

# Inhalt

<b>1</b>	<b>Numerische Methoden – Nullstellen ermitteln</b>	<b>7</b>
1.1	Bisektionsverfahren	7
1.2	Regula-falsi-Verfahren	9
1.3	Newton-Verfahren	11
<b>2</b>	<b>Lineare Algebra</b>	<b>15</b>
2.1	<b>Grundbegriffe</b>	<b>15</b>
2.1.1	Begriffe über Matrizen	15
2.1.2	Familie von Vektoren	16
2.1.3	Linearkombination	16
2.1.4	Geraden und Ebenen	18
2.2	<b>Vektorräume</b>	<b>20</b>
2.2.1	Definition	20
2.2.2	Untervektorraum	24
2.3	<b>Lineare (Un)Abhängigkeit</b>	<b>27</b>
2.3.1	Lineare (Un)Abhängigkeit – Definition und Prüfform	28
2.3.2	Schnelltests zur linearen (Un)Abhängigkeit	30
2.3.3	Erzeugendensystem	31
2.3.4	Basis	32
2.3.5	Dimension	33
2.3.6	Lineare Hülle	33
2.3.7	Rang	37
2.3.8	Koordinaten	39
2.3.9	Schaubild der Zusammenhänge	39
2.3.10	Beispiele zum Schaubild	41
2.3.11	Basiswechsel	44
2.4	<b>Eigenwerte, Eigenvektoren und Eigenräume</b>	<b>47</b>
2.4.1	Definition und Bedeutung	48
2.4.2	Eigenwerte	48
2.4.3	Vielfachheiten von Eigenwerten	51
2.4.4	Eigenvektoren	52
2.4.5	Eigenräume	52
2.4.6	Diagonalisierbarkeit	54
2.4.7	Schaubild der Zusammenhänge	57
2.5	<b>Zusammenhängende Eigenschaften – Übersicht</b>	<b>58</b>
2.6	<b>Übungsaufgaben – Lineare Algebra</b>	<b>59</b>

<b>3</b>	<b>Analysis mehrerer Veränderlicher</b>	<b>63</b>
<b>3.1</b>	<b>Differentiation, Ableitungen</b>	<b>64</b>
3.1.1	Die Partielle Ableitung	64
3.1.2	Der Gradient	67
3.1.3	Die Hesse-Matrix	68
<b>3.2</b>	<b>Taylorfunktion</b>	<b>68</b>
<b>3.3</b>	<b>Extremstellenberechnung ohne Nebenbedingung</b>	<b>70</b>
<b>3.4</b>	<b>Extremstellenberechnung mit Nebenbedingung</b>	<b>75</b>
3.4.1	Substitutionsmethode	76
3.4.2	Lagrange-Verfahren	77
<b>3.5</b>	<b>Übungsaufgaben – Analysis mehrerer Veränderlicher</b>	<b>82</b>
<b>4</b>	<b>Differentialgleichungen (DGL)</b>	<b>85</b>
<b>4.1</b>	<b>Notationen</b>	<b>86</b>
<b>4.2</b>	<b>Typisierungen</b>	<b>87</b>
4.2.1	Typisierung der DGL	87
4.2.2	Typisierung der Lösungsvarianten	90
<b>4.3</b>	<b>Lösungsansätze</b>	<b>91</b>
4.3.1	$y_h$ : Trennung der Variablen/Veränderlichen (TdV)	92
4.3.2	$y_p$ : Störgliedansatz	94
4.3.3	$y_p$ : Variation der Konstanten – VdK	95
4.3.4	Superposition von partikulären Lösungen	97
<b>4.4</b>	<b>Lösungsformel für lineare DGL erster Ordnung</b>	<b>99</b>
4.4.1	$y_h$ : Lösungsformel (homogen)	99
4.4.2	$y_p$ : Lösungsformel (partikulär)	99
4.4.3	Lösungsformel kombiniert	99
<b>4.5</b>	<b>Beispiele: Nicht-lineare DGL erster Ordnung</b>	<b>101</b>
4.5.1	Bernoulli-DGL	102
4.5.2	Eulerhomogene-DGL	104
<b>4.6</b>	<b>Übungsaufgaben – Differentialgleichungen</b>	<b>106</b>

# 1 Numerische Methoden – Nullstellen ermitteln

Nullstellen einer Funktion bestimmen (ganz gleich wie kompliziert die Funktionsvorschrift ist) stellt ein wichtiges Aufgabengebiet der Analysis dar. Nullstellen von Polynomen zweiten Grades können beispielsweise mit Hilfe der  $pq$ -Formel analytisch ermittelt werden. Für Funktionsterme, die analytisch jedoch nicht nach  $x$  aufzulösen sind, z. B. schon einfach aussehende wie

$$f(x) := e^x + x \Rightarrow e^x + x = 0 \Leftrightarrow x = ?,$$

benötigen wir numerische Verfahren, um Näherungslösungen für  $x$  zu bestimmen. Hinter allen Verfahren stecken Algorithmen, die (einfach gesagt) mit jedem Schritt den  $x$ -Wert derart anpassen, sodass nach vielen Iterationen  $f(x) \approx 0$  gilt. Dieses gefundene  $x$  *approximiert* dann die exakte Nullstelle. Im folgenden stellen wir dir drei gängige Verfahren vor.

Wir beschränken uns hier auf numerische Methoden für Funktionen mit einer Variablen.

## 1.1 Bisektionsverfahren

Das Bisektionsverfahren (Intervallhalbierungsverfahren) ist das am leichtesten zu verstehende Verfahren zur numerischen Bestimmung von Nullstellen. Das Verfahren läuft iterativ mit einem Startintervall  $[a_0, b_0]$  ab. Im Vergleich zu den anderen Verfahren, approximiert das Bisektionsverfahren die Nullstelle in dem Intervall  $[a_0, b_0]$  relativ langsam, terminiert jedoch immer.



**Voraussetzungen:** Es muss  $f(a_0) \cdot f(b_0) < 0$  gelten und  $f$  auf  $[a_0, b_0]$  stetig sein.

Die Voraussetzungen müssen geprüft werden, bevor die Iterationen beginnen. Damit wird sicher gestellt, dass überhaupt eine Nullstelle in dem Intervall existiert (das folgt aus dem *Zwischenwertsatz* – ZWS). Der Teil  $f(a_0) \cdot f(b_0) < 0$  stellt sicher, dass entweder  $f(a_0) < 0 < f(b_0)$  oder  $f(a_0) > 0 > f(b_0)$  gilt.

**Algorithmus** ( $n$  kennzeichnet dabei die  $n$ -te Iteration):

**Fall 1:** Es gilt  $f(a_0) < 0$  (und damit automatisch  $0 < f(b_0)$ , sonst wären die Voraussetzungen verletzt). Dann halten wir uns an folgendes:

1.  $x_n := \frac{a_{n-1} + b_{n-1}}{2}$  (Mittelwert des Intervalls)

2.1  $f(x_n) = 0 \Rightarrow x_n$  ist die gesuchte Nullstelle

2.2  $f(x_n) > 0 \Rightarrow a_n := a_{n-1}$  und  $b_n := x_n$

2.3  $f(x_n) < 0 \Rightarrow a_n := x_n$  und  $b_n := b_{n-1}$

**Fall 2:** Es gilt  $f(a_0) > 0$  (und damit automatisch  $0 > f(b_0)$ , sonst wären die Voraussetzungen verletzt). Dann halten wir uns an folgendes:

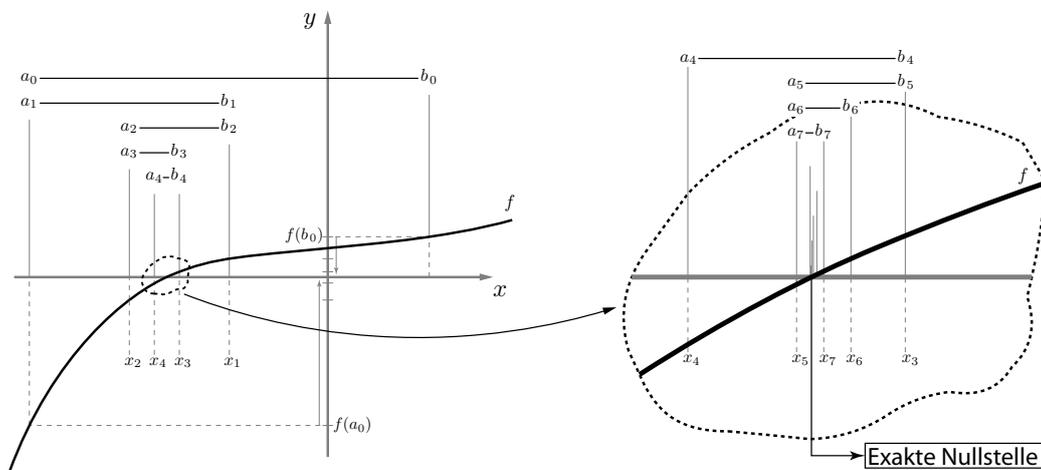
1.  $x_n := \frac{a_{n-1} + b_{n-1}}{2}$  (Mittelwert des Intervalls)

2.1  $f(x_n) = 0 \Rightarrow x_n$  ist die gesuchte Nullstelle

2.2  $f(x_n) > 0 \Rightarrow a_n := x_n$  und  $b_n := b_{n-1}$

2.3  $f(x_n) < 0 \Rightarrow a_n := a_{n-1}$  und  $b_n := x_n$

Rein qualitativ gehen wir das Verfahren einmal an der im Folgenden dargestellten Funktion  $f$  ab.



Das Anfangsintervall  $[a_0, b_0]$  ist zulässig, da  $f(a_0) < 0$ ,  $f(b_0) > 0$  und somit  $f(a_0) \cdot f(b_0) < 0$  gilt. Außerdem ist  $f$  in dem Intervall stetig. Wir können jetzt den Algorithmus starten.

Wir befinden uns im Fall 1, da  $f(a_0) < 0$ . In erster Iteration ist  $x_1$  Mittelwert von  $a_0$  und  $b_0$ . Da  $f(x_1) > 0$  gilt, setzen wir  $a_1 := a_0$  und  $b_1 := x_1$ . In zweiter Iteration gilt für den Mittelwert  $x_2$  dann  $f(x_2) < 0$ . Somit setzen wir  $a_2 := x_2$  und  $b_2 = b_1$ . Dies können wir (bis zu einer gewünschten Genauigkeit) endlos fortsetzen. Dargestellt haben wir das Verfahren bis zur 7. Iteration. Mit jedem Iterationsschritt halbiert sich das Intervall auf der  $x$ -Achse. Für  $n \rightarrow \infty$  Iterationsschritte geht die Intervallgröße also gegen 0, womit wir die Nullstelle immer besser approximieren.

**Zahlenbeispiel:** Drei Iterationsschritte

$$f(x) = x^3 + 2x + 8 = 0, \quad [a_0, b_0] = [-2,5, -1]$$

$f$  ist stetig (Polynom) und  $f(a_0) \cdot f(b_0) = -63,125 < 0$  erfüllt.

Fall 1, da  $f(a_0) = -12,625 < 0$  (und  $f(b_0) = 5 > 0$ )

$$\Rightarrow x_1 = \frac{-2,5 - 1}{2} = -1,75, \quad f(x_1) \approx -0,86 < 0 \quad \Rightarrow \quad a_1 := x_1 \text{ und } b_1 := b_0$$

$$\Rightarrow x_2 = \frac{-1,75 - 1}{2} = -1,375, \quad f(x_2) \approx 2,65 > 0 \quad \Rightarrow \quad a_2 := a_1 \text{ und } b_2 := x_2$$

$$\Rightarrow x_3 = \frac{-1,75 - 1,375}{2} = -1,5625, \quad f(x_3) \approx 1,06 > 0 \quad \Rightarrow \quad a_3 := a_2 \text{ und } b_3 := x_3$$

### 2.3.1 Lineare (Un)Abhängigkeit – Definition und Prüfform



I.u./l.a. Beispiel  
3 Vektoren aus  
 $\mathbb{R}^3$

Lineare (Un)Abhängigkeit (I.u./l.a.) ist formal durch die folgende Implikation definiert:

Die Familie  $(\vec{v}_1, \dots, \vec{v}_n)$  ist linear unabhängig, wenn

$$\lambda_1 \cdot \vec{v}_1 + \lambda_2 \cdot \vec{v}_2 + \dots + \lambda_n \cdot \vec{v}_n = \vec{0} \quad \Rightarrow \quad \lambda_1 = \lambda_2 = \dots = \lambda_n = 0 \quad (2.8)$$

gilt. Andernfalls ist  $(\vec{v}_1, \dots, \vec{v}_n)$  linear abhängig.

In Worten also: Eine Familie von Vektoren ist linear unabhängig, wenn sich der Nullvektor als Linearkombination der Vektoren nur dann ergibt, wenn alle Skalierungsfaktoren Null sind.

In anderen Worten ausgedrückt ist das gleichbedeutend mit: Eine Familie von Vektoren ist linear unabhängig, wenn sich kein einziger Vektor aus der Familie durch eine Linearkombination der verbleibenden Vektoren aus der Familie darstellen lässt.

Ab und an schreiben wir auch einfach, dass Vektoren I.u. bzw. l.a. sind, anstatt die Familie der Vektoren. Dies ist völlig legitim und wird sogar in vielen Büchern/Vorlesungen so definiert.

Der erste Teil dieser Definition sollte dir bereits von der Definition einer Linearkombination bekannt vorkommen. Äquivalent lässt sich auch die Definition (2.8) in ein LGS umschreiben:

$$\lambda_1 \vec{v}_1 + \lambda_2 \vec{v}_2 + \dots + \lambda_n \vec{v}_n = \underbrace{\begin{pmatrix} \vec{v}_1 & \dots & \vec{v}_n \end{pmatrix}}_{\text{Matrix}} \cdot \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \quad (2.9)$$

homogenes LGS

$$\text{bzw.} \quad \underbrace{\begin{pmatrix} \vec{v}_1 & \dots & \vec{v}_n \end{pmatrix}}_{\text{Matrix}} \cdot \vec{\lambda} = \vec{0} \quad \Rightarrow \quad \vec{\lambda} = \vec{0}$$

homogenes LGS

Die Vektoren  $\vec{v}_1, \dots, \vec{v}_n$  sind dabei die Spalten der Matrix.

In (2.9) sehen wir also folgendes:

**Die Spaltenvektoren einer Matrix sind genau dann linear unabhängig, wenn das zugehörige homogene LGS eindeutig lösbar ist.**

**Äquivalent: Die Spaltenvektoren einer Matrix sind genau dann linear abhängig, wenn das zugehörige homogene LGS unendlich viele Lösungen besitzt.**

Prüfe, ob die Familie  $\left( \begin{pmatrix} 1 \\ 3 \end{pmatrix}, \begin{pmatrix} -2 \\ 2 \end{pmatrix} \right)$  I.u. ist.

$$\xrightarrow{(2.9)} \left[ \begin{array}{cc|c} 1 & -2 & 0 \\ 3 & 2 & 0 \end{array} \right] \begin{array}{l} \cdot (-3) \\ \leftarrow \end{array} \rightsquigarrow \left[ \begin{array}{cc|c} 1 & -2 & 0 \\ 0 & 8 & 0 \end{array} \right]$$

$$\Rightarrow \begin{cases} 8 \cdot \lambda_2 = 0 & \Leftrightarrow \lambda_2 = 0 \\ 1 \cdot \lambda_1 - 2 \cdot 0 = 0 & \Leftrightarrow \lambda_1 = 0 \end{cases}$$

Da  $\lambda_1 = \lambda_2 = 0$  die einzige Lösung des LGS ist, ist die Familie I.u.

Die Vektoren aus dem Beispiel sind in nachfolgender Abbildung skizziert.



Hier lässt sich recht einfach erkennen, dass die beiden Vektoren  $\vec{v}_1$  und  $\vec{v}_2$  jeweils mit Null multipliziert werden müssen, um den Nullvektor darzustellen.

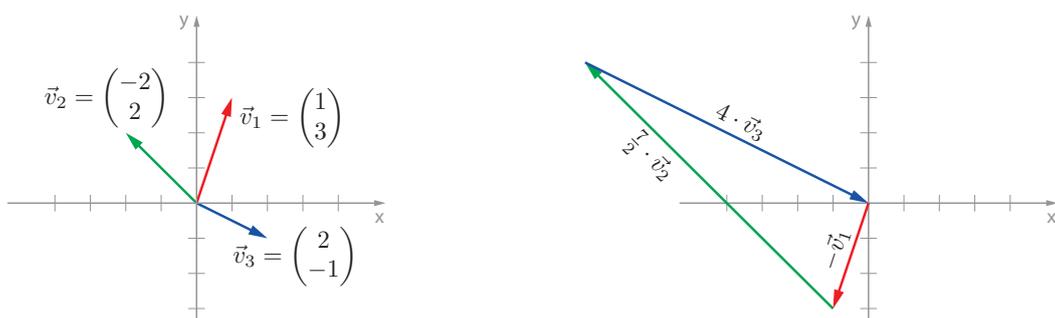
Prüfe, ob die Familie  $\left( \begin{pmatrix} 1 \\ 3 \end{pmatrix}, \begin{pmatrix} -2 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ -1 \end{pmatrix} \right)$  l.u. ist.

$$\xrightarrow{(2.9)} \left[ \begin{array}{ccc|c} 1 & -2 & 2 & 0 \\ 3 & 2 & -1 & 0 \end{array} \right] \begin{array}{l} | \cdot (-3) \\ | \leftarrow \end{array} \rightsquigarrow \left[ \begin{array}{ccc|c} 1 & -2 & 2 & 0 \\ 0 & 8 & -7 & 0 \end{array} \right]$$

$$\Rightarrow \begin{cases} \lambda_3 \in \mathbb{R} \\ 8 \cdot \lambda_2 - 7 \cdot \lambda_3 = 0 & \Leftrightarrow \lambda_2 = \frac{7}{8} \cdot \lambda_3 \\ 1 \cdot \lambda_1 - 2 \cdot \left( \frac{7}{8} \cdot \lambda_3 \right) + 2 \cdot \lambda_3 = 0 & \Leftrightarrow \lambda_1 = -\frac{1}{4} \cdot \lambda_3 \end{cases}$$

Für die reine Beantwortung der Ausgangsfrage hätten wir bereits nach dem Gauß-Algorithmus aufhören können, denn bereits da sehen wir, dass eine Variable am Ende frei wählbar bleibt und das LGS somit unendlich viele Lösungen besitzt. Also ist die Familie l.a.

Wählen wir aus dem obigen Beispiel z. B.  $\lambda_3 = 4$ , erhalten wir für die anderen Skalierungsfaktoren  $\lambda_1 = -\frac{1}{4} \cdot 4 = -1$  und  $\lambda_2 = \frac{7}{8} \cdot 4 = \frac{7}{2}$ . Dies ist in der unten stehenden Skizze verdeutlicht:



Wie die Rechnung schon gezeigt hat, lassen sich hier die drei Vektoren derart skalieren und addieren, dass sie am Ende den Nullvektor darstellen. Bei größer/kleiner gewähltem  $\lambda_3$  würde sich das Dreieck aus den Vektoren vergrößern/verkleinern.

**Ab hier sollte dir nun klar sein, dass**

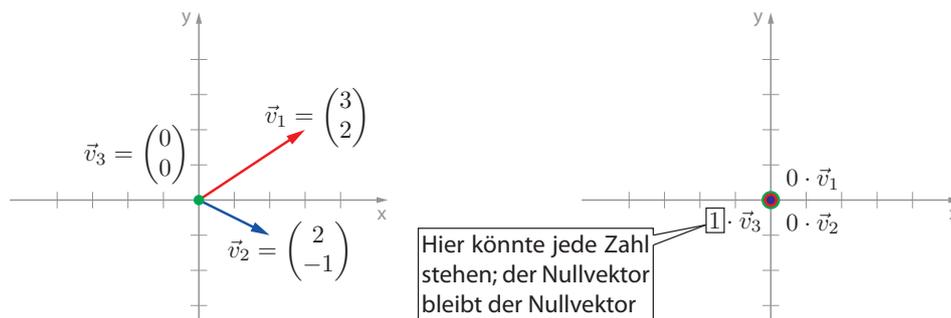
- eine LK eines Vektors durch eine l.u. Familie von Vektoren immer eindeutig ist
- eine LK eines Vektors durch eine l.a. Familie von Vektoren immer mehrdeutig ist

### 2.3.2 Schnelltests zur linearen (Un)Abhängigkeit

Wir müssen nicht immer mit der Definition (2.8) bzw. der Darstellung (2.9) arbeiten/argumentieren. Es gibt einige – teils schnellere – Methoden (wir haben diese einfach „Schnelltests“ genannt), um eine Familie von Vektoren auf lineare (Un)Abhängigkeit zu überprüfen.

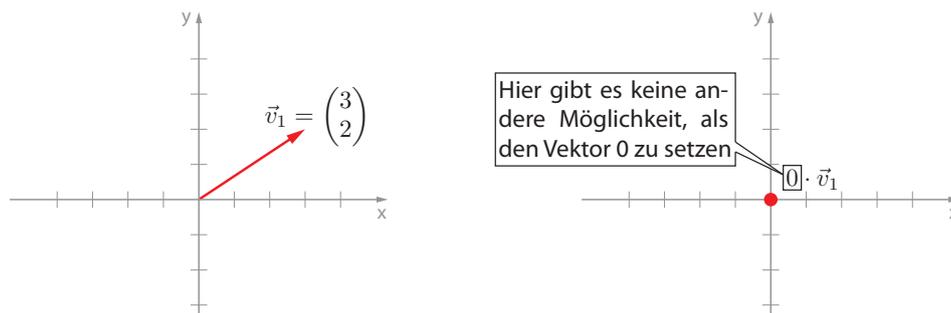
Möglicherweise darfst du diese Schnelltests in deiner Mathematik-Veranstaltung nicht offiziell zur Argumentation benutzen. Sie dienen dir allerdings trotzdem enorm fürs Verständnis!

- Wenn der Nullvektor in einer Familie enthalten ist, ist diese immer l.a., denn wir könnten den Nullvektor mit einer Zahl ungleich Null und alle anderen Vektoren aus der Familie mit Null multiplizieren. Das Ergebnis wäre der Nullvektor, obwohl wir nicht alle Skalierungsfaktoren Null gesetzt haben. Damit wäre die Familie l.a.



$$\text{Familie } \left( \underbrace{\begin{pmatrix} 3 \\ 2 \end{pmatrix}}_{\vec{v}_1}, \underbrace{\begin{pmatrix} 2 \\ -1 \end{pmatrix}}_{\vec{v}_2}, \underbrace{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}_{\vec{v}_3} \right) \text{ ist l.a.: } \begin{cases} \lambda_1 \cdot \vec{v}_1 + \lambda_2 \cdot \vec{v}_2 + \lambda_3 \cdot \vec{v}_3 = \vec{0} \\ \Rightarrow \lambda_1 = \lambda_2 = 0, \text{ aber } \lambda_3 \text{ beliebig} \end{cases}$$

- Wenn eine Familie nur aus einem einzigen Vektor ( $\neq \vec{0}$ ) besteht, dann ist diese l.u. Das ist sehr plausibel, denn wenn der Vektor nicht der Nullvektor ist, dann hat er in mindestens einer seiner Komponenten eine Zahl ungleich 0 stehen. Dann steht in dieser Komponente eine einfache Gleichung wie  $\lambda \cdot \text{„Zahl“} = 0$  mit dem eindeutigen Ergebnis  $\lambda = 0$ .



$$\text{Familie } \left( \begin{pmatrix} 3 \\ 2 \end{pmatrix} \right) \text{ ist l.u.: } \lambda \cdot \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Rightarrow \begin{cases} 1. \text{ Komp.: } \lambda \cdot 3 = 0 \Rightarrow \lambda = 0 \\ (2. \text{ Komp. } \dots \text{ nicht mehr relevant}) \end{cases}$$

- Wenn eine Familie aus zwei Vektoren besteht und diese beiden parallel zueinander stehen



# 4 Differentialgleichungen (DGL)

Jegliche Gleichungen mit Unbekannten, die uns bisher begegnet sind, konnten durch geeignete Zahlen für die Unbekannten gelöst werden, zum Beispiel

$$x^2 - 9 = 0 \Leftrightarrow x^2 = 9 \Leftrightarrow x = -3 \vee x = 3.$$

Eine **Differentialgleichung** (kurz **Diff.gleichung** oder **DGL**) ist nun etwas völlig anderes. Es ist eine Gleichung, in der eine Funktion und auch Ableitungen von dieser auftauchen. Die Lösung dieser Art von Gleichung ist also eine komplette Funktion.



Was ist eine DGL?

Gesucht sind *alle* Funktionen  $y(x)$ , die die folgende DGL erfüllen:

$$y'(x) = 2 \cdot y(x)$$

Wir suchen also alle Funktionen  $y(x)$ , die mit 2 multipliziert gleich ihrer Ableitung sind. Lösung:

$$y(x) = C \cdot e^{2x} \text{ mit } C \in \mathbb{R}, \text{ denn } y'(x) = 2 \cdot \underbrace{C \cdot e^{2x}}_{=y(x)} = 2 \cdot y(x)$$

Oft soll die Funktion  $y(x)$  noch gegebenen Anfangs-/Randwerten genügen (Anfangswertproblem (AWP) oder Randwertproblem (RWP)) (also  $y(x_0) = y_0$  soll für gegebene  $x_0, y_0$  gelten)

Gesucht ist *die* Funktion  $y(x)$ , die dem folgenden AWP genügt:

$$y'(x) = 2 \cdot y(x), y(0) = 3$$

Durch die zusätzliche Bedingung  $y(0) = 3$  werden aus unendlich vielen Lösungen (durch das  $C$ ) genau eine Lösung, die dem AWP genügt. Lösung:

$$y(x) = 3 \cdot e^{2x} \text{ denn } y'(x) = 2 \cdot \underbrace{3 \cdot e^{2x}}_{=y(x)} = 2 \cdot y(x) \text{ und } y(0) = 3 \cdot e^{2 \cdot 0} = 3 \cdot 1 = 3$$

Also: Funktion mit maximalem Definitions- und Wertebereich:

$$y: \mathbb{R} \rightarrow (0, \infty), y(x) = 3 \cdot e^{2x}$$

Von einem AWP sprechen wir normalerweise, wenn die DGL zum „Zeitpunkt“  $x_0 = 0$  irgendwelchen Startwerten genügen soll. Für  $x_0 \neq 0$  sprechen wir von einem RWP.

Dies war ein sehr einfaches Beispiel und vorallem haben wir die Lösungen vorgegeben. Wie du Lösungen von dieser und von komplizierteren DGLs □ bestimmst, erfährst du in den nächsten Unterkapiteln.

<sup>1</sup>Nicht zu jeder DGL existiert eine analytische Lösung für  $y(x)$ . Wir wollen hier jedoch nicht auf numerische Methoden zum lösen von DGLs eingehen.

## 4.1 Notationen

Zunächst wollen wir hier kurz klären, in welchen Schreibweisen DGLs auftreten können.

### Funktion mit und ohne Funktionsargument

Je nach Mathematik Veranstaltung, Büchern oder Internetquellen, kann die Funktion in der DGL mit und ohne Funktionsargument notiert sein.

$$\begin{array}{ll} \text{Ohne Funktionsargument:} & 2y' + yx = 0 \\ \text{Mit Funktionsargument:} & \Leftrightarrow 2y'(x) + y(x)x = 0 \end{array}$$

Im Fall ohne Funktionsargument erkennen wir die Funktion daran, welche der Variablen eine Ableitungskennzeichnung enthält (im Beispiel oben das  $y$ , da  $y'$  auftaucht). Die andere Variable ist dann die Variable, von der die Funktion abhängt (Im Beispiel oben das  $x$ ).

**Im Folgenden werden wir bei der Funktion *nicht* mit dazu schreiben, von welcher Variable diese abhängt. Diese Notation stellt den Standard in den allermeisten Veranstaltungen dar und du solltest dich daran gewöhnen.**

### Notation der Ableitungsterme

Im Allgemeinen solltest du wissen, dass Folgendes gilt:

$$y^{(n)} \Leftrightarrow n\text{-te Ableitung der Funktion } y \quad (4.1)$$

Verwechsle in diesem Zuge auf keinen Fall das Folgende:

$$y^{(2)} = y'' \neq y^2 = y \cdot y$$

Es kann vorkommen, dass du bei DGLs Ableitungen wie  $\dot{y}$  anstatt  $y'$  siehst. Das bedeutet lediglich, dass die Variable die Zeit  $t$  ist und  $\dot{y}$  die Ableitung nach der Zeit darstellt:

$$y(t), \quad \frac{dy}{dt}(t) = \dot{y}(t), \quad \frac{d^2y}{dt^2}(t) = \ddot{y}(t), \quad \dots$$

### Implizite und explizite Darstellung

Die explizite Darstellung einer DGL erhalten wir, wenn wir die DGL auf die höchste vorkommende Ableitung umstellen können. Falls das nicht möglich ist, kann die DGL nur in impliziter Darstellung geschrieben werden.

$$\begin{array}{l} \text{DGL 1: Implizite und explizite Darstellung: } 2y' + yx = 0 \Leftrightarrow y' = -\frac{1}{2}yx \\ \text{DGL 2: Implizite Darstellung: } y' + e^{y'} = y \quad (\text{nicht auf } y' \text{ auflösbar}) \end{array}$$



### Abstrakte Notation durch „übergeordnete“ Funktion

Manchmal ist es hilfreich (z. B. um Begriffe sauber zu definieren oder spezielle Lösungsstrategien mathematisch sauber aufzuschreiben), die DGL als „übergeordnete“ Funktion zu deklarieren ( $f$  für

explizite Darstellung und  $F$  für implizite Darstellung). Diese hängt dann von den Variablen und der gesuchten Funktion samt ihren Ableitungen ab (recht abstrakte Notation):

$$\text{In expliziter Form: } y^{(n)} = f(x, y, y', y'', \dots, y^{(n-1)}) \quad (4.2)$$

$$\text{In impliziter Form: } 0 = F(x, y, y', y'', \dots, y^{(n-1)}, y^{(n)}) \quad (4.3)$$

$$\text{DGL 1: } 2y' + yx = 0 \quad \Leftrightarrow \quad y' = -\frac{1}{2}yx =: f(x, y)$$

$$\text{DGL 2: } y' + e^{y'} = y \quad \Leftrightarrow \quad 0 = y - y' - e^{y'} =: F(x, y, y')$$

### Notation innerhalb dieses Buchs

Wir werden uns in diesem gesamten Kapitel darauf einigen, dass in den DGLs  $y$  die gesuchte Funktion darstellt,  $x$  die Variable ist und daher auch  $'$  die Ableitung kennzeichnet.

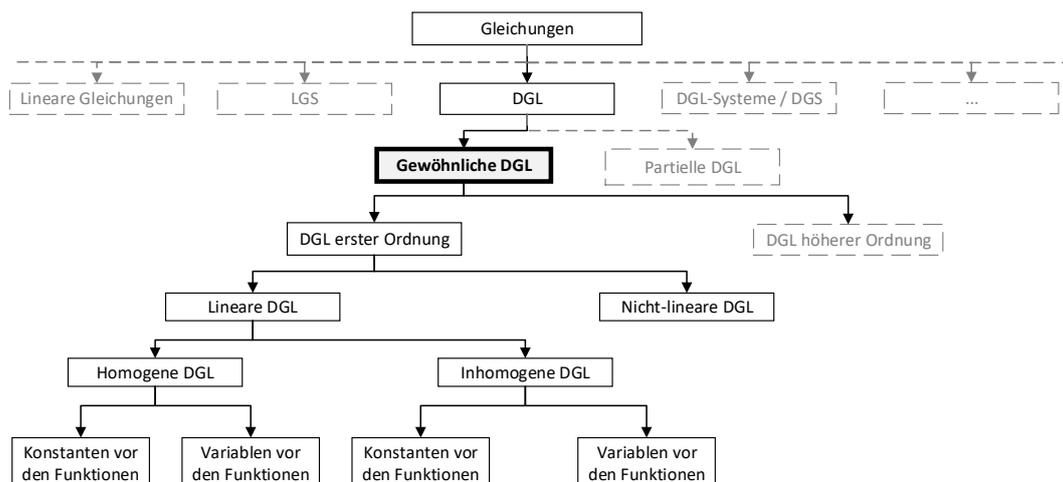
Ganz bewusst quälen wir dich schon an dieser Stelle nicht mit unnötig komplizierten Notationen, die die partiellen DGLs miteinbeziehen würden, denn wir werden diese hier nicht behandeln.

## 4.2 Typisierungen

Dieses Unterkapitel stellt im ersten Schritt das wichtigste Kapitel dar. Ohne eine vorliegende DGL korrekt zu typisieren, werden wir kein geeignetes Verfahren auswählen können, um Lösungen der DGL zu bestimmen. Außerdem ordnen wir zu der Typisierung auch die Begrifflichkeiten von Lösungsvarianten zu.

### 4.2.1 Typisierung der DGL

Zur Einführung hier für dich eine kleine Übersicht für die Strukturierung der Typisierung. Das „Baumdiagramm“ sollte selbsterklärend sein.



Die grau gestrichelten Typisierungen in der Übersicht werden wir in diesem Buch nicht behandeln.