Lineare und nichtlineare Optimierung

Vorlesung - Teil 1

Sven-Joachim Kimmerle

Physical Software Solutions &
Lehrbeauftragter (extern)
Institut für Angewandte Mathematik und Wissenschaftliches Rechnen (LRT-1)
Universität der Bundeswehr München, Neubiberg/München

Vorlesung HT 2019
Studiengänge BSc. Luft- und Raumfahrttechnik, BSc. ME

Organisatorisches

- 2-stündige Vorlesung mit 1-stündiger Übung:
  - Mo, 15:00-16:30 in 033-3131
  - Mi, 08:00-09:30 in 033-2211 (oder z.B. Di, 08:00-09:30 ?), 14-tägig ab Mi, 09.10.2019
- Vorlesung & Übung:

  Dr. habil. Sven-Joachim Kimmerle

  `sven-joachim.kimmerle@unibw.de`, Durchwahl 2129

  Sprechzeiten: flexibel nach Vereinbarung, Büro 041-2304
- Voraussetzungen:
  - Ingenieurmathematik 1-3
  - Programmierkenntnisse in einer beliebigen prozeduralen Programmiersprache (z.B. MATLAB) wünschenswert
- Prüfung: mündlich
- Internet:

  `www.unibw.de/ingmathe/teaching/linearenichtlineareoptimierung`

Empfohlene Literatur

[GK99] Geiger, C. und Kanzow, C.: *Numerische Verfahren zur Lösung unrestringierter Optimierungsaufgaben*. Springer-Verlag, Berlin/Heidelberg/New York, 1999.

[GK02] Geiger, C. und Kanzow, C.: *Theorie und Numerik restringierter Optimierungsaufgaben*. Springer-Verlag, Berlin/Heidelberg/New York, 2002.

[GL11] Gerdts, M. und Lempio, F.: *Mathematische Optimierungsverfahren des Operations Research.* DeGruyter, Berlin, 2011.

[Dr15] Dreves, A.: *Lineare und nichtlineare Optimierung*, Vorlesungsskript, BSc.-Studiengang “Luft- und Raumfahrttechnik”, Universität der Bundeswehr München, 2015.

[Ge16] Gerdts, M.: *Einführung in die lineare und nichtlineare Optimierung*, Vorlesungsskript, BSc.-Studiengang “Luft- und Raumfahrttechnik”, Universität der Bundeswehr München, 2016.

Inhaltsverzeichnis

Inhaltsverzeichnis

Motivation

- Optimierungsaufgaben finden sich:
  - in den Wirtschaftswissenschaften
  - im Management
  - in der Technik
  - in der Produktion
  - in der Bildverarbeitung
  - in der Natur
  - etc.
- Optimierung steht in engem Zusammenhang zur (mathematischen) Modellierung
- Idee: Zielfunktion wird unter Nebenbedingungen extremal
- Ohne Einschränkung betrachten wir nur Minimierungsprobleme

  Durch Multiplikation der Zielfunktion mit -1 wird ein Maximierungsproblem in ein äquivalentes Minimierungsproblem transformiert

Allgemeines Optimierungsproblem

Problem (Allgemeines Optimierungsproblem (OP))

Minimiere $f(x)$ unter den Nebenbedingungen $x \in X$.

Dabei sei $X \subseteq \mathbb{R}^n$ eine nichtleere Menge und

$f : X \to \mathbb{R}$ eine Funktion.

f heißt **Zielfunktion**.

x heißt **zulässig** für (OP), falls $x \in X$.

X heißt **zulässige Menge** von (OP).

Verschiedene Klassen von Optimierungsproblemen I

Bei einem **unrestringierten Optimierungsproblem (UOP)** gilt $X = \mathbb{R}^n$.

Oft sind die Restriktionen durch (Un-)Gleichungen gegeben:

Problem (Standard-Optimierungsproblem (SOP))

Minimiere $f(x)$ unter den Nebenbedingungen (u. d. N.)

$$g_i(x) \leq 0, \quad i = 1, \ldots, m,$$
$$h_j(x) = 0, \quad j = 1, \ldots, p.$$

bzw. in Vektornotation

$$g(x) \leq 0,$$
$$h(x) = 0.$$

Dabei seien $x \in \mathbb{R}^n$ und die Funktionen

$$f : \mathbb{R}^n \to \mathbb{R},$$
$$g = (g_1, \ldots, g_m)^\top : \mathbb{R}^n \to \mathbb{R}^m \text{ und}$$
$$h = (h_1, \ldots, h_p)^\top : \mathbb{R}^n \to \mathbb{R}^p.$$

Verschiedene Klassen von Optimierungsproblemen II

Wichtiger Spezialfall:

Problem (Lineares Optimierungsproblem (LOP) (in primaler Normalform))

Minimiere $c^\top x$ *u. d. N.*

$$x \geq 0,$$
$$Ax = b.$$

Dabei seien:

$$x, c : \in \mathbb{R}^n,$$
$$b : \in \mathbb{R}^p \text{ und}$$
$$A : \in \mathbb{R}^{p \times n}.$$

Ergibt sich aus (SOP) für

$$f(x) = c^\top x, \quad g(x) = -x, \quad h(x) = Ax - b.$$

Verschiedene Klassen von Optimierungsproblemen III

Falls im Optimierungsproblem die Zielfunktion quadratisch ist, z.B.

$$f(x) = \frac{1}{2}(x - x_{soll})^2$$

und ansonsten die Situation von (LOP) vorliegt, dann ist dies ein **linear-quadratisches Optimierungsproblem** (LQOP).

Es gibt weitere Klassen von Optimierungsproblemen (OP):

- Vektoroptimierungsprobleme
- unendlichdimensionale OP
- semi-infinite OP
- ganzzahlige OP
- gemischt-ganzzahlige OP
- nicht-differenzierbare OP
- ...

Extrema: Minima und Maxima

Definition (Minima)

- $\hat{x} \in X$ heißt **globales Minimum** von (OP), falls

$$f(\hat{x}) \leq f(x) \quad \text{für alle } x \in X. \tag{1.1}$$

- $\hat{x} \in X$ heißt **striktes globales Minimum** von (OP), falls in (1.1) "<" für alle $x \in X$ mit Ausnahme von $x = \hat{x}$ gilt.

- $\hat{x} \in X$ heißt **lokales Minimum** von (OP), falls es für ein $\varepsilon > 0$ eine Umgebung

$$U_\varepsilon(\hat{x}) := \{x \in \mathbb{R}^n \,|\, \|x - \hat{x}\| < \varepsilon\}$$

gibt mit

$$f(\hat{x}) \leq f(x) \quad \text{für alle } x \in X \cap U_\varepsilon(\hat{x}). \tag{1.2}$$

- $\hat{x} \in X$ heißt **striktes globales Minimum** von (OP), falls in (1.2) "<" für alle $x \in X \cap U_\varepsilon(\hat{x})$ mit Ausnahme von $x = \hat{x}$ gilt.

Analog werden die entsprechenden Maxima definiert.

Typische Fragestellungen

i) Existenz zulässiger Lösungen?

ii) Existenz von Optimallösungen?

iii) Eindeutigkeit der Optimallösung?

iv) Abhängigkeit der Optimallösungen von Problemdaten?

v) Welche Eigenschaften besitzen Optimallösungen, d.h. welche Bedingungen sind notwendig?

vi) Welche Bedingungen sind hinreichend dafür, dass eine zulässige Lösung eine Optimallösung ist?

vii) Welche Bedingungen sind gleichzeitig notwendig und hinreichend?

viii) Wie gewinnt man Schranken für den Optimalwert bzw. Fehlerabschätzungen für die Optimallösung?

ix) Welche Algorithmen gibt es zur Berechnung einer Optimallösung?

x) Welche numerische Eigenschaften haben diese?

i) - vii) sind eher theoretischer Natur, aber ohne diese Fragestellungen ist die Beantwortung der numerischen Aspekte viii) - x), um die es in dieser VL gehen soll, nicht möglich.

Grund-Annahme

In dieser Vorlesung setzen wir die Existenz von Optimallösungen stets voraus.

Es gilt nämlich

Theorem (Satz von Weierstraß)

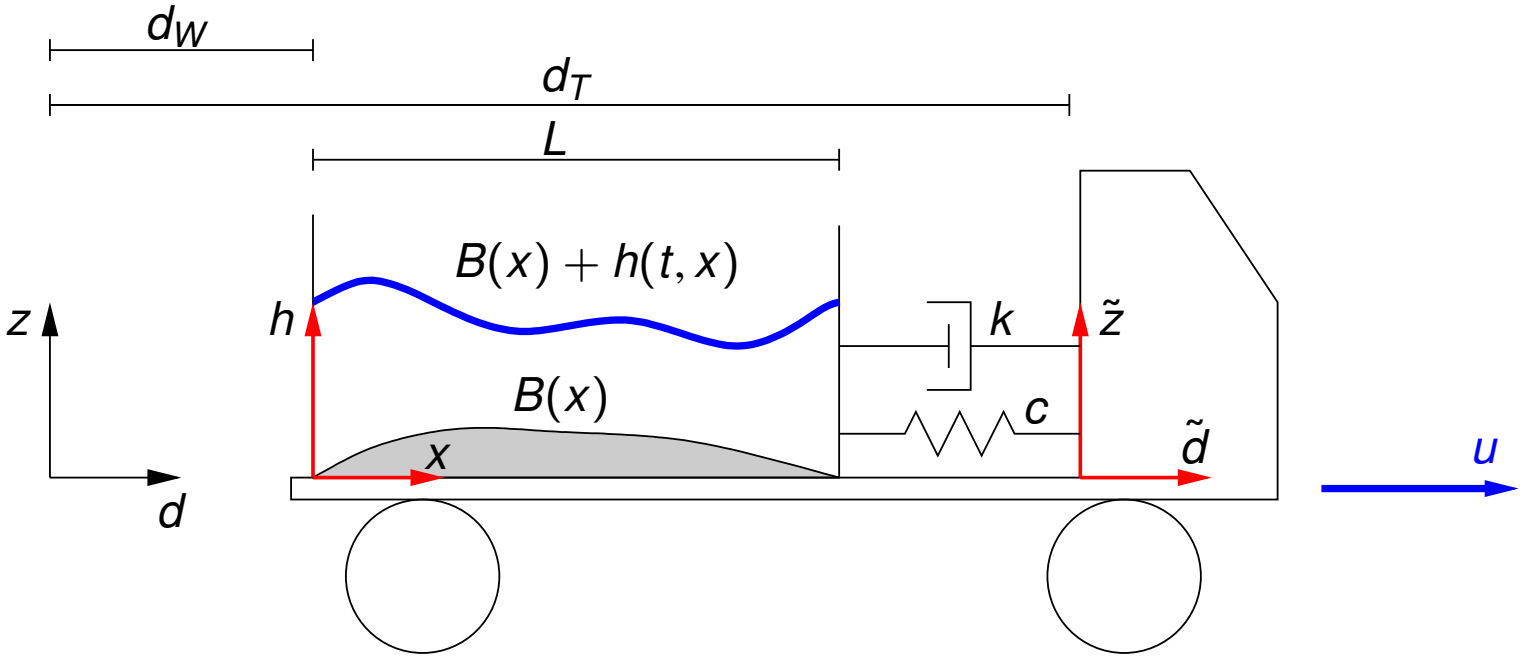
Sei $X \subseteq D \subseteq \mathbb{R}^n$ kompakt (d.h. abgeschlossen und beschränkt) und

$f : D \to \mathbb{R}$ stetig.

Dann nimmt f das Minimum (bzw. Maximum) auf X an.

Beispiele: Ein Optimalsteuerungsproblem [K., Gerdts 2015 -]

Truck with a water container: horizontal control force u, parameter final time T



m_T: mass of truck, m_W: mass of water container, d_T: distance truck, d_W: distance water container, $B(x)$: bottom elevation of container, $h(t,x)$: water level relative to bottom at time t and position x, $v(t,x)$: horizontal water velocity at time t and position x, L: length of water container, c : spring force constant, k: damper force constant

Minimierung mithilfe graphischer Darstellung

Beispiel (Funktion von Himmelblau)

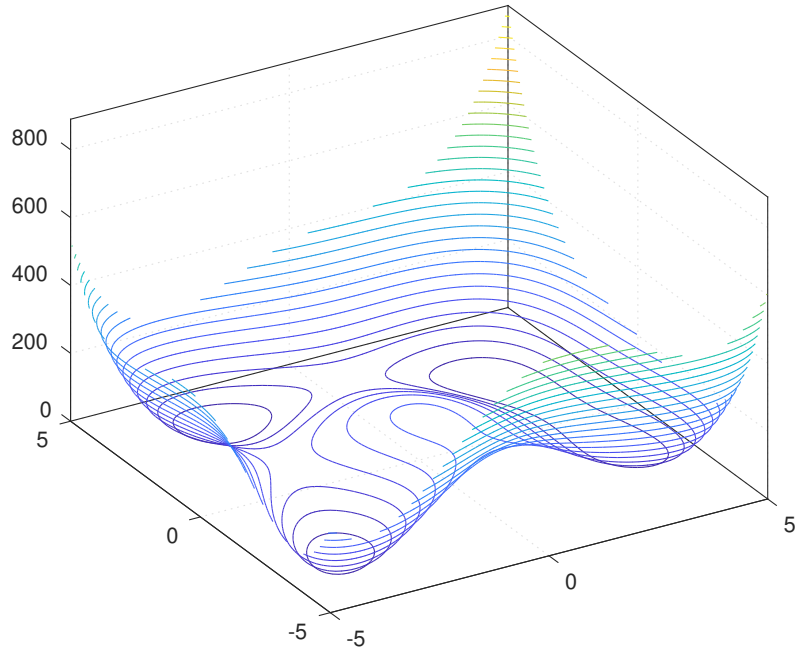
Minimiere

$$f(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2$$

über $(x, y)^\top \in \mathbb{R}^2$.

MATLAB-Code:

```
1  [x, y] = meshgrid(-5:0.1:5);
2  z = (x.^2 + y - 11).^2 + (x + y.^2 - 7).^2;
3  contour3(x, y, z, 30)
```



Höhenlinien

Eine **Höhenlinie (Niveaulinie)** $N_f(c)$ zum Niveau $c \in \mathbb{R}$ für eine Funktion $f : \mathbb{R}^n \to \mathbb{R}$ ist formal definiert als die Menge aller Punkte $x = (x_1, \ldots, x_n)^\top$ für die $f(x) = c$ gilt:

$$N_f(c) := \{x \in \mathbb{R}^n \mid f(x) = c\}, \quad c \in \mathbb{R}.$$

Auf Höhenlinien von f steht der **Gradient**

$$\nabla f(x_1, \ldots, x_n) := \begin{pmatrix} \frac{\partial f}{\partial x_1}(x_1, \ldots, x_n) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x_1, \ldots, x_n) \end{pmatrix}$$

senkrecht.

∇f ist ein Spaltenvektor. Er zeigt in die Richtung des steilsten Anstiegs.

Für reellwertige Funktionen mit einem Definitionsbereich in 1D oder 2D, kann die graphische Methode ganz hilfreich sein.

Klassisches lineares Optimierungsproblem I

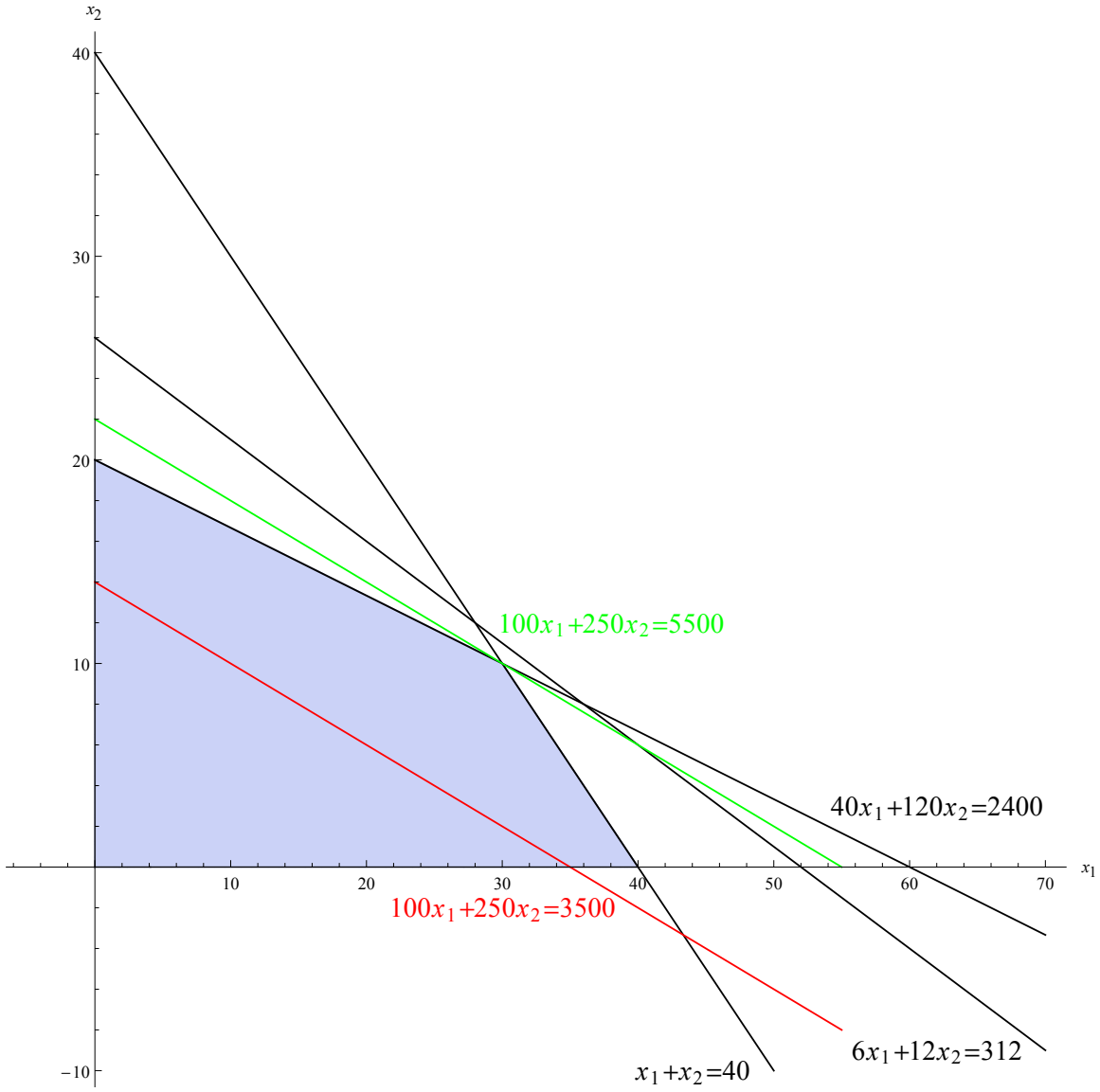
Beispiel (Agrarwirtschaft)

Ein Landwirt bewirtschaftet ein Grundstück von 40 ha Größe mit Zuckerrüben und Weizen.
Er kann hierzu 2400 € und 312 Arbeitstage einsetzen.

Die Anbaukosten betragen bei Rüben 40 €/ha und bei Weizen 120 €/ha.
Für Rüben benötigt er 6 Arbeitstage/ha und für Weizen 12 Arbeitstage/ha.
Der Reingewinn sei bei Rüben 100 €/ha und bei Weizen 250 €/ha.

Anmerkung:
Wir dürfen bereits vorneweg annehmen, dass die zur Verfügung stehenden 2400 € sofort auszugeben sind und nicht anderweitig gespart bzw. angelegt werden können und somit auch nicht in die Zielfunktion eingehen.
Wie wir sehen werden, rechtfertigt die gefundene Optimallösung diese Annahme.

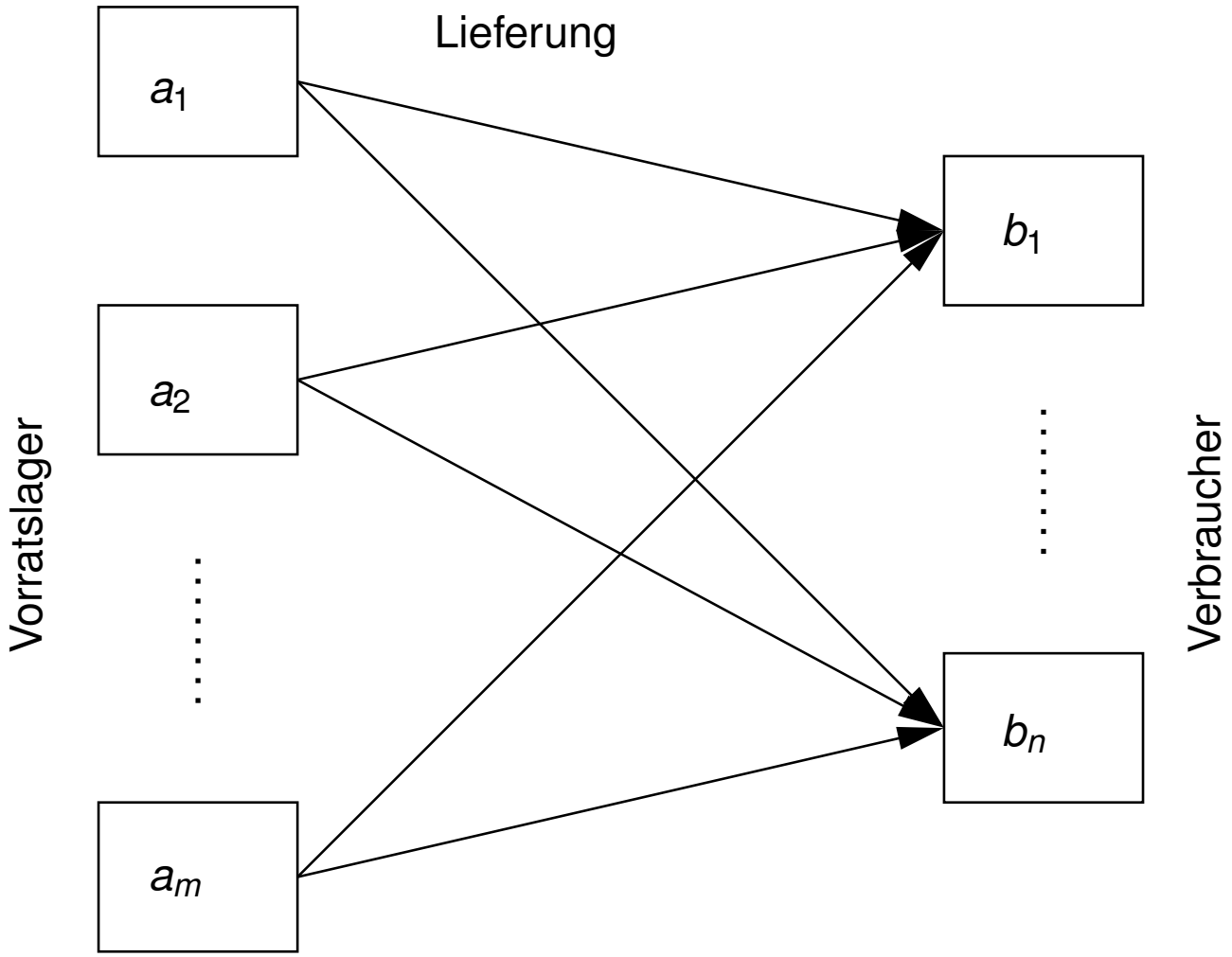
Klassisches lineares Optimierungsproblem II



Transportproblem I

Beispiel (Transportproblem)

Ein Transportunternehmer hat m Vorratslager, aus denen n Verbraucher mit einem Produkt beliefert werden können. Die Lieferkosten von Lager i zu Verbraucher j betragen c_{ij} Einheiten pro Produkteinheit. In Lager i sind a_i Einheiten des Produktes vorrätig. Verbraucher j hat einen Bedarf von b_j Einheiten. Um die Kunden nicht zu verärgern, muss der Lieferant den Bedarf der Kunden befriedigen. Andererseits möchte der Lieferant seine Lieferkosten minimieren.



Transportproblem II

Bezeichnet x_{ij} die Liefermenge von Lager i zu Verbraucher j, so führt das Problem auf das folgende Transportproblem, welches ein spezielles lineares Optimierungsproblem ist:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \quad \text{u.d.N.}$$

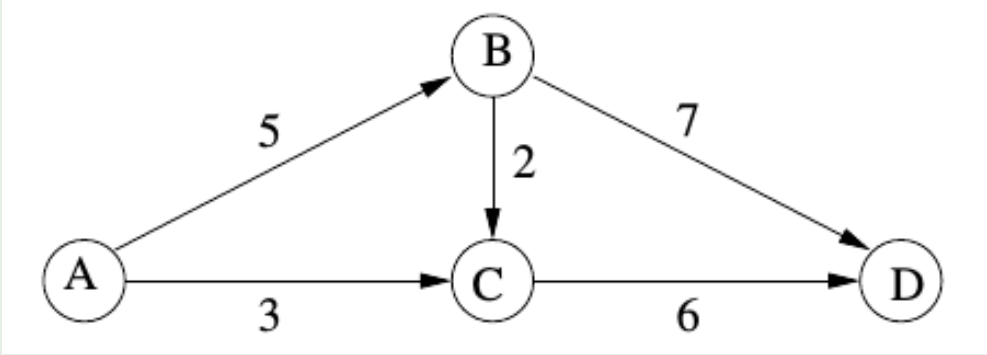
$$\begin{aligned}
\sum_{j=1}^{n} x_{ij} &\leq a_i, \quad i = 1, \ldots, m, \\
\sum_{i=1}^{m} x_{ij} &\geq b_j, \quad j = 1, \ldots, n, \\
x_{ij} &\geq 0, \quad i = 1, \ldots, m, \; j = 1, \ldots, n.
\end{aligned}$$

Die erste Nebenbedingung besagt, dass aus Lager i maximal a_i Einheiten abtransportiert werden können.
Die zweite Nebenbedingung besagt, dass der Bedarf b_j befriedigt werden muss.
Die letzte Nebenbedingung verbietet negative Liefermengen.

Graphenproblem

Beispiel (Netzwerkproblem)

Eine Firma möchte möglichst viele Waren von A nach D über das abegebildete Straßennetzwerk transportieren.



Die Zahlen neben den Kanten des Netzwerk-Graphen geben die maximale Kapazität der Kante an.

Wie kann dieses Problem mathematisch modelliert werden?

Portfoliooptimierung I

Beispiel (Portfoliooptimierungsaufgabe nach Markowitz)

Gegeben seien $j = 1, \ldots, n$ mögliche Anlagen. Jede Anlage wirft im nächsten Zeitraum einen Gewinn (bzw. Verlust) R_j ab. R_j ist allerdings nicht bekannt, aber die zugrundeliegende zufällige Verteilung.
Um einerseits den Gewinn zu maximieren und andererseits das Risiko eines Verlusts zu minimieren, wird die Anlagesumme zu prozentualen Anteilen x_j auf die n Anlagen verteilt.
Die Aufgabe eines Portfoliomanagers besteht in der optimalen Zusammenstellung des Portfolios $(x_1, \ldots, x_n)$.

Ein mögliches Ziel ist es, den Erwartungswert des Gewinns R zu maximieren:

$$E(R) = \sum_{j=1} x_j E(R_j), \quad R = \sum_{j=1}^{n} x_j R_j$$

Als Maß für das Risiko kann die Varianz von R betrachtet werden:

$$Var(R) = E(R - E(R))^2 = ER^2 - (ER)^2$$

Portfoliooptimierung II

Ein Kompromiss ist das folgende Optimierungsproblem

$$\min \; -E(R) + \alpha Var(R) \qquad \text{u. d. N.}$$

$$\sum_{j=1}^{n} x_j = 1,$$

$$x_j \geq 0, \quad j = 1, \ldots, n.$$

$\alpha \geq 0$ ist dabei ein noch zu wählender Gewichtungsparameter.

Desto kleiner α, desto größer das Risiko.

Ausgleichsrechnung und Parameteridentifikation

... später ausführlich

Inhaltsverzeichnis

Unrestringierte Optimierung

Wir betrachten hier

Problem (Unrestringiertes Optimierungsproblem (UOP))

Minimiere $f(x)$ *u. d. N.* $x \in \mathbb{R}^n$.

Dabei sei $f : \mathbb{R}^n \to \mathbb{R}$ *eine mindestens einmal stetig differenzierbare Funktion.*

Ziele:
Notwendige Bedingungen
Hinreichende Bedingungen

Wünschenswert:
Notwendige und hinreichende Bedingungen

Dann:
Numerische Verfahren (meist basierend auf notwendigen Bedingungen)

Rekapitulation I

- Der **Gradient** einer Funktion $f : \mathbb{R}^n \to \mathbb{R}$ an der Stelle $x = (x_1, \ldots, x_n)^\top \in \mathbb{R}^n$ ist definiert als Spaltenvektor

$$\nabla f(x_1, \ldots, x_n) := \begin{pmatrix} \frac{\partial f}{\partial x_1}(x_1, \ldots, x_n) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x_1, \ldots, x_n) \end{pmatrix}.$$

  Im Fall $n = 1$ ist der Gradient die 1. Ableitung von f an der Stelle x, kurz $f'(x)$.

  Anschaulich beschreibt der Gradient den Anstieg.

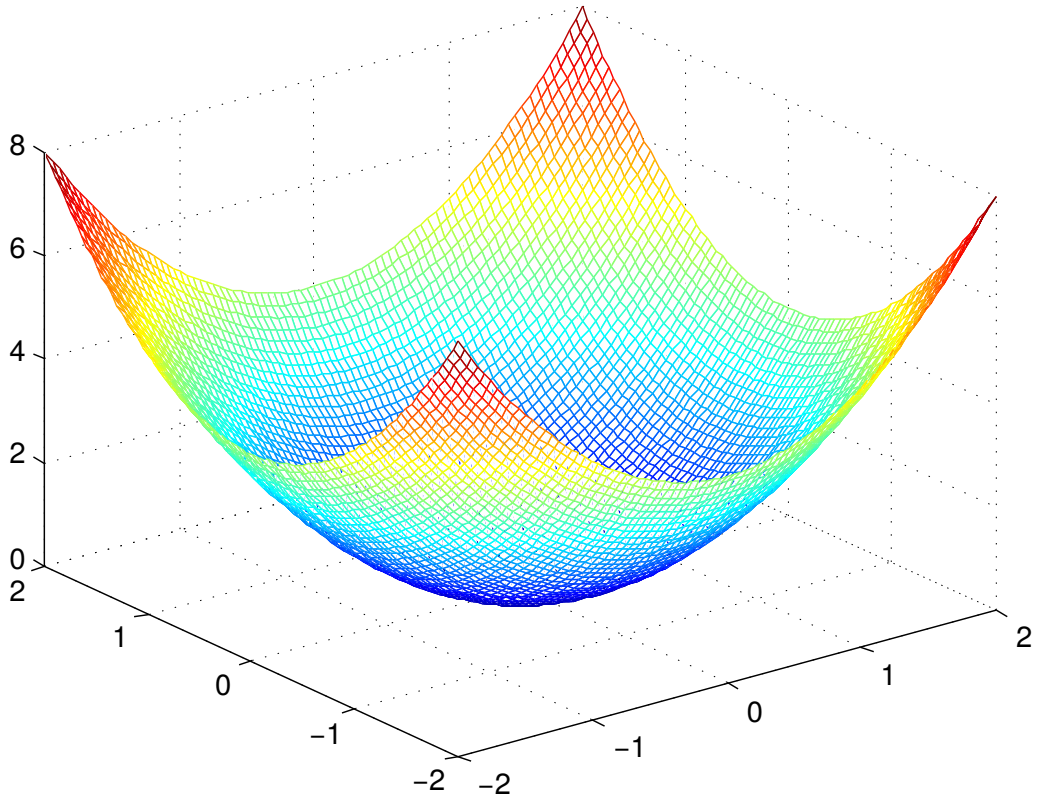
- Die **Hessematrix** einer Funktion f an der Stelle x ist definiert durch

$$H_f(x_1, \ldots, x_n) = H_f(x) = H(x) := \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1}(x) & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(x) & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n}(x) \end{pmatrix}.$$

  Im Fall $n = 1$ ist die Hessematrix die 2. Ableitung von f an der Stelle x, kurz $f''(x)$.

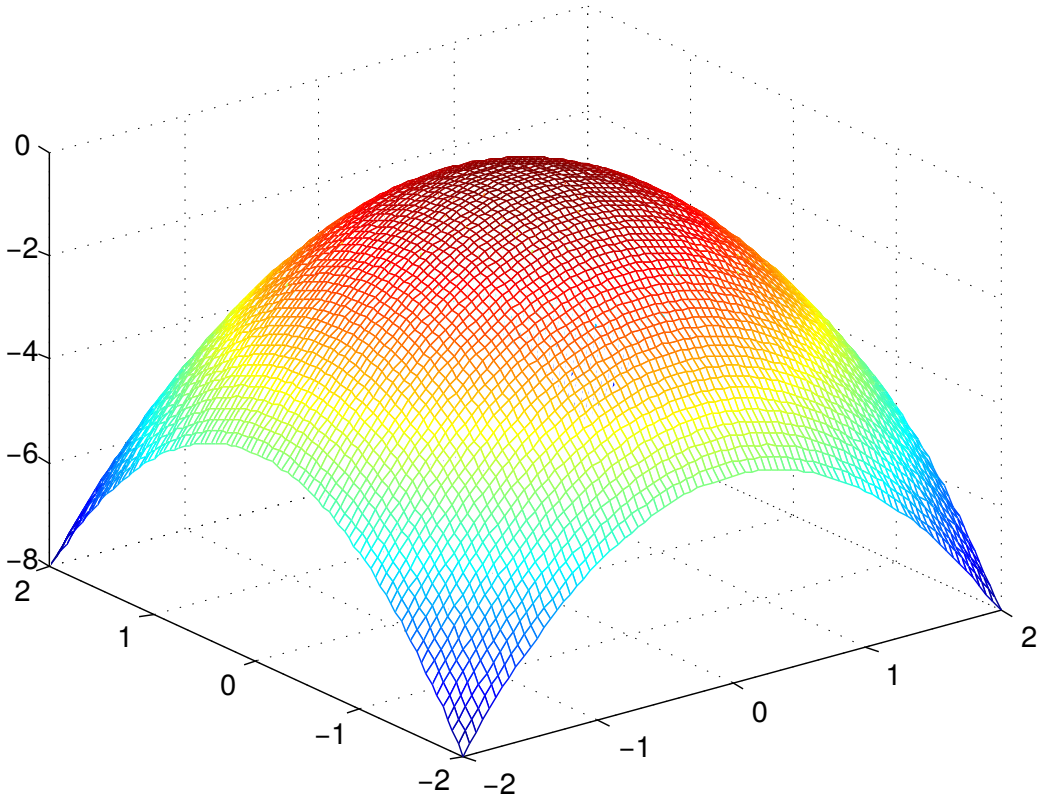
  Anschaulich beschreibt die Hessematrix die lokale Krümmung einer Funktion.

Rekapitulation II - Hessematrix I



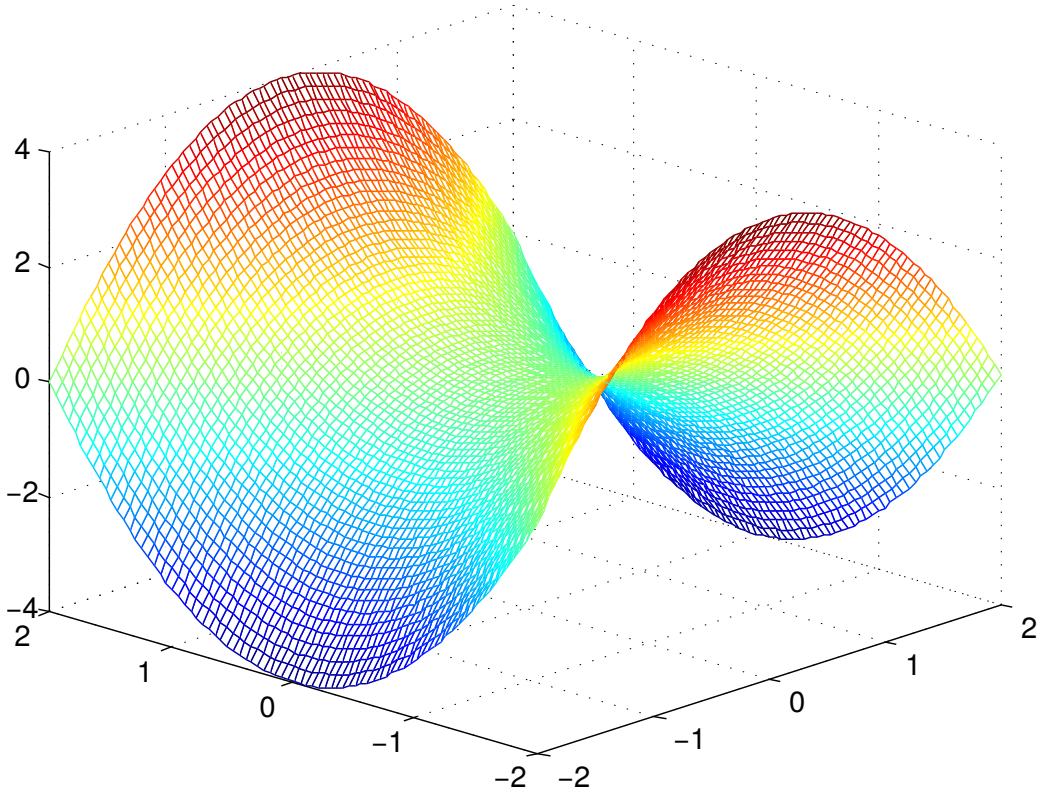
$$f(x,y) = x^2 + y^2, \qquad H_f(\hat{x}) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$

Rekapitulation II - Hessematrix II



$$f(x,y) = -x^2 - y^2, \quad H_f(\hat{x}) = \begin{pmatrix} -2 & 0 \\ 0 & -2 \end{pmatrix}$$

Rekapitulation II - Hessematrix III



$$f(x,y) = x^2 - y^2, \quad H_f(\hat{x}) = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix}$$

Rekapitulation III

Richtungsableitung

Die Richtungsableitung $f'(x;d)$ einer stetig differenzierbaren Funktion $f : \mathbb{R}^n \to \mathbb{R}$ an der Stelle $x \in \mathbb{R}^n$ in Richtung $d \in \mathbb{R}^n$ ist definiert durch

$$f'(x;d) := \lim_{t \downarrow 0} \frac{f(x+td) - f(x)}{t} = \nabla f(x)^\top d.$$

Rekapitulation IV.I - Taylorentwicklung

Taylorentwicklung

- Ist f stetig differenzierbar, dann kann f in einer Umgebung von $\hat{x}$ durch eine **affin-lineare Funktion** approximiert werden, da

$$f(x) = f(\hat{x}) + \nabla f(\hat{x})^\top (x - \hat{x}) + o(\|x - \hat{x}\|),$$

wobei

$$lim_{x \to \hat{x}} \frac{o(\|x - \hat{x}\|)}{\|x - \hat{x}\|} = 0.$$

bzw. alternativ mit $t \in \mathbb{R}$, $d \in \mathbb{R}^n$, $\xi_t \in (0, t)$

$$f(\hat{x} + td) = f(\hat{x}) + t \nabla f(\hat{x} + \xi_t d)^\top d.$$

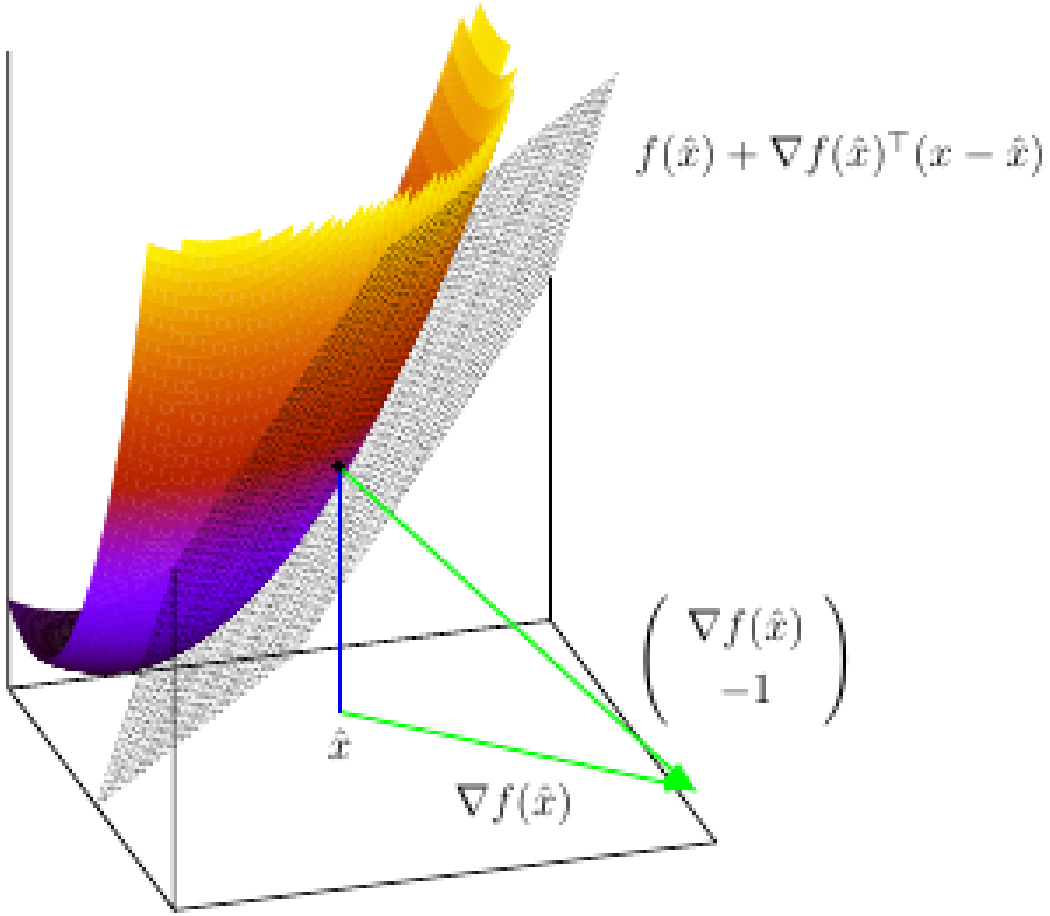
- Ist f 2mal stetig differenzierbar, dann kann f in einer Umgebung von $\hat{x}$ durch eine quadratische Funktion approximiert werden, da

$$f(x) = f(\hat{x}) + \nabla f(\hat{x})^\top (x - \hat{x}) + \frac{1}{2}(x - \hat{x})^\top H_f(\hat{x})(x - \hat{x}) + o(\|x - \hat{x}\|^2).$$

bzw. alternativ

$$f(\hat{x} + td) = f(\hat{x}) + t \nabla f(\hat{x})^\top d + \frac{t^2}{2} d^\top H_f(\hat{x} + \xi_t d) d.$$

Rekapitulation IV.II - Affin-lineare Approximation in 2D: Tangentialebene



Notwendige/hinreichende Bedingungen: Notw. Bed. 1. Ordnung

Theorem (Notwendige Bedingung 1. Ordnung)

Sei $f : \mathbb{R}^n \to \mathbb{R}$ *stetig differenzierbar und*
$\hat{x} \in \mathbb{R}^n$ *ein lokales Minimum von* f.

Dann gilt

$$\nabla f(\hat{x}) = 0.$$

Definition (Stationärer Punkt)

Jeder Punkt $x \in \mathbb{R}^n$ mit $\nabla f(x) = 0$ heißt **stationärer Punkt** von $f : \mathbb{R}^n \to \mathbb{R}$.

Stationäre Punkte sind nicht automatisch (lokale) Minima, sondern nur potentielle Kandidaten!

Die meisten numerischen Verfahren versuchen stationäre Punkte zu approximieren.

Notwendige Bedingung zweiter Ordnung

Theorem (Notwendige Bedingung 2. Ordnung)

Sei $f : \mathbb{R}^n \to \mathbb{R}$ 2mal stetig differenzierbar und $\hat{x} \in \mathbb{R}^n$ ein lokales Minimum von f.

Dann ist die Hessematrix

$$H_f(\hat{x})$$

positiv semidefinit.

Im Fall $n = 1$ gilt

$$f''(\hat{x}) \geq 0.$$

Ebenfalls findet man hierdurch nur potentielle Kandidaten für ein (lokales) Minimum.

Hinreichende Bedingung zweiter Ordnung

Theorem (Hinreichende Bedingung 2. Ordnung)

Sei $f : \mathbb{R}^n \to \mathbb{R}$ 2mal stetig differenzierbar und
$\hat{x} \in \mathbb{R}^n$ ein stationärer Punkt von f mit positiv definiter Hessematrix.

Dann ist $\hat{x}$ ein striktes lokales Minimum von f.

Im Fall $n = 1$ wird vorausgesetzt, dass

$$f''(\hat{x}) > 0.$$

Damit können wir entscheiden, ob ein Kandidat $\hat{x}$ der notwendige Bedingungen (1. und 2. Ordnung) erfüllt, in der Tat ein lokales Minimum ist.

Abstiegsrichtung

Definition (Abstiegsrichtung)

Seien $f : \mathbb{R}^n \to \mathbb{R}$ und $x \in \mathbb{R}^n$.

$d \in \mathbb{R}^n$ heißt **Abstiegsrichtung von** f **in** x, falls es ein $\bar{\alpha} > 0$ gibt mit

$$f(x + \alpha d) < f(x) \quad \text{für alle } 0 < \alpha \leq \bar{\alpha}.$$

Lemma (Hilfssatz 3.1.2)

Sei $f : \mathbb{R} \to \mathbb{R}$ stetig differenzierbar in x.

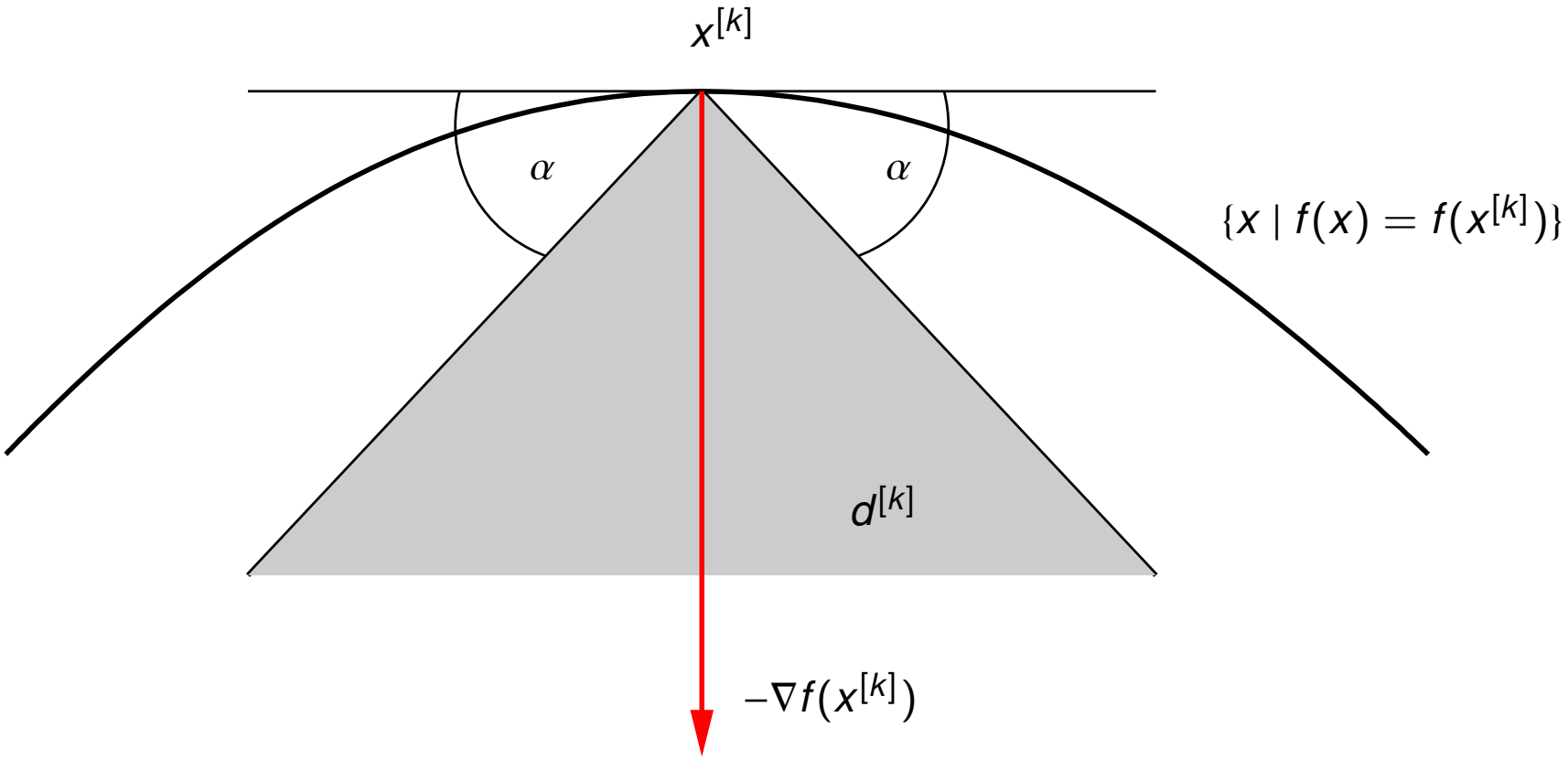
Gilt $\nabla f(x)^\top d < 0$, so ist d eine Abstiegsrichtung von f in x.

Winkelbedingung

Geometrische Interpretation des Lemma:

Winkel zwischen Gradient und Abstiegsrichtung zwischen 90° und 270°

Für effiziente Algorithmen kommt sogar nur ein geeigneter grauer Kegel für $d^{[k]}$ mit $\alpha \in (0, 90°)$ in Frage, so dass man unabhängig von k von der Niveaulinie wegbleibt (Winkelbedingung):



Abstiegsverfahren

Ziel: Verfahren zur Minimierung von $f \in C^1(\mathbb{R}^n)$

Idee: Kombiniere Abstiegsrichtungen mit einer Schrittweitensteuerung:

Algorithmus (*Allgemeines Abstiegsverfahren*)

(i) *Wähle Startpunkt* $x^{[0]} \in \mathbb{R}^n$.
Setze $i = 0$.

(ii) *Falls Abruchkriterium erfüllt, STOP.*

(iii) *Berechne Abstiegsrichtung* $d^{[i]}$ *und*
Schrittweite $\alpha_i > 0$, *so dass*

$$f(x^{[i]} + \alpha_i d^{[i]}) < f(x^{[i]}).$$

Setze $x^{[i+1]} = x^{[i]} + \alpha_i d^{[i]}$.

(iv) *Setze* $i := i + 1$ *und*
gehe zu (ii).

Beispiele für Abstiegsverfahren

- Gradientenverfahren (Verfahren des steilsten Abstiegs):

$$d^{[i]} := -\nabla f(x^{[i]})$$

  Abstiegsrichtung in nicht stationären Punkten.

  Naheliegend, aber nicht die effizienteste Wahl.

- Newtonverfahren:

$$d^{[i]} := -H_f(x^{[i]})^{-1} \nabla f(x^{[i]})$$

  Falls $H_f(x^{[i]})$ positiv definit, dann $d^{[i]}$ Abstiegsrichtung in nicht stationären Punkten.

Vergleich von Abstiegsverfahren

Liniensuche zur Schrittweitenbestimmung entlang Linie

$$\varphi(\alpha) := f(x + \alpha d)$$

Beispiel (Bsp. 3.2.2)

Minimiere $f(x_1, x_2) := x_1^2 + 10x_2^2$.

Exakte Liniensuche zur Bestimmung der Schrittweite:

$$\alpha := \mathit{argmin}\{\varphi(\alpha) \,:\, \alpha > 0\} = \mathit{argmin}\{f(x + \alpha d) \,:\, \alpha > 0\}$$

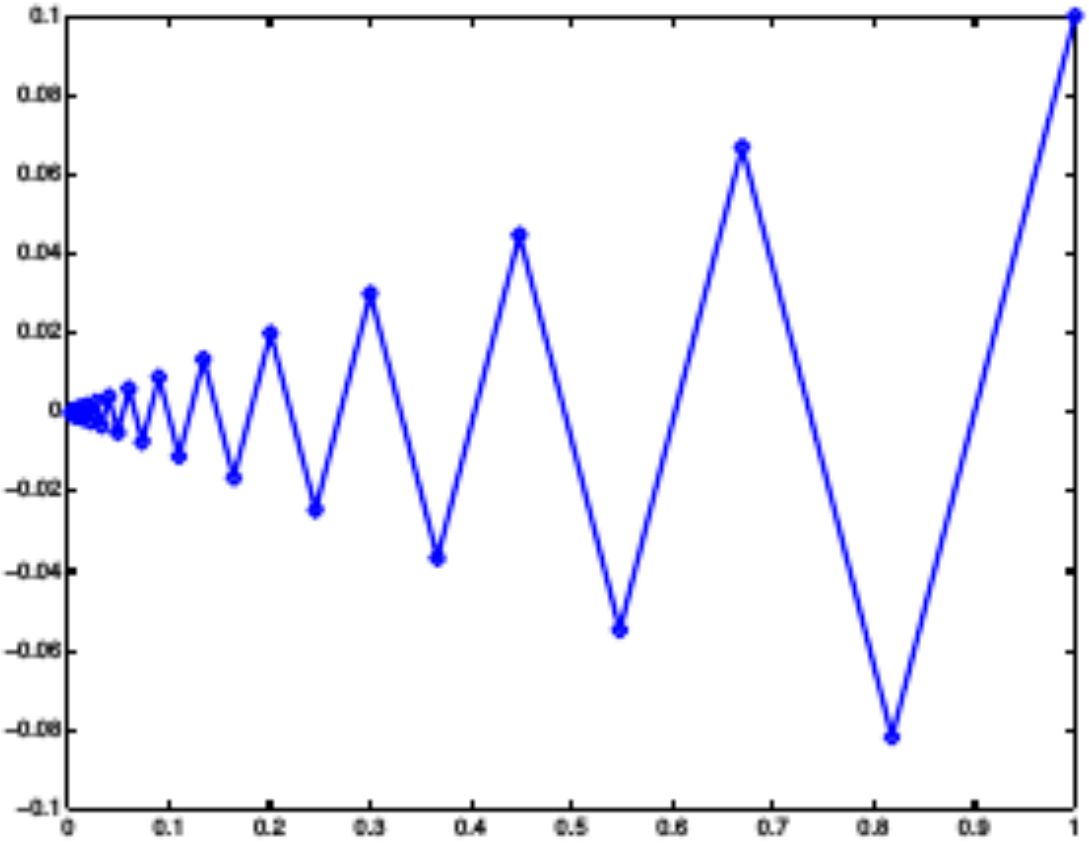
Abbruchkriterium $\|\nabla f(x)\|_2 \leq 10^{-5}$

Startvektor $x^{[0]} := (1, 0.1)^\top$

Gradientenverfahren benötigt (unskaliert) 63 Iterationen.

Newtonverfahren benötigt 1 Iteration (und löst exakt).

Konvergenz des Gradientenverfahren im Vergleich



Schrittweitenbestimmung

Algorithmus (*Armijo-Regel*)

(i) *Wähle* $\beta \in (0,1)$, $\sigma \in (0,1)$.
Setze $\alpha := 1$.

(ii) *Falls*

$$\varphi(\alpha) \leq \varphi(0) + \sigma\alpha\varphi'(0),$$

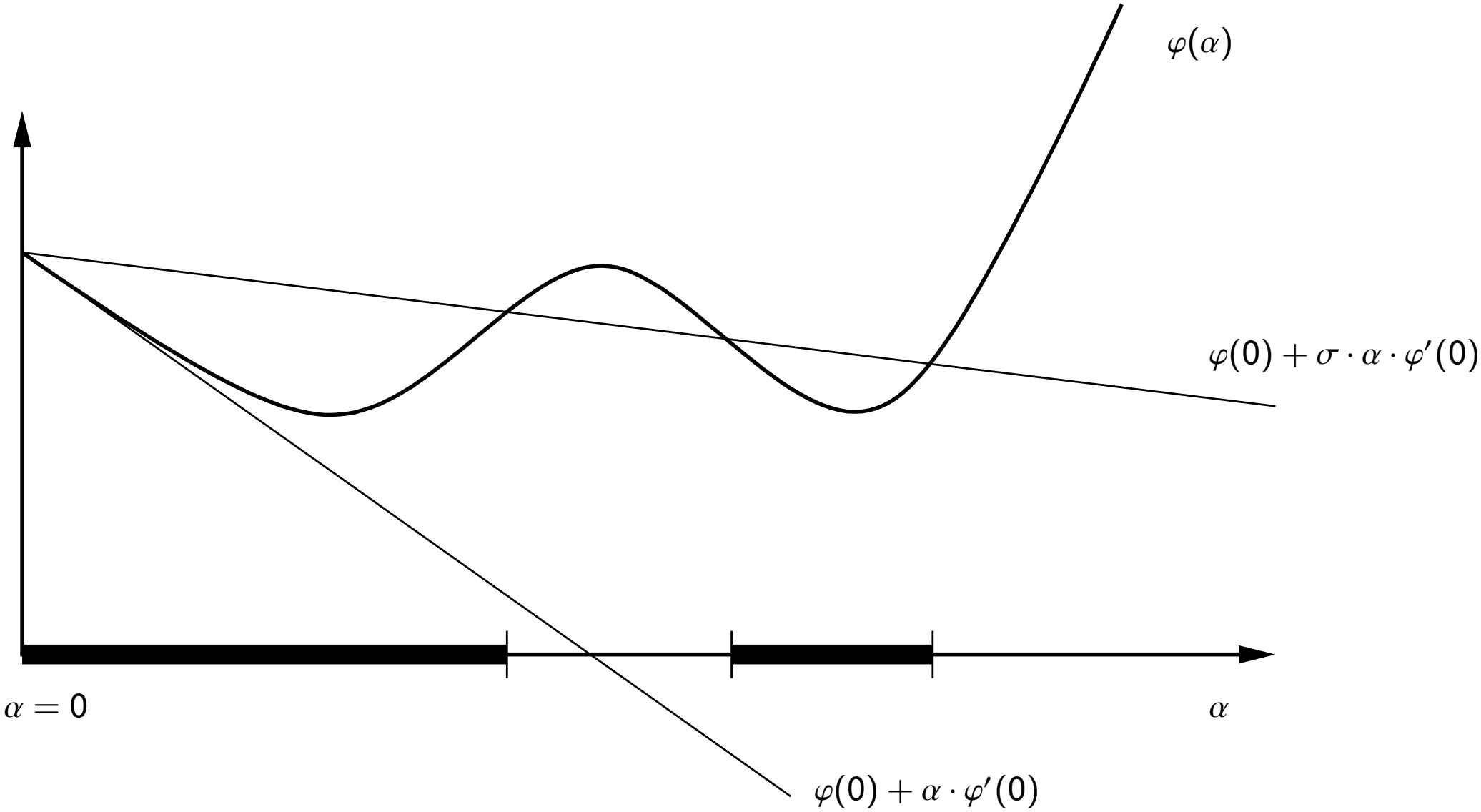
setze $\alpha_j := \alpha$ *und STOP.*

(iii) *Setze* $\alpha := \beta\alpha$.
Gehe zu (ii).

Die Armijo-Regel ist wohldefiniert.

In der Praxis typischerweise: $\beta = 0.9$, $\sigma = 0.01$

Intervalle, die Armijo-Bedingung erfüllen



Gradientenverfahren

Algorithmus (*Gradientenverfahren mit Liniensuche*)

(0) *Wähle* $x^{[0]} \in \mathbb{R}^n$,
einen Schrittweiten(reduktions)faktor $\beta \in (0,1)$ *und*
einen Steigungsfaktor $\sigma \in (0,1)$.
Setze $k := 0$.

(i) *Falls*

$$\|\nabla f(x^{[k]})\| \approx 0,$$

dann STOP.

(ii) *Berechne Suchrichtung* $d^{[k]} = -\nabla f(x^{[k]})$.

(iii) *Bestimme eine Schrittweite* α_k *mit der Armijo-Regel.*

(iv) *Setze* $x^{[k+1]} := x^{[k]} + \alpha_k d^{[k]}$,
setze $k := k + 1$ *und*
gehe zu (i).

Alternative: $d^{[k]} = -\frac{\nabla f(x^{[k]})}{\|\nabla f(x^{[k]})\|}$

Konvergenz des Gradientenverfahrens

Theorem (Satz 3.3.3 - Konvergenzsatz für Gradientenverfahren)

Wird durch das Gradientenverfahren mit Liniensuche für $f \in C^1(\mathbb{R}^n)$, $x^{[0]} \in \mathbb{R}^n$ eine nicht abbrechende Folge $\{x^{[k]}\}_{k\in\mathbb{N}}$ definiert,
so gilt für jeden Häufungspunkt x^* dieser Folge $\nabla f(x^*) = 0$, d.h. x^* ist ein stationärer Punkt von f.

Vor-/Nachteile:

- Einfach zu implementieren
- Nur **lineare Konvergenz**:
  Unter Annahmen gilt

$$f(x^{[k+1]}) - f(x^*) \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^2 (f(x^{[k]} - f(x^*)),$$

  wobei $\kappa := \lambda_{max}/\lambda_{min}$, der Quotient des größten durch den kleinsten Eigenwert von $H_f(x^*)$.
- Die **Backpropagation** im maschinellen Lernen basiert auf diesem Verfahren.

Allgemein: Konvergenzgeschwindigkeiten von Folgen

Definition (Def. 3.4.5 - (Super-)Lineare und quadratische Konvergenz)

Eine Folge $\{x^{[i]}\}_{i\in\mathbb{N}}$ **konvergiert** (mindestens)

(a) **linear** gegen $\hat{x}$, falls es ein $c \in (0,1)$ gibt, so dass

$$\|x^{[i+1]} - \hat{x}\| \leq c\,\|x^{[i]} - \hat{x}\|$$

für alle hinreichend großen i,

(b) **superlinear** gegen $\hat{x}$, falls es eine Nullfolge $\{c_i\}_{i\in\mathbb{N}}$, $c_i \downarrow 0$ gibt, so dass

$$\|x^{[i+1]} - \hat{x}\| \leq c_i\,\|x^{[i]} - \hat{x}\| \quad \text{für alle } i \in \mathbb{N},$$

(c) **in Ordnung** $p \geq 2$ gegen $\hat{x}$, falls es ein $c > 0$ gibt, so dass

$$\|x^{[i+1]} - \hat{x}\| \leq c\,\|x^{[i]} - \hat{x}\|^p \quad \text{für alle} i \in \mathbb{N},$$

d.h.

$$\|x^{[i+1]} - \hat{x}\| = O(\|x^{[i+1]} - \hat{x}\|^p).$$

Im Fall $p = 2$ heißt die Folge quadratisch konvegent.

Newtonverfahren

Idee zur Bestimmung einer Suchrichtung:
Minimiere die lokale quadratische Approximation von f um $x^{[l]}$ anstatt f selber:

$$\hat{f}(y) = f(x) + \nabla f(x)^\top (y - x) + \frac{1}{2}(y - x)^\top H_f(x)(y - x)$$

Ist H_f pos. def., dann existiert genau ein Minimum $\hat{y}$ von $\hat{f}$:

$$\nabla \hat{f}(y) = \nabla f(x) + H_f(x)(y - x) \stackrel{!}{=} 0$$
$$\Leftrightarrow \quad d = \hat{y} - x = -H_f(x)^{-1} \nabla f(x)$$

Alternative Motivation:

Löse die notwendige Bedingung $\nabla f(\hat{x}) = 0$ als Nullstellenproblem.

Man überprüft, dass d eine Abstiegsrichtung ist.

Die Voraussetzung $H_f(x)$ pos. def. ist durchaus eine Einschränkung!

Lokales Newtonverfahren

Algorithmus (*Lokales Newtonverfahren*)

(i) *Wähle* $x^{[0]} \in \mathbb{R}^n$,
setze $i = 0$.

(ii) *Falls Abbruchkriterium erfüllt, dann STOP,*

(iii) *Berechne (falls möglich)* $d^{[i]}$ *als Lösung des LGS*

$$H_f(x^{[i]})d^{[i]} = -\nabla f(x^{[i]}).$$

Setze $x^{[i+1]} := x^{[i]} + d^{[i]}$,
$i := i + 1$ *und*
gehe zu (ii).

Das lokale Newtonverfahren lässt sich
als Nullstellenverfahren zur Bestimmung eines stationären Punktes oder
als Abstiegsverfahren (wenn H_f pos. def.) mit Schrittweite $\alpha_k = 1$ interpretieren.

Exkurs: Lösung nichtlinearer Gleichungen

Nichtlineare Gleichungen treten auf bei:

- Impliziten Verfahren: Z.B. Auflösen der Vorschrift nach x_{k+1}
- Bestimmung stationärer Punkte nichtlinearer Differentialgleichungen
- Numerischen Verfahren für nichtlineare Differentialgleichungen
- und in vielen anderen Anwendungen ...

Allgemein: Sei $g \in C^1(\mathbb{R})$. Bestimme konstruktiv $\hat{x}$ so dass

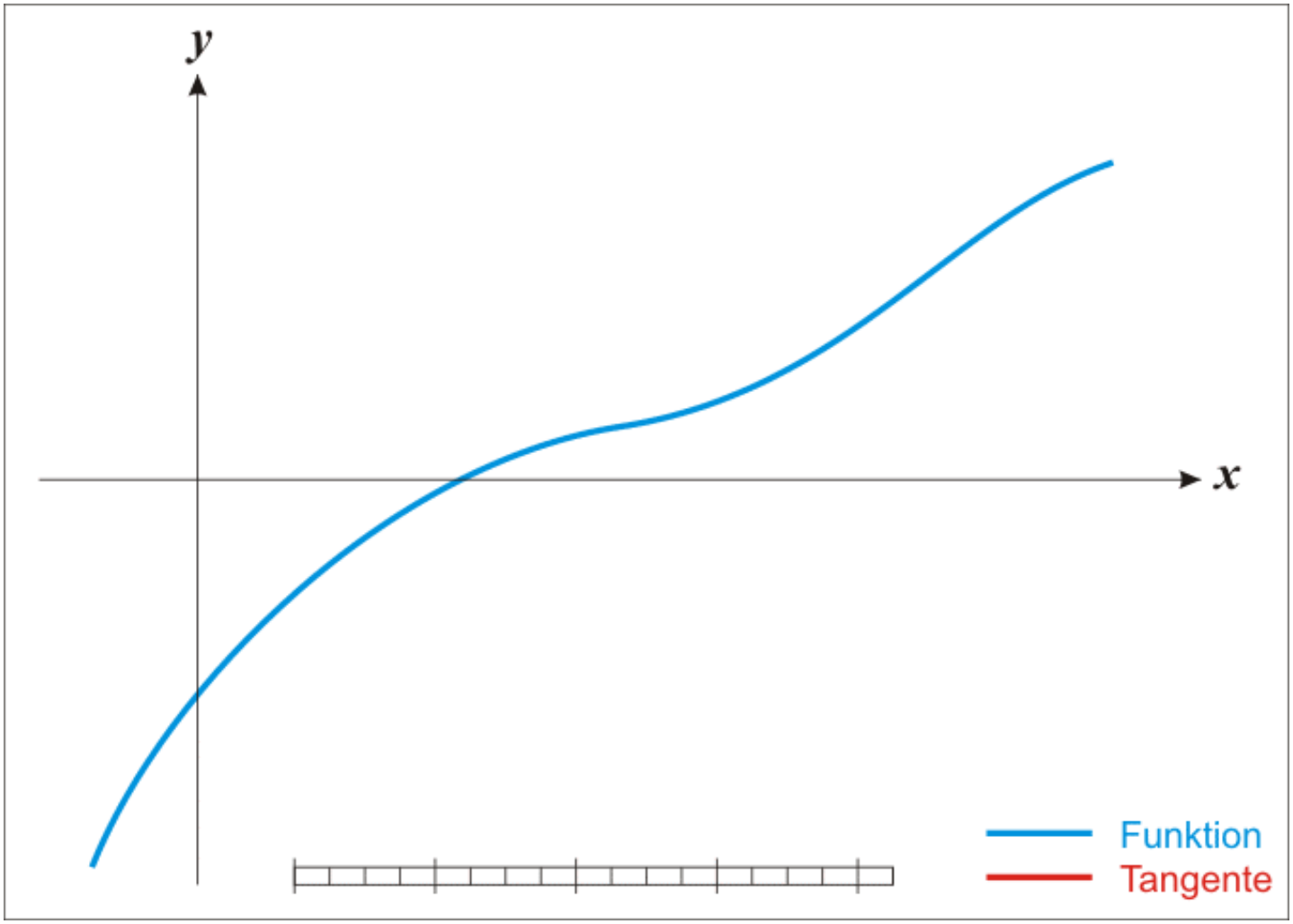
$$g(\hat{x}) = 0.$$

Newtonverfahren: (diese Folie und Animation $x_n := x^{[n]}$)

$$x_{n+1} = x_n - \frac{g(x_n)}{g'(x_n)}, \quad x_0 \in \mathbb{R} \text{ gegebener Startwert}$$

Exkurs: Newtonverfahren in 1D

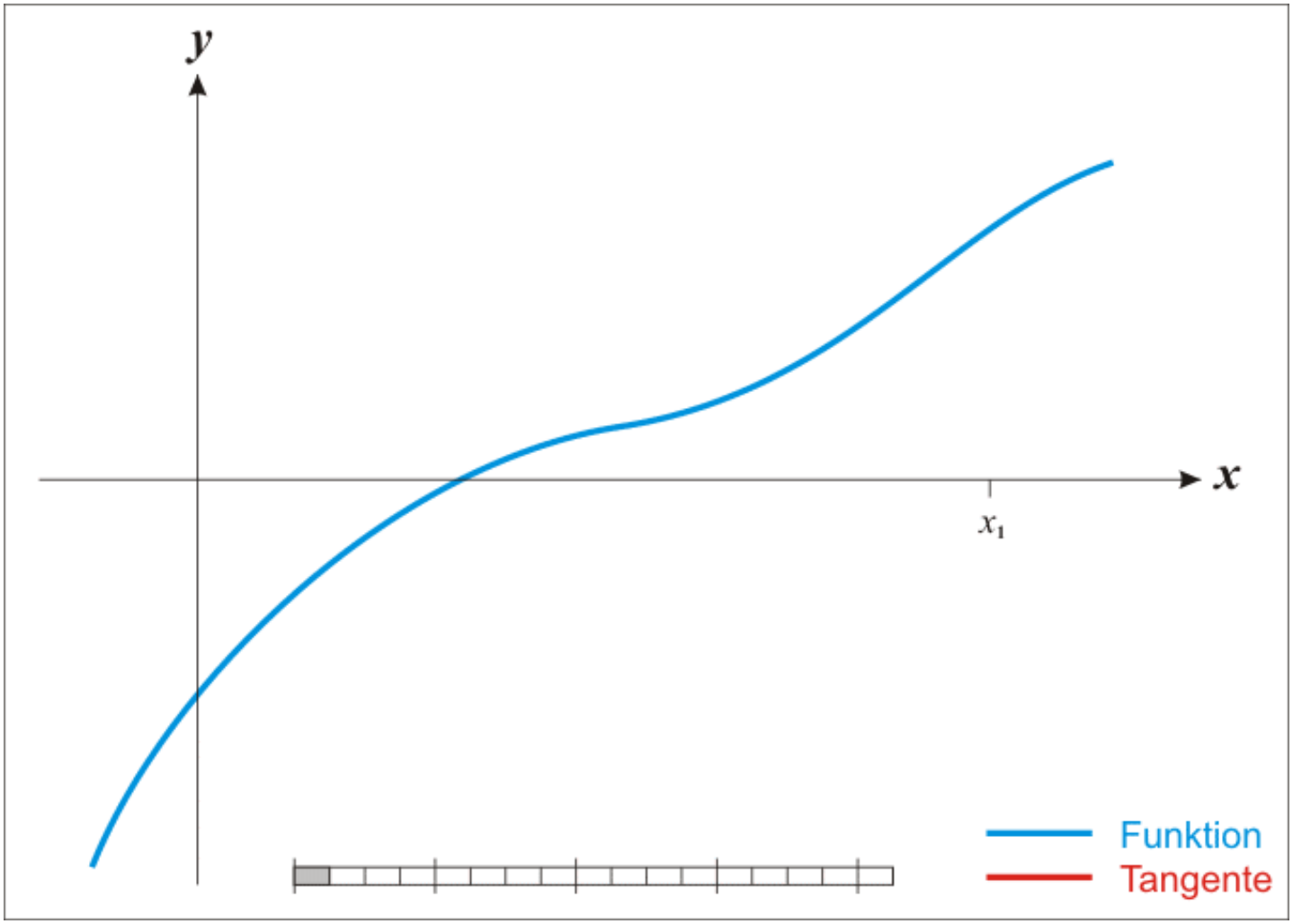
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

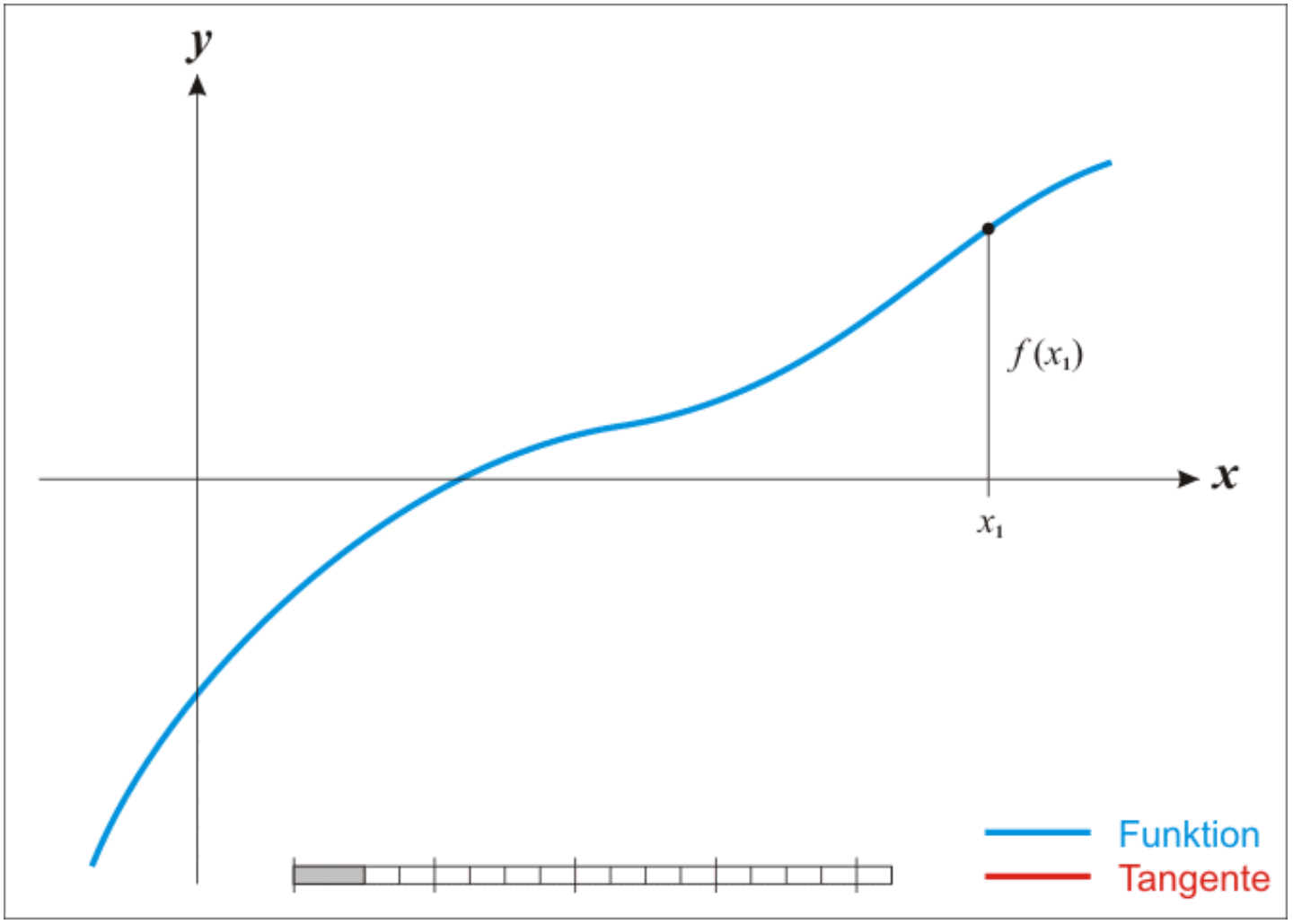
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

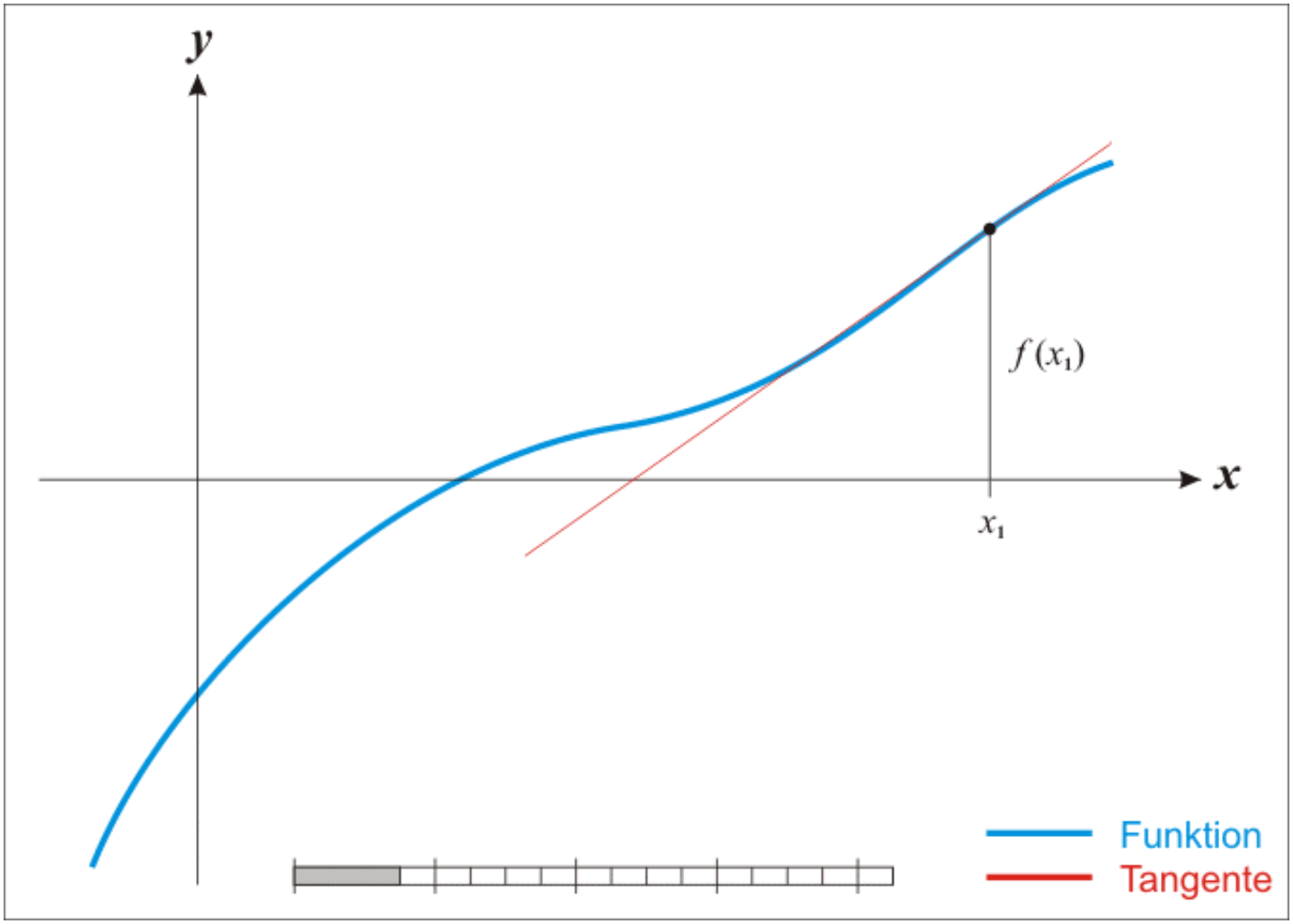
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

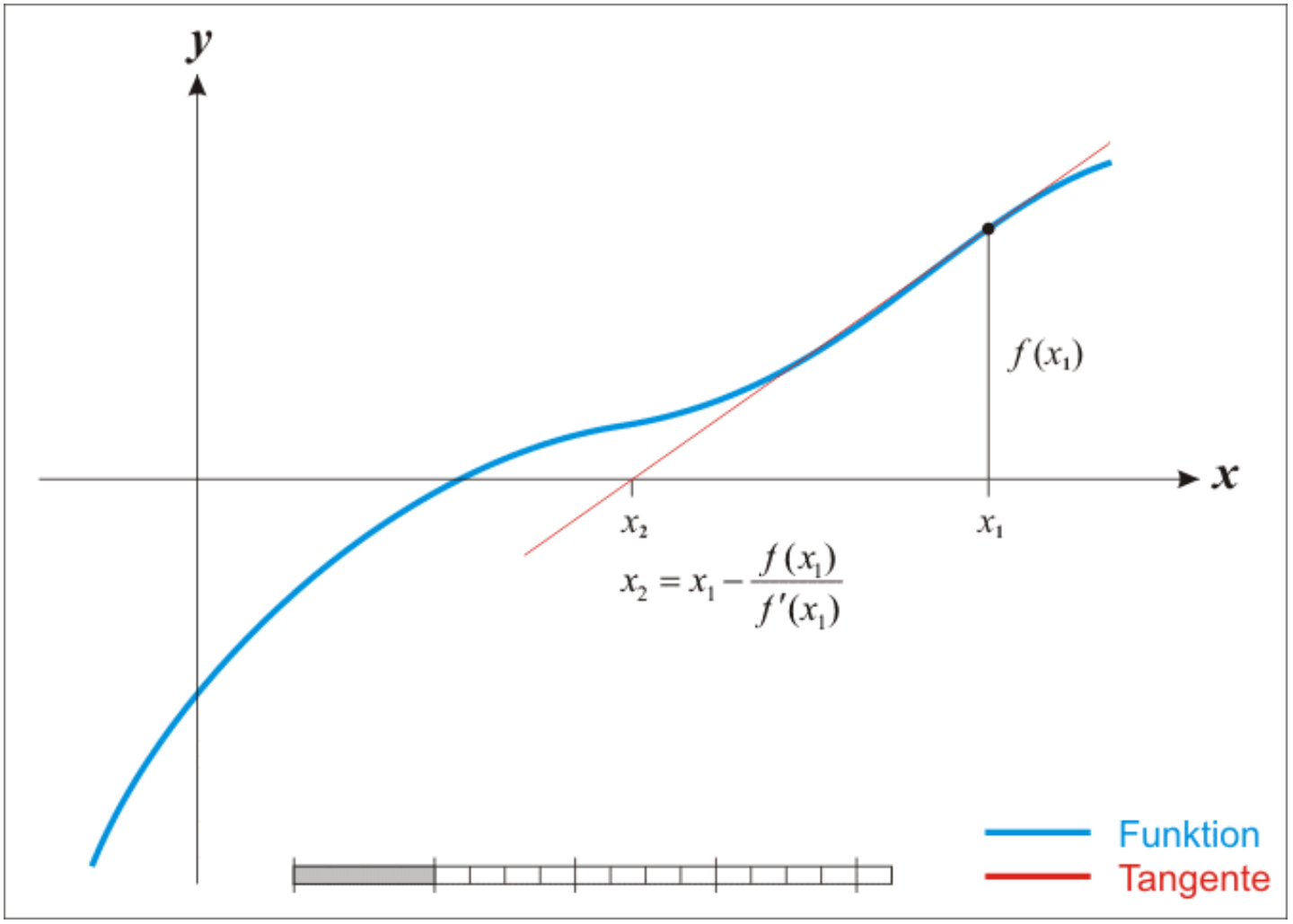
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

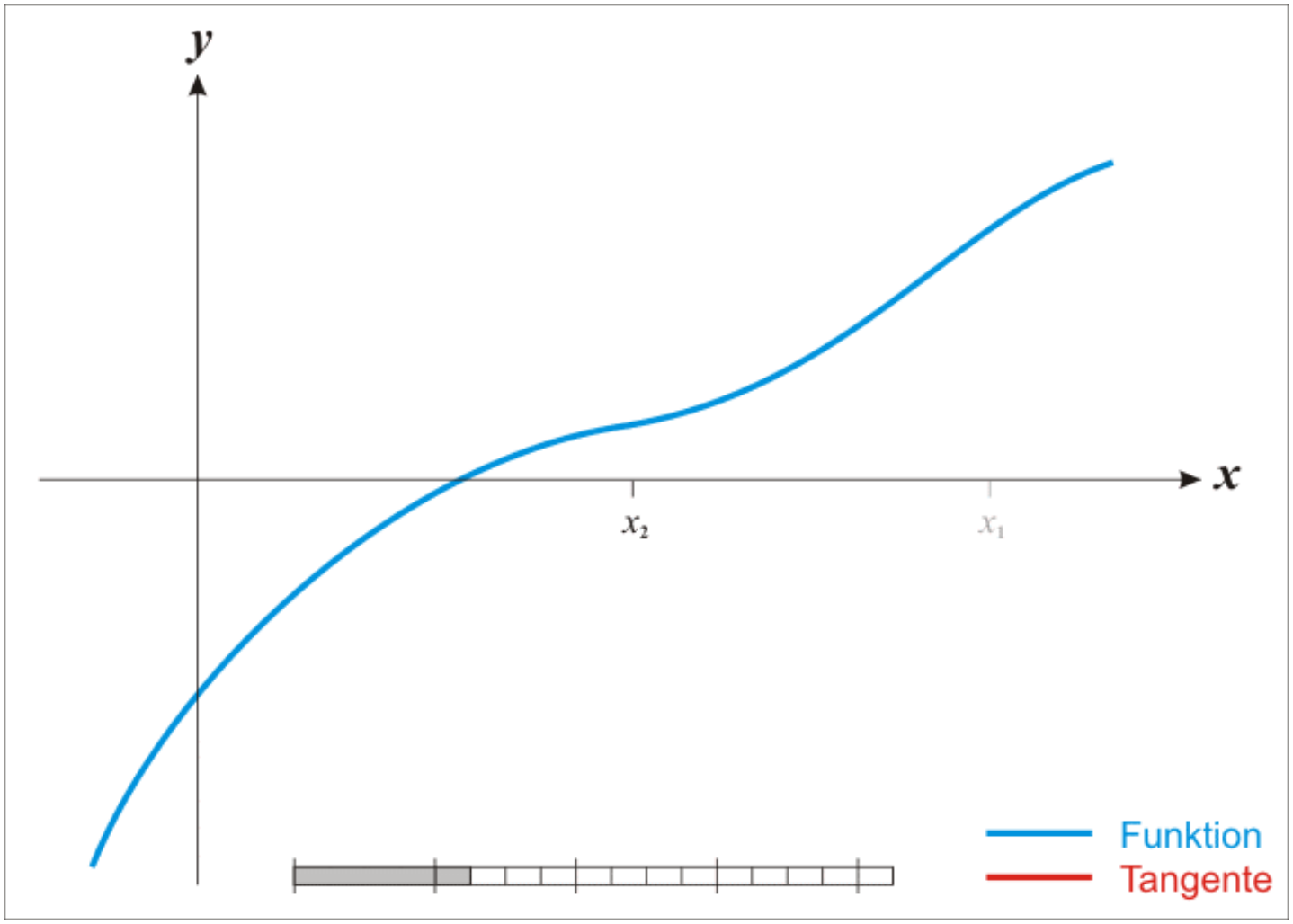
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

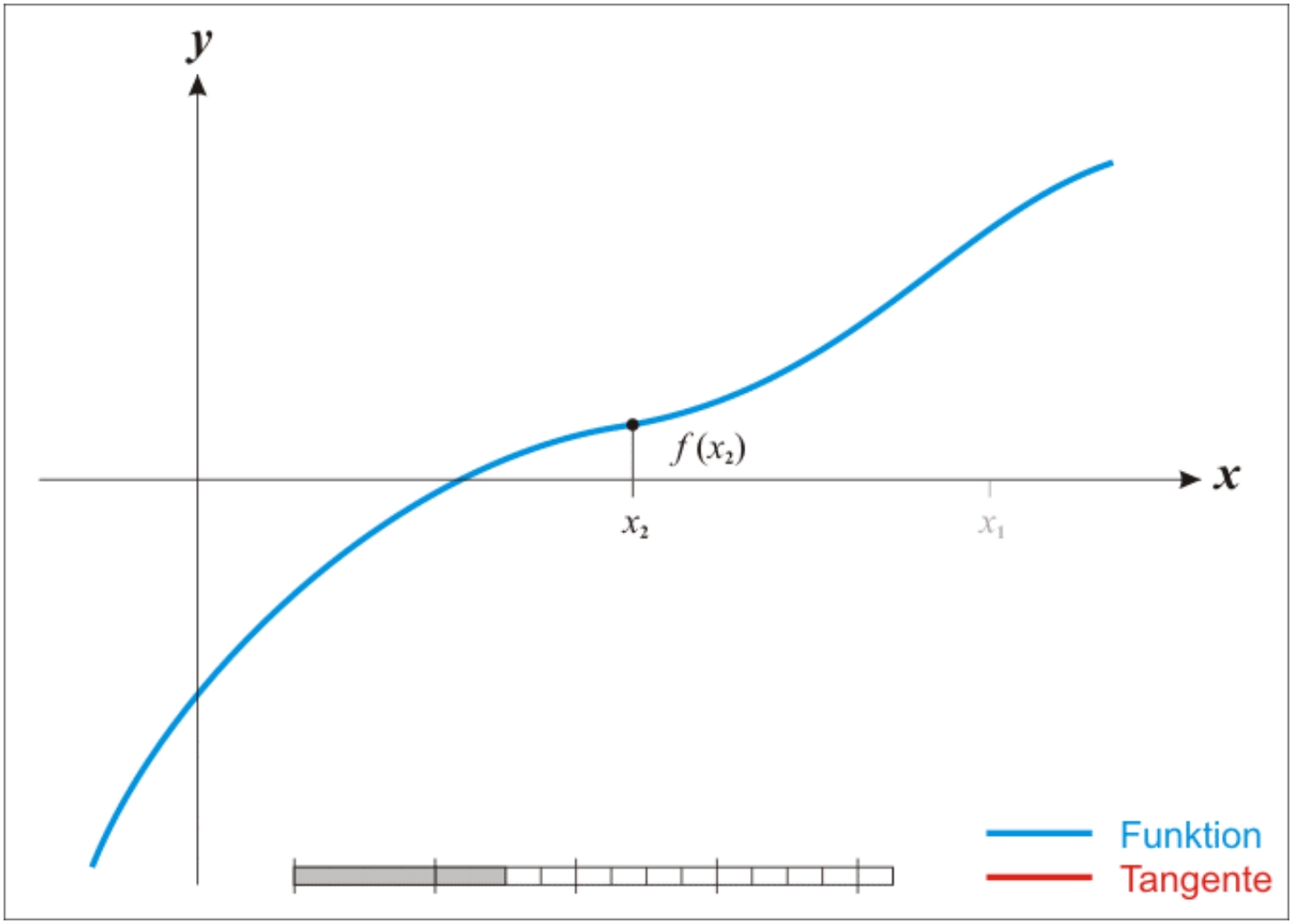
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

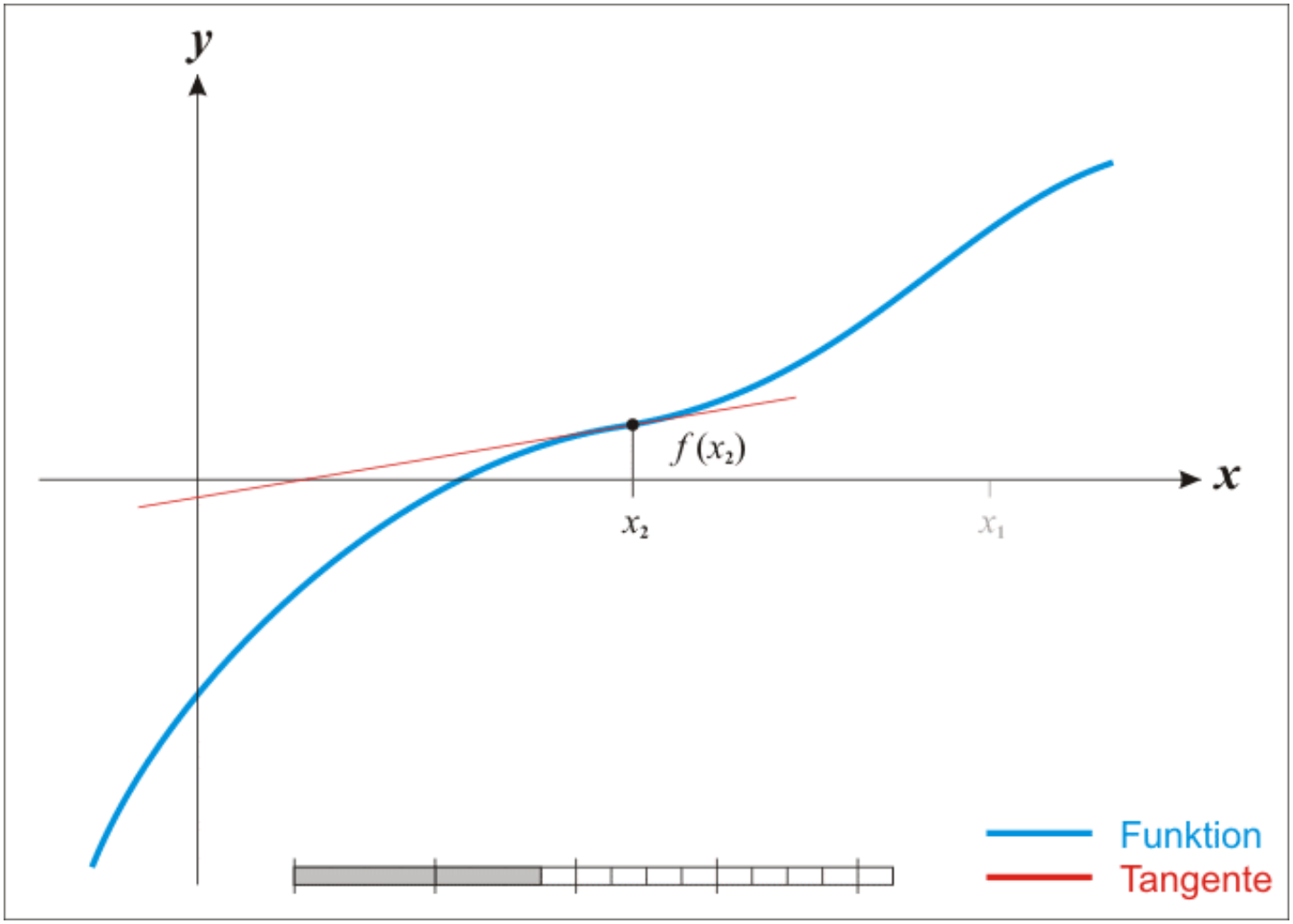
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

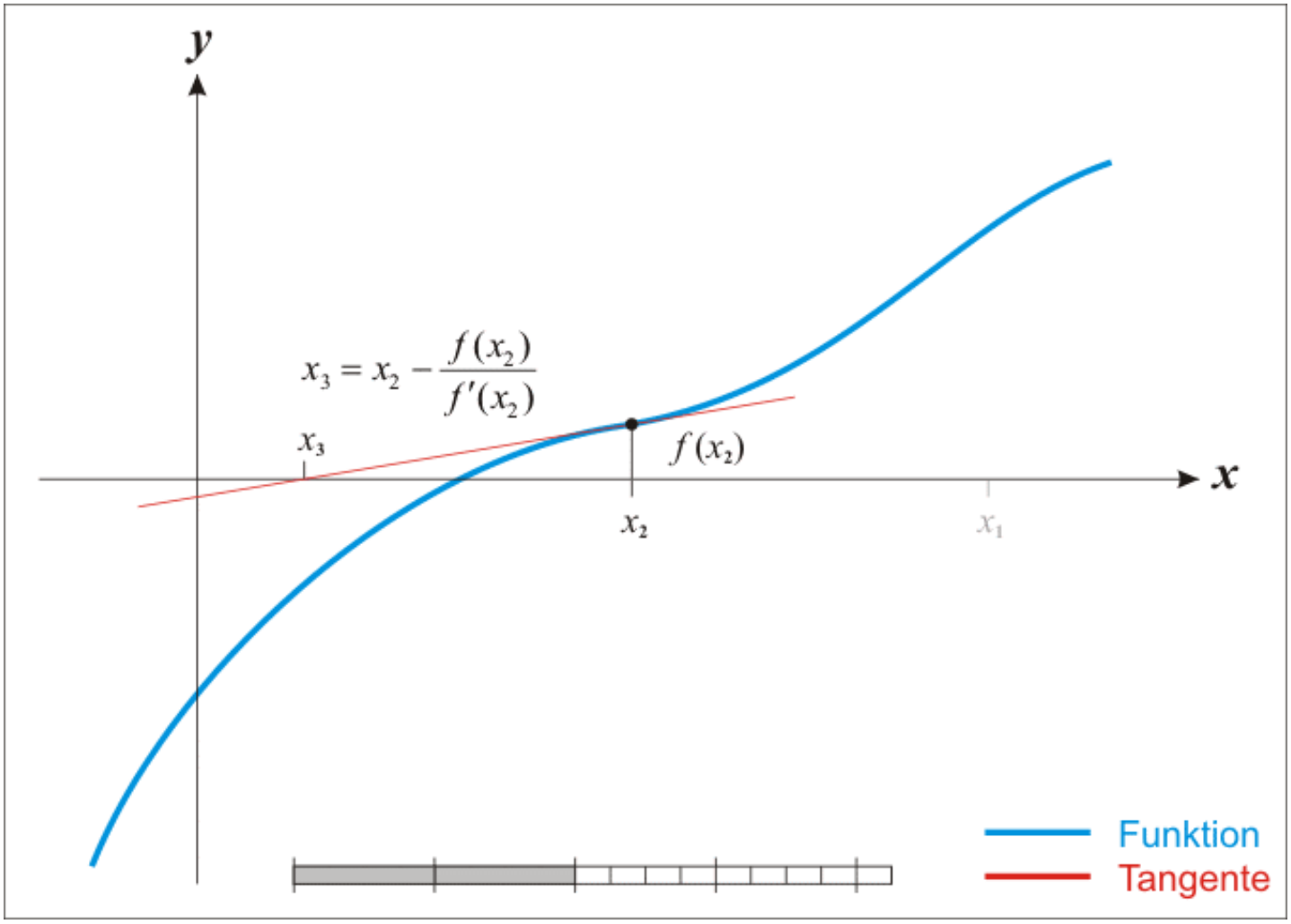
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

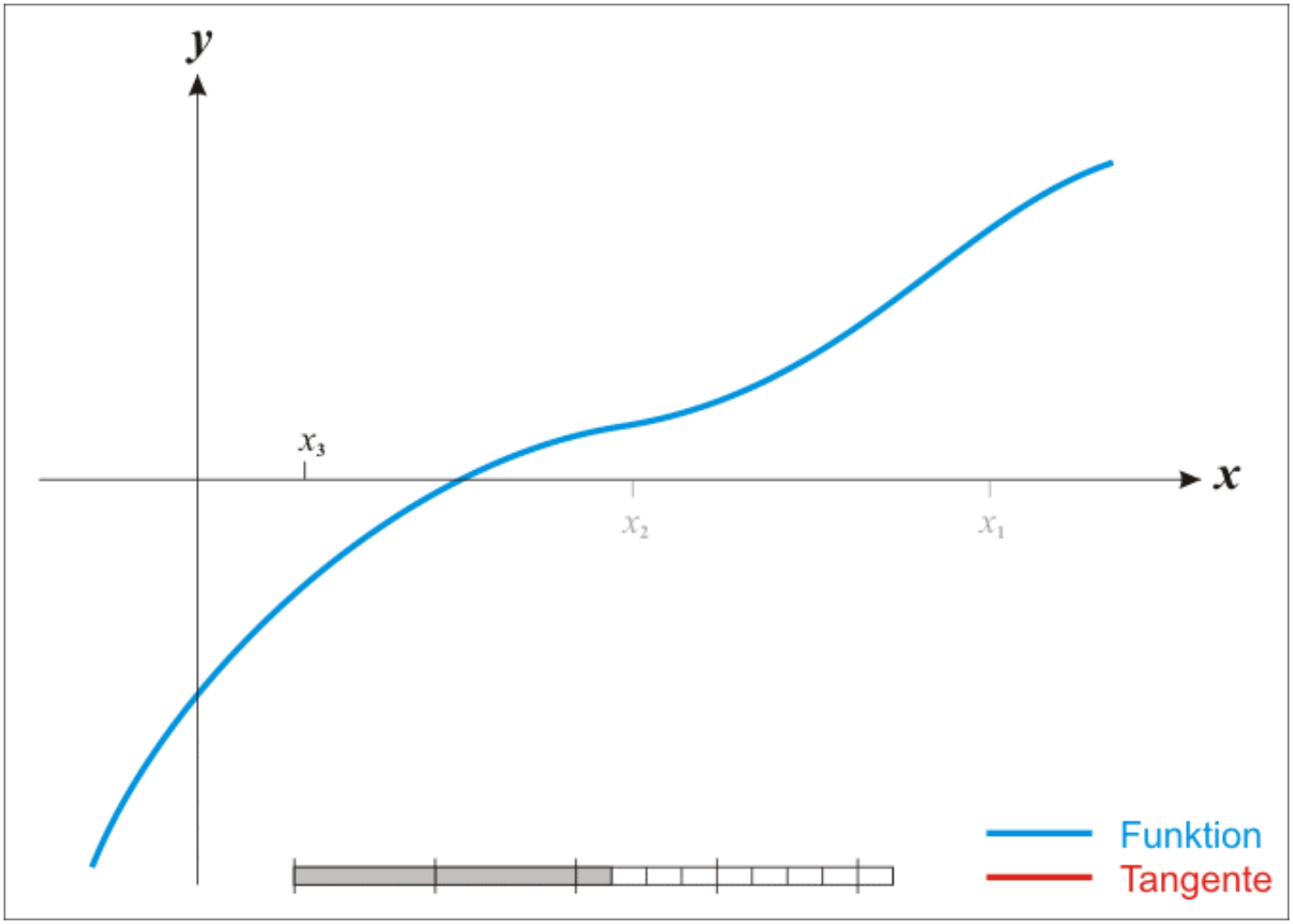
Als Beispiel betrachten wir folgende Funktion f:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

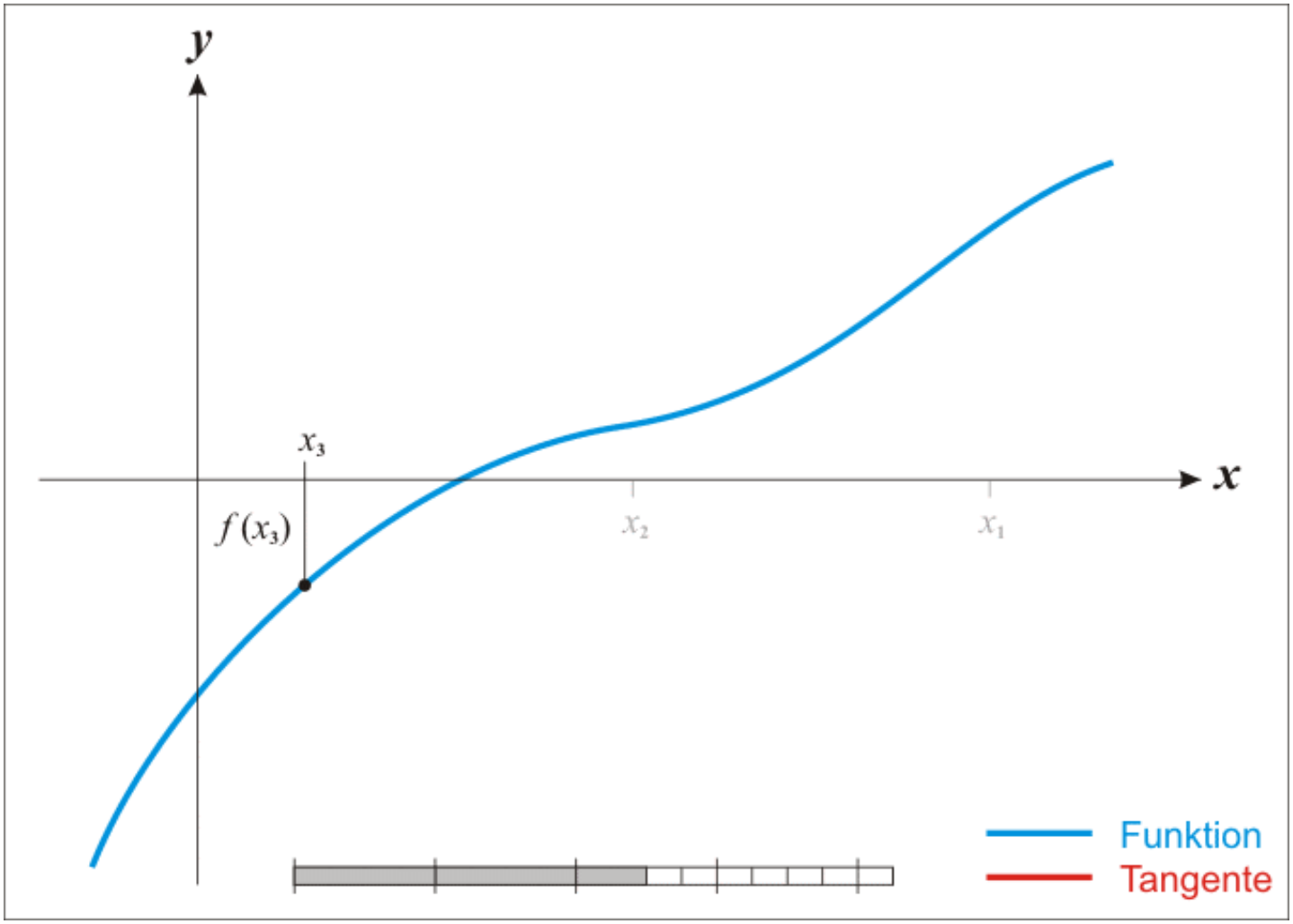
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

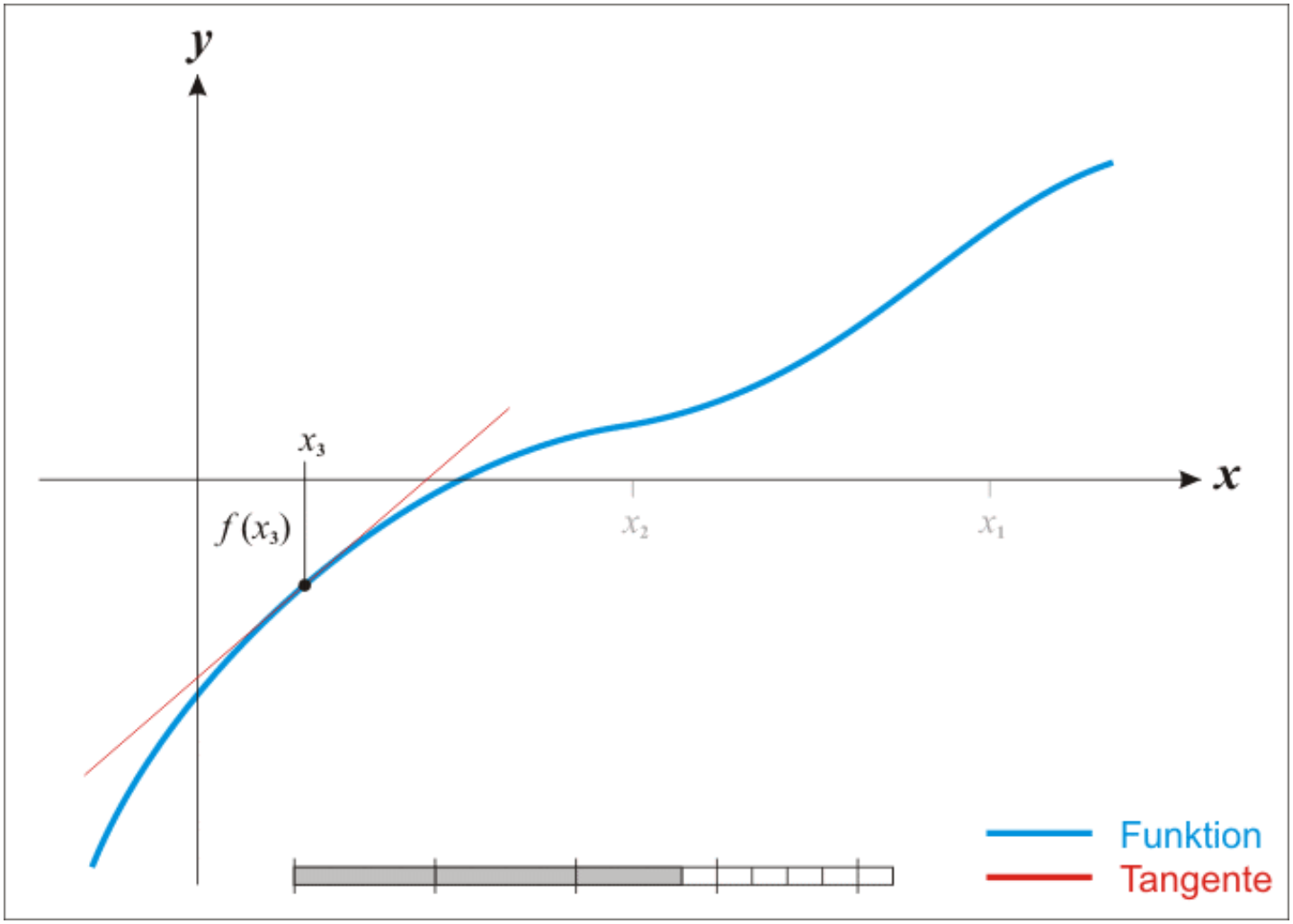
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

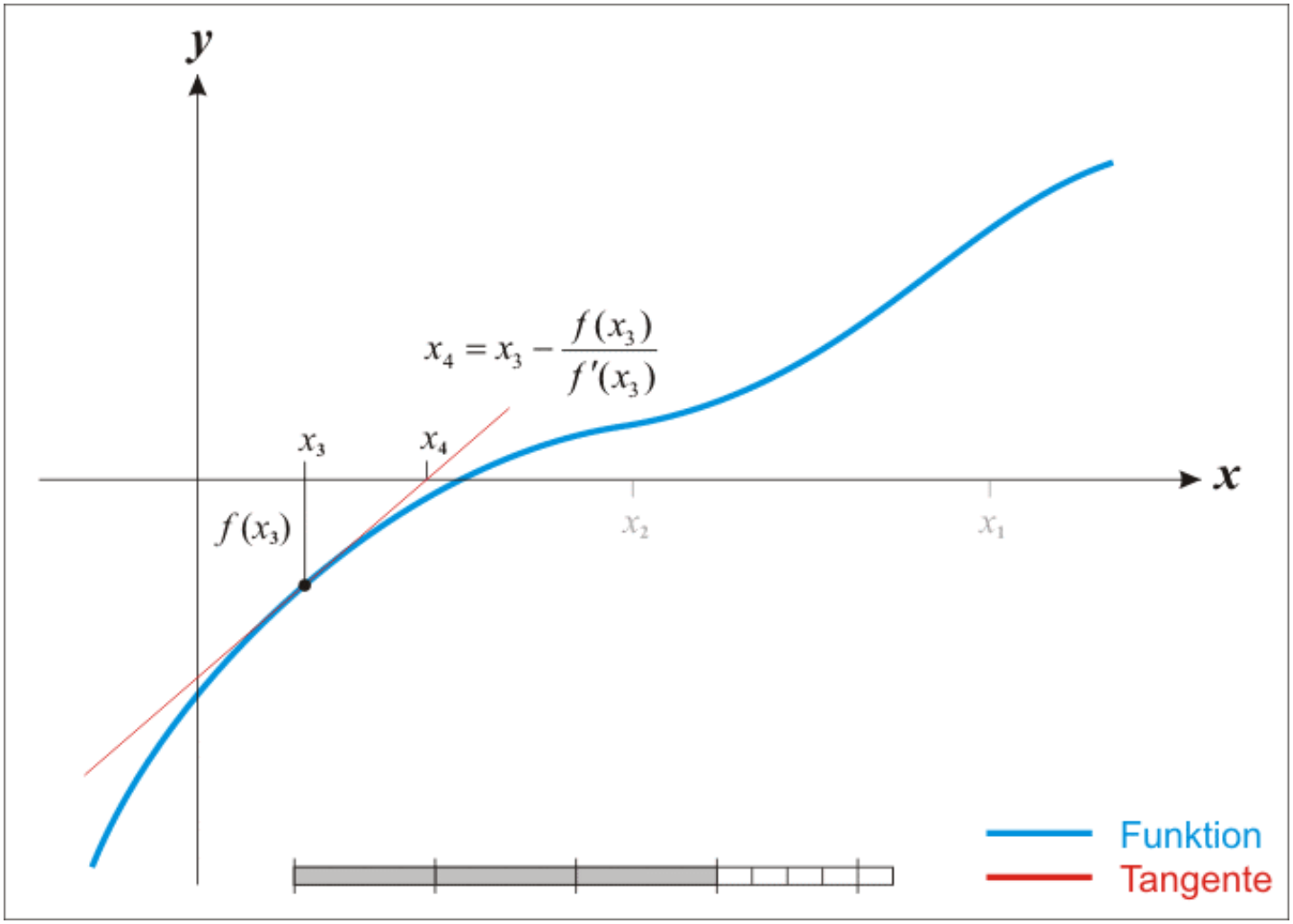
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

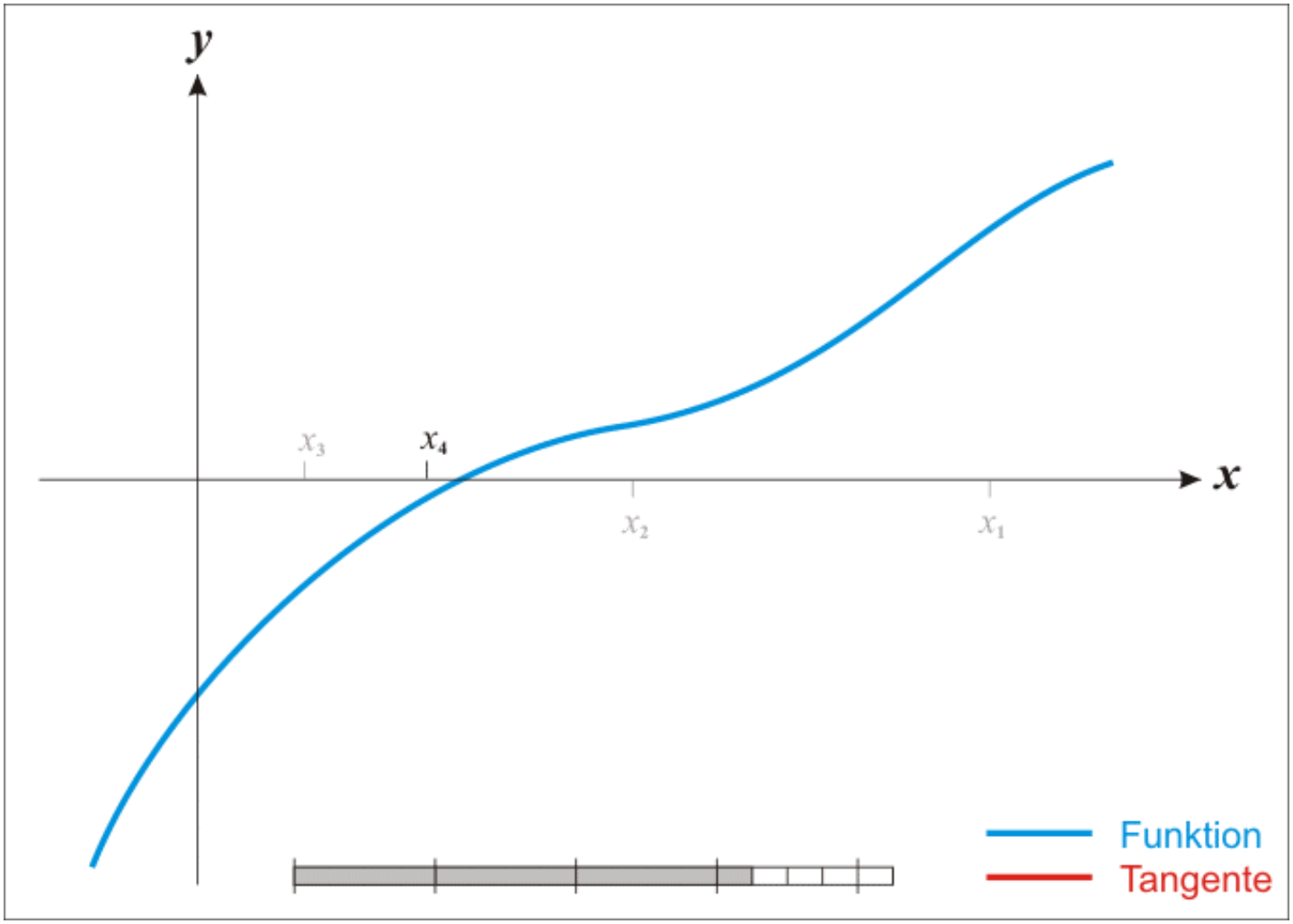
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

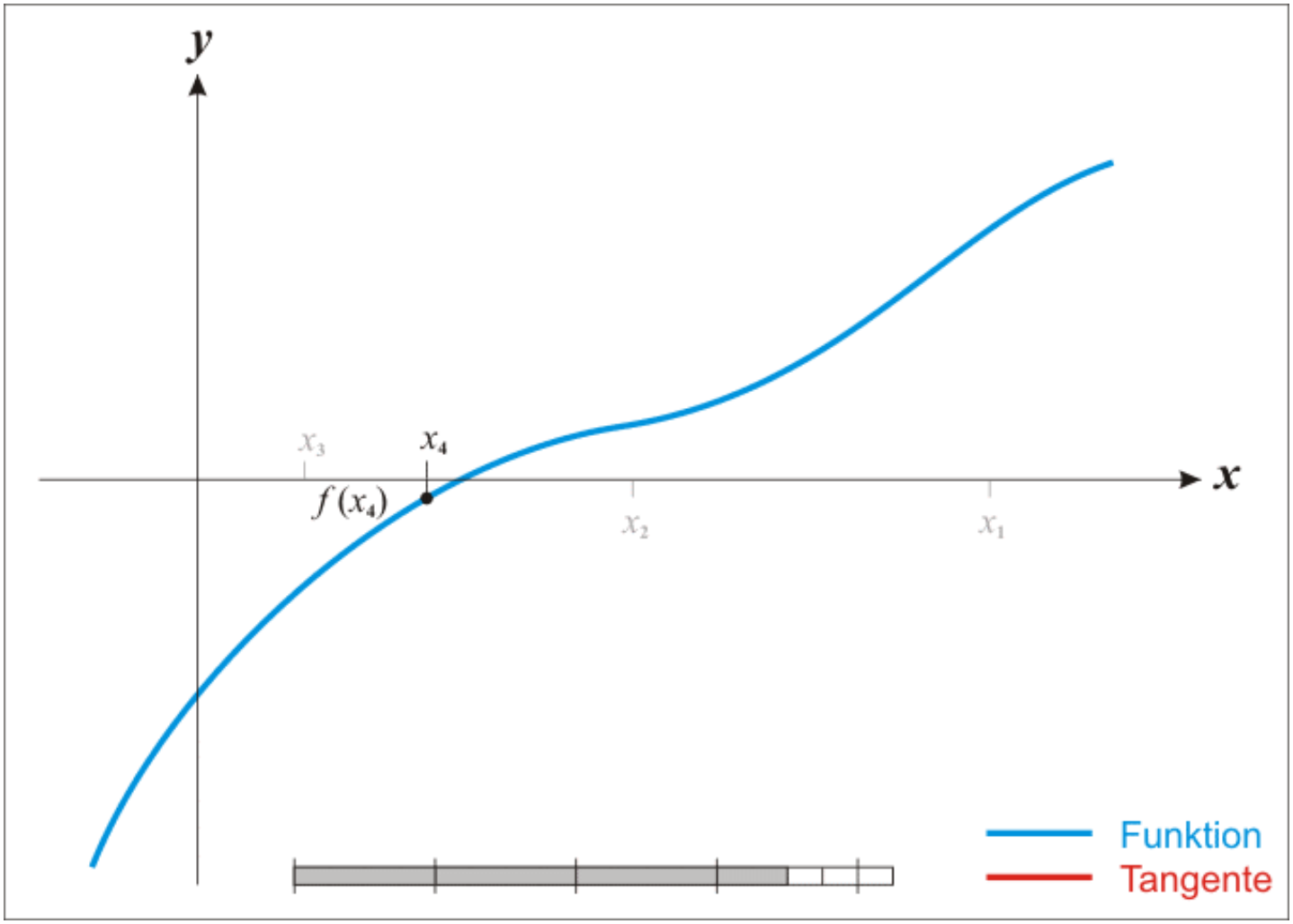
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

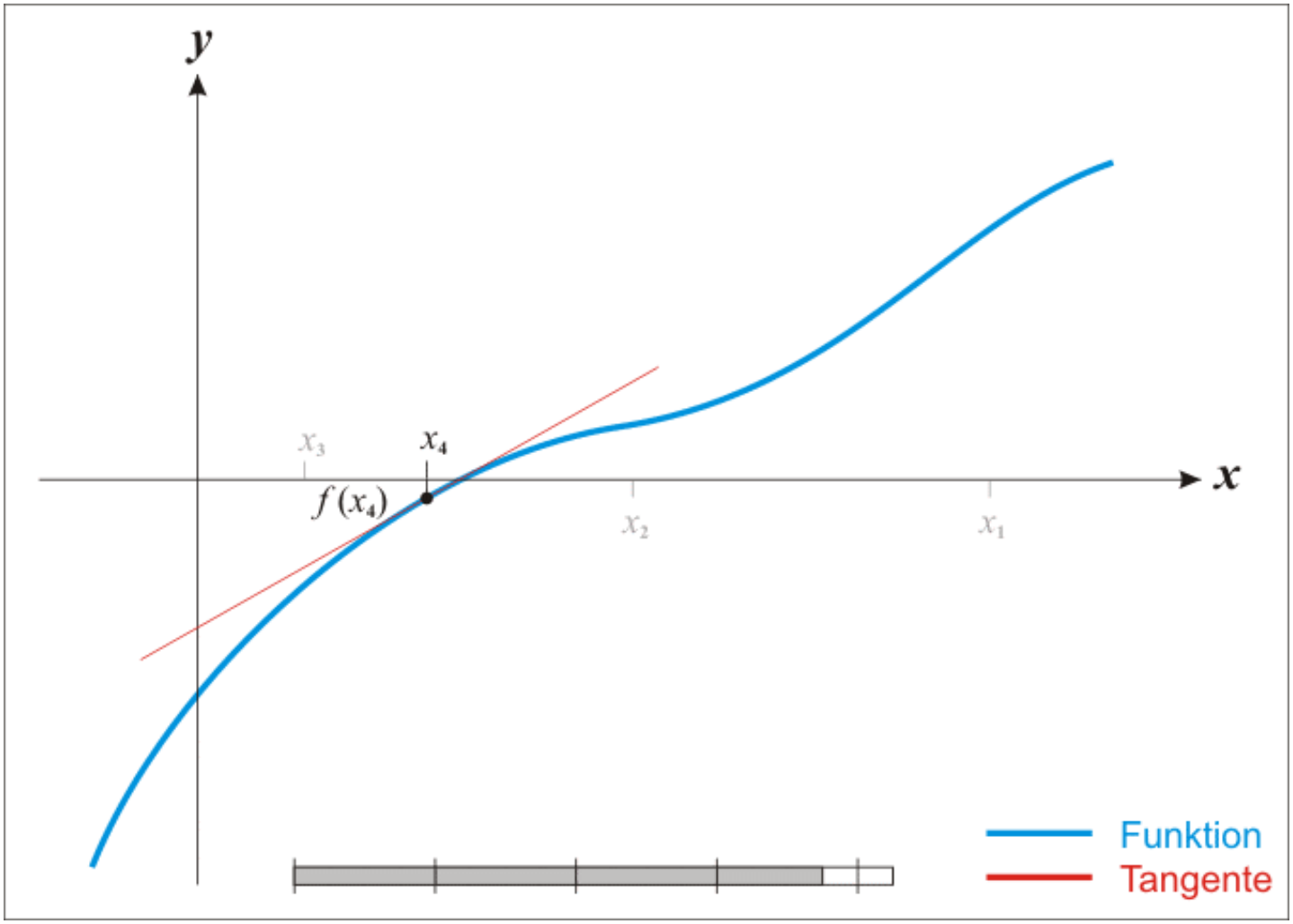
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

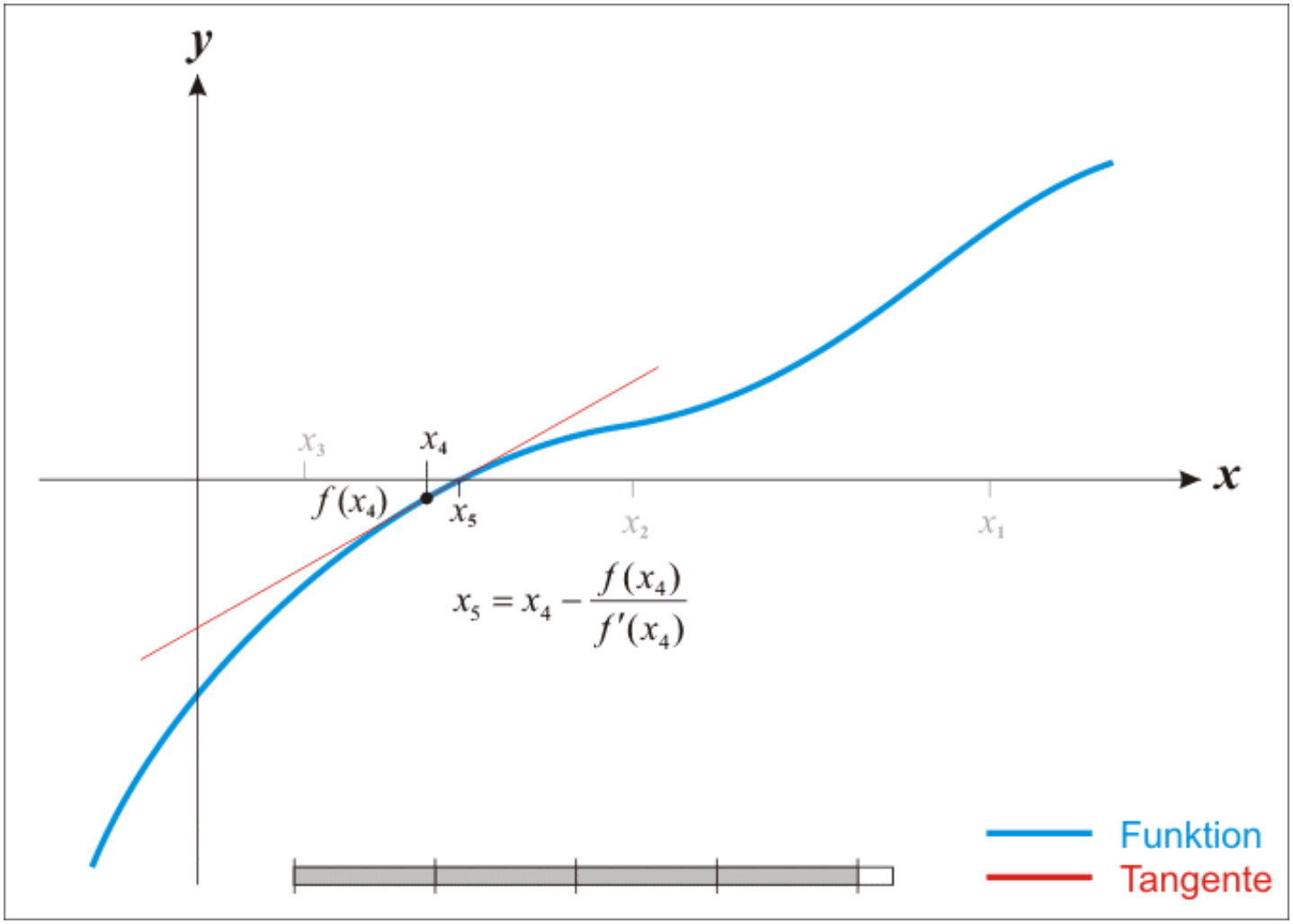
Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

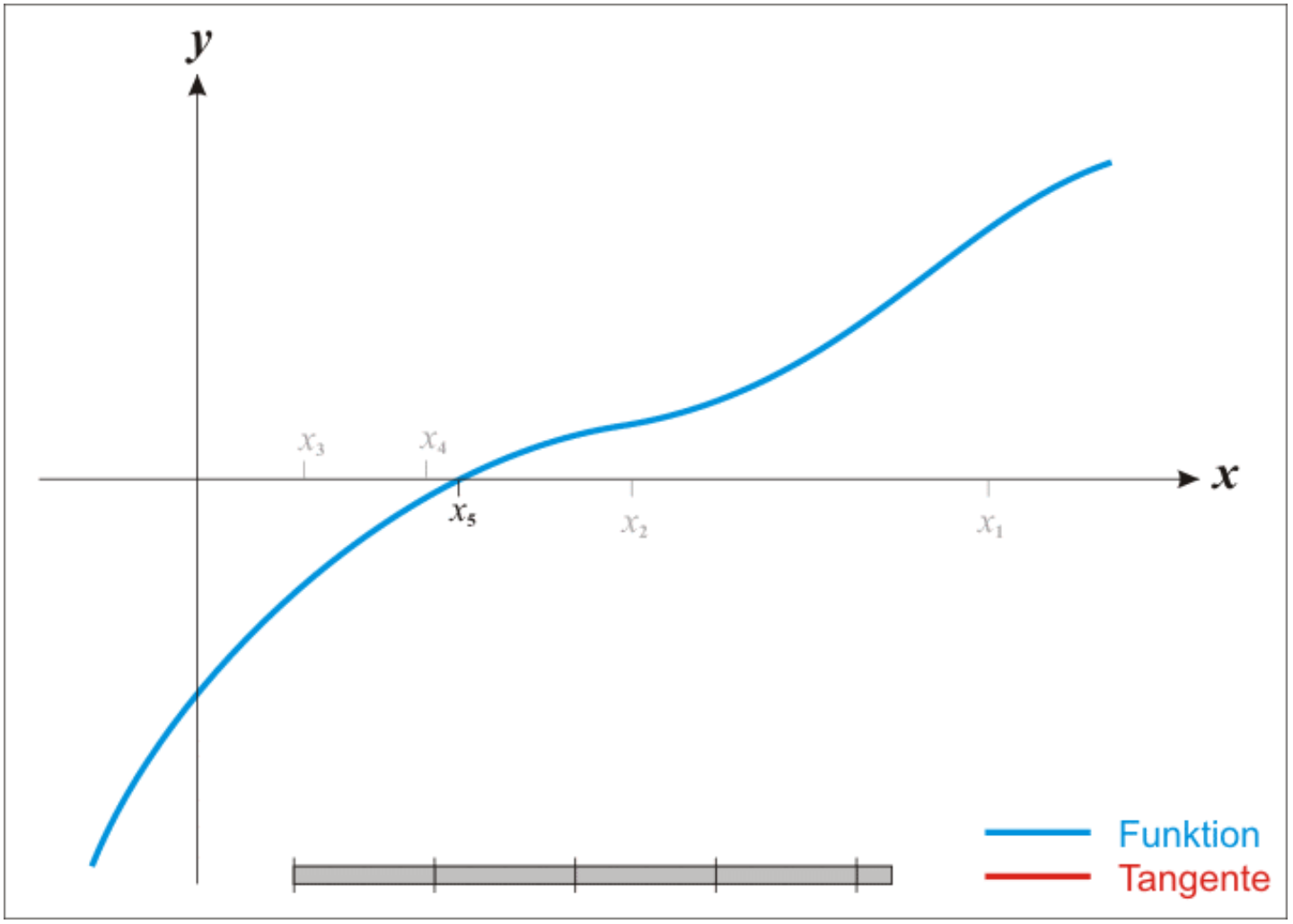
Als Beispiel betrachten wir folgende Funktion f:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Exkurs: Newtonverfahren in 1D

Als Beispiel betrachten wir folgende Funktion *f*:



Siehe https://de.wikipedia.org/wiki/Newton-Verfahren.

Konvergenz des Newtonverfahrens

Theorem (Satz 3.4.6 - Konvergenzsatz für Newtonverfahren)

Besitze f ein (lokales) Minimum $\hat{x} \in \mathbb{R}^n$.
Es gibt ein $r > 0$, so dass, wenn durch das lokale Newtonverfahren für $f \in C^2(\mathbb{R}^n)$, $x^{[0]} \in U_r\mathbb{R}^n$ eine nicht abbrechende Folge $\{x^{[k]}\}_{k\in\mathbb{N}}$ definiert wird,
diese Folge superlinear konvergiert gegen $\hat{x}$.
$\hat{x}$ ist das einzige lokale Minimum in $U_r(\hat{x})$.

Erfüllt H_f zusätzlich die Lipschitz-Bedingung

$$\|H_f(x) - H_f(y)\| \leq L\|x - y\| \quad \text{für alle } x, y \in \mathbb{R}^n,$$

so konvergiert diese Folge sogar quadratisch.

Vor-/Nachteile:

- Komplizierter zu implementieren (Lösbarkeit des LGS ?)
- Superlineare Konvergenz, oft sogar **quadratische Konvergenz**

Varianten des Newtonverfahrens I

Beim lokalen Newtonverfahren muss in jedem Schritt $H_f(x^{[i]})$ berechnet werden, was oft aufwendig (oder sogar unmöglich) sein kann

Idee: Hessematrix $H_f(x^{[i]})$ wird durch geeignete Matrix $A(x^{[i]})$ ersetzt, die in jeder Iteration durch "einfache" Rechenschritte bestimmt wird

- Vereinfachtes Newtonverfahren:

  $A = A(x^{[i]}) := H_f(x^{[0]})$ wird einmalig berechnet,
  wird einmal faktorisiert zur schnellen Lösung des Newton-LGS

- Finite-Differenzen-Approximation:

  Setze in Iteration i

$$A(x^{[i]})_{jk} = \frac{\nabla f_j(x + h^{[i]} e_k) - \nabla f_j(x)}{h^{[i]}}, \quad j,k = 1,\ldots,n,$$

  wobei $h^{[i]} \to 0$ für $i \to \infty$

Varianten des Newtonverfahrens II

- **Quasi-Newtonverfahren**:

  Starte mit symm., pos. def. Matrix $A^{[0]}$
  Berechne $A^{[i+1]}$ rekursiv aus $A^{[i]}$ durch sogenannte Update-Formeln

- Inverse Quasi-Newtonverfahren:

  Wie Quasi-Newtonverfahren, man approximiert aber $H_f^{-1}(x^{[i]})$

- Variable-Metrik-Verfahren ...

Newtonverfahren, die überall konvergieren:

- Globalisiertes Newtonverfahren:

  Was macht man, wenn Newton-LGS nicht lösbar?

Globalisiertes (oder globales) Newtonverfahren

Algorithmus (*Alg. 1.3.1 - Global. Newtonverfahren [GK99, S. 92]*)

(i) *Wähle* $x^{[0]} \in \mathbb{R}^n$, $\sigma \in (0, 1/2)$, $\beta \in (0,1)$, $\rho > 0$, $p > 2$, *setze* $i = 0$.

(ii) *Falls Abbruchkriterium erfüllt, dann STOP,*

(iii) *Berechne (falls möglich)* $d^{[i]}$ *als Lösung des LGS*

$$H_f(x^{[i]})d^{[i]} = -\nabla f(x^{[i]}).$$

Falls dieses LGS unlösbar oder $\nabla f(x^{[i]})^\top d^{[i]} > -\rho \|d^{[i]}\|^p$, *setze* $d^{[i]} = -\nabla f(x^{[i]})$.

(iv) *Bestimme Schrittweite* $\alpha_i > 0$ *mit Armijo-Verfahren.*

(v) *Setze* $x^{[i+1]} := x^{[i]} + \alpha_i d^{[i]}$, $i := i + 1$ *und gehe zu (ii).*

Geht nahe Lösung in lokales Newtonverfahren über, d.h. irgendwann immer $\alpha_i = 1$.

Im Endlichdimensionalen kann man globale, superlineare Konvergenz gegen einen stationären Punkt beweisen.

Es gibt alternative Globalisierungen (z.B. mit Bewertungsfunktion)

Inhaltsverzeichnis

Restringierte Optimierung

In diesem Abschnitt betrachten wir die schon eingeführte Problemklasse:

Problem (Standard-Optimierungsproblem (SOP))

Minimiere $f(x)$ *u. d. N.*

$$\begin{aligned} g_i(x) &\leq 0, \quad i = 1, \ldots, m, \\ h_j(x) &= 0, \quad j = 1, \ldots, p. \end{aligned}$$

bzw. in Vektornotation

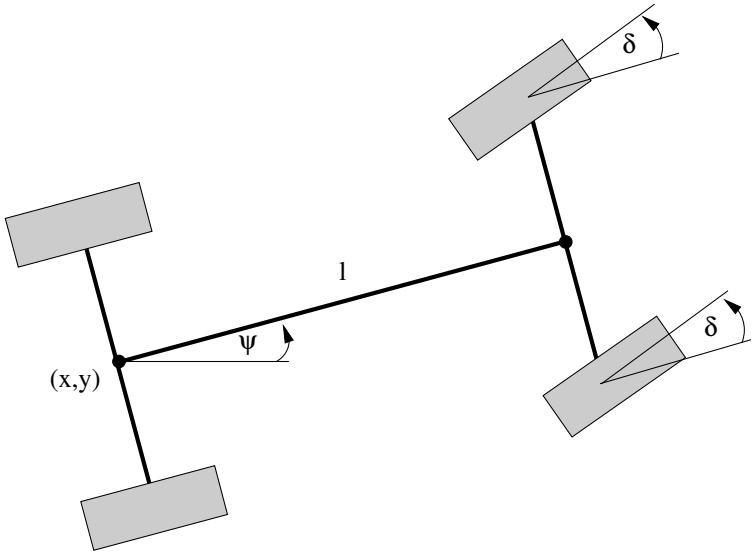
$$\begin{aligned} g(x) &\leq 0, \\ h(x) &= 0. \end{aligned}$$

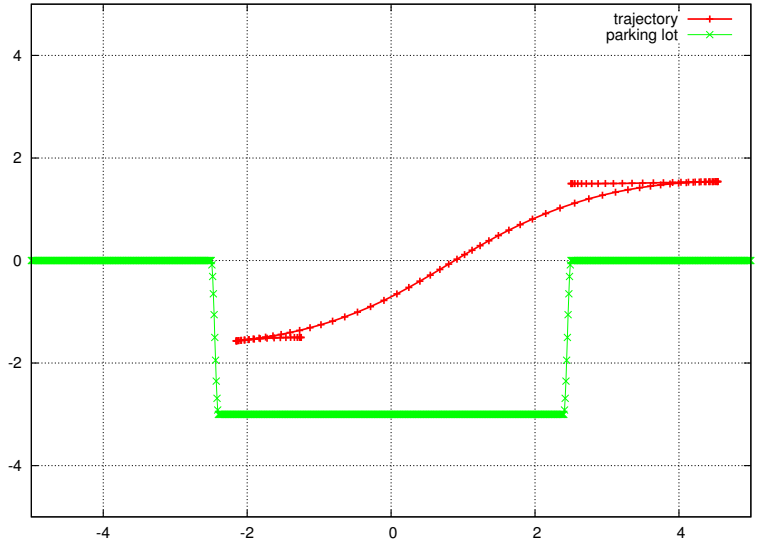
Dabei seien $x \in \mathbb{R}^n$ *und die Funktionen*

$$\begin{aligned} f &: \mathbb{R}^n \to \mathbb{R}, \\ g = (g_1, \ldots, g_m)^\top &: \mathbb{R}^n \to \mathbb{R}^m \text{ und} \\ h = (h_1, \ldots, h_p)^\top &: \mathbb{R}^n \to \mathbb{R}^p. \end{aligned}$$

Optimal control of ODE

Example: Optimal driving, e.g. a parking process:





$f(y, u, t_f) = t_f$ (final time)

Inequality constraints: Control and state constraints

Equality constraints: ODEs for states y, with initial and final conditions

Here states y and control u are time-dependent functions

Verfahren der restringierten Optimierung

- Eliminierung der Ungleichungsrestriktionen:
  - **Penalty- und Multiplikator-Verfahren**:
    Ankopplung der NB. als geeignet gewichteter **Penalty-Term** an die Zielfunktion
    dann wiederum Verfahren der unrestringierten Optimierung
    Bsp. Lagrange-Newton-Verfahren, Penalty-Verfahren, Multiplikator-Penalty-Verfahren
  - **Innere Punkte-Verfahren**:
    Mithilfe von **Barrierefunktionen** wird die Annäherung an den Rand ("die Barriere") des zulässigen Bereichs bestraft.
    Daher können Ungleichungsnebenbedingungen eliminiert werden.

- **Sequentielle quadratische Programmierung (SQP)**:

  (SOP) wird lokal approximiert durch (LQOP), das Suchrichtung liefert
  Erweiterung Lagrange-Newton-Verfahren auf Probleme mit Ungleichungs-NB.

- **Verfahren für Komplementaritätsprobleme**:

  Z.B. semiglatte Newtonverfahren, Variationsmethoden
  Man versucht die notwendigen Bedingungen umzuschreiben und direkt zu lösen

Globalisierungsstrategien in der restringierten Optimierung

- **Liniensuche**

  Z.B. Armijo-Verfahren

- **Trust-Region-Verfahren**

  Approximation auf einem Vertrauensbereich

- **Filterverfahren**

  Versuche nicht-dominierte Iterierte zu erzeugen, wobei Zielfunktionswert und Verletzung der Restriktionen betrachtet werden

Globalisierungsstrategien in der restringierten Optimierung

- **Liniensuche**

  Z.B. Armijo-Verfahren

- **Trust-Region-Verfahren**

  Approximation auf einem Vertrauensbereich

- **Filterverfahren**

  Versuche nicht-dominierte Iterierte zu erzeugen, wobei Zielfunktionswert und Verletzung der Restriktionen betrachtet werden

Wie in der unrestringierten Optimierung suche notwendige & hinreichende Optimaltätsbedingungen

Benötigte Begriffe: Lagrange-Funktion, (in)aktive Mengen

Definition (Lagrange-Funktion)

$$L(x, \lambda, \mu) := f(x) + \sum_{i=1}^{m} \lambda_i \, g_i(x) + \sum_{j=1}^{p} \mu_j \, h_j(x)$$

heißt **Lagrange-Funktion** für das Problem (SOP).

$\lambda = (\lambda_1, \ldots, \lambda_m)^\top \in \mathbb{R}^m$ und $\mu = (\mu_1, \ldots, \mu_p)^\top \in \mathbb{R}^p$ heißen **(Lagrangesche) Multiplikatoren**.

Definition (Def. 5.2.1 - (in)aktive Ungleichungsrestriktionen)

Sei x zulässig für (SOP).

Die Beschränkung $g_i(x) \leq 0$ heißt
aktiv in x, wenn $g_i(x) = 0$,
inaktiv in x, wenn $g_i(x) < 0$.

$$\mathcal{A}(x) := \{i \mid g_i(x) = 0,\ 1 \leq i \leq m\}$$

heißt **Indexmenge der in x aktiven Ungleichungsrestriktionen.**

Notwendige Bedingungen 1. Ordnung

Theorem (Satz 5.2.2 - Karush-Kuhn-Tucker-Bedingungen)

Seien f, g_i, $i = 1, \ldots, m$, und h_j, $j = 1, \ldots, p$, stetig differenzierbar und $\hat{x}$ ein lokales Minimum von (SOP).

Des Weiteren gilt die Linear Independence Constraint Qualification (LICQ) in $\hat{x}$, d.h. $\nabla g_i(\hat{x})$, $i \in \mathcal{A}(\hat{x})$, und $\nabla h_j(\hat{x})$, $j = 1, \ldots, p$, sind linear unabhängig.

Dann existieren eindeutige $\lambda \in \mathbb{R}^m$ und $\mu \in \mathbb{R}^p$, so dass gilt

$$\nabla_x L(\hat{x}, \lambda, \mu) = \qquad \text{(Stationarität)} \tag{5.3}$$

$$\nabla f(\hat{x}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\hat{x}) + \sum_{j=1}^{p} \mu_j \nabla h_j(\hat{x}) = 0, \tag{5.4}$$

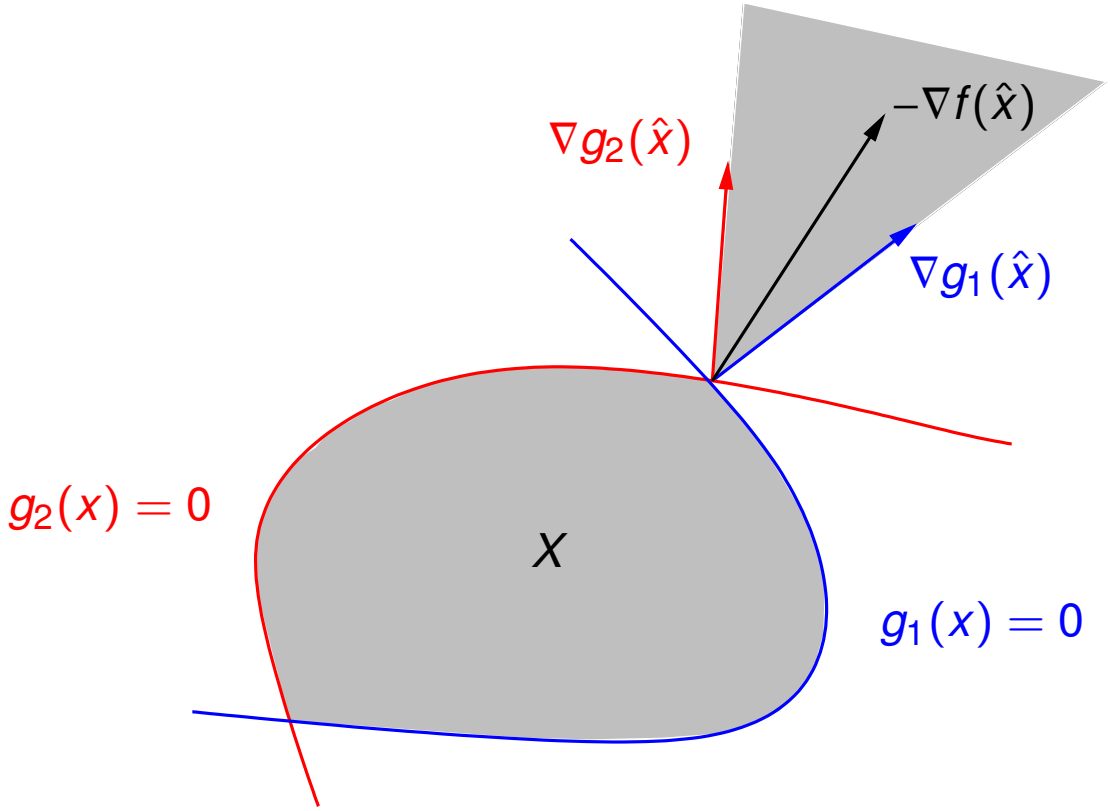
$$\lambda \geq 0, \qquad \lambda^\top g(\hat{x}) = 0, \qquad \text{(Komplementarität)} \tag{5.5}$$

$$g(\hat{x}) \leq 0, \qquad h(\hat{x}) = 0. \qquad \text{(Zulässigkeit)} \tag{5.6}$$

KKT-Punkte, d.h. Punkte $(\hat{x}, \lambda, \mu)$, die (5.3) – (5.6) erfüllen, sind Kandidaten für Optimallösungen eines (SOP).

Geometrische Motivation der KKT-Bedingungen

Bsp.: 2 Ungleichungsrestriktionen und keine Gleichungsrestriktionen



Falls in einem lokalen Minimum $\hat{x}$ die Gradienten der aktiven Ungleichungsrestriktionen linear unabhängig sind, kann der negative Gradient der Zielfunktion als nicht-negative Linearkombination der Gradienten der aktiven Beschränkungen dargestellt werden.

Regularitätsbedingungen

Falls **Regularitätsbedingungen** (constraint qualifications (CQ)) wie (LICQ) erfüllt, sind die KKT-Bedingungen notwendig.

Weitere Regularitätsbedingungen: Mangasarian-Fromowitz, Robinson, Abadie, Slater ("Existenz eines inneren Punkts") ...

Falls keine (CQ) erfüllt ist, dann gelten die KKT-Bedingungen i.A. so nicht:
Vor f tritt ein weiterer Multiplikator auf (Fritz-John-Bedingungen), der evtl. Null wird

Die (CQ) schließen isolierte Punkte bzw. Punkte, in die man nicht in hineinlaufen kann, aus.

Typischerweise hängen (CQ) von $\hat{x}$ ab $\rightsquigarrow$ schwierig *a priori* zu überprüfen

(LICQ) garantiert darüber hinaus die Eindeutigkeit der Multiplikatoren.

Kritischer Kegel

Sei (x, λ, μ) ein KKT-Punkt von (SOP) und

$$\begin{aligned} I_0(x) &:= \{i \in \{1, \dots, m\} \mid g_i(x) = 0, \lambda_i = 0\} && \text{(strikte Komplementarität verletzt)}, \\ I_>(x) &:= \{i \in \{1, \dots, m\} \mid g_i(x) = 0, \lambda_i > 0\} && \text{(strikte Komplementarität erfüllt)}. \end{aligned}$$

Dann definiere den sogenannten **kritischen Kegel**

$$\begin{aligned} T(x) := \{d \in \mathbb{R}^n \mid \quad & \nabla h_j(x)^\top d = 0, && j = 1, \dots, p, \\ & \nabla g_i(x)^\top d = 0, && i \in I_>(x), \\ & \nabla g_i(x)^\top d \leq 0, && i \in I_0(x) \qquad \}. \end{aligned}$$

Interpretation: T ist Menge der mehrdeutigen (kritischen) Richtungen

Notwendige & hinreichende Bedingungen 2. Ordnung

Theorem (Notwendige Optimalitätsbedingung 2. Ordnung)

Sei $L : \mathbb{R}^{n \times p \times m} \to \mathbb{R}$ 2mal stetig differenzierbar bzgl. x.
Sei $\hat{x}$ ein lokales Minimum von (SOP), so dass (LICQ) erfüllt ist.

Dann gilt

$$d^\top \nabla^2_{xx} L(\hat{x}, \lambda, \mu) d \geq 0 \quad \text{für alle } d \in T(\hat{x}),$$

wobei $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$ die eindeutig bestimmten Lagrangeschen Multiplikatoren aus den KKT-Bedingungen sind.

Theorem (Hinreichende Optimalitätsbedingung 2. Ordnung)

Sei $L : \mathbb{R}^{n \times p \times m} \to \mathbb{R}$ 2mal stetig differenzierbar bzgl. x.
Sei $(\hat{x}, \lambda, \mu)$ ein KKT-Punkt von (SOP) mit

$$d^\top \nabla^2_{xx} L(\hat{x}, \lambda, \mu) d > 0 \quad \text{für alle } d \in T(\hat{x}) \setminus \{0\}.$$

Dann ist $\hat{x}$ ein striktes lokales Minimum von (SOP).

Gleichungsrestringierte Pb.: Lagrangesche Multiplikatorenregel

(SOP) ohne Ungleichungsrestriktionen ergibt

Problem (Nichtlin. Optimierungspb., rein gleichungsrestringiert)

Minimiere $f(x)$ *u. d. N.*

$$h_j(x) = 0, \quad j = 1, \ldots, p.$$

Dabei seien $x \in \mathbb{R}^n$ *und die Funktionen*

$$f : \mathbb{R}^n \to \mathbb{R},$$
$$h = (h_1, \ldots, h_p)^\top : \mathbb{R}^n \to \mathbb{R}^p.$$

Hier kann man ggf. mithilfe des Satzes über implizite Funktionen auflösen, partioniere dazu $x = (y, z)^\top \in \mathbb{R}^{n-p} \times \mathbb{R}^p$ und setze $h(y, z(y)) = 0$, somit

$$\mu^\top := -\frac{\partial f}{\partial z}(\hat{y}, \hat{z}) \left(\frac{\partial h}{\partial z}(\hat{y}, \hat{z}) \right)^{-1}$$

und dann wie beim unrestringierten Optimierungsproblem vorgehen.

Lagrangesche Multiplikatorenregel

Theorem (Satz 5.1.2 - Lagrangesche Multiplikatorenregel)

Seien $f : \mathbb{R}^n \to \mathbb{R}$ und $h : \mathbb{R}^n \to \mathbb{R}^p$, stetig differenzierbar und $\hat{x}$ ein lokales Minimum von (SOP) für $m = 0$.

Des Weiteren gelte

$$Rang(h'(\hat{x})) = p,$$

wobei $h'(\hat{x})_{ij} := \frac{\partial h_i}{\partial x_j}$ die Jacobi-Matrix von h in $\hat{x}$.

Dann existiert ein $\mu \in \mathbb{R}^p$, so dass gilt

$$\nabla_x L(\hat{x}, \mu) = 0, \qquad \text{(Stationarität)}$$
$$h(\hat{x}) = 0. \qquad \text{(Zulässigkeit)}$$

Wh.:

Der Rang wird durch das Gaußsche Eliminationsverfahren bestimmt.
Für reelle Zahlen gilt Zeilenrang = Spaltenrang.
In 1D gilt $h' = \nabla h^\top$.

Lagrange-Newton-Verfahren

Anwendung des Newton-Verfahrens auf $F(x,\mu)^\top := (\nabla_x L(x,\mu), h(x))^\top = 0$ ergibt

Algorithmus (*Lagrange-Newton-Verfahren*)

(i) *Wähle* $x^{[0]} \in \mathbb{R}^n$ *und* $\mu^{[0]} \in \mathbb{R}^p$.
Setze $k := 0$.

(ii) *Falls*

$$\|F(x^{[k]}, \mu^{[k]})\| \approx 0,$$

dann STOP.

(iii) *Löse das LGS*

$$\begin{pmatrix} \nabla^2_{xx} L(x^{[k]}, \mu^{[k]}) & h'(x^{[k]})^\top \\ h'(x^{[k]}) & 0 \end{pmatrix} \begin{pmatrix} d \\ v \end{pmatrix} = - \begin{pmatrix} \nabla_x L(x^{[k]}, \mu^{[k]}) \\ h(x^{[k]}) \end{pmatrix}.$$

Setze $x^{[k+1]} := x^{[k]} + d$ *und* $\mu^{[k+1]} := \mu^{[k]} + v$.

(iv) *Setze* $k := k + 1$ *und*
gehe zu (ii).

Konvergenz des Lagrange-Newtonverfahrens

Aus der Konvergenz des lokalen Newtonverfahrens folgt die lokale Konvergenz des Lagrange-Newtonverfahrens

Hinreichende Kriterien für die Invertierbarkeit von

$$\begin{pmatrix} \nabla^2_{xx} L(x^{[k]}, \mu^{[k]}) & h'(x^{[k]})^\top \\ h'(x^{[k]}) & 0 \end{pmatrix}$$

sind:

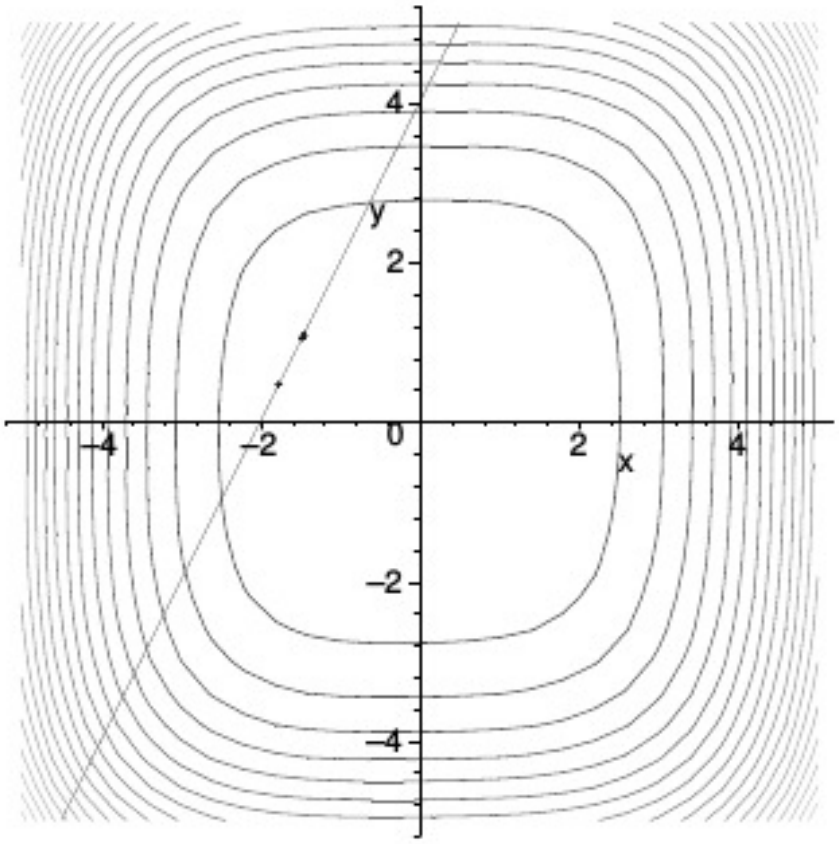
- Es gilt (LICQ) und
- die hinreichende Optimalitätsbedingung 2. Ordnung, wobei hier $T(\hat{x}) = \{d \mid h'(\hat{x})d = 0\}$.

Beispiel für Lagrange-Newton-Verfahren

Minimiere $2x_1^4 + x_2^4 + 4x_1^2 - x_1x_2 + 6x_2^2$ u.d.N. $2x_1 - x_2 = -4$, $x_1, x_2 \in \mathbb{R}$.

Beispiel für Lagrange-Newton-Verfahren

Minimiere $2x_1^4 + x_2^4 + 4x_1^2 - x_1x_2 + 6x_2^2$ u.d.N. $2x_1 - x_2 = -4, \quad x_1, x_2 \in \mathbb{R}$.




```
ITERATION  0
  ZIELFUNKTION   =          0.0000000000000000E+00
  BESCHRAENKUNG  =          0.4000000000000000E+01
  NORM KKT       =          0.0000000000000000E+00
  X=              0.0000000000000000E+00          0.0000000000000000E+00
  LAMBDA=         0.0000000000000000E+00

ITERATION  1
  ZIELFUNKTION   =          0.3425678372606001E+02
  BESCHRAENKUNG  =          0.0000000000000000E+00
  NORM KKT       =          0.4430579634007108E+02
  X=             -0.1769230769230769E+01          0.4615384615384616E+00
  LAMBDA=         0.7307692307692307E+01

ITERATION  2
  ZIELFUNKTION   =          0.2756373273045131E+02
  BESCHRAENKUNG  =          0.0000000000000000E+00
  NORM KKT       =          0.5155005942054521E+01
  X=             -0.1452391998833495E+01          0.1095216002333011E+01
  LAMBDA=         0.1660806234685793E+02

ITERATION  3
  ZIELFUNKTION   =          0.2754452288430214E+02
  BESCHRAENKUNG  =          0.0000000000000000E+00
  NORM KKT       =          0.1497798610217076E-01
  X=             -0.1467843643905216E+01          0.1064312712189567E+01
  LAMBDA=         0.1904961295898514E+02

ITERATION  4
  ZIELFUNKTION   =          0.2754452202163215E+02
  BESCHRAENKUNG  =          0.0000000000000000E+00
  NORM KKT       =          0.6772437517980240E-06
  X=             -0.1467948113419141E+01          0.1064103773161718E+01
  LAMBDA=         0.1905680332151563E+02

ITERATION  5
  ZIELFUNKTION   =          0.2754452202163215E+02
  BESCHRAENKUNG  =          0.0000000000000000E+00
  NORM KKT       =          0.3552713678800501E-14
  X=             -0.1467948118040920E+01          0.1064103763918160E+01
  LAMBDA=         0.1905680364713604E+02
```

SQP-Verfahren

Setzt man $H_k := \nabla^2_{xx} L(x^{[k]}, \mu^{[k]})$ im Gleichungssystem des Lagrange-Newton-Verfahrens, so erhält man genau die KKT-Bedingungen des quadratischen Problems

$$\begin{aligned} \min_d \quad & f'(x^{[k]})d + \frac{1}{2} d^\top H_k d \\ \text{u.d.N.} \quad & h_j(x^{[k]}) + h_j'(x^{[k]})d = 0 \qquad \text{für alle } j = 1, \ldots, p. \end{aligned}$$

Einschließlich Ungleichungen und mit $H_k := \nabla^2_{xx} L(x^{[k]}, \lambda^{[k]}, \mu^{[k]})$ erhält man

$$\begin{aligned} \min_d \quad & f'(x^{[k]})d + \frac{1}{2} d^\top H_k d \\ \text{u.d.N.} \quad & h_j(x^{[k]}) + h_j'(x^{[k]})d = 0 \qquad \text{für alle } j = 1, \ldots, p, \\ & g_i(x^{[k]}) + g_i'(x^{[k]})d \leq 0 \qquad \text{für alle } i = 1, \ldots, m. \end{aligned}$$

Man löst also in jedem Iterationsschritt ein quadratisches Problem.

Dies ist das wichtige **SQP-Verfahren (sequential quadratic programming)**.

Penalty-Verfahren I

Idee: Approximiere iterativ durch Lösung von unrestringierten Hilfsproblemen

Hilfsprobleme minimieren **Penalty-Funktion**

$$P(x;\eta_k) := f(x) + \frac{\eta_k}{2}\sum_{j=1}^{p}(h_j(x))^2 + \frac{\eta_k}{2}\sum_{i=1}^{m}(\max\{0, g_i(x)\})^2$$

für geeignete $\eta_k > 0$

Nachteile:

Gemäß Konvergenzresultat muss η_k gegen ∞ streben,
$\rightsquigarrow$ Algorithmus schlecht konditioniert;

P nicht exakt (exakt heißt es existiert $0 < \hat{\eta} < \infty$ so dass $\hat{x}$ lokales Minimum von $P(\cdot, \eta)$ für alle $\eta \geq \hat{\eta}$), und

P i.A. nicht differenzierbar

Penalty-Verfahren II

Algorithmus (*Penalty-Verfahren (rein gleichungsrestringiert)*)

(i) *Wähle* $x^{[0]} \in \mathbb{R}^n$, $\eta_0 > 0$ *und*
setze $k = 0$.

(ii) *Bestimme* $x^{[k]}$ *als Lösung von*

$$\text{Minimiere } P(x; \eta_k) \quad \text{u. d. N.} \quad x \in \mathbb{R}^n.$$

(iii) *Falls* $h(x^{[k]}) \approx 0$, *STOP.*

(iv) *Bestimme* $\eta_{k+1} > \eta_k$,
setze $k := k + 1$ *und*
gehe zu (ii).

Multiplikator-Penalty-Verfahren I

Betrachte hier nur Gleichungsrestriktionen

Betrachte anstatt Penalty-Funktion die **erweiterte Lagrange-Funktion** (Multiplikator-Penalty-Funktion)

$$L_a(x, \mu; \eta) := f(x) + \frac{\eta}{2}\|h(x)\|^2 + \mu^\top h(x)$$

für geeignete $\eta > 0$

Vorteile:
η muss nicht gegen ∞ streben.

L_a ist differenzierbar.

Nachteil:
Der optimale Lagrangeparameter μ ist unbekannt.

Multiplikator-Penalty-Verfahren II

Algorithmus (*Multiplikator-Penalty-Verfahren*)

(i) *Wähle* $x^{[0]} \in \mathbb{R}^n$, $\mu^{[0]} \in \mathbb{R}^p$, $\eta_0 > 0$, $\sigma \in (0,1)$ *und setze* $k = 0$.

(ii) *Falls* $(x^{[k]}, \mu^{[k]})$ *KKT-Punkt, STOP*

(iii) *Bestimme* $x^{[k+1]}$ *als Lösung von*

$$\text{Minimiere } L_a(x, \mu^{[k]}; \eta_k) \quad \text{u. d. N.} \quad x \in \mathbb{R}^n.$$

(iv) *Setze* $\mu^{[k+1]} := \mu^{[k]} + \eta_k h(x^{[k+1]})$.

(v) *Ist* $\|h(x^{[k+1]})\| \geq \sigma \|h(x^{[k]})\|$, *dann* $\eta_{k+1} := 10\eta_k$, *sonst* $\eta_{k+1} := \eta_k$.

(vi) *Setze* $k := k + 1$ *und gehe zu (ii).*

Das Multiplikator-Penalty-Verfahren lässt sich auch auf Ungleichungs-Nebenbedingungen erweitern.

Inhaltsverzeichnis