0)^i$$

als **Taylorpolynom** zu $f(x)$ bei x_0 .

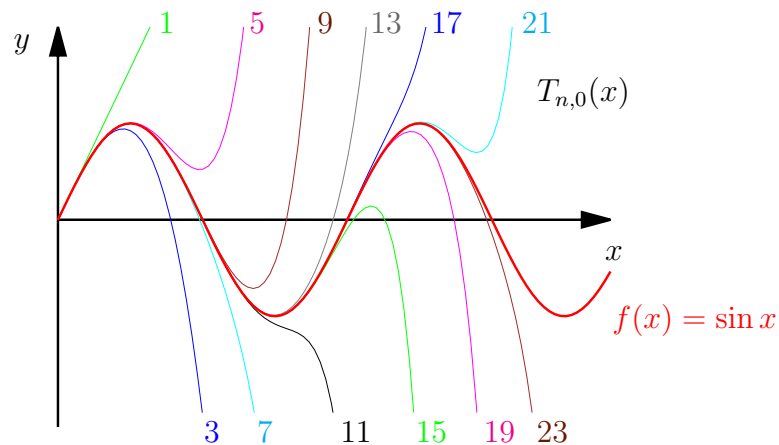
Wie gut repräsentiert das Taylorpolynom $T_{n,x_0}(x)$ die ursprüngliche Funktion $f(x)$?

Beispiel:

$$f(x) = \sin x, \quad x_0 = 0 :$$

$$\begin{aligned} f(0) &= \sin(0) = 0 \\ f'(0) &= \cos(0) = 1 \\ f''(0) &= -\sin(0) = 0 \\ f^{(3)}(0) &= -\cos(0) = -1 \\ f^{(4)}(0) &= \sin(0) = 0 \\ f^{(5)}(0) &= \cos(0) = 1 \end{aligned}$$

$$\begin{aligned} T_{n,0}(x) &= 0 + \frac{x}{1!} + 0 - \frac{x^3}{3!} + 0 + \frac{x^5}{5!} - \dots \\ &= \sum_{i=0}^m \frac{(-1)^i x^{2i+1}}{(2i+1)!} \quad \text{mit } 2m+1 = n \end{aligned}$$



CAS-Beispiel



Mittels des Aufrufes von `taylor(f(x), x, a, n)` des `mathchem`- Moduls kann man die Taylorreihenentwicklung von $f(x)$ um die Stelle a bis zur Ordnung n erzeugen.

```
from mathchem import taylor
from sympy import sin, cos, exp
from sympy.abc import x

taylor(sin(x), x, 0, 9)
```

```
taylor(sin(x), x, 1, 9)
```

```
taylor(cos(x), x, 0, 9)
```

```
taylor(exp(x), x, 0, 9)
```

```
taylor(sin(x), x, 0, 9)
```

```
taylor(sin(x), x, 3, 9)
```

Der folgende Aufruf stellt die Taylorpolynome fünfter Ordnung für $f(x) = \sin(x)$ an den Stellen $x_1 = 0, x_2 = 3$ dar.

```
import matplotlib.pyplot as plt
import numpy as np

x1 = 0
x2 = 3

xrange = np.linspace(-1, 10)

plt.plot(xrange, np.sin(xrange), "black")
plt.plot(x1, np.sin(x1), "bo")
plt.plot(xrange,
         [taylor(sin(x), x, x1, 5).subs(x, xi) for xi in xrange], "b-")
plt.plot(x2, np.sin(x2), "ro")
plt.plot(xrange,
         [taylor(sin(x), x, x2, 5).subs(x, xi) for xi in xrange], "r-")
plt.axis([-1, 10, -2, 2])
plt.show()
```

Durch den folgenden Ausdruck läßt sich illustrieren, wie gut sich $f(x) = \sin(x)$ in den verschiedenen Ordnungen der Taylorentwicklung approximieren läßt.

```
xrange = np.linspace(0, 10, 100)

plt.plot(xrange, np.sin(xrange))

for i in range(5):
    plt.plot(xrange,
             [taylor(sin(x), x, 0, 2*i+1).subs(x, xi) for xi in xrange])

plt.axis([0, 10, -2, 2])
plt.show()
```

Zum Abschluß ein Beispiel aus der physikalischen Chemie: Im Folgenden wird das Morse-Potential $V(r) = D_e [1 - e^{-a(r-r_e)}]^2$ um den Gleichgewichtsabstand r_e durch eine Taylorentwicklung zweiter Ordnung approximiert (Stichwort: harmonische Näherung).

```
from sympy import exp
```



```
from sympy.abc import r
```

```
De = 5
re = 2
a = 1.2
```

```
def V(r):
    return De*(1-exp(-a*(r-re)))*2
```

```
taylor(V(r), r, re, 2)
```

```
rrange = np.linspace(0, 7, 100)
```

```
plt.plot(rrange, [V(r).subs(r, ri) for ri in rrange])
plt.plot(rrange, [taylor(V(r), r, re, 2).subs(r,ri) for ri in rrange])
plt.axis([0, 7, -1, 8])
plt.show()
```

Fehlerbetrachtung:

Für die Differenz $R_{n,0}(x)$ zwischen dem Taylorpolynom und der ursprünglichen Funktion $f(x)$ gilt:

$$R_{n,0}(x) = f(x) - T_{n,0}(x)$$

Speziell gilt für $n = 0$:

$$\begin{aligned} R_{0,x_0}(x) &= f(x) - f(x_0) \\ &= \int_{x_0}^x 1 f'(t) dt \\ \text{part. Integration} &= [t f'(t)]_{x_0}^x - \int_{x_0}^x t f''(t) dt \\ &= x f'(x) - x_0 f'(x_0) - \int_{x_0}^x t f''(t) dt \\ &= (x - x_0) f'(x_0) + x f'(x) - x f'(x_0) - \int_{x_0}^x t f''(t) dt \\ &= (x - x_0) f'(x_0) + x \int_{x_0}^x f''(t) dt - \int_{x_0}^x t f''(t) dt \\ &= (x - x_0) f'(x_0) + \int_{x_0}^x (x - t) f''(t) dt \\ \Rightarrow f(x) &= \underbrace{f(x_0) + f'(x_0)(x - x_0)}_{T_{1,x_0}(x)} + \underbrace{\int_{x_0}^x f''(t)(x - t) dt}_{R_{1,x_0}(x)} \end{aligned}$$

S Für das **Restglied** $R_{n,x_0}(x)$ zum Taylorpolynom $T_{n,x_0}(x)$ gilt:

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

falls $f(t)$ für $t \in [x_0, x]$ $n + 1$ -mal differenzierbar ist.

B Vollständige Induktion.

3.7.2 Potenzreihen

D Unter einer **Potenzreihe** $P(x)$ versteht man ein Polynom dessen Grad unendlich ist:

$$P(x) = a_0 + a_1 x + \dots = \sum_{i=0}^{\infty} a_i x^i$$

Bricht man die Summation bei $i = n$ ab, so entsteht die **Partialsomme**

$$S_n(x) = \sum_{i=0}^n a_i x^i$$

Eine Potenzreihe heißt an einer Stelle x_0 **konvergent**, wenn die Folge der Partialsummen $S_n(x_0)$ (d.h. $S_0(x_0), S_1(x_0), S_2(x_0), \dots$) einem Grenzwert $G = \lim_{n \rightarrow \infty} S_n(x_0)$ zustrebt – andernfalls nennt man die Potenzreihe bei x_0 **divergent**.

Gibt es ein $\rho > 0$, so daß eine Potenzreihe $P(x)$ für alle $|x| < \rho$ konvergiert und für alle $|x| > \rho$ divergiert, so wird ρ als **Konvergenzradius** der Potenzreihe bezeichnet.

Betrachtet man speziell $x = 1$, so ist

$$P(x) = \sum_{i=0}^{\infty} a_i x^i = \sum_{i=0}^{\infty} a_i$$

und man spricht einfach von einer **Reihe**. Wechseln die a_i 's einer Reihe ständig ihr Vorzeichen (d.h. $a_i \cdot a_{i+1} < 0$), so spricht man von einer **alternierenden Reihe**.

S Konvergiert eine Potenzreihe $\sum_{i=0}^{\infty} a_i x^i$, dann gilt $\lim_{n \rightarrow \infty} a_n x^n = 0$, d.h. insbesondere: divergiert $a_n x^n$ oder gilt $\lim_{n \rightarrow \infty} a_n x^n = A \neq 0$, so divergiert die Reihe.

B

$$\begin{aligned} \lim_{n \rightarrow \infty} a_n x^n &= \lim_{n \rightarrow \infty} [S_n - S_{n-1}] \\ &= \lim_{n \rightarrow \infty} S_n - \lim_{n \rightarrow \infty} S_{n-1} \\ &= G - G = 0 \end{aligned}$$

S Konvergiert eine Potenzreihe für $x = x_0 \neq 0$, so konvergiert sie für jedes $|x| < |x_0|$. Divergiert die Reihe für x_0 , so divergiert sie für jedes $|x| > |x_0|$.
Gibt es ein $x_0 > 0$, für das die Reihe konvergiert, so existiert ein Konvergenzradius $\rho > 0$.

B Mittels des vorausgegangenen Satzes.

Beispiel:

Geometrische Reihe $\sum_{i=0}^{\infty} x^i$.

Für die Partialsummen läßt sich schreiben:

$$S_n(x) = 1 + x + \dots + x^n = \frac{1 - x^{n+1}}{1 - x} \quad \text{für } x \neq 1$$

$$\begin{aligned} \text{mit } \lim_{n \rightarrow \infty} S_n(x) &= \frac{1}{1 - x} \quad \text{für } x < 1 \\ \lim_{n \rightarrow \infty} S_n(x) &= \infty \quad \text{für } x \geq 1 \end{aligned}$$

damit ist der Konvergenzradius $\rho = 1$.

Gibt es weitere Konvergenzkriterien? Wie läßt sich ρ bestimmen?

CAS-Beispiel



In folgendem Beispiel wird das Konvergenzverhalten der geometrischen Reihe $\sum_{k=0}^{\infty} x^k$ für $x < 1$ untersucht. Dazu werden die Partialsumme $\sum_{k=0}^{100} x^k$ sowie die Funktion $f(x) = \frac{1}{1-x}$ graphisch dargestellt. Was fällt Ihnen auf?

```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(0, 0.999999999)

plt.plot(x, 1/(1-x), color = "red", linewidth = 5.0)
plt.plot(x, sum(x**k for k in range(101)),
         color = "black", linewidth = 3.0)
plt.axis([0, 1, 0, 20])
plt.show()
```

```
from sympy import diff, integrate, Sum, oo
from sympy.abc import x, k

diff(Sum(x**k, (k, 0, oo))).doit()
```

```
# in SymPy lassen sich Integrale von unendlichen Summen derzeit
# noch nicht berechnen
integrate(Sum(x**k, (k, 0, 100)), x).doit()
```

```
diff(1/(1-x), x)
```

```
integrate(1/(1-x), x)
```

Anmerkung: Interessieren wir uns für die Konvergenz einer Potenzreihe $P(x) = \sum_{i=0}^{\infty} a_i x^i$ für ein bestimmtes x , so können wir stattdessen die Konvergenz der Reihe $\sum_{i=0}^{\infty} A_i$ betrachten mit $A_i = a_i x^i$. D.h. alle folgenden Kriterien für Reihen sind entsprechend auf Potenzreihen übertragbar.

S Majorantenkriterium bzw. Vergleichskriterium

Konvergiert die Reihe $\sum_{i=0}^{\infty} b_i$ mit $b_i > 0$ und gilt $0 \leq a_i \leq b_i \forall i \in \mathbb{N}$, dann konvergiert

$\sum_{i=0}^{\infty} a_i$ mit

$$A = \sum_{i=0}^{\infty} a_i \leq \sum_{i=0}^{\infty} b_i = B$$

B $S_n = \sum_{i=0}^n a_i$ ist monoton wachsend und besitzt die obere Schranke B .

Beispiel:

$$\begin{aligned} e^1 &= \sum_{i=0}^{\infty} \frac{1}{i!} \\ &= 1 + \frac{1}{1} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \dots \\ &< 1 + 1 + \frac{1}{2} + \frac{1}{2 \cdot 2} + \frac{1}{2 \cdot 2 \cdot 2} + \dots = 1 + \sum_{i=0}^{\infty} \left(\frac{1}{2}\right)^i \end{aligned}$$

$$= 1 + \frac{1}{1 - \frac{1}{2}} = 1 + 2 = 3$$

S Quotienten- oder d'Alembertsches Kriterium

Sei $\sum_{i=0}^{\infty} a_i$ eine Reihe mit $a_i > 0$ für alle i und besitze die Folge $\frac{a_{i+1}}{a_i}$ den Grenzwert G , dann konvergiert die Reihe für $0 < G < 1$ bzw. divergiert für $G > 1$. Für $G = 1$ ist keine Aussage möglich.

Beispiel:

$$\sum_{i=1}^{\infty} \frac{i}{\sqrt{i} 3^i}$$

$$\begin{aligned} \lim_{i \rightarrow \infty} \frac{a_{i+1}}{a_i} &= \lim_{i \rightarrow \infty} \frac{i+1}{\sqrt{i+1} 3^{i+1}} \frac{\sqrt{i} 3^i}{i} \\ &= \lim_{i \rightarrow \infty} \frac{\sqrt{i+1}}{\sqrt{i}} \frac{1}{3} \\ &= \lim_{i \rightarrow \infty} \frac{\sqrt{i} \sqrt{1+1/i}}{\sqrt{i}} \frac{1}{3} \\ &= \lim_{i \rightarrow \infty} \sqrt{1+1/i} \frac{1}{3} \\ &= \frac{1}{3} \end{aligned}$$

d.h. die Reihe konvergiert.

S Wurzelkriterium

Sei $\sum_{i=0}^{\infty} a_i$ eine Reihe mit $a_i > 0$ und gibt es ein n , so daß $\sqrt[i]{a_i} \leq q < 1$ (q fest) für alle $i \geq n$, so ist die Reihe konvergent. Ist $\sqrt[i]{a_i} \geq 1 \forall i > n$, so ist sie divergent.

Beispiel:

$$\sum_{i=1}^{\infty} \left(\frac{3}{i}\right)^i$$

$$\begin{aligned} \text{Mit } q = 0.9 : \sqrt[i]{a_i} = \frac{3}{i} < 0.9 \quad \text{für } i > \frac{3}{0.9} \simeq n \\ \Rightarrow \sqrt[i]{a_i} < q = 0.9 \quad \text{für } i > n = 4 \end{aligned}$$

S Integralkriterium

Für die Funktion $f(x)$ gelte für $x > 1$: $f(x) \geq 0$ und weiterhin sei sie monoton fallend. Die Reihe $\sum_{i=1}^{\infty} f(i)$ konvergiert genau dann, wenn das uneigentliche Integral $\int_1^{\infty} f(x) dx$ existiert.

Beispiel:

$$\sum_{i=1}^{\infty} \frac{1}{i^2}$$

Quotienten-Kriterium:

$$\begin{aligned} \lim_{i \rightarrow \infty} \frac{a_{i+1}}{a_i} &= \lim_{i \rightarrow \infty} \left(\frac{i}{i+1} \right)^2 \\ &= 1 \end{aligned}$$

\Rightarrow keine Aussage möglich.

Integralkriterium:

$$\begin{aligned} \int_1^{\infty} \frac{1}{x^2} dx &= \lim_{a \rightarrow \infty} \left[-x^{-1} \right]_1^a \\ &= \lim_{a \rightarrow \infty} - \left[\frac{1}{a} - \frac{1}{1} \right] \\ &= 1 \end{aligned}$$

\Rightarrow konvergent.

! Alle angeführten Kriterien gehen von $a_i > 0$ aus. Mittels der folgenden Sätze lassen sich die Aussagen auf Fälle mit $a_i < 0$ übertragen.

D Eine beliebige Reihe $\sum_{i=0}^{\infty} a_i$ heißt **absolut konvergent**, falls $\sum_{i=0}^{\infty} |a_i|$ konvergiert.

S Jede absolut konvergente Reihe konvergiert, d.h. falls $\sum_{i=0}^{\infty} |a_i|$ konvergiert, so konver-

giert auch $\sum_{i=0}^{\infty} a_i$.

S Leibniz-Kriterium

Die Zahlenfolge $\{a_i\}$ sei eine monoton fallende Folge mit dem Grenzwert 0 ($\lim_{i \rightarrow \infty} a_i = 0$). Dann konvergiert die alternierende Reihe

$$\sum_{i=1}^{\infty} (-1)^i a_i = -a_1 + a_2 - a_3 + a_4 - a_5 + \dots$$

Beispiel:

$$\begin{aligned} \sum_{i=1}^{\infty} \frac{(-1)^i}{i} &= -1 + \frac{1}{2} - \frac{1}{3} + \frac{1}{4} - \dots \\ &= -\ln 2 = -0,69 \dots \end{aligned}$$

Mittels des Quotienten- bzw. Wurzelkriteriums lassen sich für Potenzreihen deren Konvergenzradien bestimmen:

S Der Konvergenzradius ρ einer Potenzreihe $\sum_{i=0}^{\infty} a_i x^i$ ist durch den Grenzwert

$$\rho = \lim_{n \rightarrow \infty} \left| \frac{a_n}{a_{n+1}} \right| \quad \text{bzw.} \quad \rho = \lim_{n \rightarrow \infty} \left(\sqrt[n]{|a_n|} \right)^{-1}$$

gegeben.

Beispiel:

$$\begin{aligned} &\sum_{i=0}^{\infty} (3x)^i \\ \rho &= \lim_{n \rightarrow \infty} \left| \frac{(3)^i}{(3)^{i+1}} \right| = \lim_{n \rightarrow \infty} \frac{1}{3} = \frac{1}{3} \end{aligned}$$

S Innerhalb des durch den Konvergenzradius festgelegten Bereiches läßt sich eine Potenzreihe differenzieren und integrieren, d.h. mit $f(x) = \sum_{i=0}^{\infty} a_i x^i$ gilt

$$f'(x) = \sum_{i=0}^{\infty} i a_i x^{i-1}$$

$$\text{bzw. } \int_a^b f(x) dx = \left[\sum_{i=0}^{\infty} \frac{1}{i+1} a_i x^{i+1} \right]_a^b$$

Beispiel:

$$\begin{aligned} f(x) &= \frac{1}{1-x} = \sum_{i=0}^{\infty} x^i \\ &= 1 + x + x^2 + x^3 + \dots \quad \text{für } |x| < 1 \\ f'(x) &= \frac{1}{(1-x)^2} = \sum_{i=0}^{\infty} i x^{i-1} \\ &= 1 + 2x + 3x^2 + 4x^3 + \dots \end{aligned}$$

S Eine Funktion $f(x)$

- a) sei in einem Intervall $[a, b]$ um $x = 0$ beliebig oft differenzierbar,
- b) das Restglied $R_n(x)$ in der Taylorschen Entwicklung verschwinde für $n \rightarrow \infty$

$$\lim_{n \rightarrow \infty} R_n(x) = 0$$

dann läßt sich die Funktion $f(x)$ in dem Intervall $[a, b]$ durch eine Potenzreihe (Taylorreihe)

$$f(x) = \sum_{i=0}^{\infty} \frac{1}{i!} f^{(i)}(0) x^i$$

entwickeln.

Kapitel 4

Funktionen mehrerer Veränderlicher

4.1 Definition, Einführung

D Seien A und B zwei beliebige Mengen. Unter einer **Funktion** oder **Abbildung** f von A nach B

$$f: A \longrightarrow B$$

versteht man eine Vorschrift, die jedem Element $x \in A$ genau ein Element $y \in B$ zuordnet

$$f: x \longmapsto y = f(x)$$

A ist der **Definitionsbereich** der Funktion

B ist der **Wertebereich** der Funktion

- $A \subset \mathbb{R}$ und $B \subset \mathbb{R}$: Funktion einer Veränderlichen (im bisherigen Sinn)
- A beliebig und $B \subset \mathbb{R}^n$: \vec{f} ist eine **vektorwertige** Funktion

$$\vec{f}(x) = \begin{pmatrix} f_1(x) \\ \vdots \\ f_n(x) \end{pmatrix}$$

Der Vektorpfeil über f wird im folgenden in der Regel weggelassen.

Die $f_i(x)$ ($i = 1, 2, \dots, n$) heißen **Koordinatenfunktionen**.

- $A \subset \mathbb{R}^m$ und B beliebig:

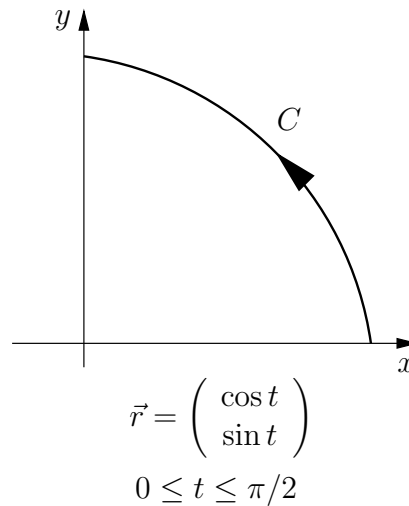
$$f = f \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = f(\vec{x})$$

Die Funktion

$$f_{x_i} = f \begin{pmatrix} x_{1,0} \\ \vdots \\ x_i \\ \vdots \\ x_{n,0} \end{pmatrix}$$

mit $x_j = x_{j,0}$ fest gehalten, außer $x_j = x_i$ heißt **partielle Funktion**

- $A \subset \mathbb{R}$ und $B \subset \mathbb{R}^n$: **Parameterdarstellung einer Kurve**, z. B. $\vec{r}(t)$, mit $\vec{r} = \begin{pmatrix} x \\ y \end{pmatrix}$ Ortsvektor



- $A \subset \mathbb{R}^n, B \subset \mathbb{R}$: **Skalares Feld**, z. B. : $T(\vec{r})$, mit T : Temperatur, \vec{r} : Ortsvektor
- $A \subset \mathbb{R}^n$ und $B \subset \mathbb{R}^n$: **Vektorfeld**, z. B. :

$$\vec{E}(\vec{r}) = \frac{1}{4\pi\epsilon_0} \frac{q\vec{r}}{|\vec{r}|^3}$$

\vec{E} : elektrisches Feld, q : Punktladung bei $\vec{r} = 0$

! Im folgenden in der Regel Einschränkung auf $A \subset \mathbb{R}^2$ und $B \subset \mathbb{R}$, d.h. eine abhängige Variable und zwei unabhängige Variablen.

N

$$f : \mathbb{R}^2 \longrightarrow \mathbb{R}$$

$$\mathbb{R}^2 \ni \vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \mapsto z = f \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}$$

alternative Notation

$$\mathbb{R}^2 \ni \vec{v} = \begin{pmatrix} x \\ y \end{pmatrix} \mapsto z = f(\vec{v}) = f \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}$$

Darstellung von Funktionen:

Analytisch:

- explizit, z.B.:

$$z = ax^2 + bxy + cy^2$$

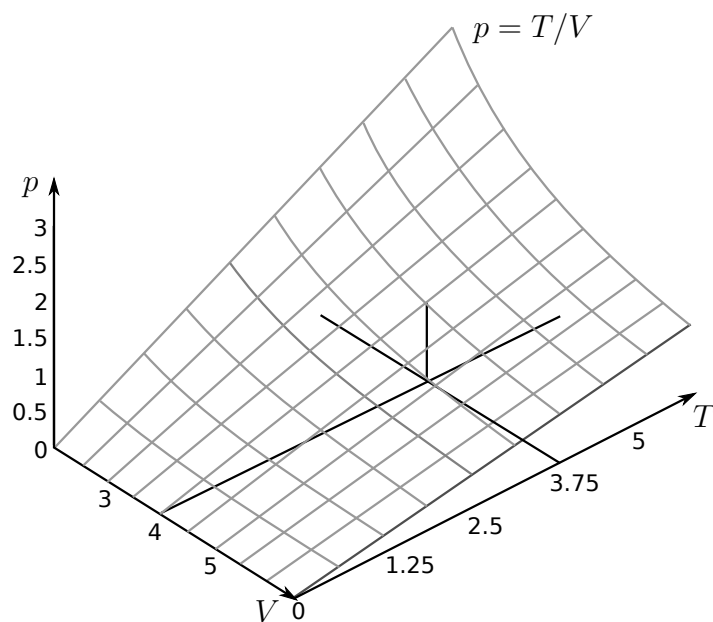
- implizit, z.B.:

$$0 = zy^2 + 2y^2 + xy$$

Graphisch:

Die folgende Beispiele für die graphische Darstellung einer Funktion $z = f(x, y)$ zweier Variablen verwendet die ideale Gasgleichung in der Form: $p = T/V$.

- 3D-Darstellung



CAS-Beispiel



Wenn Sie das Tool Jupyter Notebook verwenden, führen Sie diesen Befehl aus, um interaktive Plots zu erhalten. Ansonsten können Sie den Befehl ignorieren.

```
%matplotlib notebook
```

Der folgende Aufruf der Prozedur `plot_wireframe` liefert eine graphische Darstellung für die Funktion $f(x, y) = e^{-(x^2+y^2)}$ für den Bereich $x, y \in [-2, 2]$. Zusätzlich wird für den Punkt $(x_0, y_0) = (1, 1)$ der Funktionswert dargestellt (schwarzer Punkt).

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
import numpy as np

def f(x, y):
    return np.exp(-(x**2+y**2))

x = np.full((30, 30), np.linspace(-2, 2, 30))
y = np.transpose(np.full((30, 30), np.linspace(-2, 2, 30)))
z = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        z[i, j] = f(x[i, j], y[i, j])

fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$f(x, y)$")

ax.set_xlim([-2, 2])
ax.set_ylim([-2, 2])

ax.plot_wireframe(x, y, z,
                  label = r"$f(x, y)$", color = "red")
ax.plot([1], [1], [f(1, 1)],
        label = r"$f(1, 1)$", color = "black", marker = "+")

ax.legend()
plt.show()
```

Eine Darstellung ohne einen ausgezeichneten Punkt:

```
fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$f(x, y)$")
```

```
ax.set_xlim([-2, 2])
ax.set_ylim([-2, 2])

ax.plot_wireframe(x, y, z,
                  label = r"$f(x, y)$", color = "blue")

ax.legend()
plt.show()
```

Analog kann auf den Funktionswert ein Farbverlauf angewendet werden (die verschiedenen Bezeichnungen für Farbverläufe finden Sie in der `Matplotlib`-Dokumentation):

```
fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$f(x, y)$")

ax.set_xlim([-2, 2])
ax.set_ylim([-2, 2])

ax.plot_surface(x, y, z, cmap = "rainbow")

plt.show()
```

Hier das ideale Gasgesetz $p = T/V$ (in reduzierten Einheiten).

```
T = np.full((30, 30), np.linspace(0.1, 4, 30))
V = np.transpose(np.full((30, 30), np.linspace(0.1, 4, 30)))
p = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        p[i, j] = T[i, j]/V[i, j]

fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

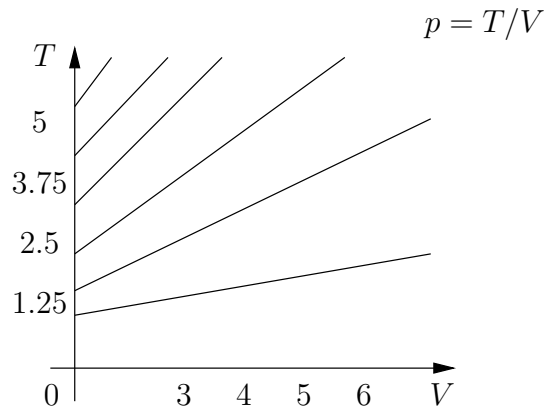
ax.set_xlabel(r"$T$")
ax.set_ylabel(r"$V$")
ax.set_zlabel(r"$p(T, V)$")

ax.set_xlim([0, 4])
ax.set_ylim([0, 4])
ax.set_zlim([0, 40])

ax.plot_surface(T, V, p, cmap = "winter")

plt.show()
```

- Isolinien



CAS-Beispiel



Eine Isoliniendarstellung $f(x, y) = e^{-(x^2+y^2)}$ liefert das folgende Beispiel, wobei der Plotbereich auf $x, y \in [-2, 2]$ und die Anzahl der Isolinien auf zehn festgelegt wurde.

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
import numpy as np

def f(x, y):
    return np.exp(-(x**2+y**2))

x = np.full((30, 30), np.linspace(-2, 2, 30))
y = np.transpose(np.full((30, 30), np.linspace(-2, 2, 30)))
z = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        z[i, j] = f(x[i, j], y[i, j])

fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$f(x, y)$")

ax.set_xlim([-2, 2])
ax.set_ylim([-2, 2])

ax.plot_wireframe(x, y, z, label = r"$f(x, y)$", color = "blue")
ax.contour(x, y, z, 10, colors = "red")
ax.contour(x, y, z, 10, colors = "red", offset = 0)
```

```
ax.legend()
plt.show()
```

Im Vergleich eine einfache Isolinien-Darstellung über den Befehl `contour`:

```
plt.figure()
plt.xlabel(r"$x$")
plt.ylabel(r"$y$")

plt.axis([-2, 2, -2, 2])

plt.contour(x, y, z, 10)

plt.show()
```

Das ideale Gasgesetz in der Isolinien-Darstellung:

```
T = np.full((30, 30), np.linspace(0.1, 4, 30))
V = np.transpose(np.full((30, 30), np.linspace(0.1, 4, 30)))
p = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        p[i, j] = T[i, j]/V[i, j]

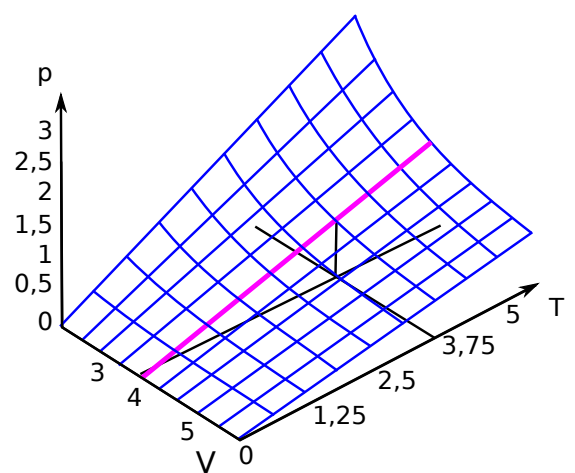
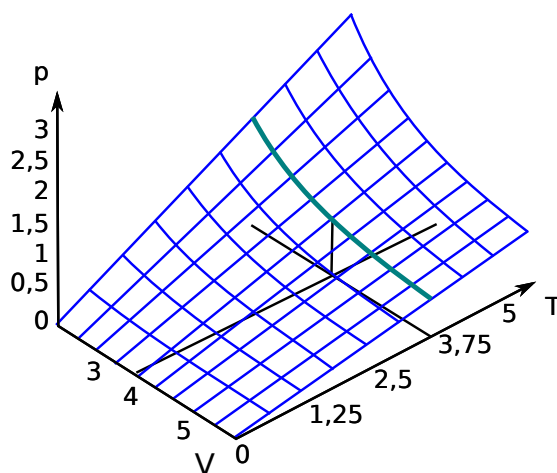
plt.figure()
plt.xlabel(r"$T$")
plt.ylabel(r"$V$")

plt.axis([0, 4, 0, 4])

plt.contour(T, V, p, 40)

plt.show()
```

- Partielle Funktionen



CAS-Beispiel



Zur Darstellung partieller Funktionen stellt `mathchem` die Funktionen `PLOT_PART_FUN_Y0` und `PLOT_PART_FUN_X0` zur Verfügung.

Hier zunächst für die Gaussfunktion $f(x, y) = e^{-(x^2+y^2)}$ die partielle Funktion $f(x, y_0)$ für festes $y = y_0 = 0.2$ sowie die partielle Funktion $f(x_0, y)$ für festes $x = x_0 = -0.4$:

```
from sympy import exp
from sympy.abc import x, y
from mathchem import PLOT_PART_FUN_Y0

PLOT_PART_FUN_Y0(exp(-(x**2+y**2)), 0.2, x, -2, 2, y, -2, 2)
```

```
from mathchem import PLOT_PART_FUN_X0

PLOT_PART_FUN_X0(exp(-(x**2+y**2)), -0.4, x, -2, 2, y, -2, 2)
```

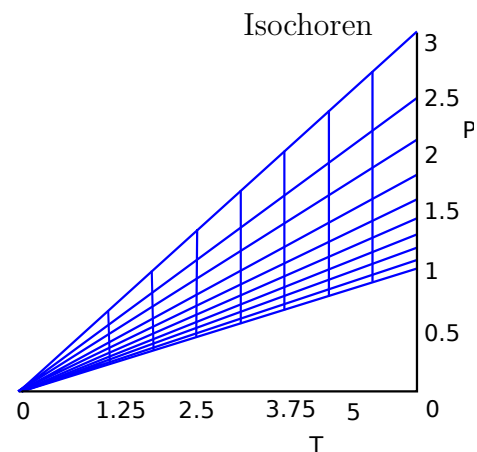
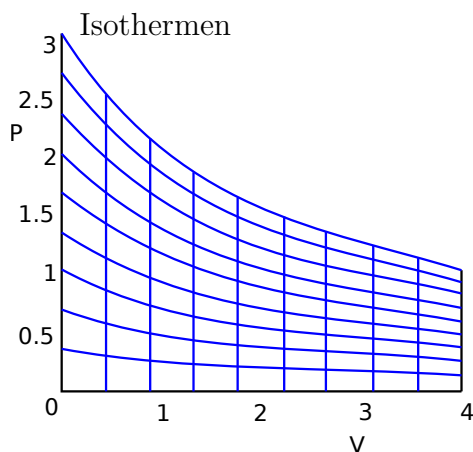
Im Folgenden die reduzierte Form der van der Waals-Gleichung, wobei intern die Variablen T und V auf x und y umgesetzt werden.

```
from sympy.abc import T, V

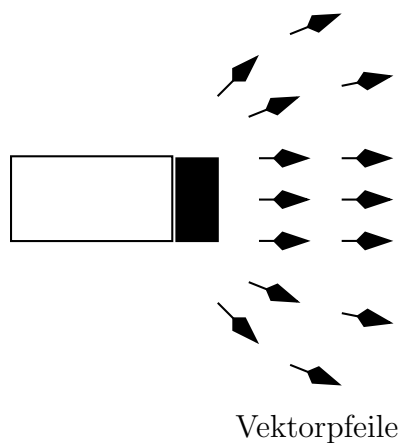
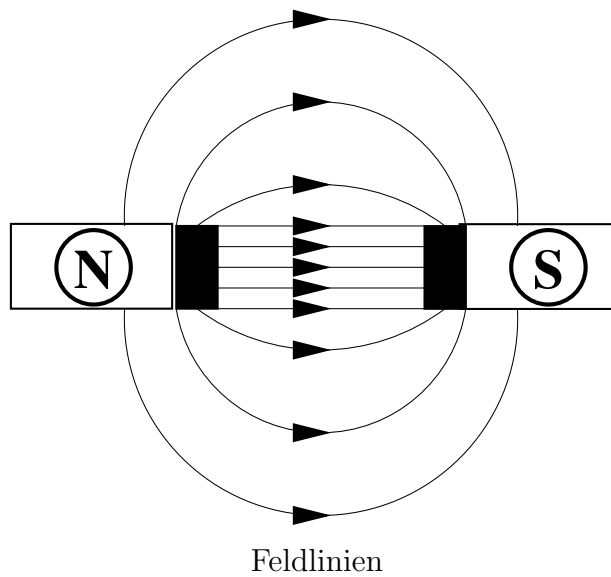
PLOT_PART_FUN_Y0((8/3)*T/(V-1/3)-3/V**2,
  1.0, T, 0.2, 2, V, 0.2, 2)
```

```
PLOT_PART_FUN_X0((8/3)*T/(V-1/3)-3/V**2,
  3.0, T, 1, 6, V, 1, 6)
```

- Kurvenscharen



- Vektorfelder



CAS-Beispiel



Darstellung eines Gradientenfeldes und der Rotation eines Vektorfeldes mit der Prozedur `PLOT_vec_xyz` des `mathchem`-Moduls (Näheres zu Gradientenfeldern und Rotation an späterer Stelle):

```
from sympy.abc import u, v, w
from mathchem import PLOT_vec_xyz

def f(u, v, w):
    return [-v*0.2, u*0.2, w*0.000000000001]

PLOT_vec_xyz(f(u, v, w), u, -4, 4, v, -4, 4, w, -4, 4)
```

```
from sympy.abc import x, y, z

def g(x, y, z):
    return [i/(x**2+y**2+z**2)**(1/2) for i in [x, y, z]]

PLOT_vec_xyz(g(x, y, z), x, -4, 4, y, -4, 4, z, -4, 4)
```

4.2 Der Begriff der Stetigkeit

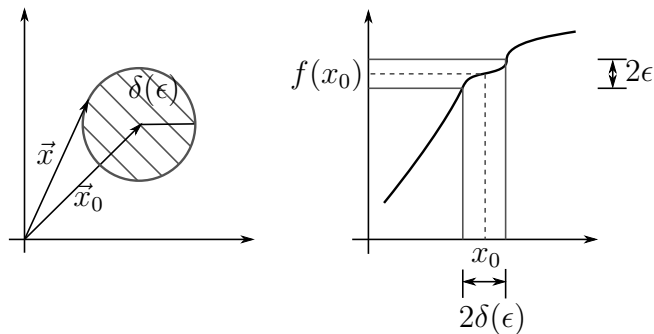
D Eine Funktion heißt **stetig** im Punkt $\vec{x}_0 \in A$, wenn es zu jedem $\epsilon > 0$ ein $\delta(\epsilon) > 0$ gibt, so daß aus

$$|\vec{x} - \vec{x}_0| < \delta \quad \forall \quad x \in A$$

die Eigenschaft

$$|f(\vec{x}) - f(\vec{x}_0)| < \epsilon$$

folgt.



S Für eine vektorwertige Funktion $\vec{f}(\vec{x})$ folgt aus der Stetigkeit jeder einzelnen Koordinatenfunktion $f_i(\vec{x})$ die Stetigkeit von $\vec{f}(\vec{x})$.

! Für die Stetigkeit einer Funktion ist die Stetigkeit ihrer partiellen Funktionen nicht hinreichend!

Beispiel:

$$f \begin{pmatrix} x \\ y \end{pmatrix} = \frac{xy}{x^2 + y^2}$$

$$f_y \begin{pmatrix} x_0 \\ y \end{pmatrix} = \frac{x_0 y}{x_0^2 + y^2}$$

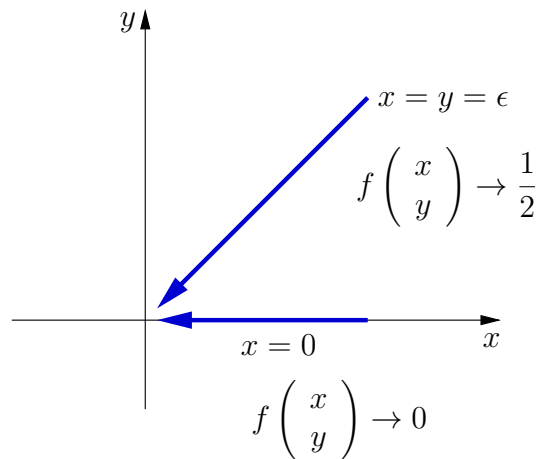
$$x_0 = 0 \implies$$

$$f_y = \frac{0y}{0 + y^2} = 0$$

d.h. die partielle Funktion ist stetig, jedoch gilt

$$f\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}\right) = \lim_{\epsilon \rightarrow 0} \frac{\epsilon\epsilon}{\epsilon^2 + \epsilon^2} = \frac{1}{2}.$$

Damit ist f nicht stetig, d. h. der Grenzwert $f(\vec{0})$ ist wegabhängig.



CAS-Beispiel



Untersuchen Sie mit folgendem Aufruf die Funktion $f(x, y) = \frac{xy}{x^2+y^2}$ auf Stetigkeit:

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
import numpy as np

def f(x, y):
    return x*y/(x**2+y**2)

x = np.full((30, 30), np.linspace(-1, 1, 30))
y = np.transpose(np.full((30, 30), np.linspace(-1, 1, 30)))
z = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        z[i, j] = f(x[i, j], y[i, j])

y0 = 0
vx = np.linspace(-1, 1, 30)
vy = y0

fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")
```

```
ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$f(x, y)$")

ax.set_xlim([-1, 1])
ax.set_ylim([-1, 1])

ax.plot_wireframe(x, y, z, label = r"$f(x, y)$",
                  color = "lime", linewidth = 0.5)

ax.plot(vx, vx, np.full(30, 0.5), label = r"$f(z)$",
        color = "red", linewidth = 3.0)
ax.plot(vx, np.full(30, vy), np.full(30, 0), label = r"$f(z)$",
        color = "blue", linewidth = 3.0)

ax.legend()
plt.show()
```

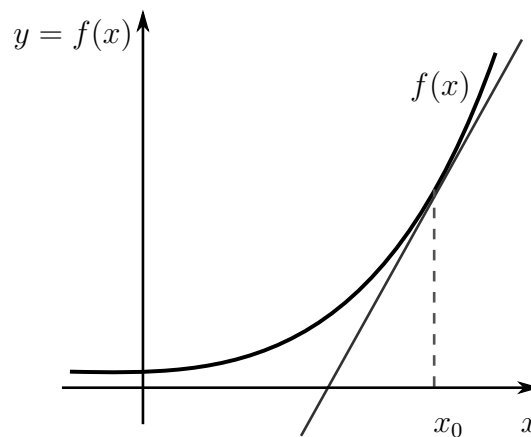
4.3 Mehrdimensionale Differentialrechnung

4.3.1 Differenzierbarkeit

D Die Funktion $f : A \subset \mathbb{R}^m \rightarrow B \subset \mathbb{R}^n$ heißt **differenzierbar** im Punkt $\vec{x} \in A$, wenn es eine lineare Abbildung, repräsentiert durch eine $n \times m$ Matrix J gibt, so daß:

$$\lim_{\Delta \vec{x} \rightarrow \vec{0}} \frac{|\vec{f}(\vec{x} + \Delta \vec{x}) - \vec{f}(\vec{x}) - J \Delta \vec{x}|}{|\Delta \vec{x}|} = 0$$

- $n = m = 1$, d.h.: $A \subset \mathbb{R}$, $B \subset \mathbb{R}$

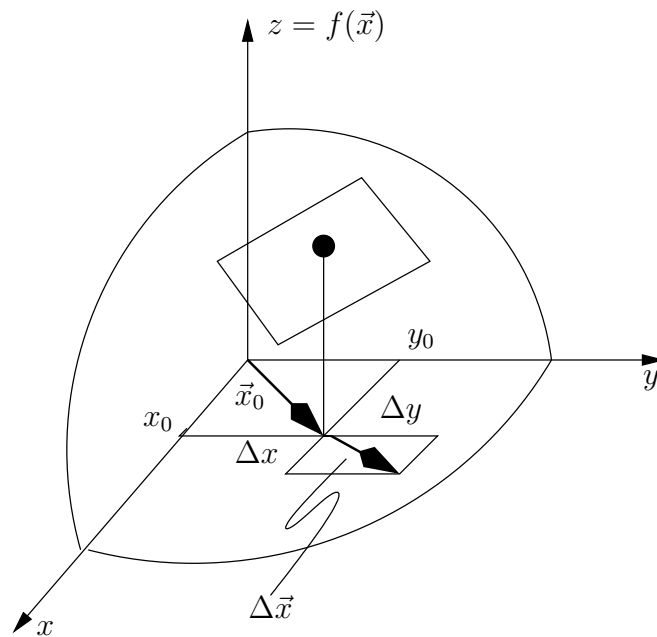


$$\Rightarrow \lim_{\Delta x \rightarrow 0} \left| \frac{f(x + \Delta x) - f(x)}{\Delta x} - J \right| = 0$$

$$\lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x} = J = f'(x)$$

\Rightarrow Die Matrix J hängt im Allgemeinen von x ab!

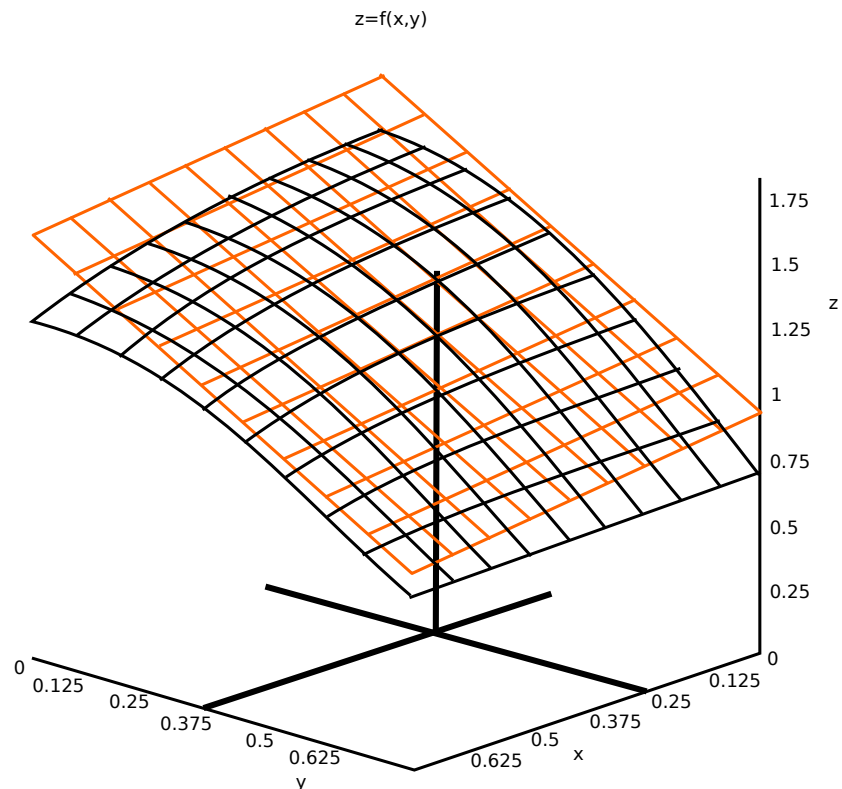
- $n = 1$, $m = 2$, d.h.: $A \subset \mathbb{R}^2$, $B \subset \mathbb{R}$



J ist nun eine Matrix der Form $J = (J_{1x} \ J_{1y})$.

Die Matrixelemente geben die Steigung der Ebene an, wenn man in die x - bzw. y -Richtung geht.

D.h. die obige Definition sagt, daß es eine **Tangentialebene** am Punkt \vec{x}_0 geben muß und diese eindeutig ist.



CAS-Beispiel



Der folgende Aufruf liefert für $f(x, y) = e^{-(x^2+y^2)}$ die Tangentialebene am Punkt $(x_0, y_0) = (0.2, 0.3)$ und stellt das Ergebnis für den Plotbereich $x, y \in [0, 1.5]$ dar:

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
import numpy as np
from sympy import exp, diff
from sympy.abc import x, y

def f(x, y):
    return exp(-(x**2+y**2))

x0 = 0.2
y0 = 0.3

def fx(x1, y1):
    return diff(f(x, y), x).subs(x, x1).subs(y, y1)

fx(x0, y0)
```

```
def fy(x1, y1):
    return diff(f(x, y), y).subs(x, x1).subs(y, y1)

fy(x0, y0)
```

```
def z(x, y):
    return fx(x0, y0)*(x-x0) + fy(x0, y0)*(y-y0) + f(x0, y0)

xrange = np.full((30, 30), np.linspace(0, 1.5, 30))
yrange = np.transpose(np.full((30, 30), np.linspace(0, 1.5, 30)))
fvalues = np.zeros((30, 30))
zvalues = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        fvalues[i, j] = f(xrange[i, j], yrange[i, j])
        zvalues[i, j] = z(xrange[i, j], yrange[i, j])

fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$z$")

ax.set_xlim([0, 1.5])
ax.set_ylim([0, 1.5])

ax.plot_wireframe(xrange, yrange, fvalues,
    label = r"$f(x, y)$", color = "red")
ax.plot_wireframe(xrange, yrange, zvalues,
```



```

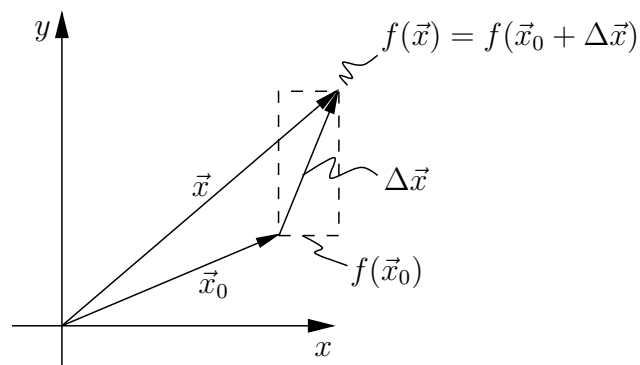
    label = r"$z(x, y)$", color = "lime")
ax.plot([x0], [y0], [f(x0, y0)], label = r"$f(x_0, y_0)$",
        color = "blue", marker = "+")

ax.legend()
plt.show()

```

4.3.2 Partielle Ableitungen

Welche Bedeutung haben die Matrixelemente von J ?



$$\begin{aligned}
 f(\vec{x}_0 + \Delta\vec{x}) &\xrightarrow{\Delta x \rightarrow 0} f(\vec{x}_0) + J\Delta\vec{x} \\
 &= f(\vec{x}_0) + (J_{1x} \ J_{1y}) \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \\
 &= f(\vec{x}_0) + (J_{1x} \ J_{1y}) \begin{pmatrix} 1 \\ 0 \end{pmatrix} \Delta x + (J_{1x} \ J_{1y}) \begin{pmatrix} 0 \\ 1 \end{pmatrix} \Delta y \\
 &= f(\vec{x}_0) + (J_{1x} \ J_{1y}) \vec{e}_x \Delta x + (J_{1x} \ J_{1y}) \vec{e}_y \Delta y \\
 &= f(\vec{x}_0) + J_{1x} \Delta x + J_{1y} \Delta y
 \end{aligned}$$

Speziell: $\Delta y = 0$

$$f(\vec{x}) = f \begin{pmatrix} x_0 + \Delta x \\ y_0 \end{pmatrix} \xrightarrow{\Delta x \rightarrow 0} f \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + J_{1x} \Delta x$$

J_{1x} ist offensichtlich die 1. Ableitung der partiellen Funktion $f \begin{pmatrix} x \\ y_0 \end{pmatrix}$ nach x , d.h.

$$J_{1x} = \frac{d}{dx} f \begin{pmatrix} x \\ y_0 \end{pmatrix}$$

Entsprechend gilt:

$$J_{1y} = \frac{d}{dy} f \begin{pmatrix} x_0 \\ y \end{pmatrix}$$

! Damit wird deutlich, daß die Matrix J im Allgemeinen vom Aufpunkt \vec{x}_0 abhängt

D Der Ausdruck

$$\begin{aligned} J_{ij} &= \lim_{\Delta \rightarrow 0} \frac{f_i(x_1, \dots, x_j + \Delta, \dots, x_m) - f_i(x_1, \dots, x_m)}{\Delta} \\ &= \left. \frac{\partial f_i}{\partial x_j} \right|_{\vec{x}} \end{aligned}$$

wird als **partielle Ableitung 1. Ordnung** der Koordinatenfunktion f_i nach der j -ten Variablen x_j im Punkt \vec{x} bezeichnet.

Die Matrix

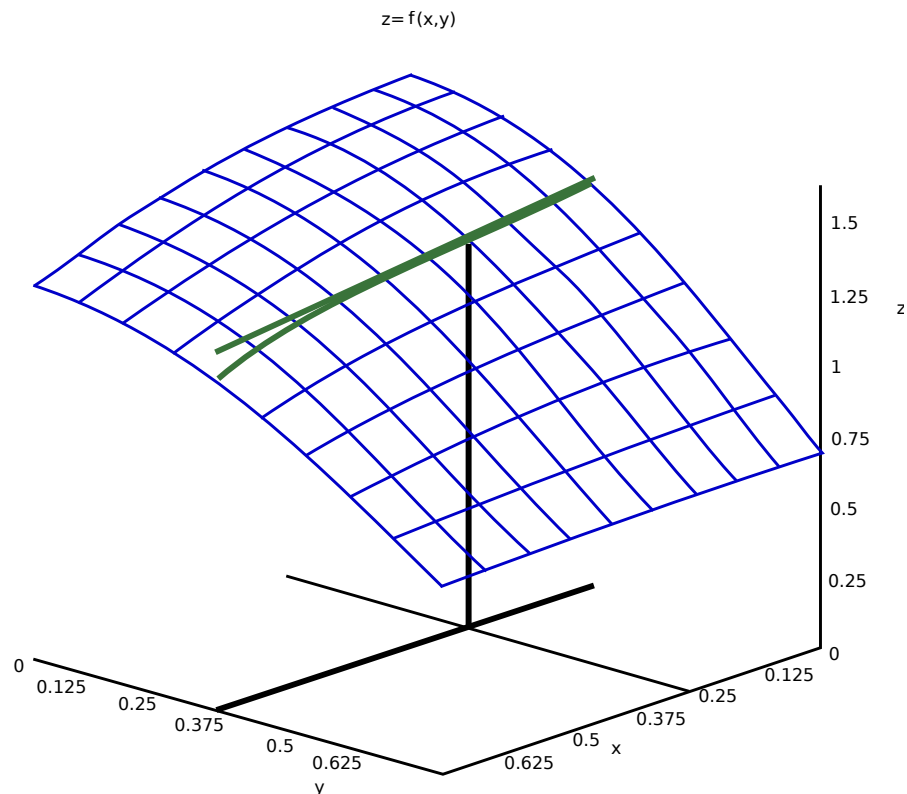
$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_m} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_m} \end{pmatrix}$$

wird als **Jacobimatrix** oder **Funktionalmatrix** bezeichnet.

N

$$\frac{\partial f}{\partial x} = \partial_x f = f_x \quad \text{usw.}$$

Partielle Ableitung $\frac{\partial}{\partial x} f(x, y_0)$:



CAS-Beispiel



Der folgende Aufruf stellt die partielle Funktion $f(x, y_0)$ zur Funktion $f(x, y) = e^{-(x^2+y^2)}$ für den Plotbereich dar.

Zusätzlich wird die Tangente der partiellen Funktion für den Punkt $(0.3, 0.3)$ gezeigt.

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
import numpy as np
from sympy import exp, diff
from sympy.abc import x, y

def f(x, y):
    return exp(-(x**2+y**2))

def dfdx(xi, yi):
    return diff(f(x, y), x).subs(x, xi).subs(y, yi)

def dfdy(xi, yi):
    return diff(f(x, y), y).subs(x, xi).subs(y, yi)

x0 = 0.3
y0 = 0.3
z0 = f(x0, y0)
dfdx0 = dfdx(x0, y0)

xrange = np.full((30, 30), np.linspace(0, 1.5, 30))
```

```

yrange = np.transpose(np.full((30, 30), np.linspace(0, 1.5, 30)))
fvalues = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        fvalues[i, j] = f(xrange[i, j], yrange[i, j])

tx = np.linspace(0, 1.5, 30)
ty = y0
tz = [z0 + dfdx0*(t-x0) for t in tx]

vx = np.linspace(0, 1.5, 30)
vy = y0
vz = [f(v, vy) for v in vx]

fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$f(x, y)$")

ax.set_xlim(0, 1.5)
ax.set_ylim(0, 1.5)

ax.plot_wireframe(xrange, yrange, fvalues,
                  color = "lime", linewidth = 0.5, label = r"$f(x, y)$")
ax.scatter([x0], [y0], z0,
           color = "black", label = r"$f(x_0, y_0)$")
ax.plot(vx, np.full(30, vy), vz,
        color = "red", linewidth = 3.0, label = r"$f(x, y_0)$")
ax.plot(tx, np.full(30, ty), tz,
        color = "blue", label = r"$f_x(x_0, y_0)$")

ax.legend()
plt.show()

```

Das mit dem Skript bereitgestellte Modul `mathchem` bietet vorgenanntes Beispiel in einer komfortableren Form an – der Übersichtlichkeit halber wird die Tangentialebene hier nicht abgebildet. Spielen Sie mit selbst gewählten Funktionen und verschaffen Sie sich einen Überblick.

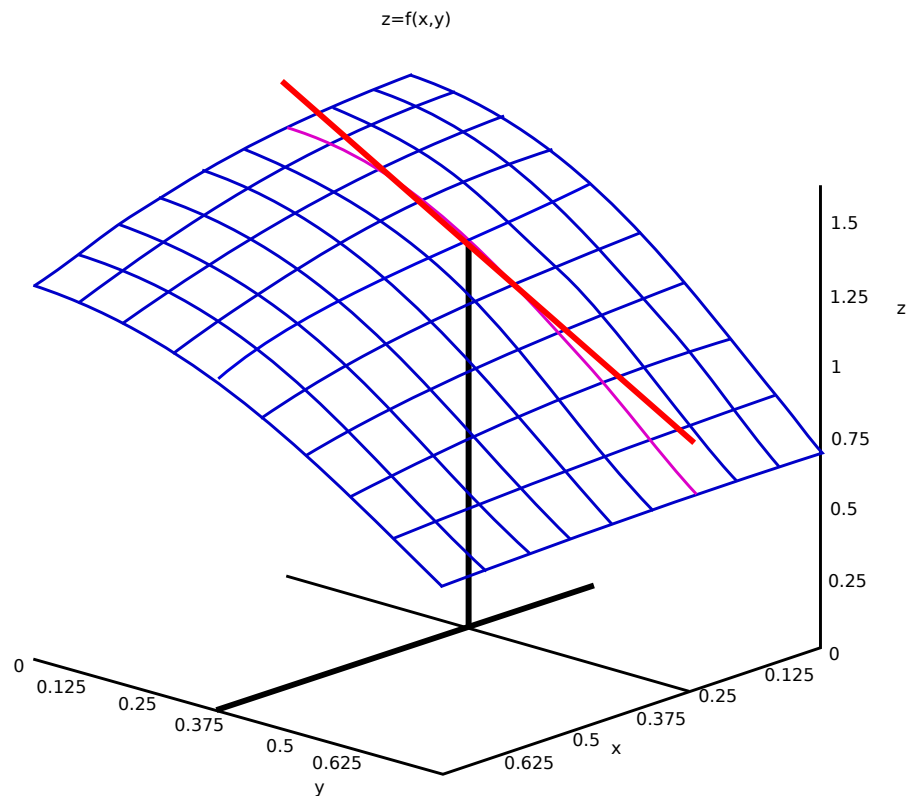
```

from mathchem import PLOT_A

PLOT_A(exp(-(x**2+y**2)), 0.3, 0.3, x, -2, 2, y, -2, 2)

```

Partielle Ableitung $\frac{\partial}{\partial y} f(x_0, y)$:



CAS-Beispiel



Die Prozedur `PLOT_B` des `mathchem`-Moduls stellt im Gegensatz zu `PLOT_A` die Funktion $f(x,y)$, beide partielle Funktionen $f(x,y_0)$ und $f(x_0,y)$ sowie die Tangentialebene dar.

```
from sympy import exp
from sympy.abc import x, y
from mathchem import PLOT_B

PLOT_B(exp(-(x**2+y**2)), 0.3, 0.3, x, -2, 2, y, -2, 2)
```

Hat man speziell $n = 1$ so hat J die Gestalt $J = \left(\frac{\partial}{\partial x_1} f \dots \frac{\partial}{\partial x_m} f \right)$.

Statt dessen läßt sich ein Spaltenvektor $\vec{\text{grad}} f$ einführen, mit

$$\vec{\text{grad}} f = \vec{\nabla} f = \begin{pmatrix} \frac{\partial}{\partial x_1} f \\ \vdots \\ \frac{\partial}{\partial x_n} f \end{pmatrix} = J^T$$

D $\vec{\nabla} f$ wird als **Gradient** von f bezeichnet.

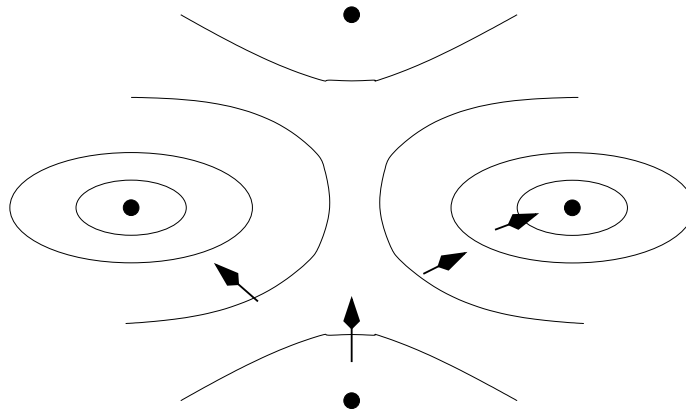
N

$$\vec{\nabla} = \begin{pmatrix} \frac{\partial}{\partial x_1} \\ \vdots \\ \frac{\partial}{\partial x_n} \end{pmatrix}$$

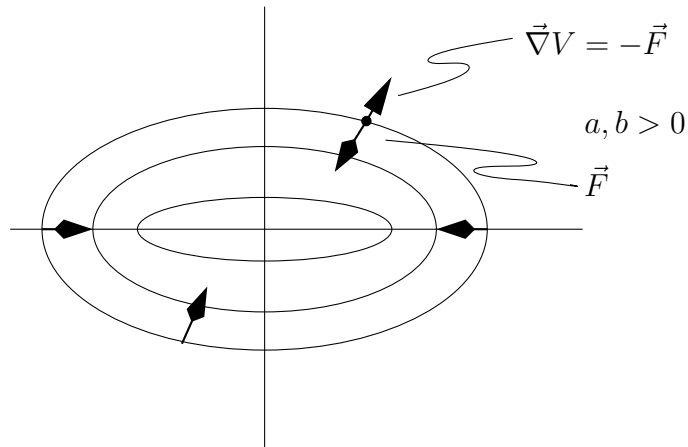
ist der **Nabla-Operator** (Differentialoperator).

Wirkt $\vec{\nabla}$ auf eine skalare Funktion $f(\vec{x})$ mit $\vec{x} \in \mathbb{R}^m$, so entsteht ein Vektor in \mathbb{R}^m – der **Gradient** von $f(\vec{x})$ – der in Richtung der stärksten Änderung von $f(\vec{x})$ weist. Seine Länge ist ein Maß für die Änderung der Funktion $f(\vec{x})$.

Beispiel:



$$\begin{aligned} V = f(\vec{x}) &= f \begin{pmatrix} x \\ y \end{pmatrix} \\ &= ax^2 + by^2 \\ \vec{\nabla} V &= \begin{pmatrix} 2ax \\ 2by \end{pmatrix} \end{aligned}$$



Hat V die Bedeutung einer potentiellen Energie, so hat $\vec{F} = -\vec{\nabla}V$ die Bedeutung einer Kraft.

CAS-Beispiel



Die folgenden Operationen erzeugen für die Funktion $f(x, y) = e^{-(x^2+y^2)}$ das Gradientenfeld für den Plotbereich $x \in [-2, 2]$ und $y \in [-2, 2]$. Berechnung des Gradienten:

```
from sympy import exp, diff
from sympy.abc import x, y

def f(x, y):
    return exp(-(x**2+y**2))

grad = [diff(f(x, y), x), diff(f(x, y), y)]
grad
```

Jetzt wird das Modul Matplotlib zur Verfügung gestellt und das Vektorfeld konstruiert:

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import axes3d
from itertools import product
import numpy as np

coord = [k for k in range(-2, 3)]
points2d = list(map(list, product(coord, coord)))

X = np.full((30, 30), np.linspace(-2, 2, 30))
Y = np.transpose(np.full((30, 30), np.linspace(-2, 2, 30)))
Z = np.zeros((30, 30))

for i in range(30):
    for j in range(30):
        Z[i, j] = f(X[i, j], Y[i, j])
```

```

fig = plt.figure()
ax = fig.add_subplot(111, projection = "3d")

ax.set_xlabel(r"$x$")
ax.set_ylabel(r"$y$")
ax.set_zlabel(r"$z$")

ax.set_xlim([-2, 2])
ax.set_ylim([-2, 2])

ax.plot_wireframe(X, Y, Z, color = "blue")

[ax.quiver(k[0], k[1], 0, grad[0].subs(x, k[0]).subs(y, k[1]),
          grad[1].subs(x, k[0]).subs(y, k[1]), 0, color = "red")
 for k in points2d]

plt.show()

```

Das mit dem Skript bereitgestellte Modul `mathchem` bietet vorgenanntes Beispiel in einer komfortableren Form an – spielen Sie mit selbst gewählten Funktionen und verschaffen Sie sich einen Überblick.

```

from mathchem import PLOT_D

PLOT_D(exp(-(x**2+y**2)), x, -2, 2, y, -2, 2)

```

4.3.3 Partielle Ableitungen höherer Ordnung

Partielle Ableitungen höherer Ordnung entstehen durch partielles Differenzieren einer partiellen Ableitung:

$$\begin{array}{ccc}
\frac{\partial}{\partial x} \left(\frac{\partial f}{\partial x} \right) = \frac{\partial^2 f}{\partial x^2} = f_{xx} & \frac{\partial}{\partial y} \left(\frac{\partial f}{\partial x} \right) = \frac{\partial^2 f}{\partial y \partial x} = f_{yx} & \\
\frac{\partial}{\partial y} \left(\frac{\partial f}{\partial y} \right) = \frac{\partial^2 f}{\partial y^2} = f_{yy} & \frac{\partial}{\partial x} \left(\frac{\partial f}{\partial y} \right) = \frac{\partial^2 f}{\partial x \partial y} = f_{xy} & \\
\text{usw.} & \underbrace{\hspace{10em}}_{\text{gemischte Ableitungen}} &
\end{array}$$

Es stellt sich jetzt noch die Frage, ob gilt:

$$f_{xy} = f_{yx}$$

S Satz von Schwarz

Sind die partiellen Ableitungen n -ter Ordnung einer Funktion f stetig im Bereich A , so ist die Reihenfolge der Differentiation mindestens bis n -ter Ordnung vertauschbar.

Beispiel:

$n = 3$:

$$\begin{aligned} f_{xy} &= f_{yx} \\ f_{xxy} &= f_{xyx} = f_{yxx} \\ \neq f_{xyy} &= f_{yyx} = f_{yxy} \end{aligned}$$

4.3.4 Extremwerte

D Falls in einem Bereich für jedes \vec{x} mit $0 < |\vec{x} - \vec{x}_0| < \delta$ die Beziehung

$$f(\vec{x}) - f(\vec{x}_0) < 0 \quad \text{bzw.} \quad f(\vec{x}) - f(\vec{x}_0) > 0$$

gilt, so bezeichnet man \vec{x}_0 als ein **lokales (relatives) Maximum bzw. Minimum**.

S $f(\vec{x})$ sei eine differenzierbare Funktion. Notwendige Bedingung für das Auftreten eines Maximums/Minimums ist das Verschwinden des Gradienten $\vec{\nabla}f = 0$, d.h. die Tangentialebene muß horizontal verlaufen.

D Ein Punkt $\vec{x} \in A$ heißt **stationär** oder **kritisch** falls

$$\begin{aligned} \vec{\nabla}f(\vec{x}) &= 0 \\ \text{d. h. } \frac{\partial f}{\partial x_i} &= 0 \quad \forall i \end{aligned}$$

Die Tatsache, daß ein Punkt kritisch ist, ist jedoch nicht ausreichend dafür, daß er ein Minimum/Maximum ist.

S Taylorreihe

Für eine Funktion zweier Veränderlicher gilt:

$$\begin{aligned} f(\vec{x} + \Delta\vec{x}) &= f(\vec{x}) + f_x \Delta x + f_y \Delta y \\ &+ \frac{1}{2!} [f_{xx} \Delta x^2 + 2f_{xy} \Delta x \Delta y + f_{yy} \Delta y^2] \\ &+ \frac{1}{3!} [f_{xxx} \Delta x^3 + \dots + f_{yyy} \Delta y^3] \\ &+ \dots + \frac{1}{n!} [f_{x^n} \Delta x^n + \dots + f_{y^n} \Delta y^n] + R_n \end{aligned}$$

falls die entsprechenden partiellen Ableitungen existieren.

Für einen **kritischen** Punkt \vec{x}_0 gilt ($\vec{\nabla} f(\vec{x}_0) = 0$):

$$\begin{aligned} f(\vec{x}_0 + \Delta\vec{x}) &= f(\vec{x}_0) + \frac{1}{2!} [f_{xx}\Delta x^2 + 2f_{xy}\Delta x\Delta y + f_{yy}\Delta y^2] + R_3 \\ &\simeq f(\vec{x}_0) + \frac{1}{2f_{xx}} [(\Delta x f_{xx} + \Delta y f_{xy})^2 + \Delta y^2 (f_{xx}f_{yy} - f_{xy}^2)] \end{aligned}$$

Annahme:

$$\begin{aligned} f_{xx}f_{yy} - f_{yx}^2 &> 0 \\ \text{d. h. } \text{Signum}(f_{xx}) &= \text{Signum}(f_{yy}) \end{aligned}$$

Falls $f_{xx} > 0, f_{yy} > 0$:

$$f(\vec{x}_0 + \Delta\vec{x}) > f(\vec{x}_0) \quad \text{d. h. } \text{Minimum}$$

Falls $f_{xx} < 0, f_{yy} < 0$:

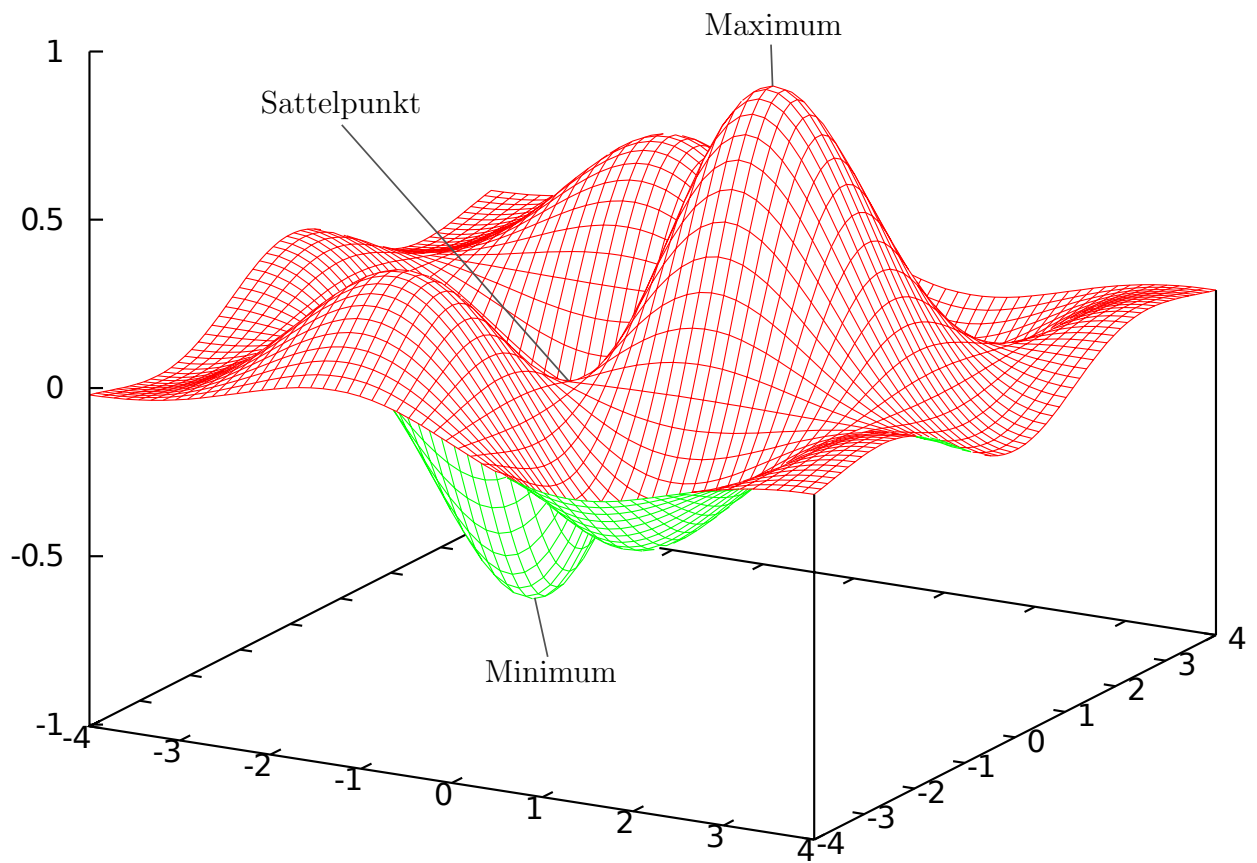
$$f(\vec{x}_0 + \Delta\vec{x}) < f(\vec{x}_0) \quad \text{d. h. } \text{Maximum}$$

Annahme:

$$f_{xx}f_{yy} - f_{yx}^2 < 0$$

$$f(\vec{x}_0 + \Delta\vec{x}) \overset{\geq}{\underset{\leq}{\neq}} f(\vec{x}_0) \quad \text{d. h. } \text{Sattelpunkt möglich}$$

Alle übrigen Fälle liegen komplizierter.



CAS-Beispiel



Identifizieren Sie für die folgenden Funktionen kritische Punkte, Minima, Maxima sowie Sattelpunkte. Nutzen Sie das Modul `mathchem`.

```
from sympy.abc import x, y
from mathchem import PLOT_A, PLOT_C, PLOT_D

def f(x, y):
    return x**2+y**2

PLOT_A(f(x, y), 1, 1, x, -2, 2, y, -2, 2)
```

```
PLOT_D(f(x, y), x, -2, 2, y, -2, 2)
```

```
PLOT_C(f(x, y), 1, 1, x, -2, 2, y, -2, 2)
```

```
from sympy import exp, sin, cos

def f(x, y):
    return exp(-(x**2+y**2))*sin(4*x)*cos(2*x)

PLOT_A(f(x, y), 1, 1, x, -2, 2, y, -2, 2)
```

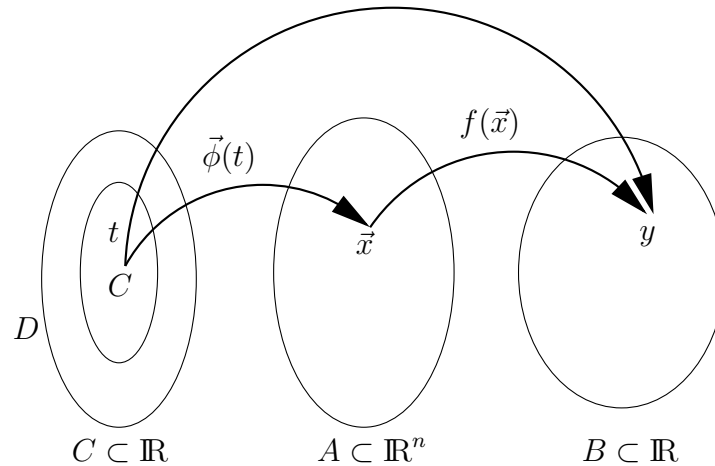
```
PLOT_D(f(x, y), x, -2, 2, y, -2, 2)
```

```
def f(x, y):  
    return sin(x)*cos(y)  
  
PLOT_A(f(x, y), 1, 1, x, -2, 2, y, -2, 2)
```

```
PLOT_D(f(x, y), x, -2, 2, y, -2, 2)
```

4.3.5 Kettenregeln für die partielle Differentiation

$$F(t) = f(\vec{x}(t)) = f(\phi_1(t), \phi_2(t), \dots, \phi_n(t))$$



S Die Funktion $f : A \subset \mathbb{R}^n \rightarrow B$ mit $\vec{x} \mapsto f(\vec{x})$ sei stetig partiell differenzierbar. ϕ_1, \dots, ϕ_n seien in D differenzierbare Funktionen $D \rightarrow A$ mit $\vec{x}(t) = (\phi_1(t), \dots, \phi_n(t)) \in A \forall t \in C$.

Dann ist die Funktion

$$F : C \rightarrow B$$

$$t \mapsto F(t) = f(\vec{x}(t)) = f(\phi_1(t), \dots, \phi_n(t))$$

differenzierbar in C und für die Ableitung $F'(t)$ gilt:

$$F'(t) = \sum_{i=1}^n f_{x_i}(\phi_1(t), \dots, \phi_n(t)) \phi_i'(t)$$

$$\frac{dF(t)}{dt} = \sum_{i=1}^n \frac{\partial f(\vec{x})}{\partial x_i} \frac{d\phi_i(t)}{dt}$$

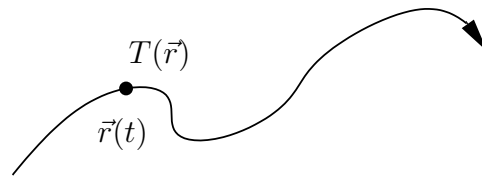
Beispiel:

$$\begin{array}{ll} \text{Temperatur} & \text{Ort} \\ T & = T(\vec{r}) \\ \vec{r} & = \vec{r}(t) = \begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} \\ t & \text{Zeit} \end{array}$$

$$\underbrace{\frac{dT}{dt}} = \underbrace{\frac{\partial T}{\partial x} \frac{dx}{dt} + \frac{\partial T}{\partial y} \frac{dy}{dt} + \frac{\partial T}{\partial z} \frac{dz}{dt}}_{\text{Geschwindigkeit in } x, y, z \text{ - Richtung}}$$

Änderung der Temperatur mit der Zeit t längs des Weges $\vec{r}(t)$

$$\begin{aligned} &= T_x v_x + T_y v_y + T_z v_z \\ &= \vec{\nabla} T \cdot \vec{v} \end{aligned}$$



B Für $n = 2$:

$$\begin{aligned} \frac{\Delta F}{\Delta t} &= \frac{F(t + \Delta t) - F(t)}{t + \Delta t - t} \\ &= \frac{f(x(t + \Delta t), y(t + \Delta t)) - f(x(t), y(t))}{\Delta t} \\ &= \frac{f(x(t + \Delta t), y(t)) - f(x(t), y(t))}{\Delta t} \\ &+ \frac{f(x(t + \Delta t), y(t + \Delta t)) - f(x(t + \Delta t), y(t))}{\Delta t} \\ &= \underbrace{\frac{f(x(t + \Delta t), y(t)) - f(x(t), y(t))}{x(t + \Delta t) - x(t)}}_{Q_1} \underbrace{\frac{x(t + \Delta t) - x(t)}{\Delta t}}_{Q_2} \\ &+ \underbrace{\frac{f(x(t + \Delta t), y(t + \Delta t)) - f(x(t + \Delta t), y(t))}{y(t + \Delta t) - y(t)}}_{Q_3} \underbrace{\frac{y(t + \Delta t) - y(t)}{\Delta t}}_{Q_4} \end{aligned}$$

Grenzwertbetrachtung: $\lim_{\Delta t \rightarrow 0} \frac{\Delta F}{\Delta t} = \frac{dF}{dt} = F'$

$$Q_1 \rightarrow \frac{\partial f(x(t), y(t))}{\partial x}$$

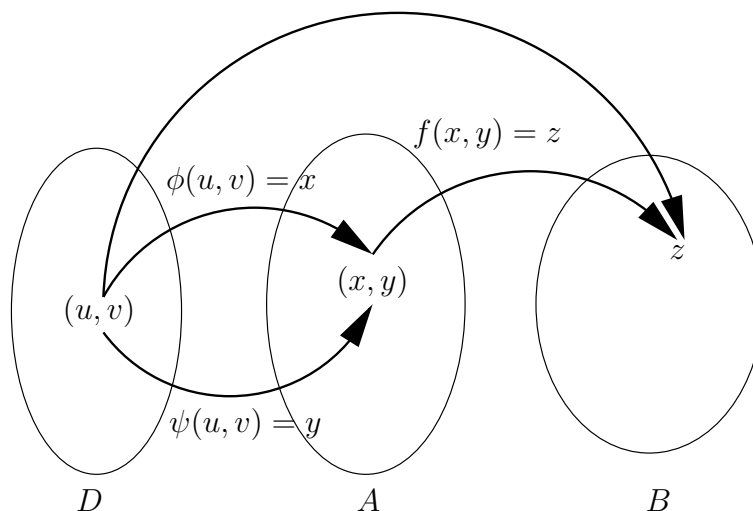
$$Q_2 \rightarrow \frac{dx}{dt} = x'$$

$$Q_3 \rightarrow \frac{f(x(t + \Delta t), y(t + \Delta t)) - f(x(t + \Delta t), y(t))}{\Delta y}$$

$$\text{strebt gegen} \quad \frac{\partial f(x(t), y(t))}{\partial y} \quad \text{für} \quad \Delta t \rightarrow 0$$

$$Q_4 \rightarrow \frac{dy}{dt} = y'$$

$$F(u, v) = f(\phi(u, v), \psi(u, v))$$



S Die Funktion $f : A \rightarrow B$ mit $(x, y) \mapsto f(x, y)$ sei stetig partiell differenzierbar. Die Funktionen

$$\begin{aligned} \phi &: (u, v) \mapsto \phi(u, v) \\ \psi &: (u, v) \mapsto \psi(u, v) \quad u, v \in D \end{aligned}$$

besitzen partielle Ableitungen ϕ_u, ϕ_v, ψ_u und ψ_v in D . Weiterhin gelte:

$$(x, y) = (\phi(u, v), \psi(u, v)) \in A \quad \forall (u, v) \in C \subset D$$

Dann besitzt auch die Funktion

$$\begin{aligned} C &\longrightarrow B \\ F &: \longmapsto F(u, v) = f(\phi(u, v), \psi(u, v)) \end{aligned}$$

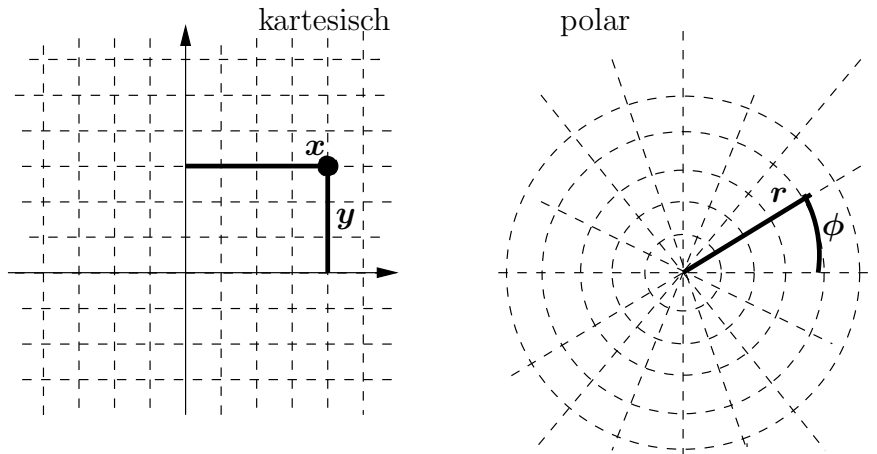
in C partielle Ableitungen F_u, F_v mit

$$\begin{aligned} F_u(u, v) &= f_x(\phi(u, v), \psi(u, v))\phi_u(u, v) + f_y(\phi(u, v), \psi(u, v))\psi_u(u, v) \\ \frac{\partial F}{\partial u} &= \frac{\partial f}{\partial x} \frac{\partial \phi}{\partial u} + \frac{\partial f}{\partial y} \frac{\partial \psi}{\partial u} \\ F_v(u, v) &= f_x(\phi(u, v), \psi(u, v))\phi_v(u, v) + f_y(\phi(u, v), \psi(u, v))\psi_v(u, v) \\ \frac{\partial F}{\partial v} &= \frac{\partial f}{\partial x} \frac{\partial \phi}{\partial v} + \frac{\partial f}{\partial y} \frac{\partial \psi}{\partial v} \end{aligned}$$

B Anwendung des letzten Satzes auf die partiellen Funktionen.

Beispiel:

Übergang von einem Koordinatensystem in ein zweites.



Transformation: $x = r \cos \phi$; $y = r \sin \phi$

$$\begin{aligned}\frac{\partial x}{\partial r} &= \cos \phi \\ \frac{\partial x}{\partial \phi} &= -r \sin \phi \\ \frac{\partial y}{\partial r} &= \sin \phi \\ \frac{\partial y}{\partial \phi} &= r \cos \phi\end{aligned}$$

$$\begin{aligned}f\left(\begin{matrix} x \\ y \end{matrix}\right) &= C e^{\alpha(x^2+y^2)} \\ F\left(\begin{matrix} r \\ \phi \end{matrix}\right) &= f\left(\begin{matrix} x(r, \phi) \\ y(r, \phi) \end{matrix}\right) \\ &= C e^{\alpha(r^2 \cos^2 \phi + r^2 \sin^2 \phi)} \\ &= C e^{\alpha r^2}\end{aligned}$$

Direktes Ableiten von F :

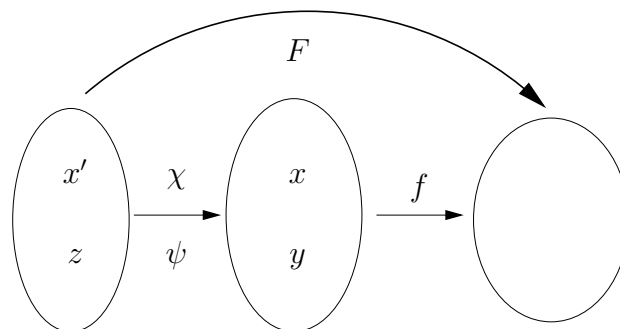
$$\begin{aligned}\frac{\partial F}{\partial r} &= 2\alpha r C e^{\alpha r^2} \\ \frac{\partial F}{\partial \phi} &= 0\end{aligned}$$

Ableiten mit obigen Beziehungen:

$$\begin{aligned}
 \frac{\partial F}{\partial r} &= \frac{\partial f}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial r} \\
 &= 2x\alpha C e^{\alpha(x^2+y^2)} \cos \phi + 2y\alpha C e^{\alpha(x^2+y^2)} \sin \phi \\
 &= 2\alpha C e^{\alpha(x^2+y^2)} [x \cos \phi + y \sin \phi] \\
 &= 2\alpha C e^{\alpha(x^2+y^2)} [r \cos^2 \phi + r \sin^2 \phi] \\
 &= 2\alpha r C e^{\alpha r^2}
 \end{aligned}$$

$$\begin{aligned}
 \frac{\partial F}{\partial \phi} &= \frac{\partial f}{\partial x} \frac{\partial x}{\partial \phi} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial \phi} \\
 &= 2x\alpha C e^{\alpha(x^2+y^2)} (-r \sin \phi) + 2y\alpha C e^{\alpha(x^2+y^2)} (r \cos \phi) \\
 &= 2\alpha r C e^{\alpha r^2} [-x \sin \phi + y \cos \phi] \\
 &= 2\alpha r C e^{\alpha r^2} [-r \cos \phi \sin \phi + r \sin \phi \cos \phi] \\
 &= 0
 \end{aligned}$$

Wechsel der Variablen



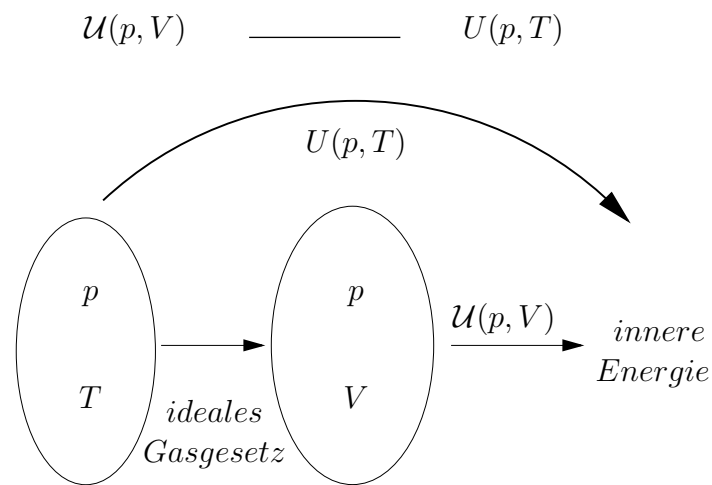
$$\left(\frac{\partial F}{\partial x'} \right)_z = \left(\frac{\partial f}{\partial x} \right)_y \left(\frac{\partial x}{\partial x'} \right)_z + \left(\frac{\partial f}{\partial y} \right)_x \left(\frac{\partial y}{\partial x'} \right)_z$$

mit $x \equiv x'$

$$\left(\frac{\partial F}{\partial x} \right)_z = \left(\frac{\partial f}{\partial x} \right)_y + \left(\frac{\partial f}{\partial y} \right)_x \left(\frac{\partial y}{\partial x} \right)_z$$

Meist (z.B. Atkins, Physikalische Chemie) wird in der Notation nicht zwischen den beiden Funktionen F und f unterschieden.

Beispiel: innere Energie



ideales Gas:

$$U(p, T) = \frac{3}{2}RT \quad \text{mit } pV = RT$$

$$U(p, V) = \frac{3}{2}pV \quad \left(\frac{\partial V}{\partial p}\right)_T = -\frac{RT}{p^2} = -\frac{V}{p}$$

$$\left(\frac{\partial U}{\partial p}\right)_T = \left(\frac{\partial U}{\partial p}\right)_V + \left(\frac{\partial U}{\partial V}\right)_p \left(\frac{\partial V}{\partial p}\right)_T$$

$$0 = \frac{3}{2}V + \frac{3}{2}p \left(-\frac{V}{p}\right) = 0$$

4.4 Mehrdimensionale Integralrechnung

4.4.1 Differentialformen

D Für beliebige Funktionen $f_1(\vec{x}), \dots, f_n(\vec{x})$ heißt der Ausdruck

$$f_1(\vec{x})dx_1 + f_2(\vec{x})dx_2 + \dots + f_n(x)dx_n$$

eine **Differentialform**.

D Erfüllen die $f_i(\vec{x})$ einer Differentialform die Bedingung:

$$\frac{\partial f_i}{\partial x_j} = \frac{\partial f_j}{\partial x_i} \quad \forall i, j$$

so spricht man von einer **exakten Differentialform**.

D Die Funktion $F(\vec{x})$ heißt **Stammfunktion** der n Funktionen $f_i(\vec{x})$ ($i = 1, \dots, n$), wenn gilt:

$$F_{x_i}(\vec{x}) = \frac{\partial F}{\partial x_i}(\vec{x}) = f_i(\vec{x}) \quad \forall i = 1, \dots, n$$

D Besitzt ein exaktes Differential eine Stammfunktion $F(\vec{x})$, so spricht man vom **vollständigen oder totalen Differential**

$$\begin{aligned} dF &= \frac{\partial F}{\partial x_1} dx_1 + \dots + \frac{\partial F}{\partial x_n} dx_n \\ &= \sum_{i=1}^n \frac{\partial F}{\partial x_i} dx_i \end{aligned}$$

Beispiel: (ideales Gas – 1 mol)

Entropie:

$$\begin{aligned} S(T, V) &= C_V \ln T + R \ln V \\ \text{Differentialform : } dS &= \frac{C_V}{T} dT + \frac{R}{V} dV \end{aligned}$$

Es gilt:

$$\begin{aligned}\left(\frac{\partial C_V}{\partial V} \frac{1}{T}\right)_T &= 0 \\ \left(\frac{\partial R}{\partial T} \frac{1}{V}\right)_V &= 0\end{aligned}$$

Es existiert eine Stammfunktion:

$$S(T, V) = C_V \ln T + R \ln V$$

Damit liegt ein vollständiges Differential vor d.h. die Differentialform ist exakt.
Innere Energie:

$$\begin{aligned}dU &= \delta Q + \delta W \\ dU &= \delta Q - p dV \\ \Rightarrow \delta Q &= dU + p dV \quad \text{Wärmeumsatz} \\ &= f_1 dU + f_2 dV\end{aligned}$$

$$\begin{aligned}\frac{\partial f_1}{\partial V} &= 0 & \frac{\partial f_2}{\partial U} &= \frac{\partial p}{\partial U} = \text{const.} \neq 0 \\ p &= \frac{1}{V} RT & U &= \frac{3}{2} RT \\ RT &= \frac{2}{3} U \Rightarrow p &= \frac{1}{V} \frac{2}{3} U \\ \Rightarrow \frac{\partial p}{\partial U} &= \frac{1}{V} \frac{2}{3} \text{ d.h. } \delta Q \text{ ist nicht exakt}\end{aligned}$$

Bedeutung:

dF gibt in 1. Ordnung die Änderung der Funktion $F(\vec{x})$ wenn sich die unabhängigen Variablen x_i um dx_i ändern.

Anwendung:

a) Lineare Approximation

Analog zur linearen Approximation von Funktion einer Veränderlicher: in der Nähe von \vec{x}_0 gibt der Ausdruck

$$\tilde{f}(\vec{x}) = f(\vec{x}_0) + \frac{\partial f}{\partial x}(\vec{x}_0)(x - x_0) + \frac{\partial f}{\partial y}(\vec{x}_0)(y - y_0)$$

eine Näherung für den Verlauf der Funktion $f(\vec{x}) \Rightarrow$ Tangentialebene.

b) Fehlerrechnung

$$df = \frac{\partial f}{\partial x_1} dx_1 + \dots + \frac{\partial f}{\partial x_n} dx_n$$

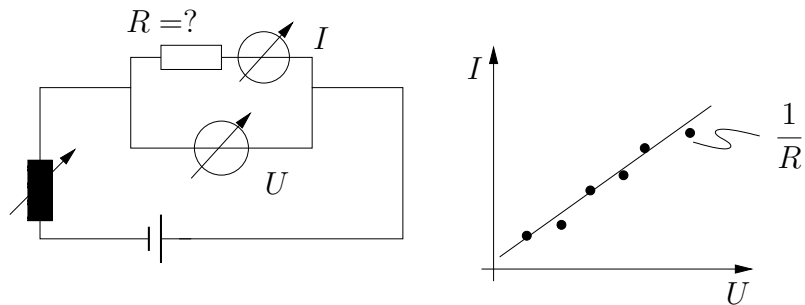
$z = f(\vec{x})$ sei eine Meßgröße, die von Meßgrößen x_i mit einem Meßfehler Δx_i abhängt. Eine Abschätzung für den "Meßfehler" Δf von f gibt:

$$\Delta f \leq \left| \frac{\partial f}{\partial x_1} \Delta x_1 \right| + \left| \frac{\partial f}{\partial x_2} \Delta x_2 \right| + \dots$$

Beispiel:

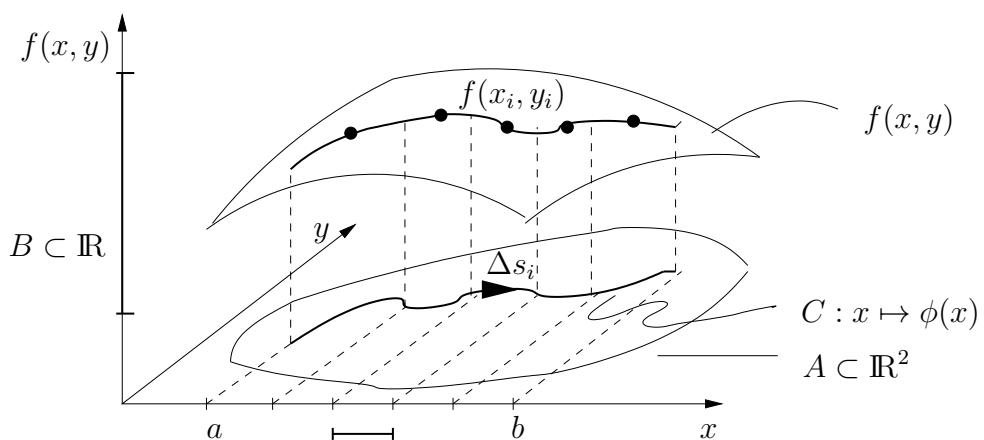
$$f \rightarrow R \quad x_1 \rightarrow U \quad x_2 \rightarrow I$$

$$R = \frac{U}{I}$$



$$\Delta R = \left| \frac{1}{I} \Delta U \right| + \left| \frac{U}{I^2} \Delta I \right|$$

4.4.2 Wegintegrale (2. Art)



S Sei f eine stetige Funktion $A \xrightarrow{\Delta x_i} B$ mit $(x, y) \mapsto f(x, y)$ und $C : y = \phi(x), a \leq x \leq b$ eine stückweise stetige Kurve mit eindeutigem Durchlaufsinne, wobei $(x, \phi(x)) \in A$ für $a \leq x \leq b$.

Teilt man die Kurve C in n Intervalle Δs_i mit zugehörigem Intervall Δx_i ($i = 1, \dots, n$) längs der x -Achse, so strebt die Summe

$$\sum_{i=1}^n f(x_i, y_i = \phi(x_i)) \Delta x_i$$

mit (x_i) den Mittelpunkten der Intervalle i für $n \rightarrow \infty$ einem Grenzwert zu:

$$\lim_{n \rightarrow \infty} \sum_{i=1}^n f(x_i, \phi(x_i)) \Delta x_i = \int_C f(x, y) dx$$

den man als **Wegintegral (2. Art)** der Funktion $f(x, y)$ über die Projektion auf die x -Achse für die Kurve bzw. den Weg C bezeichnet. Entsprechendes gilt für das Integral

$$\int_C f(x, y) dy$$

D Als **Wegintegral allgemeiner Art** über den Weg C wird die Summe

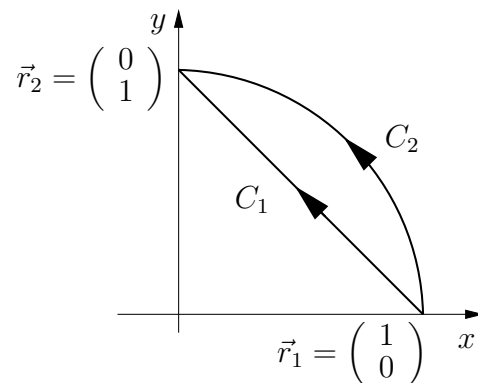
$$\int_C f_1(x, y) dx + \int_C f_2(x, y) dy$$

(in der Ebene), bzw.

$$\int_C f_1(x, y, z) dx + \int_C f_2(x, y, z) dy + \int_C f_3(x, y, z) dz$$

(im Raum) bezeichnet.

Beispiel:



$$f_1(x, y) = x - y \quad f_2(x, y) = x + y$$

parametrisierte Darstellung des Weges

Weg C_1 :

$$\begin{aligned}x &= \phi_1(y) = 1 - y & 0 \leq y \leq 1 \\y &= \psi_1(x) = 1 - x & 0 \leq x \leq 1\end{aligned}$$

$$\begin{aligned}& \int_{C_1} (x - y)dx + \int_{C_1} (x + y)dy \\&= \int_1^0 (x - (1 - x))dx + \int_0^1 (1 - y + y)dy \\&= \int_1^0 (2x - 1)dx + \int_0^1 dy \\&= [x^2 - x]_1^0 + [y]_0^1 \\&= (0 - 0) - (1 - 1) + (1 - 0) - (0 - 0) \\&= 1\end{aligned}$$

Alternativ: Parametrisierung des Weges:

$$\left. \begin{aligned}x &= \phi_1(s) = 1 - s \\y &= \psi_1(s) = s\end{aligned} \right\} \begin{aligned}0 \leq s \leq 1 \\s_1 = 0 \\s_2 = 1\end{aligned}$$

$$\begin{aligned}& \int_{C_1} (x - y)dx + \int_{C_1} (x + y)dy \\&= \int_{C_1} (1 - s - s)x'ds + \int_{C_1} (1 - s + s)y'ds \\&= \int_{C_1} (1 - 2s)(-1)ds + \int_{C_1} 1ds \\&= 2 \int_{s_1}^{s_2} s ds = [s^2]_{s_1}^{s_2} = 1\end{aligned}$$

Weg C_2 :

$$\begin{aligned}x &= \phi_2(y) = \sqrt{1 - y^2} & 0 \leq y \leq 1 \\y &= \psi_2(x) = \sqrt{1 - x^2} & 0 \leq x \leq 1\end{aligned}$$

$$\begin{aligned}& \int_{C_2} (x - y)dx + \int_{C_2} (x + y)dy \\&= \int_1^0 (x - \sqrt{1 - x^2})dx + \int_0^1 (\sqrt{1 - y^2} + y)dy\end{aligned}$$

Alternativ: Parametrisierung des Weges:

$$\left. \begin{aligned} x &= \phi_2(t) = \cos(t) \\ y &= \psi_2(t) = \sin(t) \end{aligned} \right\} \begin{aligned} 0 &\leq t \leq \pi/2 \\ t_1 &= 0 \\ t_2 &= \pi/2 \end{aligned}$$

$$\begin{aligned} & \int_{C_2} (x-y)dx + \int_{C_2} (x+y)dy \\ &= \int_{C_2} (\cos t - \sin t) \frac{dx}{dt} dt + \int_{C_2} (\cos t + \sin t) \frac{dy}{dt} dt \\ &= \int_{t_1}^{t_2} [(\cos t - \sin t)(-1) \sin t + (\cos t + \sin t) \cos t] dt \\ &= \int_{t_1}^{t_2} [-\cos t \sin t + \sin t \cos t + \sin^2 t + \cos^2 t] dt \\ &= \int_{t_1}^{t_2} dt = [t]_{t_1}^{t_2} = [t]_0^{\pi/2} = \pi/2 \end{aligned}$$

! Das Wegintegral hängt im Allgemeinen nicht nur vom Anfangs- und Endpunkt ab, sondern vom Verlauf des Integrationsweges selbst ab!

N Im Ausdruck

$$\int_C f_1(\vec{x})dx + \int_C f_2(\vec{x})dy$$

lassen sich die Funktionen f_1 und f_2 zu einem Vektor zusammenfassen:

$$\int_C \vec{f}(\vec{x}) \cdot d\vec{x} \quad \text{mit} \quad \vec{f}(\vec{x}) = \begin{pmatrix} f_1(\vec{x}) \\ f_2(\vec{x}) \end{pmatrix}$$

$$\text{und} \quad d\vec{x} = \begin{pmatrix} dx \\ dy \end{pmatrix}$$

entsprechend für 3 Dimensionen:

$$\vec{f}(\vec{x}) = \begin{pmatrix} f_1(\vec{x}) \\ f_2(\vec{x}) \\ f_3(\vec{x}) \end{pmatrix} \quad d\vec{x} = \begin{pmatrix} dx \\ dy \\ dz \end{pmatrix}$$

N Statt Wegintegral spricht man oft auch von **Kontur-** oder **Kurvenintegral**

CAS-Beispiel



Nutzen Sie die Funktion `PLOT_trajectory` des `mathchem`-Moduls, um die Kurven verschiedener Wegintegrale graphisch darzustellen:

```
from sympy.abc import u, v, t
from mathchem import PLOT_trajectory

PLOT_trajectory(-u**2-v**2+3,
                u, -1, 1, v, -1, 1, 0.2+2*t**3, t, t, -0.7, 0.7)
```

```
from sympy import sin

PLOT_trajectory(-u**2-v**2+3,
                u, -1, 1, v, -1, 1, 2*t, sin(8*t), t, -0.4, 0.4)
```

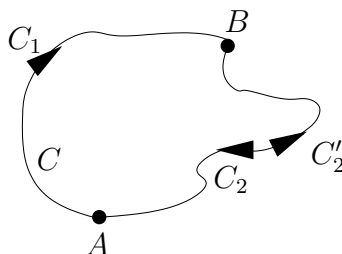
D Als **geschlossenen Integrationsweg** bezeichnet man Wege, bei denen Anfangs- und Endpunkt zusammenfallen.

Symbol:

$$\oint_C \vec{f}(\vec{x}) \cdot d\vec{x}$$

S Ist ein Wegintegral einer Funktion $\vec{f}(\vec{x})$ für einen beliebigen geschlossenen Weg gleich Null, so hängt für jedes Wegintegral von \vec{f} das Integral nur vom Anfangs- und Endpunkt ab.

B



$$\oint_C \vec{f}(\vec{x}) \cdot d\vec{x} = 0 = \int_{A \rightarrow B} \vec{f}(\vec{x}) \cdot d\vec{x} + \int_{B \rightarrow A} \vec{f}(\vec{x}) \cdot d\vec{x}$$

$$\begin{aligned}
&= \int_{\substack{C_1 \\ A \rightarrow B}} \vec{f}(\vec{x}) \cdot d\vec{x} - \int_{\substack{C'_2 \\ A \rightarrow B}} \vec{f}(\vec{x}) \cdot d\vec{x} = 0 \\
\Rightarrow \int_{\substack{C_1 \\ A \rightarrow B}} \vec{f}(\vec{x}) \cdot d\vec{x} &= \int_{\substack{C'_2 \\ A \rightarrow B}} \vec{f}(\vec{x}) \cdot d\vec{x}
\end{aligned}$$

An C_1 und $C'_2 = -C_2$ wurden nur die Forderungen gestellt, den geschlossenen Weg C zu ergeben und den Anfangs- und Endpunkt gemeinsam zu haben! $\Rightarrow \int \vec{f}(\vec{x}) \cdot d\vec{x}$ ist wegunabhängig!

? Welche Eigenschaften muß eine Funktion $\vec{f}(\vec{x})$ besitzen, damit dessen Wegintegrale nur vom Anfangs- und Endpunkt abhängen?

4.4.3 Wegintegrale für Gradientenfelder

D Eine Funktion $\vec{G}(\vec{r})$ heißt **Gradientenfeld**, falls sie sich als Gradient eines skalaren Feldes $S(\vec{x})$ darstellen läßt:

$$\vec{G}(\vec{x}) = \vec{\nabla} S(\vec{x}) = \begin{pmatrix} \frac{\partial S}{\partial x} \\ \frac{\partial S}{\partial y} \\ \frac{\partial S}{\partial z} \end{pmatrix}$$

S wird als **Potentialfunktion** bezeichnet und ist bis auf eine Konstante festgelegt.

S Das Wegintegral einer Gradientenfunktion $\vec{G}(\vec{x})$ hängt nur vom Anfangs- und Endpunkt des Weges ab.

B

$$\int_{\vec{x}_A}^{\vec{x}_E} \vec{G}(\vec{x}) \cdot d\vec{x} = \int_{\vec{x}_A}^{\vec{x}_E} \vec{\nabla} S(\vec{x}) \cdot d\vec{x}$$

$$\begin{aligned}
C &= \int_{\vec{x}_A}^{\vec{x}_E} \underbrace{\frac{\partial S}{\partial x} dx + \frac{\partial S}{\partial y} dy + \frac{\partial S}{\partial z} dz}_{\text{vollständiges oder totales Differential } dS} \\
&= \int_{S_A(\vec{x}_A)}^{S_E(\vec{x}_E)} dS \\
&= S_E - S_A
\end{aligned}$$

? Gibt es Vektorfelder, die keine Gradientenfelder sind, aber dennoch wegunabhängige Wegintegrale besitzen?

S Das Wegintegral eines Vektorfeldes $\vec{A}(\vec{x})$ ist nur dann wegunabhängig, wenn es als Gradientenfeld eines skalaren Feldes $S(\vec{x})$ dargestellt werden kann, d.h.:

$$\vec{A}(\vec{x}) = \vec{\nabla} S(\vec{x})$$

D Unter der **Rotation** eines Vektorfeldes $\vec{A}(\vec{x}) = \begin{pmatrix} A_1(\vec{x}) \\ A_2(\vec{x}) \\ A_3(\vec{x}) \end{pmatrix}$ ($\vec{A} \in \mathbb{R}^3$, $\vec{x} \in \mathbb{R}^3$)

versteht man den Ausdruck

$$\begin{aligned}
\vec{R} &= \text{rot } \vec{A}(\vec{x}) = \vec{\nabla} \times \vec{A}(\vec{x}) \\
&= \begin{pmatrix} \frac{\partial A_3}{\partial y} - \frac{\partial A_2}{\partial z} \\ \frac{\partial A_1}{\partial z} - \frac{\partial A_3}{\partial x} \\ \frac{\partial A_2}{\partial x} - \frac{\partial A_1}{\partial y} \end{pmatrix} = \begin{pmatrix} R_x \\ R_y \\ R_z \end{pmatrix}
\end{aligned}$$

d.h. es entsteht ein neues Vektorfeld $\vec{R}(\vec{x})$.

S Sei $\vec{A}(\vec{x}) = \begin{pmatrix} A_1(\vec{x}) \\ A_2(\vec{x}) \\ A_3(\vec{x}) \end{pmatrix}$ ein Vektorfeld mit stetigen partiellen Ableitungen $A_{1,x}$, $A_{1,y}$, $A_{1,z}$, $A_{2,x}$, $A_{2,y}$, $A_{2,z}$, $A_{3,x}$, $A_{3,y}$, $A_{3,z}$. Die Eigenschaft $\vec{\nabla} \times \vec{A} = 0$ ist eine notwendige und

hinreichende* Bedingung für die Wegunabhängigkeit des Wegintegrals $\int_C \vec{A}(\vec{x}) \cdot d\vec{x}$, d.h. es muß gelten:

$$\begin{pmatrix} \frac{\partial A_3}{\partial y} - \frac{\partial A_2}{\partial z} \\ \frac{\partial A_1}{\partial z} - \frac{\partial A_3}{\partial x} \\ \frac{\partial A_2}{\partial x} - \frac{\partial A_1}{\partial y} \end{pmatrix} = 0 \quad \text{oder} \quad \begin{matrix} \frac{\partial A_3}{\partial y} = \frac{\partial A_2}{\partial z} \\ \frac{\partial A_1}{\partial z} = \frac{\partial A_3}{\partial x} \\ \frac{\partial A_2}{\partial x} = \frac{\partial A_1}{\partial y} \end{matrix}$$

Damit gilt: $\vec{A}(\vec{x})$ ist genau dann ein Gradientenfeld, wenn

$$\operatorname{rot} \vec{A} = \vec{\nabla} \times \vec{A} = 0$$

ist. Mit $\vec{A} = \vec{\nabla} S$ entspricht diese Bedingung offensichtlich:

$$\begin{aligned} \frac{\partial^2 S}{\partial y \partial z} &= \frac{\partial^2 S}{\partial z \partial y} \\ \frac{\partial^2 S}{\partial z \partial x} &= \frac{\partial^2 S}{\partial x \partial z} \\ \frac{\partial^2 S}{\partial x \partial y} &= \frac{\partial^2 S}{\partial y \partial x} \end{aligned}$$

Damit hat man schließlich ein Kriterium für die Vollständigkeit einer Differentialform.

Integrabilitätsbedingung:

S Entsprechend der obigen Definition ist

$$dV = A_1 dx + A_2 dy + A_3 dz$$

ein vollständiges Differential, falls

$$\frac{\partial A_3}{\partial y} = \frac{\partial A_2}{\partial z} \quad \frac{\partial A_1}{\partial z} = \frac{\partial A_3}{\partial x} \quad \text{und} \quad \frac{\partial A_2}{\partial x} = \frac{\partial A_1}{\partial y}$$

d.h. es gibt eine Stammfunktion $V(x, y, z)$.

bzw. in zwei Dimensionen:

$$A_1 dx + A_2 dy$$

*Genau genommen muss dafür, dass die Bedingung hinreichend ist, gelten, dass die Funktion auf einem einfach zusammenhängenden Gebiet definiert ist (siehe weiterführende Literatur).

ist vollständig, falls:

$$\frac{\partial A_2}{\partial x} = \frac{\partial A_1}{\partial y}$$

Beispiele:

a) Die Kraft auf einen Körper im Weltall aufgrund der Schwerkraft ist

$$\vec{F} = -\gamma \frac{m m_E}{|\vec{r}|^2} \frac{\vec{r}}{|\vec{r}|} = -\gamma \frac{m m_E}{|\vec{r}|^2} \vec{e}_{\vec{r}}$$

m : Masse des Körpers

m_E : Masse der Erde

\vec{r} : Ortsvektor des Körpers

Ursprung: Erdmittelpunkt

γ : Gravitationskonstante

$$\vec{F} = -C \frac{1}{|\vec{r}|^2} \frac{\vec{r}}{|\vec{r}|} \quad \text{mit} \quad C = \gamma m m_E$$

Definiere:

$$\vec{G} = -\vec{F} = \frac{C}{|\vec{r}|^2} \frac{\vec{r}}{|\vec{r}|}$$

$$\text{mit} \quad \vec{r} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad \text{und} \quad |\vec{r}| = (x^2 + y^2 + z^2)^{1/2}$$

$$\begin{aligned} \vec{\nabla} \times \vec{G} &= \vec{\nabla} \times C \begin{pmatrix} x/(x^2 + y^2 + z^2)^{\frac{3}{2}} \\ y/(x^2 + y^2 + z^2)^{\frac{3}{2}} \\ z/(x^2 + y^2 + z^2)^{\frac{3}{2}} \end{pmatrix} \\ &= C \frac{(-\frac{3}{2})}{(x^2 + y^2 + z^2)^{\frac{5}{2}}} \begin{pmatrix} z2y - y2z \\ x2z - z2x \\ y2x - x2y \end{pmatrix} \\ &= C \vec{0} \end{aligned}$$

D.h. es gibt eine skalare Funktion V mit

$$\vec{G} = \vec{\nabla} V$$

Berechnung von V :

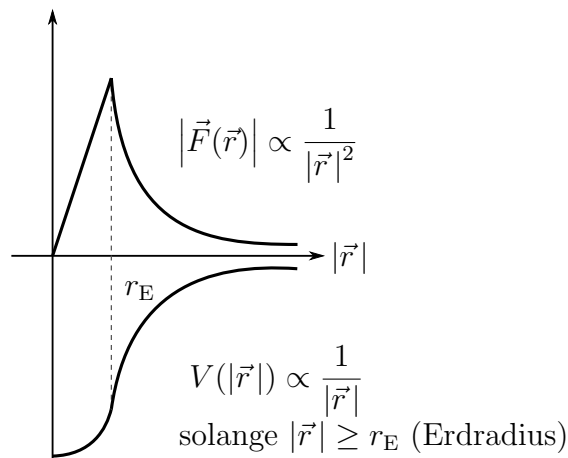
Ist man in unendlicher Entfernung von der Erde, so ist dort \vec{F} oder $\vec{G} = 0$. Da V bis auf eine Konstante fest steht, wählen wir $V(\infty) = 0$. Wie lautet $V(\vec{r}_0)$:

$$V(\vec{r}_0) = \int_{r \rightarrow \infty}^{\vec{r}_0} \vec{G}(\vec{r}) \cdot d\vec{r} + \underbrace{V(\infty)}_0$$

Der Integrationsweg ist beliebig!

$$\begin{aligned} V(\vec{r}_0) &= \int_{r \rightarrow \infty}^{\vec{r}_0} \vec{G}(\vec{r}) \cdot d\vec{r} \\ &= C \int_{(\infty,0,0)}^{(x_0,0,0)} \frac{x}{(x^2 + y^2 + z^2)^{\frac{3}{2}}} dx + C \int_{(x_0,0,0)}^{(x_0,y_0,0)} \frac{y}{(x^2 + y^2 + z^2)^{\frac{3}{2}}} dy \\ &\quad + C \int_{(x_0,y_0,0)}^{(x_0,y_0,z_0)} \frac{z}{(x^2 + y^2 + z^2)^{\frac{3}{2}}} dz \\ &= \left[\frac{-C}{(x^2 + y^2 + z^2)^{\frac{1}{2}}} \right]_{(\infty,0,0)}^{(x_0,0,0)} + \left[\frac{-C}{(x^2 + y^2 + z^2)^{\frac{1}{2}}} \right]_{(x_0,0,0)}^{(x_0,y_0,0)} \\ &\quad + \left[\frac{-C}{(x^2 + y^2 + z^2)^{\frac{1}{2}}} \right]_{(x_0,y_0,0)}^{(x_0,y_0,z_0)} \\ &= -\frac{C}{x_0} + \frac{C}{\infty} - \frac{C}{(x_0^2 + y_0^2)^{\frac{1}{2}}} + \frac{C}{x_0} - \frac{C}{(x_0^2 + y_0^2 + z_0^2)^{\frac{1}{2}}} + \frac{C}{(x_0^2 + y_0^2)^{\frac{1}{2}}} \\ &= -\frac{C}{(x_0^2 + y_0^2 + z_0^2)^{\frac{1}{2}}} \\ &= -\frac{C}{|\vec{r}_0|} \end{aligned}$$

D.h. die Potentialfunktion ist eine **eindeutige** Funktion vom Ort und hängt nur vom Abstand ab.



b) Thermodynamik:

Statt **Potentialfunktionen** spricht man von **Zustandsfunktionen**, die den Zustand eines Systems in eindeutiger Weise kennzeichnen und nicht davon abhängen, auf welchem Wege, d.h. über welche Prozeßführung der Zustand eingestellt wurde.

Innere Energie U (Gesamtenergie des Systems):

$$\text{allgemein: } U = U(T, p, V)$$

T, p, V sind über Zustandsgleichung $p = p(V, T)$ verknüpft.

z.B. ideales Gas (1 mol):

$$p = \frac{RT}{V}$$

$$U = U(T, V)$$

totales Differential:

$$dU = \left(\frac{\partial U}{\partial T} \right)_V dT + \left(\frac{\partial U}{\partial V} \right)_T dV$$

$$= C_V dT + \left(\frac{\partial U}{\partial V} \right)_T dV$$

C_V : molare spezifische Wärme bezogen auf 1 mol bei konstantem Volumen.

Für ein ideales monoatomares Gas gilt:

$$U = \frac{3}{2}RT$$

$$\text{d. h. } C_V = \frac{\partial U}{\partial T} = \frac{3}{2}R = \text{const}$$

$$\frac{\partial U}{\partial V} = 0$$

$$dU = C_V dT$$



Nicht jede thermodynamische Größe ist eine Zustandsfunktion

1. Hauptsatz der Thermodynamik:

$$dU = \underbrace{\delta Q}_{\text{Wärmemenge}} + \underbrace{\delta W}_{\text{Arbeit}}$$

$$\delta Q = dU - \delta W$$

$$= dU + pdV \quad \text{Volumenänderung}$$

$$= C_V dT + pdV \quad \text{ideales Gas}$$

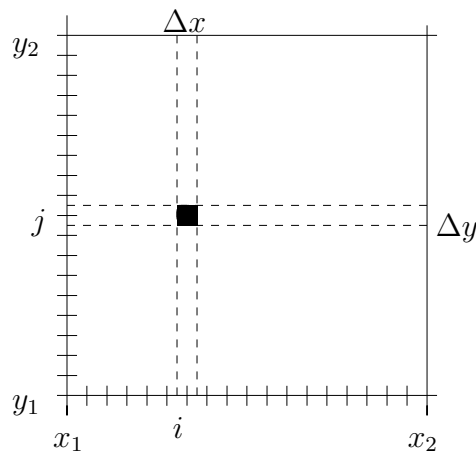
$$= C_V dT + \frac{RT}{V} dV$$

$$\frac{\partial C_V}{\partial V} = 0 \neq \frac{\partial}{\partial T} \frac{RT}{V} = \frac{R}{V}$$

$\Rightarrow \delta Q$ ist kein vollständiges Differential, da die Integrabilitätsbedingung nicht erfüllt ist.

4.4.4 Mehrfachintegrale

Problem: Im Planquadrat einer Karte sei die Höhe $h(x, y)$ angegeben. Wie groß ist die mittlere Höhe in diesem Planquadrat?



1. Schritt: Berechnung der Fläche des Planquadrats:

$$\begin{aligned}
 A &= (x_2 - x_1)(y_2 - y_1) \\
 &= \sum_{i=1}^{N_x} (y_2 - y_1) \Delta x \\
 &= \sum_{i=1}^{N_x} \left(\sum_{j=1}^{N_y} \Delta y \right) \Delta x \\
 &= \lim_{N_x \rightarrow \infty} \sum_{i=1}^{N_x} \left(\lim_{N_y \rightarrow \infty} \sum_{j=1}^{N_y} \Delta y \right) \Delta x \\
 &= \int_{x_1}^{x_2} \left(\int_{y_1}^{y_2} 1 dy \right) dx
 \end{aligned}$$

2. Schritt: Berechnung der mit der Höhe gewichteten Fläche

Analog zur Umformung für A :

$$H = \int_{x_1}^{x_2} \left(\int_{y_1}^{y_2} h(x, y) dy \right) dx$$

z.B. : $h(x, y) = ax + by$

$$H = \int_{x_1}^{x_2} \left[\int_{y_1}^{y_2} (ax + by) dy \right] dx$$

$$\begin{aligned}
&= \int_{x_1}^{x_2} \left(ax(y_2 - y_1) + \frac{1}{2}b(y_2^2 - y_1^2) \right) dx \\
&= \frac{1}{2}a(x_2^2 - x_1^2)(y_2 - y_1) + \frac{1}{2}b(x_2 - x_1)(y_2^2 - y_1^2) \\
&= (x_2 - x_1)(y_2 - y_1) \left[a \frac{x_2 + x_1}{2} + b \frac{y_2 + y_1}{2} \right]
\end{aligned}$$

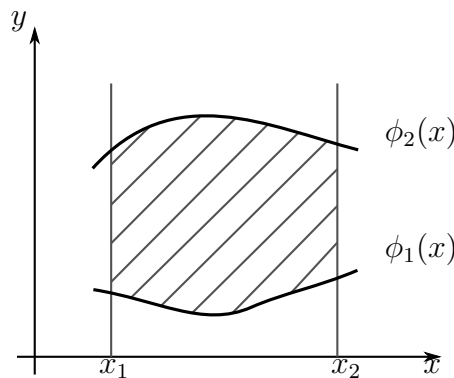
3. Schritt

$$\bar{h} = \frac{H}{A} = a \frac{x_2 + x_1}{2} + b \frac{y_2 + y_1}{2}$$

D Integrale vom Typ

$$\int_{x_1}^{x_2} \int_{y_1=\phi_1(x)}^{y_2=\phi_2(x)} f(x, y) dy dx$$

werden als **Mehrfachintegrale** bezeichnet (hier **Doppel-** oder **Zweifachintegral**) wobei von "innen" nach "außen" integriert wird (d.h. zuerst bezüglich y dann bezüglich x). Die Ausdehnung des Integrationsbereiches wird durch die Abhängigkeit der Integrationsgrenzen von den übrigen unabhängigen Variablen zum Ausdruck gebracht.



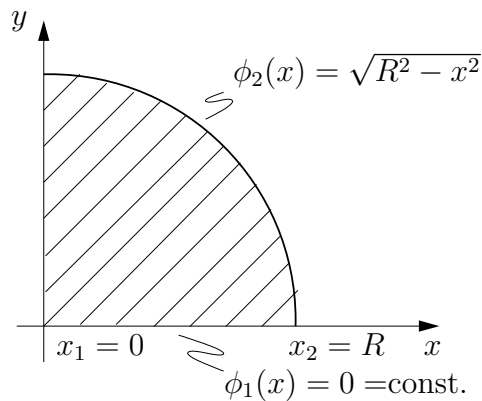
Beispiel:

Volumen einer Kugel mit Radius R :

$$h(x, y) = \sqrt{R^2 - x^2 - y^2}$$

Höhe über x, y -Ebene

$$\frac{V}{8} = \int_{x_1}^{x_2} \left[\int_0^{\sqrt{R^2 - x^2}} \underbrace{\sqrt{R^2 - x^2 - y^2}}_{= h(x, y)} dy \right] dx$$



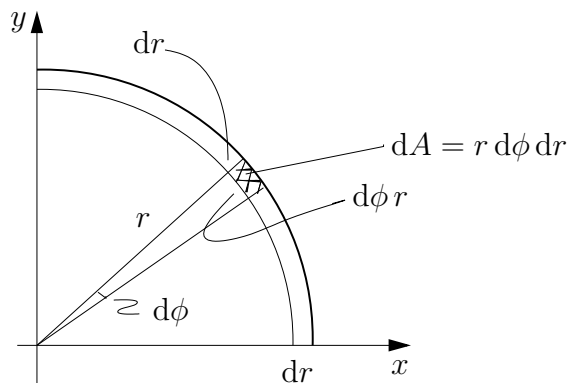
Betrachte 1. Oktant

$$\begin{aligned}
 &= \int_{x_1}^{x_2} \frac{1}{2} \left[y(R^2 - x^2 - y^2)^{\frac{1}{2}} + (R^2 - x^2) \arcsin \frac{y}{\sqrt{R^2 - x^2}} \right]_{0}^{\sqrt{R^2 - x^2}} dx \\
 &= \frac{1}{2} \int_{x_1}^{x_2} \left[\underbrace{\sqrt{R^2 - x^2}(R^2 - x^2 - (R^2 - x^2))^{\frac{1}{2}}}_{0} + (R^2 - x^2) \underbrace{\arcsin \left(\frac{\sqrt{R^2 - x^2}}{\sqrt{R^2 - x^2}} \right)}_{\frac{\pi}{2}} \right. \\
 &\quad \left. - 0(R^2 - x^2) + \underbrace{(R^2 - x^2) \arcsin \frac{0}{\sqrt{R^2 - x^2}}}_{0} \right] dx \\
 &= \frac{1}{2} \int_{x_1}^{x_2} (R^2 - x^2) \frac{\pi}{2} dx \\
 &= \frac{\pi}{4} \int_0^R (R^2 - x^2) dx \\
 &= \frac{\pi}{4} \left[xR^2 - \frac{1}{3}x^3 \right]_0^R \\
 &= \frac{2}{3} \frac{\pi}{4} R^3 \\
 \Rightarrow V &= \frac{4}{3} \pi R^3
 \end{aligned}$$

was zu erwarten war!

Berechnung mittels Polarkoordinaten:

$$\begin{aligned}
 \frac{V}{8} &= \int \tilde{h}(r, \phi) dA(r, \phi) \\
 &= \int_0^R \int_0^{\frac{\pi}{2}} \tilde{h}(r, \phi) r d\phi dr \\
 &= \int_0^R \int_0^{\frac{\pi}{2}} \sqrt{R^2 - r^2} r d\phi dr
 \end{aligned}$$



$$\begin{aligned}
 &= \frac{\pi}{2} \int_0^R \sqrt{R^2 - r^2} r \, dr \\
 &= \frac{\pi}{2} \left(-\frac{1}{3} \right) \left[(R^2 - r^2)^{\frac{3}{2}} \right]_0^R \\
 &= -\frac{\pi}{6} \left[0 - (R^2)^{\frac{3}{2}} \right] \\
 &= \frac{\pi}{6} R^3 \\
 &= \frac{1}{8} \frac{4\pi}{3} R^3 \\
 \Rightarrow V &= \frac{4}{3} \pi R^3
 \end{aligned}$$

S Für ein Mehrfachintegral gilt beim Übergang von einem Koordinatensystem in ein zweites, z.B. $(x, y) \rightarrow (r, \phi)$

$$\begin{aligned}
 \int_{x_1}^{x_2} \int_{g_1(x)}^{g_2(x)} f(x, y) \, dy \, dx &= \int_{r_1}^{r_2} \int_{h_1(r)}^{h_2(r)} f(x(r, \phi), y(r, \phi)) |D| \, d\phi \, dr \\
 &= \int_{r_1}^{r_2} \int_{h_1(r)}^{h_2(r)} F(r, \phi) |D| \, d\phi \, dr
 \end{aligned}$$

mit der Jacobischen Determinante

$$D = \begin{vmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \phi} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \phi} \end{vmatrix} = \frac{\partial x}{\partial r} \frac{\partial y}{\partial \phi} - \frac{\partial x}{\partial \phi} \frac{\partial y}{\partial r}$$

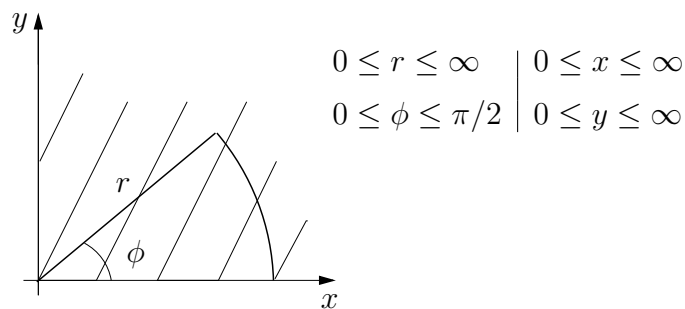
Beispiel: kartesische \rightarrow polare Koordinaten:

$$\begin{aligned}
 x &= r \cos \phi \\
 y &= r \sin \phi
 \end{aligned}$$

$$D = \begin{vmatrix} \cos \phi & -r \sin \phi \\ \sin \phi & r \cos \phi \end{vmatrix} = r \cos^2 \phi + r \sin^2 \phi \\ = r$$

Anwendung:

$$I = \int_0^\infty e^{-x^2} dx \\ I^2 = \int_0^\infty e^{-x^2} dx \int_0^\infty e^{-y^2} dy \\ = \int_0^\infty \left[\int_0^\infty e^{-y^2} dy \right] e^{-x^2} dx \\ = \int_0^\infty \int_0^\infty e^{-y^2} e^{-x^2} dx dy \\ = \int_0^\infty \int_0^\infty e^{-(x^2+y^2)} dx dy$$



$$= \int_0^\infty \int_0^{\pi/2} e^{-r^2} r dr d\phi \\ = \frac{\pi}{2} \int_0^\infty e^{-r^2} r dr \\ = \frac{\pi}{2} \frac{1}{2} \underbrace{\left[e^{-r^2} \right]_0^\infty}_{-1} \\ = \frac{\pi}{4} \\ \Rightarrow I = \frac{\sqrt{\pi}}{2}$$

Anwendung in der physikalischen Chemie - Orbitalmodelle: hier Wasserstoff 1s-Orbital

$$\Psi_{1s}(\vec{r}) = \Psi(r, \theta, \phi) = N e^{-\frac{r}{a_0}}$$

mit a_0 Bohrscher Radius.

$$W(\vec{r}) = |\Psi_{1s}|^2 = N^2 e^{-\frac{2r}{a_0}}$$

hierbei ist $|\Psi|^2$ die Aufenthaltswahrscheinlichkeit eines Elektrons beim Radius r . N ist ein Normierungsintegral, d.h. die Funktion muß noch normiert werden, und zwar so, daß die Aufenthaltswahrscheinlichkeit im gesamten Raum W_{ges} gleich eins ist.

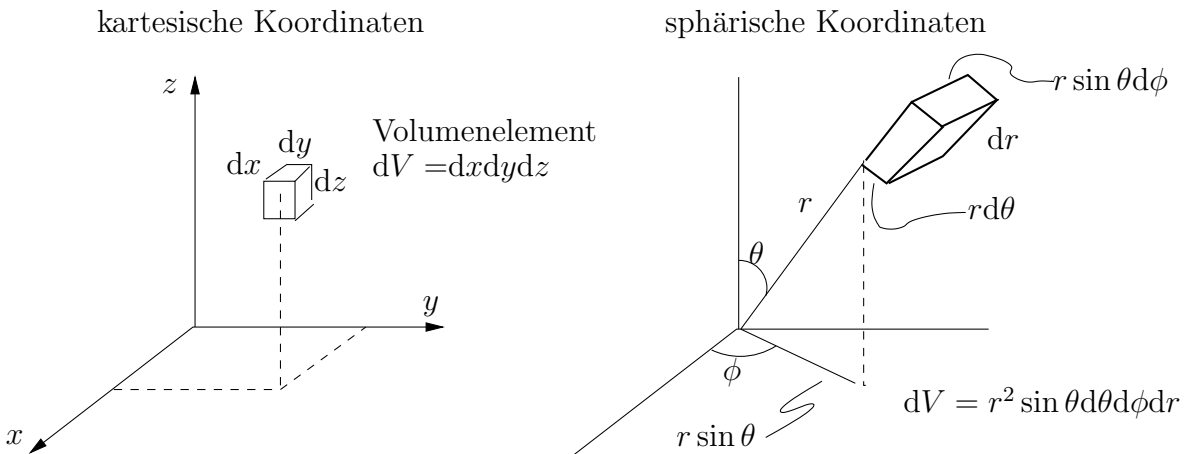
Fordere also:

$$\int W(\vec{r})dr = 1$$

$$\Rightarrow \int_0^R \int_0^\pi \int_0^{2\pi} N^2 e^{-\frac{2r}{a_0}} d\phi \sin \theta d\theta r^2 dr = 1$$

$$\Rightarrow N = \left[\int_0^R \int_0^\pi \int_0^{2\pi} e^{-\frac{2r}{a_0}} d\phi \sin \theta d\theta r^2 dr \right]^{\frac{1}{2}}$$

Analoge Vorgehensweise bei 3-fach Koordinaten



Volumen einer Kugel mit Radius R :

- kartesische Koordinaten

$$\begin{aligned} \frac{V}{8} &= \int_0^R \int_0^{\sqrt{R^2-x^2}} \int_0^{\sqrt{R^2-x^2-y^2}} 1 dx dy dz \\ &= \int_0^R \int_0^{\sqrt{R^2-x^2}} [z]_0^{\sqrt{R^2-x^2-y^2}} dx dy \\ &= \int_0^R \int_0^{\sqrt{R^2-x^2}} \sqrt{R^2-x^2-y^2} dx dy \\ \text{s.o.} \\ &= \frac{1}{8} \frac{4\pi}{3} R^3 \end{aligned}$$

- Kugelkoordinaten

$$\begin{aligned}\int f(r, \theta, \phi) dV(r, \theta, \phi) &= \int_0^R \int_0^{2\pi} \int_0^\pi \underbrace{f(r, \theta, \phi)}_{=1} r^2 \sin \theta d\theta d\phi dr \\ &= \int_0^R \int_0^{2\pi} \int_0^\pi r^2 \sin \theta d\theta d\phi dr \\ &= 2 \cdot 2\pi \left[\frac{r^3}{3} \right]_0^R \\ &= \frac{4\pi}{3} R^3\end{aligned}$$

Index

- Abbildung 19
abelsch 15
abgeschlossen 14
Ableitung 163
Absolutbetrag 34, 35
Abstand eines Punktes zu einer Geraden 56
Abstand zweier Punkte 55
Additionstheorem 152
algebraisches Komplement 86
alternierende Reihe 216
antisymmetrische Funktion 118
Antisymmetrische Matrix 66
- Bandmatrix 65
Basis 59
Basistransformation 61
bijektiv 19
Bildmenge 19
Bildmenge einer Funktion 19
Bildpunkt 19
Boltzmann Verteilung 158
- charakteristische Gleichung 105
charakteristische Matrix 105
charakteristisches Polynom 105
- Definitionsbereich 19
dekadischer Logarithmus 159
Diagonalelement 64
Diagonalisierung 114
Diagonalmatrix 65
Differentialform 257
Differentialquotient 163
Differenzmenge 10
Dimension 59
Doppelintegral 271
Drehspiegelung 103
Drehung 102
Dupel 12
- Ebene 57
echte Teilmenge 7
Eigenvektor 105
Eigenwert 105
Eigenwertproblem (EWP) 105
eindeutig 19
Einheitskreis 150
Einheitsmatrix 65
Einheitsvektor 50
Element 6
elementare Umformung 68
endliche Gruppe 15
entartete Eigenwerte 105
Eulersche Formel 35
exakte Differentialform 257
Exponent 150
Exponentialfunktion 156
Extremstelle 175
- Folgen 128
Fundamentalsatz der Algebra 146
Funktion 19
Funktionalmatrix 240
Funktionswert 19
- ganze rationale Funktionen 143
ganze Zahlen 26
Gauß Verfahren 79
Gauß-Funktion 157
Gauß-Jordan Verfahren 78
Gaußsche Zahlenebene 34
gebrochen rationale Funktionen 148
geometrische Reihe 217
geordneter Körper 29
geordnetes Paar 12
Gerade 55
gerade Funktion 118
geschlossener Integrationsweg 263

- Gradient 244
 Gradientenfeld 264
 Graph 20
 Grenzwert 130
 Grenzwert einer Zahlenfolge 128
 Gruppe 15
 Gruppenaxiome 15
 Gruppentafel 15

 Hauptachsentransformation 114
 Hauptsatz der Differential- und Integralrechnung 193
 hebbare Unstetigkeit 140
 Hessesche Normalform 57
 homogenes LGS 95

 Identität 101
 imaginäre Zahl i 32
 Imaginärteil einer komplexen Zahl 32
 inhomogenes LGS 95
 injektiv 19
 Intervalle 30
 inverse Matrix 77
 inverses Element 17
 Inversion 102
 irrationale Zahlen 29

 Jacobimatrix 240

 kartesisches Koordinatensystem 40
 kartesisches Produkt 12
 Kettenregel 169
 Koeffizientenmatrix 95
 kommutativ 15
 kommutative Gruppe 43
 komplex konjugierte Zahl 34
 komplexe Zahlen 32
 Komponenten 41
 Komposition 125
 Konturintegral 262
 Konvergenzradius 216
 Koordinaten 40
 Koordinatenfunktion 223
 Koordinatentransformation 101
 Kreisfunktionen 150
 Kreuzprodukt zweier Vektoren 52

 kritischer Punkt 175, 247
 Kurvenintegral 262
 Körper 27

 Laplacescher Entwicklungssatz 93
 leere Menge 6
 linear abhängig 59
 linear unabhängig 59
 linearer Vektorraum 58
 lineares Gleichungssystem (LGS) 95
 Linearfaktoren 147
 Linearkombination 47
 linksinverses Element 15
 linksneutrales Element 15
 Logarithmusfunktion 159
 Länge eines Vektors 50

 Matrix 64
 Mehrfachintegral 271
 Menge 6
 Mittelwertsatz der Differentialrechnung 180
 Morse-Potential 157

 n-dimensionaler Raum 40
 n-Tupel 12
 Nabla-Operator 244
 natürliche Zahlen 24
 natürlicher Logarithmus 159
 neutrales Element 16
 Normierung 50
 Nullfolge 128
 Nullmatrix 65
 Nullstellen 147
 Nullvektor 43

 obere Dreiecksmatrix 65
 Obermenge 7
 Ordnung 24
 Ordnung einer Gruppe 15
 orthogonal 51
 orthogonale Matrix 81
 orthonormale Basisvektoren 63
 orthonormierte Vektoren 81
 Ortsvektor 41

 Parameterdarstellung einer Kurve 224
 Partialbruchzerlegung 200

- Partialsomme 216
partielle Ableitung 240
partielle Funktion 224
partielle Integration 195
partikuläre Lösung 95
Peano-Axiome 24
Periode 119
periodische Funktion 119
Permutation 83
Phase 35
Pol mit Vorzeichenwechsel 139
Polarkoordinaten 35
Polstelle 139
Polynom 143
Potentialfunktion 264
Potenzfunktionen 150
Potenzreihe 216
Prinzip der vollständigen Induktion 24
Produktregel 167, 195
punktsymmetrische Funktion 118
- quadratisch 64
Quotientenregel 168
- Rang einer Matrix 75
rationale Zahlen 27
Realteil einer komplexen Zahl 32
Rechtssystem 53
reelle Zahlen 29
Regel von Sarrus 85
reguläre Matrix 68
Reihe 216
Restglied 215
Rotation eines Vektorfeldes 265
- Sattelpunkt 175
Satz von Rolle 180
Satz von Schwarz 246
Schmidtsches Orthogonalisierungsverfahren 62
Schnittmenge 9
Signumfunktion 117
Simpsonregel 210
singuläre Matrix 68
Skalar 58
skalares Feld 224
- skalares Produkt 48
Spaltenvektor 41
Spatprodukt dreier Vektoren 54
Spiegelung 102
Sprung 139
Sprungstelle 139
Stammfunktion 191
stationärer Punkt 247
Stetigkeit 138
Stufenfunktion 117
Substitutionsregel 196
Summenregel 195
surjektiv 19
Symmetrieoperationen 101
symmetrische Funktion 118
Symmetrische Matrix 65
- Tangentialebene 237
Taylorpolynom 213
Teilfolge 129
Teilmenge 7
Terrassenpunkt 175
totales Differential 257
Transformationsmatrix 101
Translation 103
Transponierte Matrix 66
Transposition 83
Trapezregel 209
trigonometrische Funktionen 150
Tripel 12
triviale Lösung 95
- Umkehrfunktion 20
unendlich dimensional 59
ungerade Funktion 118
unitäre Matrix 82
unstetig 139
Unterdeterminante 85
untere Dreiecksmatrix 65
Untergruppe 15
- Vektor 40, 58
Vektorfeld 224
vektorwertige Funktion 223
Venn-Diagramm 7
verallgemeinertes EWP 105

- Vereinigungsmenge 8
- Verkettung 125
- Verknüpfung 14
- Verknüpfungstafel 14
- vollständiges Differential 257

- Wegintegral 260
- Wegintegral allgemeiner Art 260
- Wendepunkt 176
- Wertebereich einer Funktion 19
- Wertetabelle 117
- Wertevorrat 19
- Wurzelfunktion 150


- Zahlenfolgen 128
- Zahlengerade 30
- Zeilen- und Spaltenrang einer Matrix 74
- Zeilenvektor 41
- zweite Ableitung 172
- Zwischenwertsatz 142

Liste der CAS Beispiele

1	Mengen	6
2	Mengenoperationen	8
3	Mengen – Schnittmenge	9
4	Mengen – Differenzmenge	10
5	Mengen – Rechenregeln	11
6	Mengen – kartesisches Produkt	13
7	Funktionen – surjektive, injektive und bijektive Funktionen	22
8	Summationen	25
9	Lösen von Gleichungen	30
10	Komplexe Zahlen	33
11	Komplexe Zahlen – komplexe Konjugation	34
12	Komplexe Zahlen – Polardarstellung – 1	35
13	Komplexe Zahlen – Polardarstellung – 2	37
14	Vektoren – graphische Darstellung	44
15	Vektoren – Rechenregeln	46
16	Vektoren – Skalarprodukt	48
17	Vektoren – dreidimensionale Geraden	56
18	Matrizen – Umformen – 1	69
19	Matrizen – Umformen – 2	71
20	Matrizen – Umformen – 3	71
21	Matrizen – Umformen – 4	72
22	Matrizen – Umformen – 5	75
23	Matrizen – Umformen – 6	78
24	Matrizen – Orthogonalität	81
25	Matrizen – Determinanten – 1	85
26	Matrizen – Determinanten – 2	86
27	Matrizen – Determinanten – 3	87
28	Matrizen – Determinanten – 4	90
29	Matrizen – Determinanten – 5	91
30	Matrizen – Determinanten – 6	92
31	Matrizen – LGS	98
32	Matrizen – Gleichung – 1	107
33	Matrizen – Gleichung – 2	112
34	Funktionen – Eigenschaften	121
35	Funktionen – Rechengesetze	126
36	Funktionen – Zahlenfolgen	129

37 Funktionen – Grenzwert	136
38 Funktionen – Geradengleichung	144
39 Funktionen – Polynome	147
40 Funktionen – Gebrochen rationale Funktionen	148
41 Funktionen – Trigonometrische Funktionen	153
42 Funktionen – Elementare Funktionen	158
43 Funktionen – Ableitungen	165
44 Funktionen – Differentiationsregeln	172
45 Funktionen – Höhere Ableitungen	173
46 Funktionen – Kurvendiskussion	177
47 Funktionen – Newton-Verfahren zur Nullstellensuche	183
48 Funktionen – Unbestimmte Integrale	194
49 Funktionen – Uneigentliche Integrale	206
50 Funktionen – Trapez- und Simpsonregel	210
51 Funktionen – Taylorentwicklung	213
52 Funktionen – Potenzreihen	217
53 Graphen – Plot	226
54 Graphen – Isolinien	228
55 Graphen – Funktion	230
56 Graphen – Vektorfelder	231
57 Graphen – Stetigkeit	234
58 Graphen – Tangentialebene	238
59 Graphen – Partielle Funktionen und Ableitungen – 1	241
60 Graphen – Partielle Funktionen und Ableitungen – 2	243
61 Graphen – Gradientenfeld	245
62 Graphen – Min-Max-Kriterium, Sattelpunkt	249
63 Graphen – Wegintegrale (2. Art)	263

Bibliothek mathematischer Funktionen

Einige Beispiele erfordern ein bereitgestelltes Python-Modul. Dieses kann hier  geladen werden. Speichern Sie die Datei auf Ihrem Computer (rechter Mausklick – Speichern unter: `mathchem.py`) und legen Sie sie in einem Ordner Ihrer Wahl ab. Diesen Ordner müssen Sie ggf. noch zur Variable `PYTHONPATH` hinzufügen. Öffnen Sie dazu ein Terminal-Fenster und geben Sie folgendes ein:

Windows:

```
setx PYTHONPATH "%PYTHONPATH%;Pfad_zu_mathchem"
```

macOS und Linux:

```
echo 'export PYTHONPATH=$PYTHONPATH:Pfad_zu_mathchem' >> ~/.profile
source ~/.profile
```

Alternativ können Sie die Variable auch in `.bash_profile` oder `.bashrc` speichern, dies bleibt Ihnen überlassen.

In Python kann diese Datei mittels `import mathchem` geladen werden. Um alle Beispiele ausführen zu können, benötigen Sie außerdem die Module [NumPy](#), [SciPy](#), [Matplotlib](#), [SymPy](#) und [mpmath](#).