Lineare und nichtlineare Optimierung

Priv.-Doz. Dr. Sven-Joachim Kimmerle

Vorlesung Herbsttrimester 2023
Bachelor Luft- und Raumfahrttechnik

Überblick

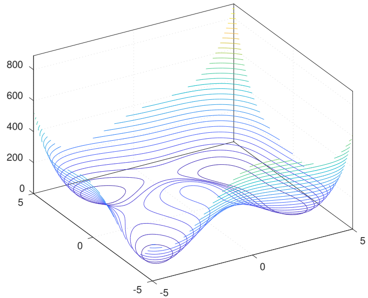
1. Einführung
   - Dozent
   - Thema der Lehrveranstaltung
   - Organisatorisches
   - Einleitung
2. Unrestringierte Optimierung
3. Wichtige Verfahren der numerischen Optimierung
4. Restringierte Optimierung
5. Zusammenfassung - Ausblick und Wiederholung

Dozent: Sven-Joachim Kimmerle

- 2000: Vordiplom in Mathematik & Vordiplom in Physik (U Heidelberg)
- 2002: Maîtrise in Mathematik (U Paris 7, Frankreich)
- 2004: Diplom in Mathematik (U Heidelberg)
- 2004-2009: Forschungszentrum MATHEON, Berlin
- 2009: Promotion in Mathematik (HU Berlin)
- 2010: Toyota/U Ottawa, Ottawa, Kanada
- 2011-2018: Postdoc & Vertretungsprofessor, UniBw München, Neubiberg
- 2019: Habilitation in Mathematik (UniBw München, Neubiberg)
- Seit 2018: Physical Software Solutions GmbH, Münsing & Ottobrunn
- Seit 2019: zusätzlich Lecturer an verschiedenen Hochschulen

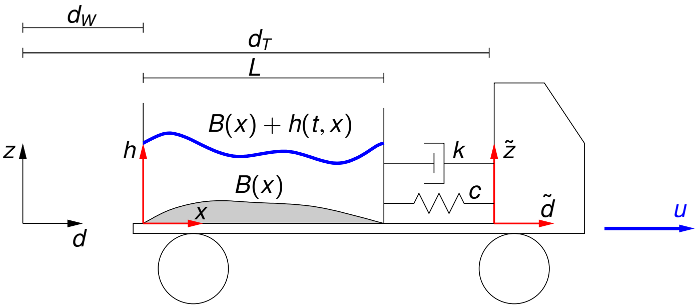
Was ist Optimierung?

- Ein konkretes Problem, das sich mathematisch formulieren lässt, habe mehrere Lösungen (vielleicht sogar unendlich viele). Welche Lösung ist die beste d.h. optimale?
- Idee: Betrachte Zielfunktion, diese wird unter Nebenbedingungen extremal



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

LKW mit Flüssigkeitscontainer: horizontale Steuerung (Kraft) u, Parameter Fahrzeit T



m_T: Masse LKW, m_W: Masse Flüssigkeitscontainer, d_T: Wegstrecke LKW, d_W: Wegstrecke Container, $B(x)$: Bodenprofil Container, $h(t,x)$: Flüssigkeitsspiegel relativ zum Bodenprofil zur Zeit t an der Position x, $v(t,x)$: horizontale Flüssigkeitsgeschwindigkeit zur Zeit t an der Position x, L: Länge des Containers, c : Federkonstante, k: Dämpfungskonstante

ODE-PDE Optimalsteuerungsproblem

Optimalsteuerungsproblem

Minimiere als Zielfunktion $\mathcal{J}$ eine Linearkombination von

- Fahrzeit T ($\rightsquigarrow$ Gewicht α_0);
- Abweichung zu einem gewünschten, z.B. konstanten Flüssigkeitsspiegel $\|h - h_d\|^2_{L^2(Q)}$ ($\rightsquigarrow$ Gewicht α_1);[a]
- Aufwand ("Kosten") der Steuerung $\|u\|^2_{L^2([0,T])}$ ($\rightsquigarrow$ Gewicht α_2);
- Abweichung zu Endpositionen $\|d_T(T) - d_T^T\|^2 + \|d_W(T) - d_W^T\|^2$ und -geschwindigkeiten $\|d_T'(T) - (d_T')^T\|^2 + \|d_W'(T) - (d_W')^T\|^2$ ($\rightsquigarrow$ Gewichte α_3, α_4)

unter den Nebenbedingungen

- gekoppelte gewöhnliche und partielle Differentialgleichungen (ODE & PDE) gelten mit Anfangs- und Randbedingungen
- Steuerbeschränkungen
  $u(t) \in \mathcal{U} := [u_{min}, u_{max}], \quad U_{ad} = \left\{ u \in L^2(0,T) \,\middle|\, u(t) \in \mathcal{U} \right\}.$
- Zustandsbeschränkungen ($0 < \underline{h} \leq h \leq \overline{h}$)

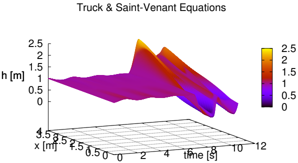
---

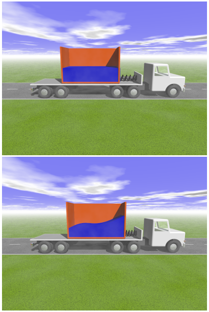
[a] L^2 ist ein Funktionenraum, Details spielen hier keine Rolle.

Numerische Optimalsteuerung: Nur Fahrzeit in der Zielfunktion [Gerdts/K. 2015]

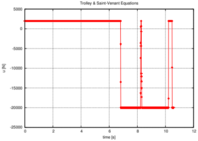
Ziel: Optimales Bremsen

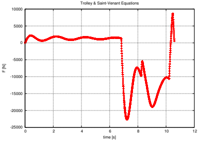
- bei kürzester Fahrzeit (Bremsung in letzter Minute)





Resultat: $T = 10.6$ s, $\epsilon = 0.453$, Steuerung u (links), Feder-Dämpfer-Kraft (rechts)



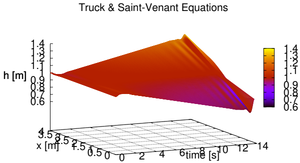


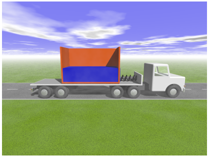
($\alpha_0 = 0.1$, $\alpha_1 = 0$, $\alpha_2 = 0$, $\alpha_3 = 0$, $\alpha_{4a} = 0$, $\alpha_{4b} = 100$, $d_T(T) = 100$, $d_W(T) = 95$, $d'_T(0) = 10$, $d'_W(0) = 10$, $N = 1500$, $M = 50$, $L = 4$, $m_T = 2000$, $m_W = 4000$, $h_0 = 1$, $c = 40000$, $k = 10000$, $g = 9.81$, $u_{min} = -20000$, $u_{max} = 2000$)

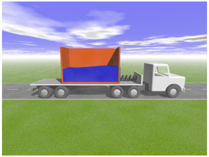
Numerische Optimalsteuerung: Kompromiss in der Zielfunktion [Gerdts/K. 2015]

Ziel: Optimales Bremsen

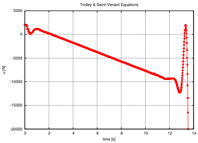
- bei kürzester Fahrzeit,
- mit minimalem Steueraufwand und
- mit minimaler "Aufschaukelung"

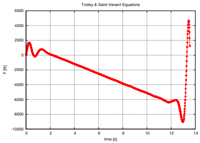






Resultat: $T = 13.8$ s, $\epsilon = 0.644$, Steuerung u (links), Feder-Dämpfer-Kraft (rechts)





($\alpha_0 = 0.1$, $\alpha_1 = 1$, $\alpha_2 = 0.001$, $\alpha_3 = 0$, $\alpha_{4a} = 0$, $\alpha_{4b} = 100$, $d_T(T) = 100$, $d_W(T) = 95$, $d_T'(0) = 10 = d_W'(0)$, $N = 1000$, $M = 30$, $L = 4$, $m_T = 2000$, $m_W = 4000$, $h_0 = 1$, $c = 40000$, $k = 10000$, $g = 9.81$, $u_{min} = -20000$, $u_{max} = 2000$)

Anwendungen der Optimierung I

- Optimierungsaufgaben finden sich:
  - in den Wirtschaftswissenschaften
  - im Management
  - in der Technik
  - in der Produktion
  - in der Bildverarbeitung
  - in den Naturwissenschaften
  - in der Datenanalyse
  - etc.
- Optimierung steht in engem Zusammenhang zur (mathematischen) Modellierung

Anwendungen der Optimierung II

- Beispiele in den Anwendungsbereichen:
  - Minimierung von Herstellungskosten
  - Minimierung von Transportkosten
  - Ausgleichsrechnung, Approximation
  - Optimalsteuerung
  - Parameteridentifikation, z.B. in Differentialgleichungen
  - etc.

Ziele der Lehrveranstaltung

Vermittlung von Grundlagen aus dem Gebiet der linearen und nichtlinearen Optimierung:

- Überblick über verschiedene Arten von Optimierungsproblemen
- Kennenlernen grundlegender Methoden der Optimierung
- Verständnis der Theorie, die dahinter steckt
- Analyse und Implementierung von Algorithmen
- Praktisches Lösen von Optimierungsproblemen aus verschiedenen Anwendungsgebieten

Organisatorisches

- 2 V + 1 Ü (seminaristischer Unterricht)

  Mo, 15:00-17:15 in 2111-033 (Pausen flexibel n.V.)
  endgültige Vorlesungs- und Übungszeiten nach Vereinbarung beim ersten Treffen!
  erstes Treffen am Mo, 02.10.2023 um 15:00

- Webseite (Folien, Übungsblätter & anderes Material):
  `www.unibw.de/ingmathe/teaching/linearenichtlineareoptimierung`

- ggf. virtuell über Zoom
  Details und Zugangsschlüssel folgen bei Bedarf

- Sprechzeiten & Kontakt
  - Nach der Lehrveranstaltung bzw. flexibel nach Vereinbarung
  - Email: `sven-joachim.kimmerle@unibw.de`

Benotung

- **Prüfungsform**
  - Mündliche Prüfung am Ende des Trimester
  - Bitte achten Sie grundsätzlich auch auf Ankündigungen des Prüfungsamts!
  - Prüfungsrelevant ist insbesondere der in der Vorlesung behandelte und in den Übungen vertiefte Stoff.

Geplanter Arbeitsaufwand

Laut Modulhandbuch sind für 3 ECTS Punkte
etwa folgende Aufwände von Ihnen zu erwarten:

- 36 Stunden Anwesenheit (24 Stunden Vorlesung, 12 Stunden Übung)
- 18 Stunden Vor- und Nachbereitung der Übung sowie Übungsaufgaben
- 36 Stunden Nachbereitung der Vorlesung und Prüfungsvorbereitung

Insgesamt also 90 Stunden Workload

Voraussetzungen & Verwendbarkeit

Voraussetzungen:

- Höhere Mathematik I–III
- Programmierkenntnisse in einer beliebigen prozeduralen Programmiersprache (z.B. MATLAB) wünschenswert

Verwendbarkeit:

- Anwendungen des erlangten Wissens in allen Studienschwerpunkten

Empfohlene Literatur

- **Alt, W.: *Nichtlineare Optimierung*. Vieweg, 2002.** $\longleftarrow$ **!**
- Geiger, C. und Kanzow, C.: *Numerische Verfahren zur Lösung unrestringierter Optimierungsaufgaben*. Springer, 1999.
- Geiger, C. und Kanzow, C.: *Theorie und Numerik restringierter Optimierungsaufgaben*. Springer, 2002.
- Nocedal, J. und Wright, S.: *Numerical Optimization*. Springer, 2. Aufl., 2006.
- Gerdts, M.: *Einführung in die lineare und nichtlineare Optimierung*, Vorlesungsskript, BSc.-Studiengang “Luft- und Raumfahrttechnik”, Universität der Bundeswehr München, 2016.

Weitere Quellen folgen im Lauf der Lehrveranstaltung.

Empfohlene Software

Im Praktikum verwenden wir Matlab und dabei auch die *Optimization Toolbox*.

Installation von Matlab:

- Download der Matlab-Studentenversion (kostenlos für UniBw M-Studierende) beim RZ
- Kostenlose Alternative: Scilab http://www.scilab.org/products/scilab/download

Kopierverbot

Alle im Rahmen dieser Vorlesung zur Verfügung gestellten Materialien wurden von mir mit einem Passwort geschützt, das nur den Hörern dieser Veranstaltung bekannt gegeben wurde. Jegliche Form der Weitergabe ist untersagt!

Sven-Joachim Kimmerle

Motivation: Konzept

- 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).

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.)*

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

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

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

Klassen von Opt.pb. I (in Vektornotation)

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.)

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

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

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

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:

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

Ergibt sich aus (SOP) für

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

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** (QOP).

Klassen von Optimierungsproblemen III

Es gibt weitere Klassen von Optimierungsproblemen (OP):

- Vektoroptimierungsprobleme
- unendlichdimensionale OP
- semi-infinite OP
- ganzzahlige (diskrete) OP
- gemischt-ganzzahlige OP
- nicht-differenzierbare OP
- stochastische OP
- ...

Diese betrachten wir hier nicht näher.

Extrema: Minima und Maxima

Definition (Minima)

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

$$f(\hat{x}) \leq f(x) \quad \text{für alle } x \in X. \tag{4.1}$$

- $\hat{x} \in X$ heißt **striktes globales Minimum** von (OP), falls in (4.1) "<" 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

$$f(\hat{x}) \leq f(x) \quad \text{für alle } x \in X \cap U_\varepsilon(\hat{x}). \tag{4.2}$$

- $\hat{x} \in X$ heißt **striktes globales Minimum** von (OP), falls in (4.2) "<" 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 diesem Modul 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: Minimierung mithilfe graphischer Darstellung

Beispiel (Funktion von Himmelblau)

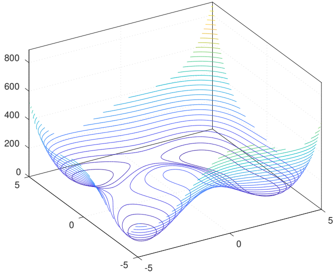
Minimiere

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

über $(x, y)^\top \in \mathbb{R}^2$.

<u>MATLAB-Code:</u>

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

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

Gradient

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

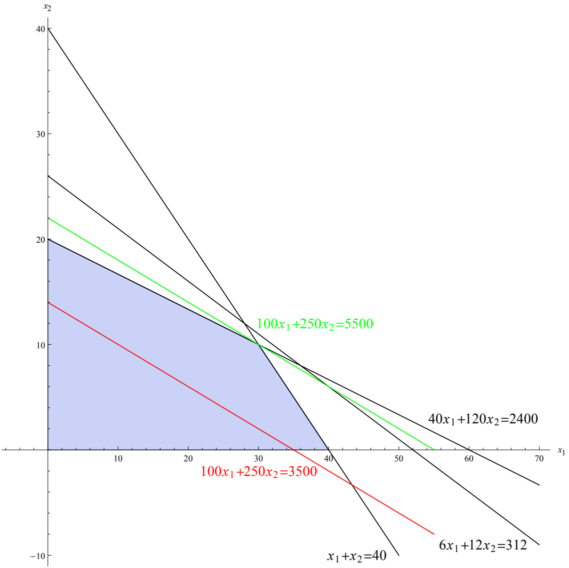
Beispiel (Agrarwirtschaft)

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.

Klassisches lineares Optimierungsproblem II



Transportproblem I

Beispiel (Transportproblem)

Ein Transportunternehmer hat m Vorratslager, aus denen n Verbraucher mit einem Produkt beliefert werden können.

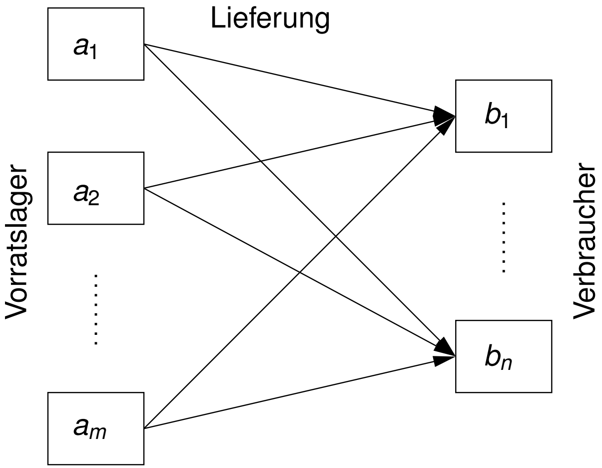
Die Lieferkosten von Lager i zu Verbraucher j betragen c_{ij} Einheiten pro Produkteinheit.

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



Transportproblem III

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:

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

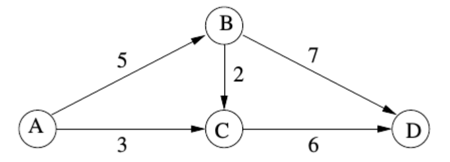
$$\begin{aligned} \sum_{j=1}^{n} x_{ij} &\leq a_i, \quad i = 1, \ldots, m, \\ \sum_{i=1}^{m} x_{ij} &\geq b_j, \quad j = 1, \ldots, n, \\ x_{ij} &\geq 0, \quad i = 1, \ldots, m, \; j = 1, \ldots, n. \end{aligned}$$

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.

Beispiel (Netzwerkproblem)

Eine Firma möchte möglichst viele Waren von A nach D über das abgebildete 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 $(x_1, \ldots, x_n)$.

Portfoliooptimierung II

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 III

Ein Kompromiss ist das folgende Optimierungsproblem

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

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

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

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

Desto kleiner α, desto größer das Risiko.

(Nicht-)Lineare Ausgleichsrechnung: Ausfallzeiten unter Temperaturbelastung

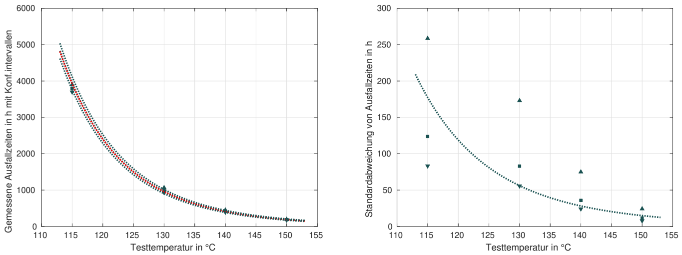


Abbildung: Ausgleichsrechnung über verschiedene Temperaturen T im Vergleich mit Messpunkten (Quadrate) samt Konfidenzintervallgrenzen zu $q = 90\%$ (Dreiecke). Links für $\mu_{krit}(T) \pm \sigma_{krit}(T)$, in Rot eine Referenzkurve. Rechts $\sigma_{krit}(T)$. [K., Dvorsky, Ließ 2018]

Modellbasierter Ansatz

$$f(T) = t_0 \exp\left(\left(\frac{T_a}{T - T_\infty}\right)^d\right)$$

Klassifikation von Optimierungsproblemen

Optimierungsprobleme können z. B. folgendermaßen klassifiziert werden:

- **restringiert** oder **unrestringiert** (d.h. mit oder ohne Nebenbedingungen)
- **global** oder **lokal**
- **eindimensional** oder **mehrdimensional**
- **stetig** oder **diskret**
- **deterministisch** oder **stochastisch**

In dieser Lehrveranstaltung keine diskreten und wenig zu stochastischen Optimierungsproblemen

Restringiert vs. unrestringiert

- In den meisten praktischen Anwendungen kommen Nebenbedingungen vor, die Optimierungsprobleme sind restringiert.
- Einige Anwendungen führen aber auch zu unrestringierten Problemen.
- Restringierte Probleme können durch eine Reihe von unrestringierten Hilfsproblemen gelöst werden.

$\rightsquigarrow$ Hauptunterteilung in dieser Vorlesung:
Zuerst unrestringierte Probleme, dann restringierte Probleme

Thematischer Aufbau

1. **Einführung** und eindimensionale Probleme
2. Grundlagen der **unrestringierten Optimierung** und evtl. ableitungsfreie Verfahren
3. Gradienten-, CG- und (Quasi-)Newton-**Verfahren**
4. Beispiele und theoretische Grundlagen der **restringierten Optimierung**
5. **Lineare Optimierung**: Simplex-Verfahren und Innere-Punkt-Methode
6. **Quadratische Optimierung** mit der Active-Set-Methode
7. Allgemeine **nichtlineare Optimierung** mit der SQP-Methode sowie Lagrange- und Penaltyverfahren

Überblick

1. Einführung
2. Unrestringierte Optimierung
   - Unrestringierte Optimierung in 1D
   - Unrestringierte Optimierung in mehreren Dimensionen
3. Wichtige Verfahren der numerischen Optimierung
4. Restringierte Optimierung
5. Zusammenfassung - Ausblick und Wiederholung

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.*

Idee:
Suche zunächst nach lokalen Minima (aus denen man unter bestimmten Bedingungen globale Extrema erhält).

Ziele:
Notwendige Bedingungen
Hinreichende Bedingungen
Wünschenswert: Notwendige & hinreichende Bedingungen

Numerische Verfahren (meist basierend auf notwendigen Bedingungen)

Unrestringierte Optimierung für $n = 1$

Wir betrachten hier zunächst

Problem (Unrestringiertes Optimierungsproblem (UOP))

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

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

Idee:
Suche zunächst nach lokalen Minima (aus denen man unter bestimmten Bedingungen globale Extrema erhält).

Ziele:
Notwendige Bedingungen
Hinreichende Bedingungen
Wünschenswert: Notwendige & hinreichende Bedingungen

Numerische Verfahren (meist basierend auf notwendigen Bedingungen)

Theorie in 1D

Eine zweimal stetig differenzierbare Funktion $f : \mathbb{R} \to \mathbb{R}$ besitzt ein lokales Minimum bei $\overline{x}$.

Dann gilt:

- $f'(\overline{x}) = 0$ (Notwendige Bedingung 1. Ordnung)
- $f''(\overline{x}) \geq 0$ (Notwendige Bedingung 2. Ordnung)

Gilt umgekehrt $f'(\overline{x}) = 0$ und $f''(\overline{x}) > 0$,
so ist $\overline{x}$ ein lokales Minimum von f.
(Hinreichende Bedingung)

Verfahren in 1D

- $\overline{x}$ kann in den seltensten Fällen analytisch berechnet werden.
- Verfahren, die beide Ableitungen benutzen, sind meist sehr schnell,
  aber nicht alle Funktionen sind zweimal stetig differenzierbar und Ableitungen sind oft schwer zu berechnen.
- Ableitungsfreie Methoden sind zwar langsamer, aber für eine breitere Klasse von Problemen einsetzbar.

Beispiel: Newton-Verfahren in 1D

- **Annahme:** $f(x)$ ist zweimal stetig differenzierbar; $f'(x)$ und $f''(x)$ können analytisch berechnet werden.
- **Strategie:** Bestimme iterativ eine Nullstelle der ersten Ableitung; starte dazu von einem Wert, in dessen Nähe man ein Minimum vermutet.
- **Verfahren:** Beginne mit Startwert $x^{[0]}$

$$\text{Für } k = 0, 1, 2, \ldots : \qquad x^{[k+1]} = x^{[k]} - \frac{f'(x^{[k]})}{f''(x^{[k]})}$$

- **Konvergenz:** Das Newton-Verfahren konvergiert lokal quadratisch, d.h. es gilt

$$|\overline{x} - x^{[k+1]}| \leq c \cdot |\overline{x} - x^{[k]}|^2,$$

falls f'' in einer Umgebung der Nullstelle invertierbar und Lipschitz-stetig.

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 <u>konstruktiv</u> $\hat{x}$ so dass

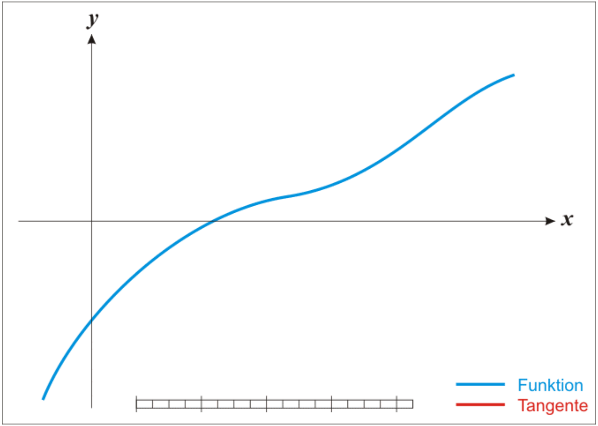
$$g(\hat{x}) = 0.$$

Newton-Verfahren:

$$x^{[k+1]} = x^{[k]} - \frac{g(x^{[k]})}{g'(x^{[k]})}, \quad x^{[0]} \in \mathbb{R} \text{ gegebener Startwert}$$

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

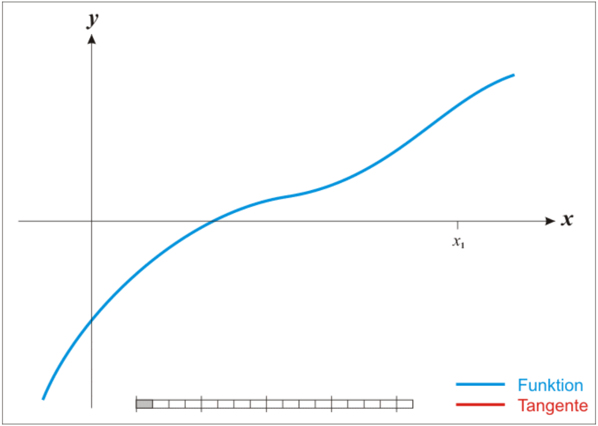


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

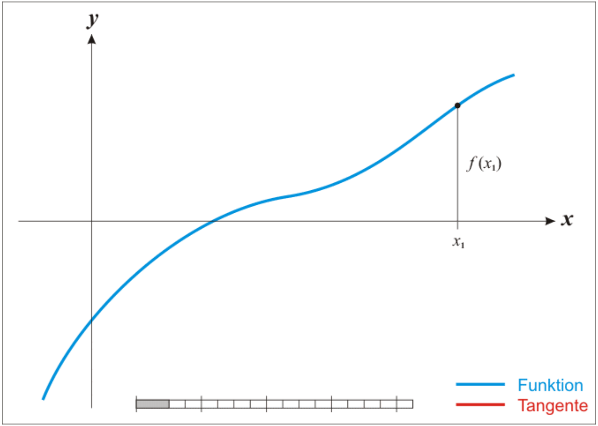


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

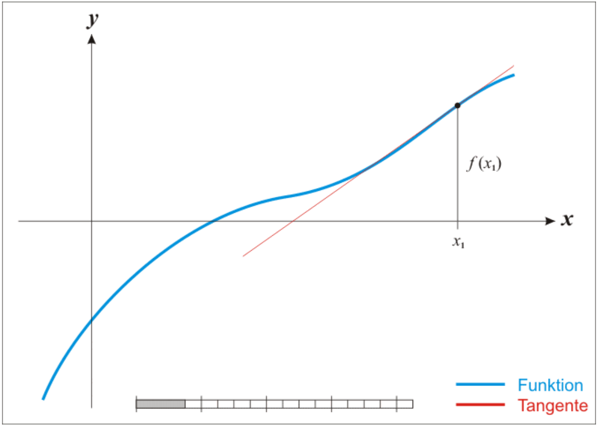


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

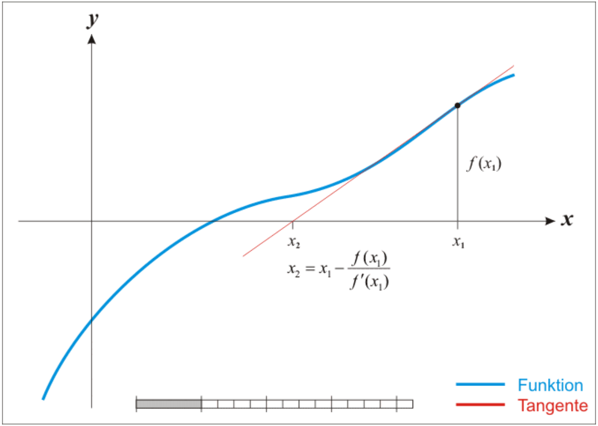


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

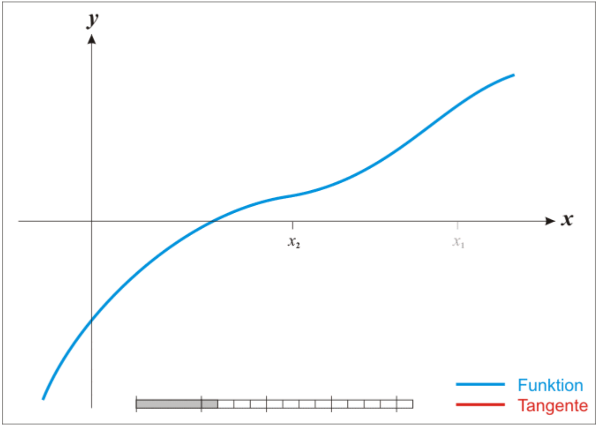


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

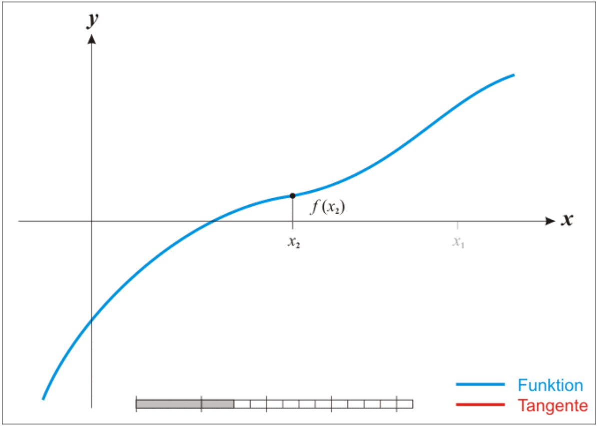


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

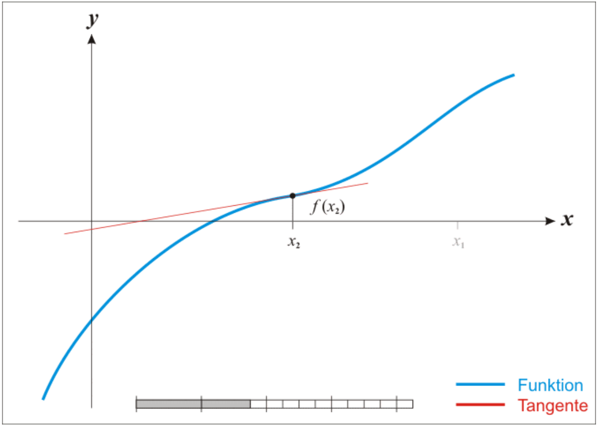


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

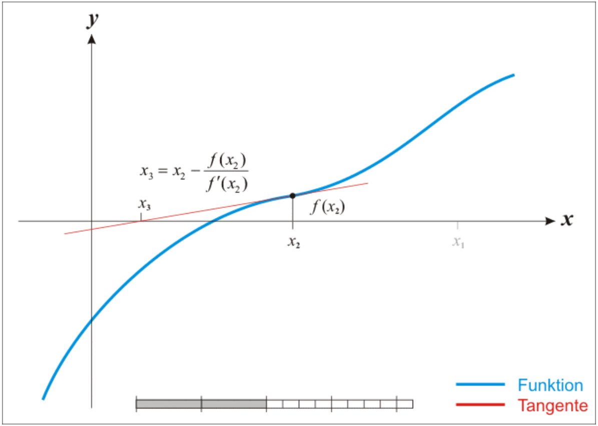


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

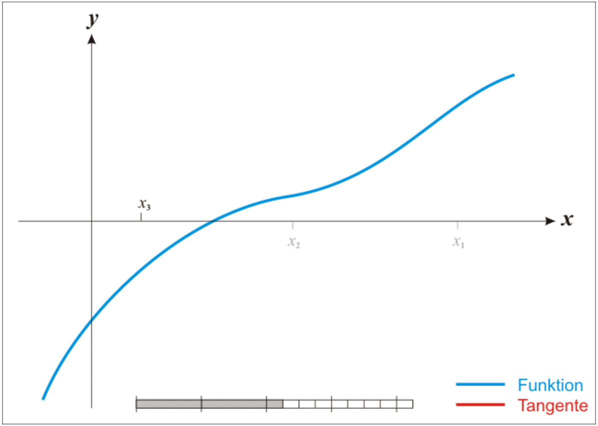


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

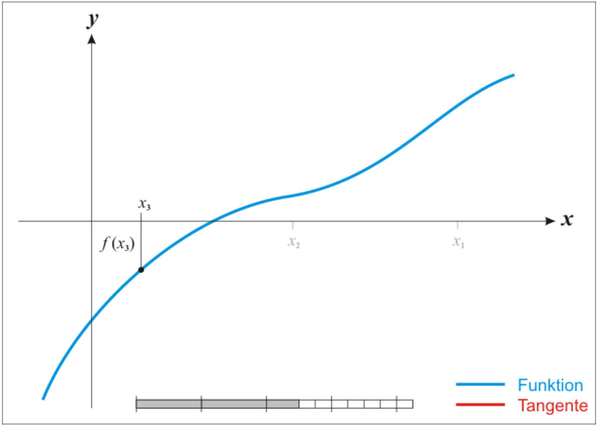


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

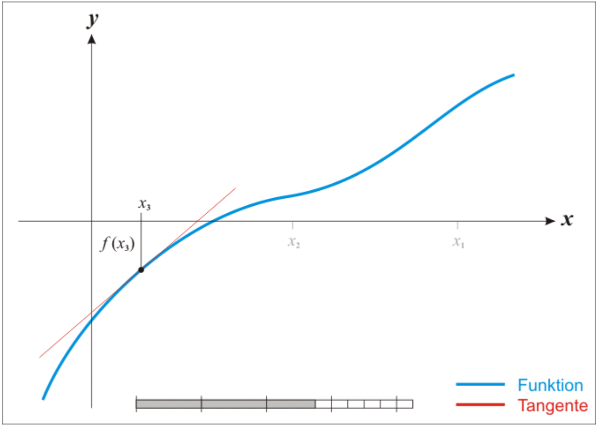


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

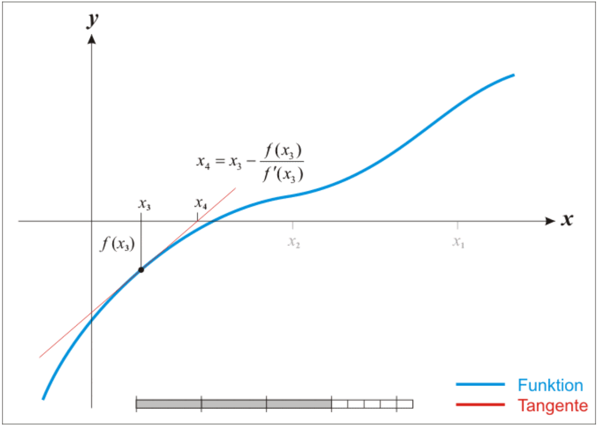


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

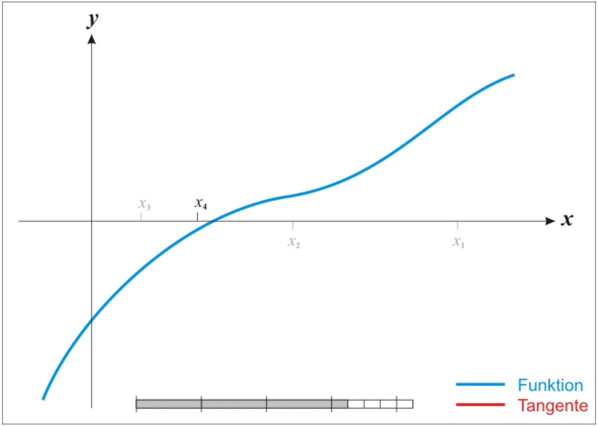


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

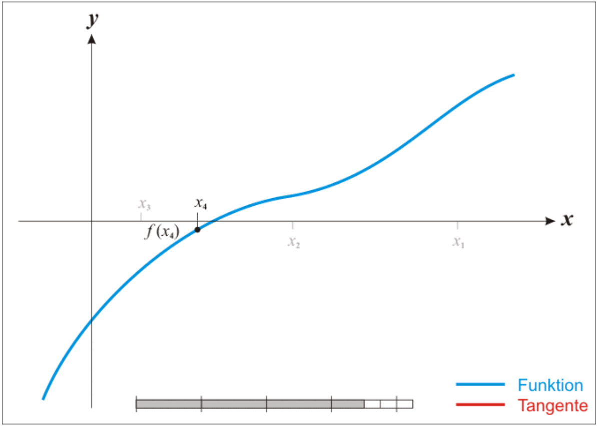


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

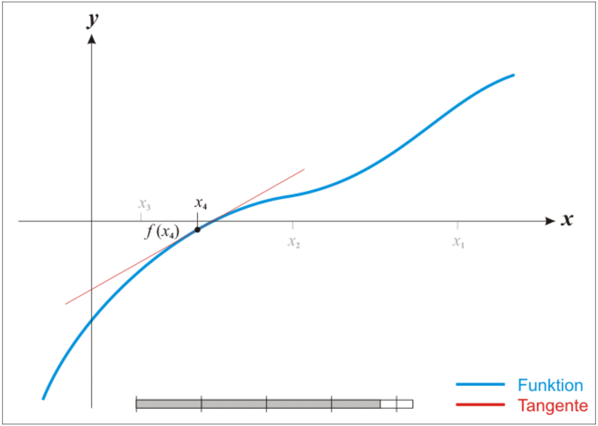


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

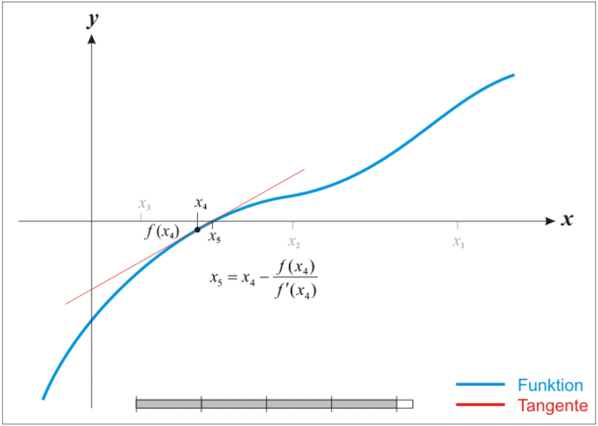


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):

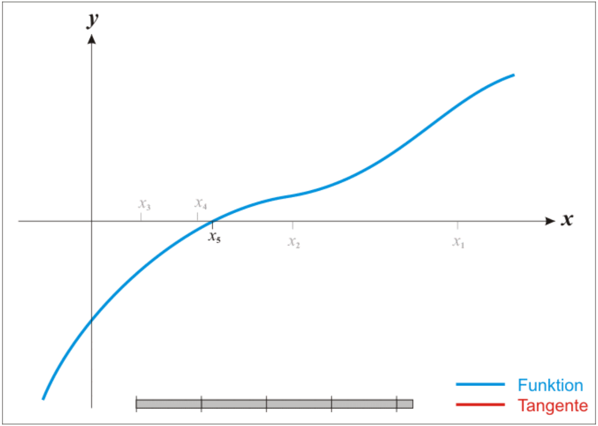


(hier $x_n := x^{[n]}$)

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

Newton-Verfahren in 1D

Als Beispiel betrachten wir folgende Funktion f (in der Optimierung betrachtet man typischerweise f' !):



(hier $x_n := x^{[n]}$)

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

Probleme & Modifikationen Newton-Verfahren

- **$f''(x)$ kann nicht explizit berechnet werden.**

  $\Longrightarrow$ Ersetze $f''(x)$ durch den Differenzenquotienten oder eine Formel zum numerischen Differenzieren.

- **Das Newton-Verfahren divergiert.**

  $\Longrightarrow$ Kombination des Newton-Verfahrens mit einem sicheren Verfahren wie z.B. dem Bisektions-Verfahren („safeguarding“) oder Reduktion der Schrittweite (gedämpftes Newton-Verfahren)

- **Das Verfahren konvergiert gegen einen Sattelpunkt.**

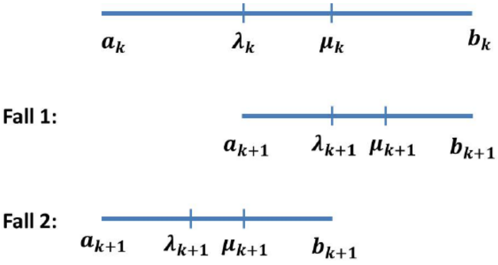
  $\Longrightarrow$ Beginne mit einem anderen Startwert oder führe einige Iterationen mit einem langsamen Verfahren durch.

Bsp.: Verfahren des goldenen Schnitts

- Annahme: $f(x)$ stetig mit Minimum in $[a, b]$
- Teile $[a, b]$ im Verhältnis des goldenen Schnitts, d. h.

$$\lambda = a + 0.382 \cdot (b - a) \text{ und}$$
$$\mu = a + 0.618 \cdot (b - a)$$

- Falls $f(\lambda^{[k]}) > f(\mu^{[k]})$ gehe zu Fall 1 ansonsten zu Fall 2 und wende das Verfahren rekursiv an



Goldener Schnitt: Implementierung in Matlab


```
function x = Golden(f,a,b,iter)
% Beispielaufruf: x=Golden(@(x)(x-1)^2,-1,3,10)

g=(sqrt(5)-1)/2;  % Verhaeltnis des Goldenen Schnitts 0.618
lambda=a+(1-g)*(b-a);
mu=a+g*(b-a);
f_lambda=f(lambda);f_mu=f(mu); % Funktionswerte fuer Fallunterscheidung
for i=1:iter
    disp(['a: ',num2str(a),' , b: ',num2str(b),' , f(lam): ',num2str(f_lambda),' , f(mu): ',num2str(f_mu)]);
    if f_lambda>f_mu
        a=lambda;lambda=mu;mu=a+g*(b-a);
        f_lambda=f_mu;f_mu=f(mu);
    else
        b=mu;mu=lambda;lambda=a+(1-g)*(b-a);
        f_mu=f_lambda;f_lambda=f(lambda);
    end
end
x=(a+b)/2; %Ausgabewert
end
```

Goldener Schnitt: Konvergenz

- Sichere Konvergenz des Verfahrens, falls $f(x)$ im betrachteten Intervall konvex oder unimodal ist
- Relativ langsame Konvergenz:
  $|b_{k+1} - a_{k+1}| = 0.618 \cdot |b_k - a_k|$
  Zum Vergleich Bisektionsverfahren:
  $|b_{k+1} - a_{k+1}| = 0.5 \cdot |b_k - a_k|$
- Im Gegensatz zum Gradientenverfahren benötigt das Verfahren des goldenen Schnitts keine Ableitungen.
- Durch das besondere Teilungsverhältnis des goldenen Schnitts spart man in jedem Schritt eine Funktionsauswertung
- Weitere Verfahren:
  - Sukzessive parabolische Interpolation: Verbesserung der Konvergenzgeschwindigkeit durch Verwendung von Funktionswerten
  - Sekantenverfahren als Alternative zum Newton-Verfahren

Unrestringierte Optimierung für $n \geq 1$

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.*

Idee:
Suche zunächst nach lokalen Minima (aus denen man unter bestimmten Bedingungen globale Extrema erhält).

Ziele:
Notwendige Bedingungen
Hinreichende Bedingungen
Wünschenswert: Notwendige & hinreichende Bedingungen

Numerische Verfahren (meist basierend auf notwendigen Bedingungen)

Rekapitulation: Gradient

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

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

  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.

  Der Gradient steht senkrecht auf Niveaulinien.

Rekapitulation: Hesse-Matrix I

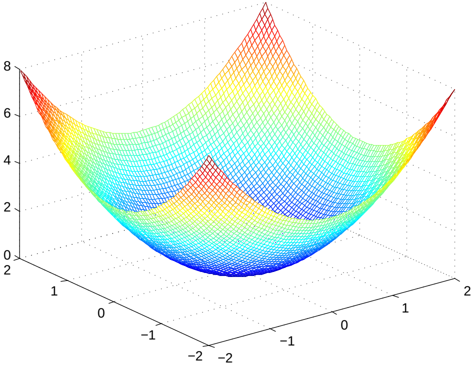
- Die **Hesse-Matrix** einer Funktion f an der Stelle x ist definiert durch

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

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

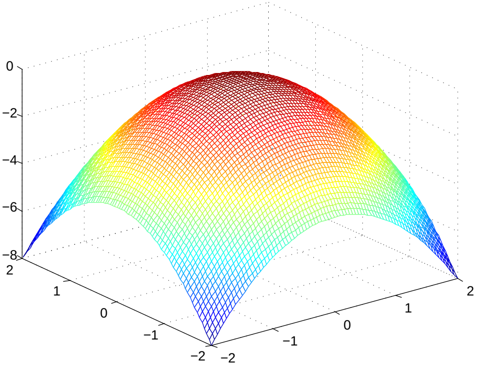
Anschaulich beschreibt die Hesse-Matrix die lokale Krümmung einer Funktion.

Rekapitulation: Hesse-Matrix II



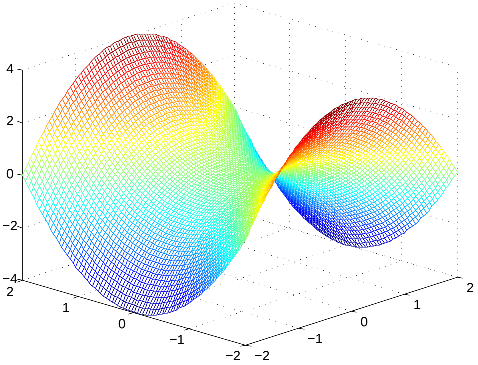
$$f(x_1, x_2) = x_1^2 + x_2^2, \qquad H_f(\hat{x}_1, \hat{x}_2) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$

Rekapitulation: Hesse-Matrix III



$$f(x_1, x_2) = -x_1^2 - x_2^2, \quad H_f(\hat{x}_1, \hat{x}_2) = \begin{pmatrix} -2 & 0 \\ 0 & -2 \end{pmatrix}$$

Rekapitulation: Hesse-Matrix IV



$$f(x_1, x_2) = x_1^2 - x_2^2, \quad H_f(\hat{x}_1, \hat{x}_2) = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix}$$

Rekapitulation: Richtungsableitung

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: Taylorentwicklung I

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

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

Rekapitulation: Taylorentwicklung II

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

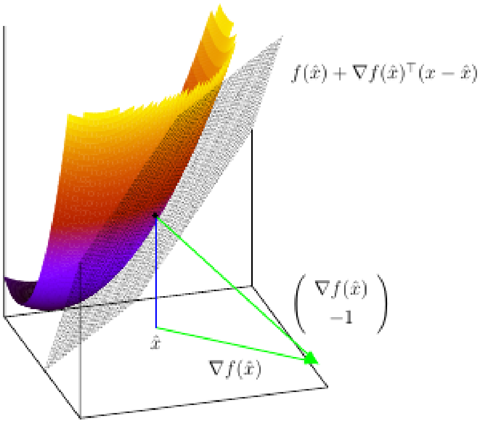
$$f(x) = f(\hat{x}) + \nabla f(\hat{x})^\top (x - \hat{x}) + \frac{1}{2}(x - \hat{x})^\top H_f(\hat{x})(x - \hat{x}) + o(\|x - \hat{x}\|^2).$$

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: 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 Hesse-Matrix

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

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 Hesse-Matrix.

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.

Stationäre Punkte

Stellen $\overline{x}$ mit $\nabla f(\overline{x}) = (0, 0, \ldots, 0)$ heißen **stationäre Punkte**.

Abhängig von den Eigenschaften der Hesse-Matrix erhält man:

- **positiv definit:** lokales Minimum
- **negativ definit:** lokales Maximum
- **indefinit:** Sattelpunkt
- **singulär:** alles möglich (z.B. bei positiv oder negativ semidefiniter Hesse-Matrix)

Konvexe Mengen und Funktionen

- Eine **Menge** $X \subseteq \mathbb{R}^n$ heißt **konvex**, falls zu zwei beliebigen Punkten auch die gesamte Verbindungsstrecke zu x gehört, d. h.
$$\forall x, y \in X : \forall \lambda \in [0,1] : \; x + \lambda(y - x) \in X$$
- Sei $X \subseteq \mathbb{R}^n$ konvex. Eine **Funktion** $f : X \to \mathbb{R}$ heißt **konvex**, falls für zwei beliebige Punkte der Graph unterhalb der Verbindungsstrecke liegt, d. h.
$$\forall x, y \in X : \forall \lambda \in [0,1] : \; f(x + \lambda(y-x)) \leq f(x) + \lambda(f(y) - f(x))$$
- Eine **Funktion** $f : X \to \mathbb{R}$ heißt **streng konvex**, falls
$$\forall x, y \in X : \forall \lambda \in [0,1] : \; f(x + \lambda(y-x)) < f(x) + \lambda(f(y) - f(x))$$
- Eine **Funktion** $f : X \to \mathbb{R}$ heißt **konkav**, falls
$$\forall x, y \in X : \forall \lambda \in [0,1] : \; f(x + \lambda(y-x)) \geq f(x) + \lambda(f(y) - f(x))$$

Aussagen über konvexe Funktionen

Konvexitätskriterium:

Sei $X \subseteq \mathbb{R}^n$ eine konvexe Menge und $f : X \to \mathbb{R}$ zweimal stetig differenzierbar.

Dann gilt:

$$f \text{ ist konvex auf } X \;\Leftrightarrow\; \forall x \in X : H_f(x) \text{ ist positiv semidefinit}$$

Folgerung:

Sei $X \subseteq \mathbb{R}^n$ eine konvexe Menge und $f : X \to \mathbb{R}$ eine konvexe Funktion.

Dann ist jeder stationäre Punkt von f ein lokales und auch ein globales Minimum von f.

Beispiele zu konvexen Mengen und Funktionen

Konvexe Mengen sind beispielsweise:

- $\mathbb{R}^n$
- Alle Polyeder, insbesondere Rechtecke, Quader und höherdimensionale Quader
- Alle kreis- und kugelförmigen Mengen

Folgende Funktionen sind konvex auf $\mathbb{R}^2$:

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

Lineare Funktionen/Optimierungsprobleme

Sei $c \in \mathbb{R}^n$. Dann ist ein unrestringiertes lineares Optimierungsproblem gegeben durch:

$$\min_{x \in \mathbb{R}^n} f(x) = c^T x = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

Wegen $\nabla f(x) = (c_1, c_2, \ldots, c_n)^\top$ haben unrestringierte lineare Optimierungsprobleme keine stationären Punkte.

Quadratische Funktionen/Optimierungspb.

Sei Q eine symmetrische $n \times n$ Matrix und $q \in \mathbb{R}^n$. Dann ist ein unrestringiertes quadratisches Optimierungsproblem geg. durch:

$$\min_{x \in \mathbb{R}^n} f(x)$$

mit

$$f(x) = \frac{1}{2} x^T Q x + q^T x$$

Falls Q positiv semidefinit ist, so ist $f(x)$ konvex.

Zweidimensionaler Fall: $Q = \begin{pmatrix} q_{11} & q_{21} \\ q_{21} & q_{22} \end{pmatrix}$ und $q = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$:

$$\frac{1}{2} x^T Q x + q^T x = \frac{1}{2} q_{11} x_1^2 + q_{21} x_1 x_2 + \frac{1}{2} q_{22} x_2^2 + q_1 x_1 + q_2 x_2$$