Numerische Mathematik

Hochschule
Bonn-Rhein-Sieg

Dr. Marco Hülsmann

Vorlesung WS 2019/20 & SS 2020, Hochschule Bonn-Rhein-Sieg, FB 02

Organisatorisches

- **Modulverantwortlicher:** Prof. Dr. Andreas Priesnitz
- **Dozent:** Dr. Marco Hülsmann

Kontakt

Dr. Marco Hülsmann

Lehrkraft für besondere Aufgaben
Hochschule Bonn-Rhein-Sieg
Fachbereich 02 (Informatik) & Fachbereich 03 (EMT)
Grantham-Allee 20, 53757 Sankt Augustin

- **Büro:** B-112 (FB03, EMT)
- **Telefon:** 02241/865-391
- **Email:** `marco.huelsmann@h-brs.de`

Sprechstunde:

donnerstags, 17-18 Uhr, C-180

Stundenplan

- **Vorlesung:** Do, 13:30-15:00, C-116
  Beginn: 10.10.2019, Ende: 23.01.2020
- **Übung:** Mi, 15:15-16:45, C-116
  Beginn: 10.10.2019, Ende: 23.01.2020

Voraussetzungen

- Mathematische Grundlagen
- Lineare Algebra
- Analysis (auch im Mehrdimensionalen!)
- Programmierung

Sie müssen die grundlegenden Sachen Aussagenlogik, Ableitungsregeln und das Lösen von Linearen Gleichungssystemen beherrschen!!!

Zu weiterführenden Themen wie Eigenwerte, Integralrechnung, Differentialgleichungen wird es Wiederholungen geben!

Übungen

- Die Übungsaufgaben werden montags eine Woche vor der Übung auf LEA hochgeladen.
- Sie brauchen keine Aufgaben bearbeiten und abgeben. Es gibt auch keine Vorleistungstests.
- In der Übungsstunde ist dennoch stets Ihre Mitarbeit gefragt. Die Übung soll keine zweite Vorlesung sein!
- Es wird auch Programmieraufgaben geben.

Programmieraufgaben

- Grundsätzlich können Sie jede Programmiersprache verwenden, die Sie möchten, falls Sie die Programmieraufgaben vorab oder später eigenständig lösen möchten.
- Vorgeführt werden Skripte in den Skriptsprachen *python* und *octave*. Beides können Sie sich online als Paket kostenlos herunterladen.
  Tipp: Falls Sie unter Windows arbeiten, installieren Sie sich zusätzlich *NotePad++*!

Prüfung

- **Prüfungsform:** Mündliche Prüfung (30 min)
- **Zulassung zur Prüfung:** Sie sind automatisch zugelassen.

Weitere Informationen zur mündlichen Prüfung gegen Ende des Semesters!!!

Sie haben das letzte Wort!

Haben Sie noch Fragen zum Ablauf bzw. Wünsche/Erwartungen etc.???

Numerische Mathematik (Numerik)

- ist ein Teilgebiet der Mathematik und beschäftigt sich mit der **approximativen** (näherungsweisen) Lösung von kontinuierlichen mathematischen Problemen
- Hauptziel der Numerik ist die Konstruktion und Analyse von **Algorithmen**
- durch effiziente numerische Verfahren und die heutigen Rechnerarchitekturen ist es möglich, große Probleme, d.h. mit vielen Unbekannten, in akzeptabler Zeit zu lösen
- Zwei Fragestellungen:
  1. Wie kann man den **Rechenaufwand** durch geeignete Verfahren effizient reduzieren?
  2. Da ein Rechner nur mit endlich vielen Zahlen umgehen kann, treten **Rundungsfehler** auf. Wie sind diese Rundungsfehler effizient zu handhaben?

Geschichte der Numerik

- als erster Numeriker überhaupt gilt **Archimedes von Syrakus** (3. Jahrhundert v. Chr.)
  $\Rightarrow$ numerische (approximative) Berechnung der Kreiszahl π
- **Gauß** (17./18. Jahrhundert): bei Gauß-Elimination können erhebliche Rundungsfehler und ein hoher Rechenaufwand auftreten
  $\Rightarrow$ Entwicklung des Gauß-Seidel-Verfahrens, Erhöhung der **Konvergenzgeschwindigkeit**
- Entwicklung von **Rechenmaschinen** (die erste im Jahre 1930 durch **Konrad Zuse**)
- Mathematische und technische Weiterentwicklung im 19. Jahrhundert durch **John von Neumann**
- Heute sind numerische Verfahren in Industrie und Wissenschaft (z.B. **Finite-Elemente-Methode**) Alltagswerkzeug

Überblick I

Überblick I

Überblick I

Überblick I

8 Interpolation

- 8.1 Polynominterpolation: Lagrange-, Newton- und Tschebyscheff-Polynome, Hermite-Interpolation
- 8.2 Diskrete und schnelle Fouriertransformation
- 8.3 Radiale Basisfunktionen
- 8.4 Spline-Interpolation

9 Numerische Lösung gewöhnlicher Differentialgleichungen

- 9.1 Gewöhnliche Differentialgleichungen, Analytische Lösung
- 9.2 Ein- und Mehrschrittverfahren
- 9.3 Konsistenz, Stabilität und Konvergenz
- 9.4 Runge-Kutta-Verfahren
- 9.5 Differentialgleichungen höherer Ordnung

Überblick I

Numerische Fehleranalyse

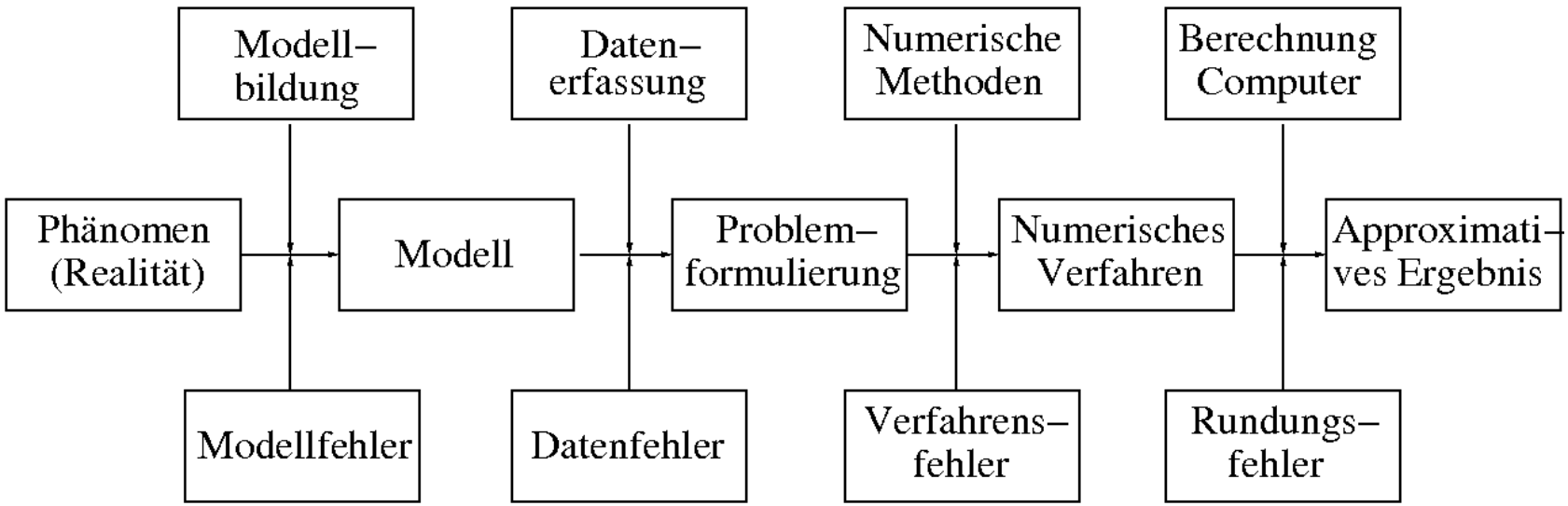
Ziel der Numerik:

möglichst realitätsnahe Modellierung und Simulation von Phänomenen

Unterschiedliche Fehlertypen:

- **Modellfehler:** Modell ist nur vereinfachte Darstellung der Realität
- **Datenfehler:** Meßfehler/Meßungenauigkeiten
- **Verfahrensfehler:** Diskretisierungs-/Abbruchfehler, aufgrund des Ersatzes des Modells durch numerische Approximation
- **Rundungsfehler:** Rechner kann nur mit endlich vielen Zahlen rechnen

Numerische Simulation: Fehlerquellen



- **a-priori-Genauigkeit:** Genauigkeit wird vorgegeben, also von vornherein angefordert
- **a-posteriori-Genauigkeit:** Genauigkeit der Lösung wird hinterher geschätzt (Numerische Fehleranalyse)

Fehlermaße

Tatsächlicher (realer) Wert: x, Näherungswert: $\hat{x}$

Notwendig:

Abstandsmaß $d(x, y)$ zwischen zwei Werten x und y

gegeben durch *Norm* $\|\cdot\|$: $d(x, y) = \|x - y\|$

Fehlerbegriffe:

- **Absoluter Fehler:** $\|x - \hat{x}\|$
- **Relativer Fehler:** $\frac{\|x-\hat{x}\|}{\|x\|}$

Ziel:

Fehler *möglichst klein*, Einführung einer *Toleranz* $\varepsilon > 0$ mit $\|x - \hat{x}\| \leq \varepsilon,\ \varepsilon \leq \mathrm{eps}$ (Maschinengenauigkeit)

Normen

Sei $x = (x_1, ..., x_n) \in \mathbb{R}^n,\ n \in \mathbb{N}_{\geq 2}$:

- **Maximumsnorm:**

$$\|x\|_\infty := \max_{i=1,...,n} |x_i|$$

- **Euklidische Norm:**

$$\|x\|_2 := \sqrt{\sum_{i=1}^{n} x_i^2}$$

- **1-Norm:**

$$\|x\|_1 := \sum_{i=1}^{n} |x_i|$$

Rundung

$\hat{x} = rd(x)$ mit Rundungsfehlern behafteter Wert

Numerische Probleme bei Rundung (Gleitkommazahlen):

- Auslöschung
- Exponentenunterlauf
- Exponentenüberlauf

Rundung geschieht durch bereits aus der Grundschule bekannte Rundung $\tilde{rd}(x)$ (*Aufrunden* ab 5, sonst *Abrunden*)

andere bekannte Rundungsmethode: *Abschneiden*

Gleitkommazahlen

Definition 1.1: (Gleitkommazahl)

Die gerundete Zahl

$$rd(x) = \begin{cases} \operatorname{sign}(x) \cdot 0.\alpha_1\alpha_2...\alpha_t \cdot B^e, & 0 \leq \alpha_{t+1} < \frac{B}{2} \\ \operatorname{sign}(x) \cdot 0.\alpha_1\alpha_2...(\alpha_t + 1) \cdot B^e, & \frac{B}{2} \leq \alpha_{t+1} \leq B-1 \\ \operatorname{sign}(x) \cdot 0.1 \cdot B^{e+1}, & \frac{B}{2} \leq \alpha_{t+1} \leq B-1, \\ & \alpha_1 = ... = \alpha_t = B-1 \end{cases}$$

heißt *Gleitkommazahl*.

Kondition und Fehlerfortpflanzung

Betrachte die Rechenvorschrift $y = \varphi(x)$, wobei x ein Eingabewert und y das Ergebnis ist. Die Funktion

$$\varphi : \mathbb{R} \to \mathbb{R}, \; x \mapsto y = \varphi(x)$$

sei mindestens einmal differenzierbar.

Definition 1.2: (Relative Konditionszahl)

Die Zahl

$$c := \varphi'(x) \cdot \frac{x}{y}$$

heißt *relative Konditionszahl* von y bzgl x (oder φ). Sie beschreibt den Einfluß einer Störung in x auf das Ergebnis y.

Verallgemeinerung des Konditionsbegriffs

Für $x \in \mathbb{R}^n$ und $y \in \mathbb{R}^m$ betrachte die differenzierbare Funktion

$$\varphi : \mathbb{R}^n \to \mathbb{R}^m, \; x \mapsto y = \varphi(x)$$

mit Komponentenfunktionen $\varphi_i : \mathbb{R}^n \to \mathbb{R}$ so, daß
$y_i = \varphi_i(x_1, ..., x_n)$, $i = 1, ..., m$.
Für die relativen Fehler ergibt sich:

$$\varepsilon_{y_i} = \sum_{j=1}^{n} c_{ij} \varepsilon_{x_j}, \; c_{ij} = \frac{x_j}{y_i} \frac{\partial \varphi_i(x)}{\partial x_j}$$

Kondition der arithmetischen Grundoperationen

Sei $z = x * y,\ * \in \{+, -, \times, /\}$

- **Multiplikation:** $|\varepsilon_z| \leq |\varepsilon_x| + |\varepsilon_y|$ (numerisch gutartig/ gut konditioniert)
- **Division:** $|\varepsilon_z| \leq |\varepsilon_x| + |\varepsilon_y|$ (numerisch gutartig/ gut konditioniert)
- **Addition/Subtraktion:**
  - Addition mit gleichen Vorzeichen bzw. Subtraktion mit verschiedenen Vorzeichen:
    $|\varepsilon_z| \leq \max\{|\varepsilon_x|, |\varepsilon_y|\}$ (numerisch gutartig/gut konditioniert)
  - Addition mit verschiedenen Vorzeichen bzw. Subtraktion mit gleichen Vorzeichen:
    Auslöschung möglich (sog. *numerische Katastrophe*)

Matrizen und lineare Gleichungssysteme

Lineare Gleichungssysteme:

Das effiziente Lösen insbesondere großer linearer Gleichungssysteme (LGS) ist eine der wichtigsten mathematischen Problemstellungen!

Entstehung linearer Gleichungssysteme:

Nichtlineare Systeme gehen durch *Linearisierung* in lineare Gleichungssysteme über, kontinuierliche Probleme werden *diskretisiert* (z.B. Differential- und Integralrechnung)

Anwendungsbereiche von linearen Gleichungssystemen

- **Wettervorhersage:** System von zeitabhängigen partiellen Differentialgleichungen zur Bestimmung von Windgeschwindigkeit, Druck, Feuchtigkeit, Temperatur, ...
  - Raum wird durch Gitter approximiert (Meßwerte an den Gitterpunkte)
  - Zeit wird ebenfalls diskretisiert (in *Zeitschritte* unterteilt), pro Zeitschritt wird ein LGS mit 4 Millionen Unbekannten gelöst

  Prognosen müssen schnell erstellt werden $\Leftrightarrow$ Sehr effiziente Löser erforderlich!
- **Klimavorhersagen:** typischer Vorhersagezeitraum: $\sim$ 100 Jahre, Modell zur Simulation der Ozeane erforderlich, Abstand zweier Gitterpunkte in Äquatornähe: $\approx$ 600 km
- **Windkanal:** Simulation von Großraumflugzeugen mit bis zu 15 Millionen Gitterpunkten

Ineffiziente und effiziente Verfahren

Betr. LGS mit n Unbekannten.

- **Cramersche Regel:** Explizite Darstellungsform, Determinantenberechnungen erfolderlich, Rechenzeit steigt proportional zu $n!$; Zahlenbeispiel: $n = 50, n! \approx 10^{50}$
- **Gaußsches Eliminationsverfahren**: Aufwand proportional zu n^3, insbesondere für Bandmatrizen effizient
- **Numerische Näherungsverfahren (Überrelaxationsverfahren)**: Aufwand proportional zu $n^{\frac{3}{2}}$
- **Mehrgitterverfahren:** Aufwand proportional zu n

Aber: Auch die Rechnerleistung ist in den letzten Jahrzehnten erheblich gestiegen!

Matrizen und lineare Gleichungssysteme

Aufgabenstellung:

Sei $A \in \mathbb{R}^{n \times n}$, $n \in \mathbb{N}$, eine quadratische Matrix und

$$f : \mathbb{R}^n \to \mathbb{R}^n, x \mapsto Ax$$

die zugehörige lineare Abbildung. Löse für einen Vektor $b \in \mathbb{R}^n$ *(rechte Seite)* das lineare Gleichungssystem (LGS)

$$f(x) = Ax = b$$

Lösen des LGS:

mit Einsetzungs-/Gleichsetzungsverfahren (Mittelstufe) oder allgemein mit dem *Gaußschen Eliminationsverfahren*

Numerische Fragestellungen

1. Was kann passieren, wenn die rechte Seite gestört ist?
   Wie kann man Konditionsanalysen durchführen?
   Problem: Analytischer Zusammenhang $x = \varphi(A, b)$ nicht gegeben!
2. Andere Möglichkeit zur Bestimmung der Lösung x mithilfe der Inversen A^{-1} (falls existent). Zur Erinnerung:

   $$Ax = b \text{ eind. loesbar } \Leftrightarrow A \text{ invertierbar } \Leftrightarrow \det(A) \neq 0$$

   Problem: Bestimmung von A^{-1} im allgemeinen äußerst rechenaufwendig! Wie kann man mithilfe effizienter numerischer Verfahren den Rechenaufwand signifikant reduzieren?

Wichtige Arten von Matrizen in der Numerik

Bereits die Gestalt der Matrix kann sowohl zu numerische gutartigen Problemstellungen führen als auch den Rechenaufwand erheblich reduzieren. Wir betrachten insbesondere:

- Tridiagonalmatrizen
- Obere-/Untere Dreiecksmatrizen
- Hessenberg-Matrizen
- Bandmatrizen
- Blockdiagonalmatrizen (auch Blocktridiagonalmatrizen)

Matrixnormen und Operatornormen

- Sog. *Normen* ordnen in der Mathematik Objekten gewisse *Größen* zu, beispielsweise einem Vektor aus dem $\mathbb{R}^n$ oder $\mathbb{C}^n$ seine Länge.
- Dies ist auch für Matrizen möglich (sog. *Matrixnormen*). Mithilfe der sog. *Operatornorm* einer Matrix lassen sich numerische Konditionsaussagen treffen.

Normen und Matrixnormen

Definition 2.1: (Norm)

Die Abbildung $||\cdot|| : \mathbb{R}^n \to \mathbb{R}$ heißt *Norm*, falls

(i) $||x|| > 0$ für $x \in \mathbb{R}^n \backslash \{0\}$, $||x|| = 0 \Leftrightarrow x = 0$ (Definitheit)

(ii) $||\alpha x|| = |\alpha| \cdot ||x||$ für $\alpha \in \mathbb{R}$, $x \in \mathbb{R}^n$ (Homogenität)

(iii) $||x + y|| \leq ||x|| + ||y||$ für $x, y \in \mathbb{R}^n$ (Dreiecksungleichung)

Für (zunächst allgemein rechteckige) Matrizen $A \in \mathbb{R}^{m \times n}$, $m, n \in \mathbb{N}$, lassen sich auch Normen für Matrizen definieren. Diese sog. *Matrixnormen* müssen auch die Eigenschaften (i)–(iii) aus Definition 2.1 erfüllen. Weiterhin müssen sie mit einer gegebenen Vektornorm *verträglich* bzw. zu dieser *passend* sein.

Operatornormen

Definition 2.2: (verträgliche/passende Matrixnorm)

Für Matrizen $A \in \mathbb{R}^{m\times n}$, $m, n \in \mathbb{N}$, heißt eine Matrixnorm $||A||$ *mit/zu* den Vektornormen $||\cdot||_a$ im $\mathbb{R}^n$ und $||\cdot||_b$ im $\mathbb{R}^m$ *verträglich/passend*, falls $\forall_{x\in\mathbb{R}^n}\ ||Ax||_b \leq ||A|| \cdot ||x||_a$.
Schreibe für quadratische Matrizen $A \in \mathbb{R}^{n\times n}$ lediglich $\forall_{x\in\mathbb{R}^n}\ ||Ax|| \leq ||A|| \cdot ||x||$, d.h. die Matrixnorm ist *submultiplikativ.*

Definition 2.3: (Operatornorm)

Für quadratische Matrizen $A \in \mathbb{R}^{n\times n}$ ist die zu einer gegebenen Vektornorm $||\cdot||$ gehörige *Operatornorm* (auch *Grenzennorm*) definiert durch $\mathrm{lub}(A) := \max\limits_{x\neq 0} \dfrac{||Ax||}{||x||} = \max\limits_{||x||=1} ||Ax||$.
Sie ist mit ihrer Vektornorm verträglich.

Wichtige Matrixnormen

Korollar 2.1:

$\mathrm{lub}(A)$ ist submultiplikativ.

Wichtige Matrixnormen:

- Zeilensummennorm $||A||_\infty := \max\limits_{1\le i\le n} \sum\limits_{j=1}^{n} |a_{ij}|$

  zur Maximumsnorm gehörige Operatornorm
- Spaltensummennorm $||A||_1 := \max\limits_{1\le j\le n} \sum\limits_{i=1}^{n} |a_{ij}|$

  zur 1-Norm gehörige Operatornorm
- Frobeniusnorm/1-Norm/Schur-Norm $||A||_F := \left(\sum\limits_{i,j=1}^{n} a_{ij}^2 \right)^{\frac{1}{2}}$

  mit der euklidischen Norm verträglich, aber nicht zugehörige Operatornorm

Eigenwerte und Eigenvektoren

Wiederholung aus Lineare Algebra
Definition Eigenwerte und Eigenvektoren:

Sei $A \in \mathbb{R}^{n \times n}$ eine quadratische Matrix. Ein Skalar $\lambda \in \mathbb{R}$ heißt *Eigenwert* von A zum *Eigenvektor* $v \in \mathbb{R}^n \backslash \{0\}$, falls

$$A \cdot v = \lambda \cdot v$$

Es gilt $A \cdot v = \lambda v \Leftrightarrow (A - \lambda \cdot E_n) \cdot v = 0$. Die Matrix $A - \lambda \cdot E_n$ ist nicht regulär (nicht invertierbar). D.h., die Lösung des LGS ist mehrdeutig, und somit gibt es unendlich viele Eigenvektoren. Der Nullvektor ist per Definition als Eigenvektor ausgeschlossen!

Charakteristisches Polynom

Definition 8.29: (Charakteristisches Polynom)

Das Polynom $\chi_A(\lambda) := \det(A - \lambda \cdot E_n)$ heißt *charakteristsiches Polynom* von A.

Die Eigenwerte sind genau die Nullstellen von χ_A. Nach dem Fundamentalsatz der Algebra exisiteren stets m Eigenwerte (mit Vielfachheiten gezählt), die auch komplex sein können. Falls $\lambda \in \mathbb{C}$ Eigenwert einer reellen Matrix A ist, dann auch der dazu konjugiert-komplexe Eigenwert $\bar{\lambda}$.

Wichtige Aussagen zu Eigenwerten/-vektoren

- Die Matrix $A - \lambda \cdot I$ ist singulär. Die Nullstellen von χ_A sind genau die Eigenwerte von A.
- Die Matrizen $A^T A$ und AA^T haben dieselben Eigenwerte. Ist v ein Eigenvektor von $A^T A$, dann ist Av Eigenvektor von AA^T.
- Symmetrische Matrizen haben reelle Eigenwerte und eine Orthonormalbasis aus Eigenvektoren.

Wichtige Aussagen zu Eigenwerten/-vektoren

- Ist λ Eigenwert von A mit Eigenvektor v, dann ist für $k \in \mathbb{N}$ λ^k Eigenwert von A^k mit Eigenvektor v.
- A ist invertierbar $\Leftrightarrow \lambda = 0$ ist kein Eigenwert von A. Falls $\lambda_1, ..., \lambda_n$ die Eigenwerte von A sind, so sind $\frac{1}{\lambda_1}, ..., \frac{1}{\lambda_n}$ die Eigenwerte von A^{-1}.
- Es gilt: $\det(A) = \prod_{i=1}^{n} \lambda_i$ und $\mathrm{sp}(A) = \sum_{i=1}^{n} \lambda_i$.
- Falls die Eigenvektoren von A eine Basis bilden, so ist $D = V^{-1}AV$ eine Diagonalmatrix, auf deren Diagonalen die Eigenwerte von A stehen. Die Matrix V enthält die Eigenvektoren von A. Die Umkehrung gilt auch.

Berechnung von Eigenwerten

Betrachte $A = \begin{pmatrix} 1 & 3 \\ 4 & 2 \end{pmatrix}$. Dann ist $A - \lambda I = \begin{pmatrix} 1-\lambda & 3 \\ 4 & 2-\lambda \end{pmatrix}$.

Dann gilt

$$\begin{aligned} \chi_A(\lambda) = \det(A - \lambda I) &= (1-\lambda)(2-\lambda) - 12 \\ &= 2 - 3\lambda + \lambda^2 - 12 = \lambda^2 - 3\lambda - 10. \end{aligned}$$

Da $\chi_A(\lambda)$ ein quadratisches Polynom in λ ist, können dessen Nullstellen mithilfe der *pq*-Formel berechnet werden:

$$\begin{aligned} \lambda_{1,2} &= \frac{3}{2} \pm \sqrt{\frac{9}{4} + 10} = \frac{3}{2} \pm \frac{7}{2} \\ &\Rightarrow \quad \lambda_1 = 5, \ \lambda_2 = -2, \end{aligned}$$

also sind $\lambda_1 = 5$ und $\lambda_2 = -2$ die Eigenwerte von A.

Berechnung von Eigenvektoren

Zur Bestimmung der Eigenvektoren ist das lineare Gleichungssystem $(A - \lambda I)v = 0$ zu lösen.
Es seien $v_1 := \begin{pmatrix} v_{11} \\ v_{12} \end{pmatrix} \in \mathbb{R}^2$ und $v_2 := \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix} \in \mathbb{R}^2$
Eigenvektoren zu λ_1 bzw. λ_2.
Es gilt:

$$A - \lambda_1 I = A - 5I = \begin{pmatrix} -4 & 3 \\ 4 & -3 \end{pmatrix}, \; A - \lambda_2 I = A + 2I = \begin{pmatrix} 3 & 3 \\ 4 & 4 \end{pmatrix}.$$

Für v_1 muß gelten:

$$-4v_{11} + 3v_{12} = 0 \Leftrightarrow v_{11} = \frac{3}{4} v_{12}.$$

Die Spektralnorm

$$||A||_{sp} := ||A||_2 := \left(\max\{|\lambda| \mid \lambda \text{ Eigenwert von } AA^T\}\right)^{\frac{1}{2}}$$

ist die zur euklidischen Norm gehörige Operatornorm.
Falls A symmetrisch ist, so gilt $AA^T = A^2$, und somit

$$||A||_2 = (\max\{|\lambda| \mid \lambda \text{ Eigenwert von } A\})$$

Falls A symmetrisch positiv definit ist, so sind alle Eigenwerte > 0, und es folgt

$$||A||_2 = \lambda_{\max}$$

(größter Eigenwert von A)

Konditionszahl von Matrizen

Lemma 2.1:

Betr. das eindeutig lösbare LGS $Ax = b$ mit einer Störung Δb.
Dann gilt für den absoluten und relativen Fehler bzgl. x:

(i) $||\Delta x|| \leq ||A^{-1}|| \cdot ||\Delta b||$

(ii) $\frac{||\Delta x||}{||x||} \leq ||A|| \cdot ||A^{-1}|| \cdot \frac{||\Delta b||}{||b||}$

Definition 2.4: (Konditionszahl einer Matrix)

Für $A \in \mathbb{R}^{n \times n}$ invertierbar (regulär) ist

$$\mathrm{cond}(A) := \mathrm{lub}(A) \cdot \mathrm{lub}(A^{-1})$$

die *zu A gehörige Konditionszahl*.

Konditionsabschätzungen bei Störung auf A

Lemma 2.2:

Sei $F \in \mathbb{R}^{n\times n}$ mit $||F|| < 1$. Dann ist $I + F$ regulär, und es gilt

$$||(I+F)^{-1}|| \leq \frac{1}{1-||F||}$$

Satz 2.1:

Sei $A \in \mathbb{R}^{n\times n}$ regulär. Sei weiterhin $B = A(I+F)$ mit $||F|| < 1$. Betr. das LGS $Ax = b$. Sei Δx so, daß $(A+\Delta A)(x+\Delta x) = B(x+\Delta x)$ ($\Delta A = AF$ Störung auf A). Dann gilt $\frac{||\Delta x||}{||x||} \leq \frac{||F||}{1-||F||}$. Falls $\mathrm{cond}(A)\frac{||B-A||}{||A||} < 1$, so gilt

$$\frac{||\Delta x||}{||x||} \leq \frac{\mathrm{cond}(A)}{1-\mathrm{cond}(A)\frac{||B-A||}{||A||}} \frac{||B-A||}{||A||}$$

Konditionsabschätzungen bei Störung auf A und b

Satz 2.2:

Bei Störungen ΔA und Δb, wobei die Störung auf A so klein ist, daß $||A^{-1}|| \cdot ||\Delta A|| < 1$ ist, gelten die folgenden Konditionsabschätzungen:

(i) $$||\Delta x|| \leq \frac{||A^{-1}||}{1 - ||A^{-1}|| \cdot ||\Delta A||} \left(||\Delta b|| + ||A^{-1}|| \cdot ||\Delta A|| \cdot ||b|| \right)$$

(ii) $$\frac{||\Delta x||}{||x||} \leq \frac{\mathrm{cond}(A)}{1 - \mathrm{cond}(A) \frac{||\Delta A||}{||A||}} \left(\frac{||\Delta A||}{||A||} + \frac{||\Delta b||}{||b||} \right) \text{ für } b \neq 0$$

Gauß-Elimination

Seien $n \in \mathbb{N}$. Betrachte ein Lineares Gleichungssystem mit n Gleichungen und n Unbekannten $x_1, ..., x_n \in \mathbb{R}$, ist gegeben durch

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n &= b_2 \\ &\vdots \quad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + ... + a_{nn}x_n &= b_n \end{aligned}$$

Das Zielsystem ist eine obere Dreiecksmatrix

$$R = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ 0 & \ddots & \vdots \\ 0 & 0 & r_{nn} \end{pmatrix}, \; Rx = c, c = \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix}$$

Eliminationsverfahren

Wende eine Folge S von Elementarmatrizen (Matrixoperationen) auf die erweiterte Koeffizientenmatrix $(A|b)$ an: $S \cdot (A|b) = (R|c)$ (Eliminationsverfahren an der Tafel!). Es gibt drei Fälle:

(i) Es existiert keine Lösung. Dann steht in der c-Spalte von $(R|c)$ ein Pivotelement.

(ii) Es existiert genau eine Lösung. Dann enthält R keine Nullzeilen.

(iii) Es existieren unendlich viele verschiedene Lösungen. Dann enthält R Nullzeilen, und man kann $|\bar{P}|$ Variablen frei wählen, wobei $\bar{P}$ die Menge der Nicht-Pivotelemente ist.

Pivotstrategien

Im k-ten Eliminationsschritt verwendet man oft sog. *Pivotstrategien*, um das Verfahren numerisch stabiler zu gestalten:

- **Spalten-Pivotwahl**: Wähle k-te Spalte als Pivotspalte.
  Pivotelement: $\max\limits_{1\leq i\leq n} \left|a_{ik}^{k-1}\right|$
- **Zeilen-Pivotwahl**: Wähle k-te Zeile als Pivotzeile.
  Pivotelement: $\max\limits_{1\leq j\leq n} \left|a_{kj}^{k-1}\right|$
- **Total-Pivotwahl:** Pivotelement: $\max\limits_{1\leq i,j\leq n} \left|a_{ij}^{k-1}\right|$
- **Diagonale Pivotwahl:** Pivotelement: a_{kk}^{k-1}
  (dann sind keine Zeilen- bzw. Spaltenvertauschungen möglich)

Nachteil:

Pivotsuche erhöht den Rechenaufwand!

LR-Zerlegung

Satz 2.3: (LR-Zerlegung)

Jede reguläre Matrix $A \in \mathbb{R}^{n\times n}$ besitzt, falls bei der Gauß-Elimination diagonale Pivotwahl möglich ist, eine eindeutige Zerlegung der Form $A = LR$, wobei

$$L = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \ell_{21} & 1 & 0 & \cdots & 0 \\ \ell_{31} & \ell_{32} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \cdots & 1 \end{pmatrix}, R = \begin{pmatrix} r_{11} & r_{12} & r_{13} & \cdots & r_{1n} \\ 0 & r_{22} & r_{23} & \cdots & r_{2n} \\ 0 & 0 & r_{33} & \cdots & r_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & r_{nn} \end{pmatrix}$$

Allgemeine LR-Zerlegung

Definition 2.5: (Permutationsmatrix)

Eine sog. *Permutationsmatrix* $P \in \mathbb{R}^{n \times n}$ ensteht durch Vertauschen der Zeilen- bzw. Spaltenvektoren der Einheitsmatrix $E_n \in \mathbb{R}^{n \times n}$.

Sind beim Gauß-Verfahren Zeilenvertauschungen erforderlich, so ergibt sich die LR-Zerlegung als

$$PA = LR$$

wobei P eine Permutationsmatrix ist

$\mathcal{O}$-Notation

Definition 2.6: ($\mathcal{O}$-Notation)

Sei $f : \mathbb{N} \to \mathbb{R}_+$ eine Funktion. Dann wird definiert:

(i) $\mathcal{O}(f) := \{g : \mathbb{N} \to \mathbb{R}_+ \mid \exists_{c>0}\, \exists_{n_0 \in \mathbb{N}}\, \forall_{n \geq n_0}\, g(n) \leq c \cdot f(n)\}$, d.h. g wächst asymptotisch nicht schneller als f.

(ii) $o(f) := \{g : \mathbb{N} \to \mathbb{R}_+ \mid \forall_{c>0}\, \exists_{n_0 \in \mathbb{N}}\, \forall_{n \geq n_0}\, g(n) < c \cdot f(n)$, d.h. g wächst asymptotisch langsamer als f, also $\frac{g(n)}{f(n)}$ ist eine Nullfolge.

Rechenaufwand der LR-Zerlegung

Der Rechenaufwand der LR-Zerlegung ist gegeben durch

- $\frac{1}{3}n^3 - \frac{1}{3}n$ Punktoperationen
- $\frac{1}{3}n^3 - \frac{1}{2}n^2 + \frac{1}{6}n$ Strichoperationen

Man sagt, der Aufwand ist *in der Größenordnung von* n^3 oder *in* $\mathcal{O}(n^3)$.

Wiederholung aus Lineare Algebra: Definitheit von Matrizen

Definition: (Definitheit)

Sei A eine $n \times n$-Matrix.

(i) A heißt *positiv definit (pd)*, falls $\forall_{x \in \mathbb{R}^n \setminus \{0\}} \ \langle x, Ax \rangle > 0$.

(ii) A heißt *positiv semidefinit (psd)*, falls $\forall_{x \in \mathbb{R}^n \setminus \{0\}} \ \langle x, Ax \rangle \geq 0$.

(iii) A heißt *negativ (semi-)definit (nd/nsd)*, falls $-A$ positiv (semi-)definit ist.

(iv) A heißt *indefinit*, falls $\exists_{x,y \in \mathbb{R}^n} \ \langle x, Ax \rangle > 0 \ \wedge \ \langle y, Ay \rangle < 0$.

Wiederholung aus Lineare Algebra: Definitheit und Eigenwerte

Im Falle symmetrischer Matrizen gibt es eine Orthonormalbasis aus Eigenvektoren, und alle Eigenwerte sind reell. Falls also alle Eigenwerte echt positiv (negativ) sind, ist die Matrix pd (nd). Ist mindestens ein Eigenwert 0, und sind alle anderen Eigenwerte echt positiv (negativ), so ist die Matrix psd (nsd). Ist mindestens ein Eigenwert echt positiv und alle anderen echt negativ, so ist die Matrix indefinit.
Hat die Matrix Diagonalgestalt, so sind die Eigenwerte direkt auf der Diagonalen ablesbar.

Cholesky-Zerlegung

Satz 2.4: (Cholesky-Zerlegung)

Sei $A \in \mathbb{R}^{n \times n}$ symmetrisch und positiv definit (spd). Dann besitzt A eine eindeutige Zerlegung der Form $A = LL^T$, wobei

$$L = \begin{pmatrix} \ell_{11} & 0 & 0 & \cdots & 0 \\ \ell_{21} & \ell_{22} & 0 & \cdots & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \cdots & \ell_{nn} \end{pmatrix}$$

Der Rechenaufwand einer Cholesky-Zerlegung ist nur halb so hoch wie der einer LR-Zerlegung. Es sind ca. $\frac{1}{6}n^3$ Multiplikationen, $\frac{1}{2}n^2$ Divisionen und n Wurzeloperationen erforderlich. L kann auch aus den Eigenwerten und Eigenvektoren von A berechnet werden

Rundungsfehler bei der Gauß-Elimination

Definition 2.7: (Skalierung einer Matrix)

Die Matrixmultiplikation

$$D_1 A D_2,$$

wobei D_1 und D_2 Diagonalmatrizen sind, heißt *Skalierung* von A.

Man wählt oft D_1 und D_2 so, daß für $\tilde{A} := D_1 A D_2$ gilt:

$$\forall_{i\ell=1,\ldots,n} \sum_{k=1}^{n} |\tilde{a}_{ik}| \approx \sum_{j=1}^{n} |\tilde{a}_{j\ell}|,$$

also so, daß die Summe der Beträge in den Zeilen/Spalten ungefährt dieselbe Größenordnung haben. A und $\tilde{A}$ heißen *äquilibriert*.

Der Fixpunktsatz von Banach

Satz 3.1: (Fixpunktsatz von Banach)

Sei R ein vollständiger metrischer Raum, $\Omega \subseteq R$ mit Metrik $d : R \times R \to \mathbb{R}$. Falls

(i) $f : \Omega \to \mathbb{R}$ eine *Kontraktion* ist, d.h., f auf Ω Lipschitz-stetig ist mit Lipschitz-Konstante $P < 1$, also

$$\exists_{P<1} \, \forall_{x,y\in\Omega} \; d(f(x), f(y)) \leq P \cdot d(x, y)$$

(ii) eine abgeschlossene Teilmenge $K \subseteq \Omega$ existiert mit $f(K) \subseteq K$, d.h. f ist *selbstabbildend*,

dann konvergiert die rekursiv definierte Folge $x^{(n+1)} = f(x^{(n)})$, $x^{(0)} \in K$ (bzw. $x^{(0)} \in \Omega, x^{(1)} \in K$), $n \in \mathbb{N}$ gegen einen Fixpunkt $x^* \in K$ von f, d.h. $f(x^*) = x^*$. Dieser Fixpunkt ist eindeutig.

Fixpunktsatz von Banach: Fehlerabschätzungen

Satz 3.1: (Fixpunktsatz von Banach, Forts.)

Es gelten die Fehlerabschätzungen

$$\begin{aligned} d(x^*, x^{(n)}) &\leq \frac{P^n}{1-P} d(x^{(1)}, x^{(0)}) \\ d(x^*, x^{(n)}) &\leq \frac{P}{1-P} d(x^{(n)}, x^{(n-1)}) \end{aligned}$$

Fixpunktsatz für iterative Lösung von LGSn

Betrachte das Fixpunktproblem $x = Bx + g$ mit zugehöriger Iterationsvorschrift

$$x^{(n+1)} = Bx^{(n)} + g,\ x^{(0)} \in \mathbb{R}^n,\ B \in \mathbb{R}^{n \times n},\ g \in \mathbb{R}^n$$

Für eine Vektornorm gelte $||Bx|| < P||x||,\ P < 1$, für alle $x \in \mathbb{R}^n$. P sei obere Schranke für die Operatornorm $||B|| < 1$. Dann gelten die Aussagen von Satz 3.1 für $K = \Omega = \mathbb{R}^n$ und $f(x) = Bx + g$.

Jacobi-Verfahren

Aus $Ax = b$ wird Fixpunktiteration $x^{(k+1)} = Bx^{(k)} + g$ mit

$$B = D^{-1}(D - A),\ D = \mathrm{diag}(A),\ g = D^{-1}b$$

Komponentenschreibweise:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j\neq i}^{n} a_{ij} x_j^{(k)} \right),\ i = 1, ..., n$$

Die neue Iteration $x^{(k+1)}$ wird ausschließlich mit der alten Iteration $x^{(k)}$ berechnet. Daher handelt es sich um ein *Gesamtschrittverfahren*. Damit es konvergiert, muß $||D^{-1}(D - A)|| < 1$ gelten.

Spektralradius und das Lemma von Schur

Definition 3.1: (Spektralradius)

Die nichtnegative Zahl

$$\rho(A) := \max\{|\lambda| \mid \lambda \text{ Eigenwert von } A\}$$

heißt *Spektralradius* von A.

Lemma 3.1: (Schur)

Jede Matrix $A \in \mathbb{R}^{n\times n}$ läßt sich mithilfe einer orthogonalen Matrix $Q \in \mathbb{R}^{n\times n}$ so zerlegen, daß

$$R = Q^T AQ$$

eine obere Dreiecksmatrix ist.

Matrixnorm und Spektralradius

Satz 3.2:

Sei $||\cdot||$ eine Operatornorm mit passender Vektornorm. Es gilt einerseits für $A \in \mathbb{R}^{n\times n}$

$$\rho(A) \leq ||A||$$

und andererseits

$$\forall_{\varepsilon>0}\ \exists_{||\cdot||_\varepsilon}\ ||A||_\varepsilon \leq \rho(A) + \varepsilon$$

Bei der Konvergenzanalyse kann man daher statt $||B|| < 1$ auch $\rho(B) < 1$, was in vielen Fällen wesentlich leichter ist.

Konvergenz des Jacobi-Verfahrens

Man kann zeigen: Falls A regulär und

$$\max_{i=1,\ldots,n} \sum_{j=1, j\neq i}^{n} \frac{|a_{ij}|}{|a_{ii}|} < 1$$

(sog. *starkes Zeilensummenkriterium*) oder A *diagonaldominant*, d.h.

$$\forall_{i=1,\ldots,n} \; |a_{ii}| > \max_{j=1,\ldots,n,\; j\neq i} |a_{ij}|,$$

dann konvergiert das Jacobi-Verfahren gegen die eindeutige Lösung des LGS.

Gauß-Seidel-Verfahren

Aus $Ax = b$ wird Fixpunktiteration $x^{(k+1)} = Bx^{(k)} + g$ mit

$$B = -(D + C_1)^{-1} C_2,\ g = (D + C_1)^{-1} b$$

Dabei sind $D = \mathrm{diag}(A)$, C_1 der linke untere Teil von A (sonst 0) und C_2 der rechte obere Teil von A (sonst 0). Es gilt $A = D + C_1 + C_2$. Komponentenschreibweise:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right),\ i = 1, ..., n$$

Die neue Iteration $x^{(k+1)}$ wird nicht ausschließlich mit der alten Iteration $x^{(k)}$, sondern auch mit den bereits berechneten Komponenten der neuen Iteration, berechnet. Daher handelt es sich um ein *Einzelschrittverfahren*. Damit es konvergiert, muß $|| - (D + C_1)^{-1} C_2|| = ||(D + C_1)^{-1} C_2|| < 1$ gelten.

Konvergenz des Gauß-Seidel-Verfahrens

Man kann zeigen: Falls A regulär und

$$\max_{i=1,\ldots,n} \sum_{j=1, j\neq i}^{n} \frac{|a_{ij}|}{|a_{ii}|} < 1$$

(sog. *starkes Zeilensummenkriterium*) oder A *diagonaldominant*, d.h.

$$\forall_{i=1,\ldots,n} \; |a_{ii}| > \max_{j=1,\ldots,n,\; j\neq i} |a_{ij}|,$$

dann konvergiert das Gauß-Seidel-Verfahren gegen die eindeutige Lösung des LGS.
I.a. konvergiert das Gauß-Seidel-Verfahren schneller als das Jacobi-Verfahren, da bereits berechnete Komponenten der neuen Iteration, also bessere Näherungen, im gleichen Schritt verwendet werden.

Relaxation

Idee: Gewichte den *Defekt* $v^{(k)}$ bei der Iteration

$$x^{(k+1)} = x^{(k)} + v^{(k)}$$

mit einem Gewicht $0 < \omega < 2$ so, daß der Spektralradius der Iterationsmatrix minimal wird!

- $\omega < 1$: Unterrelaxation
- $\omega > 1$: Überrelaxation

SOR-Verfahren

Gauß-Seidel-Relaxation: Komponentenschreibweise:

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right),$$

Matrixschreibweise:

$$x^{(k+1)} = -(D+\omega C_1)^{-1}(\omega C_2 - (1-\omega)D)x^{(k)} + \omega(D+\omega C_1)^{-1}b$$

Successive **O**ver-**R**elaxation
Man kann zeigen: Wenn A spd ist, dann konvergiert das SOR-Verfahren für alle $0 < \omega < 2$ mit

$$\omega_{opt} = \frac{2}{1+\sqrt{1-\rho(D^{-1}(D-A))}} \in (1,2)$$

Krylow-Methoden

Die sog. *Krylow-Methoden* oder *Krylow-Unterraum-Methoden* gehen auf den russischen Schiffsbauingenieur und Mathematiker Alexei Nikolajewitsch Krylow zurück, welcher die sog. *Krylow-Unterräume* einführte:

$$\mathcal{K}_m := \mathrm{Span}(r^{(0)}, Ar^{(0)}, A^2 r^{(0)}, ..., A^{m-1} r^{(0)}), \ m \in \mathbb{N}$$

Dabei ist $r^{(0)} := b - Ax^{(0)}$ das sog. *Residuum*.
Fast alle Krylow-Methoden finden eine bessere Approximation $x_m \in x_0 + \mathcal{K}_m$, wobei das Residuum $r^{(m)} := b - Ax^{(m)}$ auf einem Unterraum von $\mathcal{K}_m$ orthogonal steht (sog. *Galerkin-Bedingung*).

Methode des steilsten Abstiegs

Anstatt das Lineare Gleichungssystem $Ax = b$ zu lösen, minimiert man die Funktion

$$f(x) = \frac{1}{2}\langle x, Ax\rangle - \langle b, x\rangle$$

mit $\nabla f(x) = Ax - b$
Verfahren:

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} r^{(k)}$$

wobei $\alpha^{(k)} \in \mathbb{R}$ in jeder Iteration so gewählt wird, daß

$$\frac{\partial}{\partial x^{(k)}} f(x^{(k+1)}) = 0$$

Dann geht das Verfahren in jeder Iteration in Richtung des steilsten Abstiegs von f ($-\nabla f(x^{(k)})$).

Methode der konjugierten Richtungen

Die Methode des steilsten Abstiegs sucht wiederholt in denselben Suchrichtungen. Wähle orthogonale Suchrichtungen $d^{(0)}, ..., d^{(k-1)}$ und führe in jeder Suchrichtung genau einen Schritt durch, so daß die entsprechende Komponente i von x (Projektion von x bzgl. $d^{(i)}$) exakt bestimmt wird.
Es ergibt sich

$$\alpha^{(k)} = -\frac{\langle d^{(k)}, e^{(k)} \rangle}{\langle d^{(k)}, d^{(k)} \rangle}$$

wobei $e^{(k)} = x^{(k)} - x^*$ der Fehler der k-ten Iteration ist.

A-Orthogonalität

Definition 3.2: (A-Orthogonalität)

Zwei Vektoren $v, w \in \mathbb{R}^n$ heißen *A-orthogonal* oder *A-konjugiert*, falls $\langle v, Aw \rangle = 0$

Kombiniert man die Methode der konjugierten Richtung mit dem Ansatz der Methode des steilsten Abstiegs, die Funktion f zu minimieren, allerdings in Richtung $d^{(k)}$, so erhält man

$$\alpha^{(k)} = \frac{\langle d^{(k)}, r^{(k)} \rangle}{\langle d^{(k)}, Ad^{(k)} \rangle}$$

und die Suchrichtungen sind somit A-orthogonal.

Methode der konjugierten Gradienten

Die Methode der konjugierte Gradienten (cg) ist die Methode der konjugierten Richtungen, wobei die Suchrichtungen über die Konjugation der Residuen konstruiert werden. Es gilt

$$\langle r^{(k)}, r^{(\ell)} \rangle = \langle \nabla f(x^{(k)}), \nabla f(x^{(\ell)}) \rangle = 0, \ k \neq \ell$$

Die Residuen bzw. die Suchrichtungen erzeugen den Krylow-Raum

$$\begin{aligned} D_k &= \mathrm{Span}(r^{(0)}, ...r^{(k-1)}) = \mathrm{Span}(d^{(0)}, Ad^{(0)}, ..., A^{k-1}d^{(0)}) \\ &= \mathrm{Span}(r^{(0)}, Ar^{(0)}, ..., A^{k-1}r^{(0)}) \end{aligned}$$

und da $AD_k \subseteq D_{k+1}$ und $r^{(k+1)}$ orthogonal zu D_{k+1}, ist $r^{(k+1)}$ A-orthogonal zu D_k. Es gilt $\alpha^{(k)} = \dfrac{\langle r^{(k)}, d^{(k)} \rangle}{\langle d^{(k)}, Ad^{(k)} \rangle}$, wobei die Suchrichtungen $d^{(k)}$ aus den Residuen $r^{(k)}$ und den vorherigen Suchrichtungen rekursiv bestimmt werden.

Rechenaufwand und Kondition von Krylow-Methoden

Rechenaufwand besteht hauptsächlich aus Matrix-Vektor-Multiplikationen, die aus $\mathcal{O}(m)$ sind, wobei m die Anzahl an Matrixeinträgen $\neq 0$ sind. Daher verwendet man Krylow-Methoden insbesondere für sog. *dünnbesetzte Matrizen (sparse matrices)*.
Die Bestimmung der Suchrichtungen führt zu einem hohen Aufwand beim Gram-Schmidt-Verfahren im Falle der Methode der konjugierten Richtungen. Bei der Methode der konjugierten Gradienten ist das Gram-Schmidt-Verfahren aufgrund der A-Orthogonalität der Residuen einfacher.
Im Falle schlecht konditionierter Matrizen verwendet man eine sog. *Präkonditionierung*, d.h. man multipliziert A von links mit einer invertierbaren Matrix P, so daß $\mathrm{cond}(PA) << \mathrm{cond}(A)$ und am besten auch $\rho(PA) << \rho(A)$. Dann kann eine wesentlich bessere Konvergenz erwartet werden.

Newton-Verfahren

Betrachte differenzierbare Funktion $f : [a, b] \to \mathbb{R}$ mit $f(a)f(b) < 0$, d.h., $f(a)$ und $f(b)$ haben unterschiedliches Vorzeichen.
Sei weiterhin $f'(x) \neq 0$ für $x \in [a, b]$. Dann ist f auf $[a, b]$ streng monoton, und nach dem Zwischenwertsatz hat die Gleichung

$$f(x) = 0$$

genau eine Lösung $\xi \in (a, b)$. Näherung durch Nullstellen von Tangenten. Iteration des **Newton-Verfahrens**:

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}$$

mit Startwert $x^{(0)} \in [a, b]$ (extrem startwertabhängig!!!).

Fixpunktsatz von Banach, Spezialfall $\mathbb{R}^1$

Lemma 4.1:

Sei $f : [a, b] \to \mathbb{R}$ mit $f([a, b]) \subseteq [a, b]$, $x^{(0)} \in [a, b]$. Dann ist durch $x^{(k)}$ eine Folge definiert. Falls f auf $[a, b]$ stetig ist, dann existiert ein Fixpunkt von f in $[a, b]$.

Satz 4.2: (Fixpunktsatz von Banach, Spezialfall $\mathbb{R}^1$)

Sei $f : [a, b] \to \mathbb{R}$ stetig mit $f([a, b]) \subseteq [a, b]$, $x^{(0)} \in [a, b]$. Weiterhin gelte auf $[a, b]$ eine Lipschitz-Bedingung mit Lipschitz-Konstante $P < 1$. Dann ist durch $x^{(k)}$ eine Folge $x^{(k)} \subseteq [a, b]$ definiert mit $\lim\limits_{k\to\infty} x^{(k)} = x^*$, wobei x^* der (eindeutige) Fixpunkt von f in $[a, b]$ ist. Es gilt die Fehlerabschätzung $|x^* - x^{(k)}| \leq \frac{P^{(k)}}{1 - P}|x^{(1)} - x^{(0)}|$

Anziehungspunkt und Konvergenzordnung

Definition 4.1: (Anziehungspunkt)

Ein Fixpunkt von $f : [a, b] \to \mathbb{R}$ heißt *Anziehungspunkt* der Iteration $x^{(k+1)} = f(x^{(k)})$, wenn für alle Startwerte $x^{(0)}$ in einer Umgebung von x^* gilt: $\lim_{k\to\infty} x^{(k)} = x^*$

Definition 4.2: (Konvergenzordnung)

Eine Folge $x^{(k)}$ *konvergiert mit Konvergenzordnung* $p \in \mathbb{N}$ gegen einen Grenzwert x^*, falls

$$\forall_{k\in\mathbb{N}} \; ||x^{(k+1)} - x^*|| \leq P||x^{(k)} - x^*||^p, \; P < 1$$

Bei $p = 1$ spricht man von *linearer*, bei $p = 2$ von *quadratischer* und bei $p = 3$ von *kubischer* Konvergenz.

Konvergenz des Newton-Verfahrens

Satz 4.2: (Konvergenz des Newton-Verfahrens)

Sei $f : [a, b] \to \mathbb{R}$ in einer Umgebung von x^* stetig differenzierbar. Dann gilt:

(i) Falls $|f'(x^*)| < 1$, dann ist x^* Anziehungspunkt der Fixpunktiteration.

(ii) Falls $|f'(x^*)| > 1$, dann ist für keinen Startwert $x^{(0)}$ durch $x^{(k+1)} = f(x^{(k)})$ eine gegen x^* konvergente Folge definiert, es sei denn, es ist (zufällig) $x^* = x^{(k)}$ für ein $k \in \mathbb{N}$.

(iii) Falls $|f'(x^*)| = 1$, so kann sowohl Konvergenz als auch Divergenz vorliegen.

Beachte: $g(x) = 0 \Leftrightarrow f(x) = x$ mit $f(x) = g(x) + x,\ f'(x) = g'(x) + 1.$

Bisektionsverfahren

Betrachte stetige Funktion $f : [a, b] \to \mathbb{R}$ mit $f(a)f(b) < 0$, d.h., $f(a)$ und $f(b)$ haben unterschiedliches Vorzeichen (oBdA $f(a) < 0, f(b) > 0$, sonst betrachte $-f$).
Nach dem Zwischenwertsatz hat die Gleichung

$$f(x) = 0$$

mindestens eine Lösung in $x^* \in (a, b)$.
Wähle Intervallmittelpunkt $m := \frac{1}{2}(a + b)$. Falls $f(m) = 0$: fertig!
Falls $f(m) < 0$, wähle m als neue **linke** Intervallgrenze. Falls $f(m) > 0$, wähle m als neue **rechte** Intervallgrenze.
Iterative Fortführung: **Bisektionsverfahren**, R-lineare Konvergenz, d.h. $||x^{(k)} - x^*|| \leq \alpha_k$, wobei α_k eine monoton fallende Nullfolge ist, $k \in \mathbb{N}_0$.

Regula falsi

Es gelten dieselben Voraussetzungen wie beim Bisektionsverfahren. Betrachte Gerade durch die Punkte $(a, f(a))$ und $(b, f(b))$ (Sekante!). Berechne die Nullstelle c dieser Sekante und prüfe, ob diese im Intervall $[a, c]$ oder $[c, b]$ liegt.
Iterative Fortführung: **Regula-falsi-Methode**.

Satz 4.3: (Konvergenz der Regula-falsi-Methode)

Es sei x^* die einzige Nullstelle einer mindestens dreimal stetig differenzierbaren Funktion $f : [a, b] \to \mathbb{R}$, $[a, b] \subseteq \mathbb{R}$. Weiterhin gelte $f'(x^*) \neq 0$ sowie $f''(x^*) \neq 0$. Dann konvergiert die Regula-falsi-Methode linear.

Nichtlineare Gleichungssysteme

Sei $f : D \to \mathbb{R}^n$ stetig differenzierbar, $D \subseteq \mathbb{R}^n$. Löse das nichtlineare Gleichungssystem

$$f(x) = \begin{pmatrix} f_1(x_1, ..., x_n) \\ \vdots \\ f_n(x_1, ..., x_n) \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$$

Dabei sind $f_j : D \to \mathbb{R},\ j = 1, ..., n$ reelle Funktionen (auch *Funktionale* genannt).

Ansatz wie bei Jacobi und Gauß-Seidel: Sei $j = 1, ...n$:

$$f_j(x_1^{(k)}, x_2^{(k)} ..., x_{j-1}^{(k)}, x_j^{(k+1)}, x_{j+1}^{(k)}, ..., x_n^{(k)}) \text{ (Jacobi)}$$

$$f_j(x_1^{(k+1)}, x_2^{(k+1)} ..., x_{j-1}^{(k+1)}, x_j^{(k+1)}, x_{j+1}^{(k)}, ..., x_n^{(k)}) \text{ (Gauss} - \text{Seidel)}$$

Löse also jeweils eine nichtlineare Gleichung in $\mathbb{R}$ mit Unbekannter $x_j^{(k+1)}$, sog. *Jacobi-Newton* bzw. *Gauß-Seidel-Newton* (*lokale Linearisierung*).

Newton-Verfahren im $\mathbb{R}^n$

Globale Linearisierung: Analog zu 1D-Newton:

$$x^{(k+1)} = x^{(k)} - (Df(x^{(k)}))^{-1} f(x^{(k)})$$

mit der Jacobi-Matrix

$$\begin{aligned} Df(x^{(k)}) &= \left(\frac{\partial f_i}{\partial x_j}(x^{(k)}) \right)_{ij=1,\dots,n} \\ &= \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(x^{(k)}) & \cdots & \frac{\partial f_1}{\partial x_n}(x^{(k)}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1}(x^{(k)}) & \cdots & \frac{\partial f_n}{\partial x_n}(x^{(k)}) \end{pmatrix} \in \mathbb{R}^{n \times n} \end{aligned}$$

In der Praxis **niemals** Inversenbestimmung, sondern Lösen des LGS $Df(x^{(k)})v^{(k)} = -f(x^{(k)})$, dann Iteration $x^{(k+1)} = x^{(k)} + v^{(k)}$

Varianten des Newton-Verfahrens

Vereinfachtes Newton-Verfahren:

$$x^{(k+1)} = x^{(k)} - (Df(x^{(0)}))^{-1} f(x^{(k)})$$

Quasi-Newton-Verfahren:
Ersatz von $Df(x^{(k)})$ durch Approximation $H^{(k)}$

Konvergenz des allgemeinen Newton-Verfahrens

Lemma 4.2:

Sei $f : D \to \mathbb{R}^n$, $D \subseteq R^n$ konvex. $Df(x)$ existiere auf D. Weiterhin gebe es ein $\gamma \in \mathbb{R}$ mit

$$\forall_{x,y \in D} \; ||Df(x) - Df(y)|| \leq \gamma ||x - y||$$

Dann gilt:

$$\forall_{x,y \in D} \; ||f(x) - f(y) - Df(y)(x - y)|| \leq \frac{\gamma}{2} ||x - y||^2$$

Zur Erinnerung: D ist konvex, falls für alle $x, y \in D$ die Verbindungsstrecke

$$\{\lambda x + (1 - \lambda)y = y + \lambda(x - y) \;\mid\; 0 \leq \lambda \leq 1\} \subseteq D$$

Der Satz von Newton-Mysovskhik

Satz 4.4: (Newton-Mysovskhik)

Seien $f : D \to \mathbb{R}^n$, $D \subseteq R^n$ konvex, stetig differenzierbar und $Df(x)$ invertierbar für $x \in D$. Weiterhin gebe es Konstanten $\alpha, \beta, \gamma > 0$ so, daß $\frac{\beta\gamma}{2} < 1$ und

(i) $\forall_{x,y \in D} \; ||Df(x) - Df(y)|| \leq \gamma ||x - y||$

(ii) $\forall_{x \in D} \; ||Df(x)^{-1}|| \leq \beta$

(iii) $||Df(x^{(0)})^{-1} f(x^{(0)})|| \leq \alpha$

(iv) $0 < h_0 := \frac{\alpha\beta\gamma}{2} < 1$, $r = \frac{\alpha}{1-h_0}$ so, daß
$\bar{S}_r(x^{(0)}) := \{x \mid ||x - x^{(0)}|| \leq r\} \subseteq D$.

Dann ist die Folge $x^{(k)}, k \in \mathbb{N}$, der Newton-Iteration wohldefiniert und $\forall_{k \in \mathbb{N}} \; x^{(k)} \in \bar{S}_r(x^{(0)})$, und das Newton-Verfahren konvergiert gegen ein $x^* \in \bar{S}_r(x^{(0)})$ mit $f(x^*) = 0$.

Fehlerabschätzung beim Satz von Newton-Mysovskhik

Satz 4.3: (Newton-Mysovskhik, Forts.)

Es gilt die Fehlerabschätzung

$$\forall_{k\in\mathbb{N}} \; ||x^{(k+1)} - x^{(k)}|| \leq \frac{\beta\gamma}{2}||x^{(k)} - x^*||^2$$

Die Konvergenz des Newton-Verfahrens ist also quadratisch.

Der Satz von Newton-Kantorovich

Satz 4.5: (Newton-Kantorovich)

Seien $f : D \to \mathbb{R}^n$, $D \subseteq R^n$ konvex, stetig differenzierbar und $Df(x^{(0)})$ invertierbar für $x \in D$. Weiterhin gebe es Konstanten $\alpha_0, \gamma_0 > 0$ so, daß

(i) $\forall_{x,y \in D} \; ||Df(x^{(0)})^{-1}(Df(x) - Df(y))|| \leq \gamma_0 ||x - y||$

(ii) $||Df(x^{(0)})^{-1} f(x^{(0)})|| \leq \alpha_0$

(iii) $0 < h_0 := \alpha_0 \gamma_0 < \frac{1}{2}$, $r = \frac{1-\sqrt{1-2h_0}}{\gamma_0}$ so, daß
$\bar{S}_r(x^{(0)}) := \{x \mid ||x - x^{(0)}|| \leq r\} \subseteq D$.

Dann ist die Folge $x^{(k)}, k \in \mathbb{N}$, der Newton-Iteration wohldefiniert, $Df(x)$ ist für alle $x \in D$ invertierbar, und $\forall_{k \in \mathbb{N}} \; x^{(k)} \in \bar{S}_r(x^{(0)})$, und das Newton-Verfahren konvergiert gegen ein $x^* \in \bar{S}_r(x^{(0)})$ mit $f(x^*) = 0$. Die Konvergenz ist quadratisch.

Approximation vs. Interpolation

Problemstellung: Gegeben sei eine Reihe von Daten (experimentelle Meßdaten, statistische Evaluationsdaten o.ä), die in Abhängigkeit einer bestimmten Größe gemessen wurden (z.B. Ort, Zeit usw.)
Sei Ω der Eingaberaum (Ort, Zeit usw.) und $\mathcal{D}$ der Datenraum, in dem die Meßdaten entstanden sind.
Ziel: Finde numerisch gut handhabbare (d.h. berechenbare/auswertbare) und je nach Anwendung analytische und möglichst glatte Funktion $f : \Omega \to \mathcal{D}$

- **Approximation:** Alle Daten sollen möglichst gut durch f erfaßt werden, d.h. für alle $x \in \Omega, d = d(x) \in \mathcal{D}$ muß gelten: $||f(x) - d(x)||$ minimal
- **Interpolation:** Alle Daten sollen durch f genau getroffen werden, d.h. für alle $x \in \Omega, d = d(x) \in \mathcal{D}$ muß gelten: $f(x) = d(x)$

Fehlermaße

Sei $g : \Omega \to \mathcal{D}$ diejenige Funktion, die das Zustandekommen der Meßdaten exakt beschreibt. Eine derartige Funktion ist in der Realität oftmals nicht zugänglich. Die Approximationsfunktion f muß so gewählt werden, daß $||g - f||$ minimal wird, z.B. für die folgenden Normen:

(i) $$||g - f||_\infty = \max_{x \in [a,b]} ||g(x) - f(x)||$$

(ii) $$||g - f||_2 = \left(\int_a^b |g(x) - f(x)|^2 \, dx \right)^{\frac{1}{2}}$$

Die Regressionsgerade

Gegeben seien die Punkte $(x_i, y_i) \in \mathbb{R}^2$ für $i = 1, ..., n$, $n \in \mathbb{N}$. Die *Regressionsgerade* ist diejenige Gerade

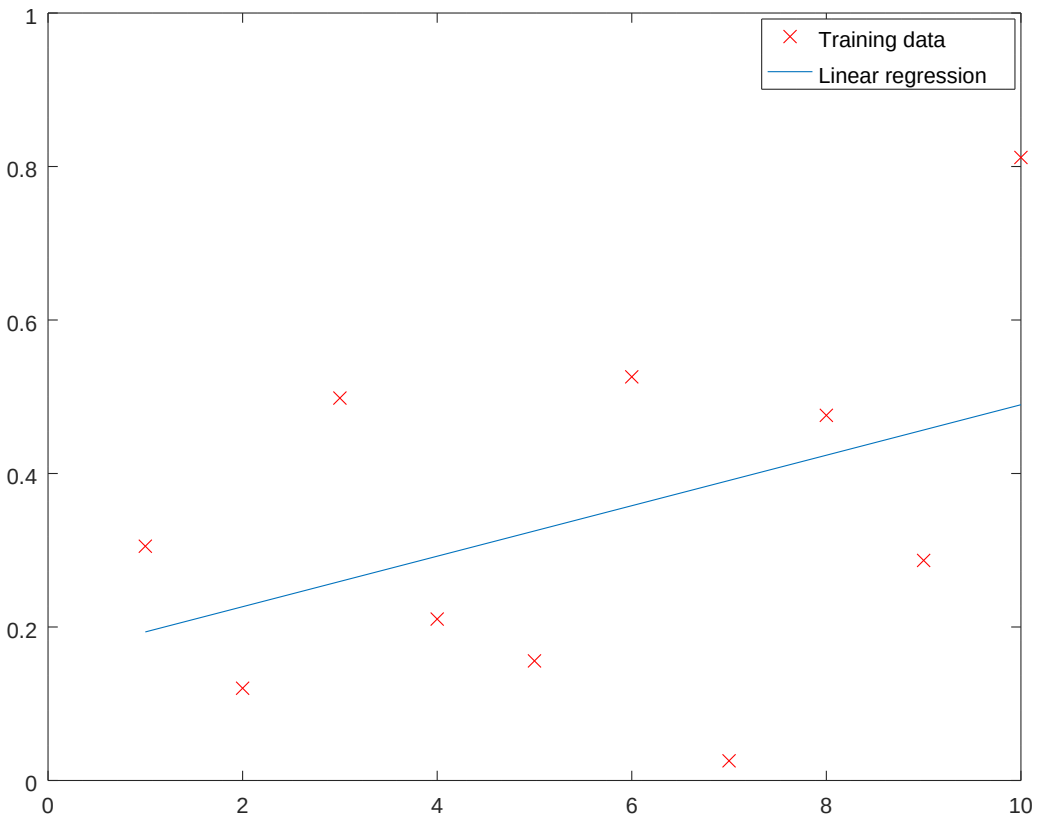
$$g(x) = ax + b$$

welche die Abstände $||y_i - g(x_i)||$ für alle $i = 1, ..., n$ minimiert. Die Koeffizienten der Regressionsgeraden sind gegeben durch

$$\begin{aligned} b &= \bar{y} - a\bar{x}, \\ a &= \frac{\frac{1}{n}\sum_{i=1}^{n} x_i y_i - \bar{x}\bar{y}}{\frac{1}{n}\sum_{i=1}^{n} x_i^2 - \bar{x}^2} = \frac{\mathrm{Cov}(x, y)}{\mathrm{Var}(x)} \end{aligned}$$

Dabei sind $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ und $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ die Mittelwerte der Daten, $\mathrm{Cov}(x, y)$ die *Kovarianz* und $\mathrm{Var}(x)$ deren *Varianz*.

Beispiel einer Regressionsgeraden



Lineare Ausgleichsrechnung

Erweiterung der Regressionsgeraden auf den $\mathbb{R}^n$:
Betrachte m Datenpunkte aus dem $\mathbb{R}^{n+1}$, wobei $m > n$:

$$\begin{aligned}(x_1, y_1) &= (x_{11}, ..., x_{1n}; y_1)\\ (x_2, y_2) &= (x_{21}, ..., x_{2n}; y_2)\\ &\vdots\\ (x_m, y_m) &= (x_{m1}, ..., x_{mn}; y_m)\end{aligned}$$

und das überbestimmte LGS $A\alpha = y$, wobei $A \in \mathbb{R}^{m\times n}$ die Matrix ist, die die x_{ij} enthält, $y = (y_1, ..., y_m) \in \mathbb{R}^m$,
$\alpha = (\alpha_1, ..., \alpha_n) \in \mathbb{R}^n$ (hat i.a. keine Lösung)

Lineares Ausgleichsproblem

Löse das Minimierungsproblem

$$\min_{\alpha \in \mathbb{R}^n} ||y - A\alpha||$$

Suche $\alpha \in \mathbb{R}^n$ so, daß der Abstand zwischen y und $A\alpha$, also die Norm des Residuums, minimal wird.

- Euklidische Norm $||\cdot||_2$: **Methode der Kleinsten Quadrate**
- Maximumsnorm $||\cdot||_\infty$: **Diskretes Tschebyscheff-Problem**

Normalengleichungen

Definition 5.1: (Normalengleichungen)

Seien $A \in \mathbb{R}^{m \times n}, m > n, y \in \mathbb{R}^m, \alpha \in \mathbb{R}^n$. Die durch das LGS

$$A^T A\alpha = A^T y,\ A^T A \in \mathbb{R}^{n \times n},\ A^T y \in \mathbb{R}^n$$

gegebenen linearen Gleichungen heißen *Normalengleichungen*.

Löst man das überbestimmte LGS $(e\ A)\bar{\alpha} = y$ mit $e = (1, ..., 1)^T \in \mathbb{R}^m$ und $\bar{\alpha} = (\bar{\alpha}_0, \bar{\alpha}_1, ..., \bar{\alpha}_n)$, so werden die Datenpunkte durch eine n-dimensionale Regressionshyperebene getrennt.

Pseudoinverse

Lemma 5.1:

Es gilt $\forall_{y \in \mathrm{Bild}(A)} \; \exists!_{x_1 \in \ker(A)^\perp} \; Ax_1 = y$, d.h., es gibt eine wohldefinierte lineare Abbildung

$$f : \mathrm{Bild}(A) \to \mathbb{R}^n, \; \forall_{y \in \mathrm{Bild}(A)} \; Af(y) = y, \; f(y) \in \ker(A)^\perp$$

Man kann also nach einem Urbild zu $y \in \mathbb{R}^m$ suchen.

Definition 5.2: (Pseudoinverse)

Die zusammengesetzte Abbildung $f \circ \bar{P} : \mathbb{R}^m \to \mathbb{R}^n, \; y \mapsto f(\bar{P}(y))$, wobei $\bar{P} \in \mathbb{R}^{m \times m}$ diejenige Matrix ist, die $\mathbb{R}^m$ auf $\mathrm{Bild}(A)$ projiziert (wg. $\bar{P}y \in \mathrm{Bild}(A)$ und Lemma 5.1 ist die Abbildung wohldefiniert und linear), ist eine $n \times m$-Matrix und heißt *Pseudoinverse* von A. Bez.: $A^+ y := f(\bar{P}(y))$

Eigenschaften der Pseudoinversen

Im Spezialfall $A \in \mathbb{R}^{n \times n}$ stimmt die Pseudoinverse mit der bekannten Inversen A^{-1} überein.

Satz 5.1: (Eigenschaften der Pseudoinversen)

Die Pseudoinverse A^+ einer $m \times n$-Matrix A hat die folgenden Eigenschaften:

(i) A^+A und AA^+ sind symmetrisch

(ii) $AA^+A = A$ und $A^+AA^+ = A^+$

Satz 5.2: (Eindeutigkeit der Pseudoinversen)

Eine $n \times m$-Matrix ist mit den Eigenschaften aus Satz 5.1 eindeutig bestimmt. A^+ wird dann auch *Moore-Penrose-Inverse* genannt.

Berechnung der Pseudoinversen

Mithilfe der Eigenschaften aus Satz 5.1 läßt sich eine Pseudoinverse eindeutig ermitteln.

Spezialfälle:

- $A^T A$ invertierbar $\Rightarrow \; A^+ = (A^T A)^{-1} A^T$
- AA^T invertierbar $\Rightarrow \; A^+ = A^T (AA^T)^{-1}$

$\alpha^* = A^+ y$ ist Ausgleichslösung des überbestimmten LGS $A\alpha = y$, d.h., α^* minimiert $||y - A\alpha||$

QR-Zerlegung

Sei $A \in \mathbb{R}^{m \times n}$. Es ist dabei völlig irrelevant, ob $m > n$, $m < n$ oder $m = n$ gilt.
Die *QR-Zerlegung* ist eine Zerlegung der Form $A = Q \cdot R$, wobei $Q \in \mathbb{R}^{m \times m}$ eine orthogonale Matrix ($Q^T Q = E_m$) und R eine obere Dreiecksmatrix ist.
Anwendung für überbestimmte Gleichungssysteme:
Betr. LGS $Ax = QRx = b$. Löse zunächst $Qz = b$, berechne also $z = Q^T b$, und anschließend $Rx = z$ durch Rückwärtseinsetzen.
Weitere wichtige Anwendung: Eigenwertprobleme
Die QR-Zerlegung ist ein numerisch äußerst stabiles Verfahren!
Eine QR-Zerlegung kann z.B. durch **Householder-Spiegelungen** oder **Givens-Rotationen** realisiert werden.

Householder-Spiegelungen

Definition 5.3: (Householder-Matrix)

Die orthogonale Matrix

$$H = E_m - 2nn^T,$$

welche eine eine Spiegelung um eine Ebene mit Normalenvektor n beschreibt, heißt *Householder-Matrix*. Dabei ist $E_m - nn^T$ eine Projektionsmatrix auf die Ebene (orthogonale Projektion).

Die Bestimmung der Matrix Q ergibt sich als Produkt von Householdermatrizen H_i:

$$Q^T = \prod_{i=1}^{n-1} H_{n-i} = H_{n-1} \cdots H_1, \; Q^T A = R$$

d.h., die Linksmultiplikation von Householdermatrizen bewirkt das Zusatandekommen einer oberen Dreiecksmatrix.

Projektion und Spiegelung

Lemma 5.2:

(i) $P = E_m - nn^T$ beschreibt tatsächlich eine Projektion auf die Ebene mit Normalenvektor $n \in \mathbb{R}^m$. Dementsprechend ist H tatsächlich eine Spiegelung an dieser Ebene.

(ii) H ist orthogonal.

Wähle $n := a_{*1} + \operatorname{sign}(a_{11})||a_{*1}||e_1$, wobei a_{*1} der erste Spaltenvektor von A und e_1 der erste Einheitsvektor ist. In diesem Fall: $\operatorname{sign}(0) = 1$. Dies bewirkt, daß die erste Spalte von A zu einem Vielfachen des ersten Einheitsvektors wird! Beispiel an der Tafel!

Givens-Rotationen

Definition 5.4: (Givens-Matrix)

Eine Matrix $G \in \mathbb{R}^{m \times m}$ heißt *Givens-Matrix*, falls sie eine Drehung um den Winkel $\varphi \in [0, 2\pi)$ beschreibt. also falls sie die $2 \times 2 -$ *Matrix*

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix} = \begin{pmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{pmatrix}$$

enthält, mit $c = \frac{a}{\sqrt{a^2+b^2}}$ und $s = \frac{b}{\sqrt{a^2+b^2}}$, falls $\begin{pmatrix} a \\ b \end{pmatrix}$ der Vektor ist, der mit der x-Achse den Winkel φ einnimmt. Wenn man ihn um $-\varphi$ dreht, landet man auf der x-Achse, d.h. eine Komponente des Vektors wird zu Null.

Eine Givens-Matrix dreht so, daß genau ein Eintrag von A auf Null gesetzt wird. Beispiele an der Tafel!

Eigenwerte in der Informatik: Bildverarbeitung

Bei Gesichtserkennungen in der Bildverarbeitung werden relevante Informationen zu einer bestimmten Klasse von bekannten Objekten ermittelt (Mustererkennung). Dabei werden Bilder als sog. *Pixelvektoren* des $\mathbb{R}^n$ aufgefaßt. Die Bilder sind also Punkte $b_1, ..., b_M \in \mathbb{R}^n$, wobei $n \in \mathbb{N}$ die Anzahl an Pixeln und $m \in \mathbb{M}$ die Anzahl an Bildern ist.

Frage:

Kann man den Raum der Gesichtsbilder als Unterraum des Bilderraums repräsentieren und das Auftreten neuer/ähnlicher Gesichter erkennen?

Antwort:

Ja, und zwar durch die Bestimmung der Kovarianzmatrix der Gesichtsbildvektoren (Abweichungen vom Durchschnittsbild). Die Eigenvektoren dieser Matrix spannen den gesuchten Unterraum auf.

Eigenwerte in der Informatik: Page-Ranking

Das sog. *PageRank-Verfahren* nach Larry Page modelliert das Verhalten eines Internetbenutzers, der über Links verbundene Webseiten aufruft.

Hauptanwendung: Seitenbewertung für Google.

Prinzip: Das Gewicht einer Seite (PageRank) soll umso höher sein, je mehr Seiten mit möglichst hohem Eigengewicht auf die Seite verlinken.

Definition: PageRank einer Webseite a:

$$PR(a) = \frac{q}{T} + (1-q)\sum_{i=1}^{n} \frac{PR(p_i)}{L(p_i)},$$

- $p_1, ..., p_n$: Anzahl Seiten, die auf a verlinken,
- T: Gesamtzanzahl an Seiten
- $L(p)$: Anzahl Seiten, auf die die Webseite p verlinkt,
- $q \in [0,1]$: Wahrscheinlichkeit eines zufälligen Seitenwechsels,
- $1-q \in [0,1]$: Wahrscheinlichkeit der Benutzung eines Links.

Google-Matrix

- **Normierte Adjazenzmatrix** A: $a_{ij} = 1$ falls Seite i und j durch Link verbunden sind, sonst 0,
- **Linkmatrix** L**:**

$$L_{ij} := \begin{cases} \frac{1}{L(p_i)}, & a_{ij} = 1, \\ 0 & \text{sonst} \end{cases},$$

- **Dangling-Nodes-Vektor** w**:**

$$w_i = \begin{cases} 1, & L(p_i) = 0 \\ 0 & \text{sonst} \end{cases},$$

- **Einsvektor** e: $e = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbb{R}^T$.

Google-Matrix: Eigenwertproblem des PageRanks-Verfahrens

$$P := (1-d)\left(L + \frac{1}{T} w e^T\right) + \frac{d}{T} e e^T.$$

sog. **Google-Matrix.**
Der **PageRank-Vektor** PR ist Eigenvektor von P^T zum Eigenwert 1:

$$P^T(PR) = PR.$$

Eigenwertproblem

Bestimme zu gegebener quadratischen Matrix $A \in \mathbb{R}^{n \times n}$ ein Skalar $\lambda \in \mathbb{C}^n$ und einen Vektor $v \in \mathbb{C}^n \backslash \{0\}$ so, daß

$$A \cdot v = \lambda \cdot v$$

Falls A symmetrisch ist, gilt $\lambda \in \mathbb{R}$ und $v \in \mathbb{R}^n \backslash \{0\}$, und die Eigenvektoren bilden eine Orthonormalbasis des $\mathbb{R}^n$.
Da es unendlich viele Eigenvektoren gibt, wählt man diejenigen mit $||v|| = 1$ (Lösung des zugehörigen LGS mehrdeutig), falls nötig mit Gram-Schmidt, z.B. sind die Eigenvektoren der Einheitsmatrix zum Eigenwert $\lambda = 1$ alle Vektoren aus $\mathbb{R}^n \backslash \{0\}$.

Gerschgorin-Kreise

Satz 6.1: (Gerschgorin)

Sei $A \in \mathbb{C}^{n \times n}$ eine quadratische Matrix. Dann gilt:

(i) Jeder Eigenwert von A liegt in der Vereinigung der *Zeilenkreise*

$$\left\{ z \in \mathbb{C} \;\middle|\; |z - a_{ii}| \leq \sum_{j=1,\ j \neq i}^{n} |a_{ij}|,\ i = 1, \ldots, n \right\}$$

(ii) Jeder Eigenwert von A liegt in der Vereinigung der *Spaltenkreise*

$$\left\{ z \in \mathbb{C} \;\middle|\; |z - a_{jj}| \leq \sum_{i=1,\ i \neq j}^{n} |a_{ij}|,\ j = 1, \ldots, n \right\}$$

Numerische Bedeutung der Gerschgorin-Kreise

Ist A also diagonaldominant, so sind die Radien im Vergleich zum Mittelpunkt klein. Überschneiden sich die Kreise nicht oder nur geringfügig, so liegen die Eigenwerte isoliert vor, und das Problem wird gut konditioniert sein. Überschneiden sich die Kreise, so wird das Problem eher schlecht konditioniert sein.

Einfache Vektoriteration

Sei $A \in \mathbb{C}^{n\times n}$ diagonalisierbar mit

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$$

also λ_1 sei der sog. *dominante* Eigenwert.
Betrachte die sog. *einfache Vektoriteration*

$$x^{(k+1)} = Ax^{(k)} = A^{k+1}x^{(0)},\ k \in \mathbb{N}_0,\ x^{(0)} \in \mathbb{C}^n$$

Die Verhältnisse der Komponenten $\frac{x_j^{(k+1)}}{x_j^{(k)}}, j = 1, ..., n$, konvergieren für $k \to \infty$ gegen λ_1.

Verbesserte Vektoriteration nach von Mises

Betrachte Normierung der $x^{(k)}$:

$$z^{(k)} := \frac{x^{(k)}}{||x^{(k)}||}, \; x^{(k+1)} := Az^{(k)}, k = 0, 1, 2, ...$$

Dann konvergiert $||x^{(k)}||$ für $k \to \infty$ gegen $|\lambda_1|$ und $z^{(k)}$ gegen den zugehörigen Eigenvektor v_1.
Berechnung von λ_1 über *Rayleigh-Quotient*

$$\lambda_1 = \frac{\langle v_1, Av_1 \rangle}{\langle v_1, v_1 \rangle}$$

Deflation

Nachteil der Vektoriteration: Nur Bestimmung des betraglich größten Eigenwerts möglich! Ausweg: sog. *Deflation*, d.h. Projektion in den Orthogonalraum zu $\mathrm{Span}(v_1)$:

$$x^{(k+1)} = Ax^{(k)} - \langle Ax^{(k)}, v_1 \rangle \cdot v_1$$

Die Vektoriteration konvergiert dann gegen den betraglich zweitgrößten Eigenwert λ_2.

Inverse Vektoriteration nach Wielandt

Falls λ eine bekannte gute Näherung für einen bestimmten Eigenwert λ_j ist, also falls $\forall_{k=1,\ldots,n,\ k\neq j}|\lambda_j - \lambda| << |\lambda_k - \lambda|$. Betrachte dann die Vektoritation

$$x^{(k+1)} = (A - \lambda I)^{-1} x^{(k)},\ \ k = 0, 1, 2\ldots$$

Diese konvergiert gegen den betraglich größten Eigenwert von $(A - \lambda I)^{-1}$, also gegen $\frac{1}{\lambda_j - \lambda}$.

Hessenberg-Matrizen

Definition 6.1: (Hessenberg-Matrix)

Eine *Hessenberg-Matrix* ist eine obere Dreiecksmatrix, die auf der Nebendiagonalen direkt unterhalb der Diagonalen noch voll besetzt sein kann.

Man kann durch Givens-Rotationen oder Gauß-Elimination mit geeigneter Pivotsuche (ähnlich der LR-Zerlegung) jede Matrix auf Hessenberg-Form bringen!

Bestimmung der Eigenwerte von Hessenberg-Matrizen

Sei B eine Hessenberg-Matrix. Betrachte die Lösung des Linearen Gleichungssystems

$$(B - \lambda I)x = \alpha e_1$$

wobei $e_1 = (1, 0, ..., 0)^T \in \mathbb{R}^n$ und $\alpha = \alpha(\lambda) \in \mathbb{R}$.
Die Cramersche Regel liefert, daß $\alpha(\lambda)$ ein Vielfaches des charakteristischen Polynoms $\det(B - \lambda I)$ ist. Berechne die Nullstellen von $\alpha(\lambda)$ mithilfe des Newton-Verfahrens!

QR-Verfahren

Bestimme mithilfe der Methoden aus Abschnitt 5.3 (Householder/Givens) eine QR-Zerlegung der Matrix A, deren Eigenwerte zu bestimmen sind:

$$A^{(0)} = A = QR = Q^{(0)}R^{(0)}$$

Bestimme dann $A^{(1)} := R^{(0)}Q^{(0)}$ und wieder eine neue QR-Zerlegung $A^{(1)} = Q^{(1)}R^{(1)}$. Allgemein konvergiert das Verfahren

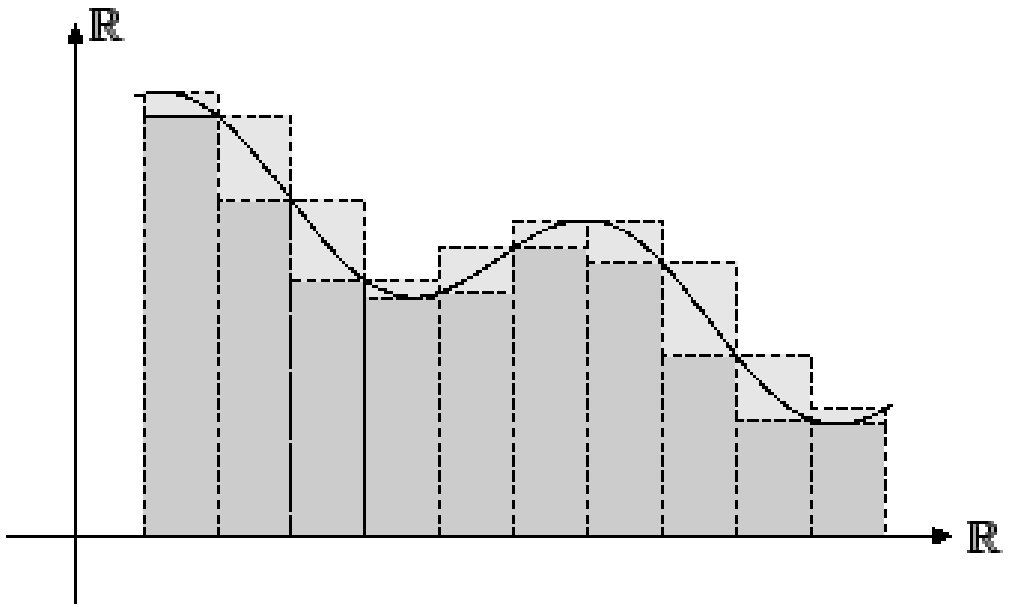
$$A^{(i)} = Q^{(i)}R^{(i)}, \; A^{(i+1)} = R^{(i)}Q^{(i)}$$

gegen eine obere Dreiecksmatrix, auf deren Diagonale alle Eigenwerte von A stehen. Die Produktmatrix $P^{(i)} := Q^{(0)} \dots Q^{(i)}$ konvergiert gegen die Matrix, in deren Spalten die Eigenvektoren von A stehen.

Wiederholung: Integralrechnung

- **Motivation:** Flächenberechnung (durch Kurven begrenzte Flächen)
- **Wichtige Integralbegriffe:** Riemann-Integral, Lebesgue-Integral (stochastische Anwendungen: Verallgemeinerung des Riemann-Integrals)
- **Treppenfunktionen:** Flächen unter einer Kurve werden mit Treppenfunktionen approximiert, die Summe der enstehenden Rechtecksflächen ist eine Approximation für das Integral
- **Integral:** Grenzwert der Summe der Rechtecksflächen für Anzahl Stützstellen gegen ∞ (Anzahl Stützstellen legt die Feinheit der Zerlegung des Intervalls fest)
- **Integralrechnung:** 'Umkehrung der Differentialrechnung'

Approximation durch Treppenfunktionen



Stammfunktion

Definition 7.1: (Stammfunktion)

Sei $f : \mathbb{R} \to \mathbb{R}$ stetig. Eine auf $D(f)$ differenzierbare Funktion $F : \mathbb{R} \to \mathbb{R}$, $D(F) \supseteq D(f)$, heißt *Stammfunktion* von f, falls

$$\forall_{x \in D(f)} \; F'(x) = f(x)$$

Korollar 7.1:

Es gibt unendlich viele Stammfunktionen. Diese unterscheiden sich lediglich um eine additive Konstante $c \in \mathbb{R}$.

Unbestimmtes Integral

Definition 7.2: (Unbestimmtes Integral)

Die Menge aller Stammfunktionen von $f : \mathbb{R} \to \mathbb{R}$ heißt *unbestimmtes Integral* von f:

$$\int f(x)\ dx = F(x) + c,\ F'(x) = f(x),\ c \in \mathbb{R}$$

Satz 7.1: (Linearität des Integrals)

Das unbestimmte Integral ist linear, d.h., für $f, g : \mathbb{R} \to \mathbb{R}$ stetig und $\alpha, \beta \in \mathbb{R}$ gilt:

$$\int \alpha f(x) + \beta g(x)\ dx = \alpha \int f(x)\ dx + \beta \int g(x)\ dx$$

Integrationsrechenregeln

Satz 7.2: (Partielle Integration)

Seien $f, g : \mathbb{R} \to \mathbb{R}$ stetig differenzierbar. Dann gilt:

$$\int f'(x)g(x)\ dx = f(x)g(x) - \int f(x)g'(x)\ dx$$

Satz 7.3: (Substitutionsregel)

Seien $g : [a, b] \to [c, d]$ stetig differenzierbar und $f : [c, d] \to \mathbb{R}$ stetig. Dann gilt:

$$\int f(g(x))g'(x)\ dx = \int f(z)\ dz \Big|_{z=g(x)}$$

Vorgehensweisen zur Integralberechnung

- **Partielle Integration:** Entscheide, welche Funktion aufzuleiten ist und welche abzuleiten. Multipliziere Stammfunktion der aufzuleitende Funktion mit der anderen Funktion. Dann kommt ein Minuszeichen und das Integral über die Stammfunktion der aufzuleitenden Funktion mal die Ableitung der abzuleitenden Funktion.
- **Substitution:** Substituiere $z = g(x)$. Bilde $\frac{dz}{dx} = g'(x) \Rightarrow dx = \frac{1}{g'(x)} dz$. Ersetze im Integral $g(x)$ durch z und dx durch $\frac{1}{g'(x)} dz$.

Integration von Potenzreihen

Satz 7.4: (Integration von Potenzreihen)

Sei $f(x) := \sum_{n=0}^{\infty} a_n (x - x_0)^n$ eine Potenzreihe um den Entwicklungspunkt x_0. R sei ihr Konvergenzradius. Dann gilt

$$\int f(x)\, dx = \sum_{n=0}^{\infty} \frac{a_n}{n+1} (x - x_0)^{n+1}$$

Diese Potenzreihe hat ebenfalls Konvergenzradius R.

Integration gebrochen rationaler Funktionen

Sei $f(x) = \frac{Z(x)}{N(x)}$, wobei $Z(x)$ und $N(x)$ Polynome sind.

- Falls $\deg(Z) < \deg(N)$, so ist eine Partialbruchzerlegung durchzuführen. Wir betrachten hier nur den Fall, daß $N(x)$ nur reelle Nullstellen hat. Dann besteht die Partialbruchzerlegung aus Termen der Form $\frac{A}{x+d}, A, d \in \mathbb{R}$, und es gilt
$$\int \frac{A}{x+d}\, dx = \log|x+d| + c$$
- Falls $\deg(Z) \geq \deg(N)$, so ist zunächst eine Polynomdivision durchzuführen. Auf das Restpolynom ist dann wieder eine Partialbruchzerlegung anzuwenden, und es muß wieder wie oben beschrieben vorgegangen werden.

Treppenfunktionen

Definition 7.3: (Treppenfunktion)

Eine Funktion $T : [a, b] \to \mathbb{R}$ heißt *Treppenfunktion*, falls

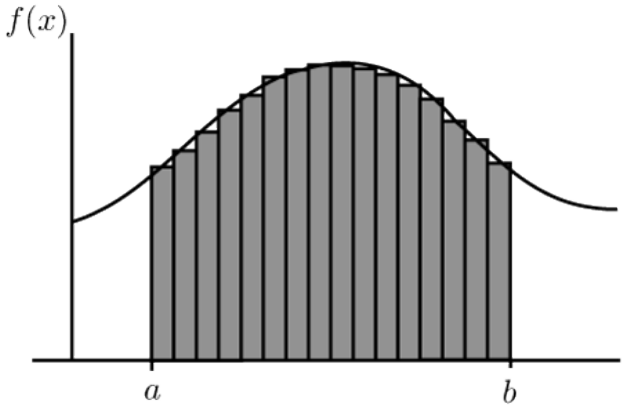
$$\exists_{x_0,x_1,\ldots,x_m \in [a,b]} \; a = x_0 < x_1 < \ldots < x_m \wedge \forall_{i=1,\ldots,m} \; T|_{(x_{i-1},x_i)} \text{ konstant}$$

($m \in \mathbb{N}$). Die Aufteilung des Intervalls $[a, b]$ in obige Teilintervalle heißt *Zerlegung* Z_m der *Feinheit*

$$L(Z_m) := \max_{i=1,\ldots,m} (x_i - x_{i-1})$$

(Länge des größten Teilintervalls).

Integral als Flächenapproximation



Ziel: Approximation der Fläche, die eine Funktion $f : [a, b] \to \mathbb{R}$ mit der $x-$Achse einschließt, durch Integral einer Treppenfunktion, also die Summe der Rechtecksflächen. Dabei sollten sowohl die Wahl der Stützstellen $x_0, ..., x_m$ als auch die Höhe a_i einer Treppenstufe $i \in \{1, ..., m\}$ irrelevant sein (es sollte lediglich $\min(f(x_{i-1}), f(x_i)) \leq a_i \leq \max(f(x_{i-1}), f(x_i))$ für $i = 1, ..., m$ gelten).

Bestimmtes Integral einer Treppenfunktion

Definition 7.4: (Bestimmtes Integral einer Treppenfunktion)

Seien $T : [a, b] \to \mathbb{R}$ eine Treppenfunktion, $m \in \mathbb{N}$, und es gelte

$$\forall_{i=1,\ldots,m}\ T(x) =: a_i,\ (x \in (x_{i-1}, x_i))$$

Dann heißt

$$\int_a^b T(x)\ dx := \sum_{i=1}^{m} a_i(x_i - x_{i-1})$$

bestimmtes Integral der Treppenfunktion T auf dem Intervall $[a, b]$ (Sprechweise: 'von a nach/bis b').

Das bestimmte (Riemann-)Integral

Definition 7.5: (Riemann-integrierbar, bestimmtes Integral)

Eine Funktion $f : [a, b] \to \mathbb{R}$ heißt *Riemann-integrierbar* auf dem Intervall $[a, b]$, falls für jede Folge von Zerlegungen Z_n mit $\lim_{n\to\infty} L(Z_n) = 0$ und für jede Wahl von Punkten $\xi_i \in [x_{i-1}, x_i]$ gilt:

$$\sum_{i=1}^{\infty} f(\xi_i)(x_i - x_{i-1}) < \infty$$

Der Grenzwert heißt *bestimmtes Integral von f* auf dem Intervall $[a, b]$ (Sprechweise: 'von a nach/bis b'). Schreibweise: $\int_a^b f(x)\, dx$

Hauptsatz der Infinitesimalrechnung

Satz 7.5: (Hauptsatz der Infinitesimalrechnung)

Sei $f : [a, b] \to \mathbb{R}$ stetig. Dann ist $F(x) := \int_a^x f(\xi)\, d\xi$ eine Stammfunktion von f, und für jede beliebige Stammfunktion F gilt:

$$\int_a^b f(x)\, dx = F(b) - F(a)$$

Schreibweisen:

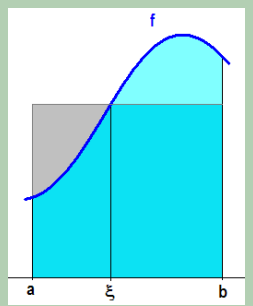
$$\int_a^b f(x)\, dx = [F(x)]_a^b = F(x)|_a^b$$

Mittelwertsatz der Integralrechnung

Satz 7.6: (Mittelwertsatz der Integralrechnung)

Sei $f : [a, b] \to \mathbb{R}$ stetig. Dann gilt:

$$\exists_{\xi \in [a,b]} \; f(\xi) = \frac{1}{b-a} \int_a^b f(x) \; dx$$



Der Flächeninhalt unter der Kurve auf dem Intervall $[a, b]$ entspricht dem Flächeninhalt des Rechtecks mit Länge $b - a$ und Breite $f(\xi)$.

Eigenschaften des Integrals

Es gilt:

(i) $\forall_{c \in [a,b]} \int_a^b f(x)\ dx = \int_a^c f(x)\ dx + \int_c^b f(x)\ dx$

(ii) $\int_a^a f(x)\ dx = 0$

(iii) $\int_a^b f(x)\ dx = -\int_b^a f(x)\ dx$

Trapezregel

Ziel: Approximation der zu integrierenden Funktion $f : [a.b] \to \mathbb{R}$ durch Trapez mit Flächeninhalt

$$T = (b-a)\frac{f(b)+f(a)}{2}$$

Dreiteilung des Intervalls $[a, b]$ in Teilintervalle der Länge $h > 0$:

$$T(h) := h\left(\frac{1}{2}(f(a)+f(b)) + f(a+h) + f(a+2h)\right)$$

Beliebig viele Stützstellen $a + ih, i = 0, ..., n \;\Rightarrow\; h = \frac{b-a}{n}$:

$$T(h) = h\left(\frac{1}{2}(f(a)+f(b)) + \sum_{i=1}^{n-1} f(a+ih)\right)$$

Simpsonregel

Ziel: Approximation der zu integrierenden Funktion $f : [a.b] \to \mathbb{R}$ durch Parabel $P(x)$ (Polynom vom Grad 2) so, daß $P(a) = f(a)$, $P(b) = f(b)$ und $P\left(\frac{a+b}{2}\right) = f\left(\frac{a+b}{2}\right)$:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right) =: S(h)$$

Betrachte allgemein $n = 2m$ Teilintervalle der Länge $h = \frac{b-a}{n}$, also $2m+1$ Stützstellen. Sei $f_i = f(a+ih)$ für $i = 0, ..., 2m$:

$$\int_a^b f(x)\,dx \approx \frac{h}{3}(f_0+f_{2m+1})+\frac{2h}{3}(f_2+f_4+...+f_{2m})+\frac{4h}{3}(f_1+f_3+...+f_{2m-1}) =$$

Fehlerabschätzung bei der Trapezregel

Sei f auf $[a, b]$ mindestens 2x stetig differenzierbar und
$M := \max_{x \in [x_0, x_0+h]} |f''(x)|$

$$|T_{\text{err}}(h)| := \left| \int_{x_0}^{x_0+h} f(x)\,dx - \frac{h}{2}(f(x_0) + f(x_0+h)) \right| \leq \frac{1}{12} \cdot M \cdot h^3$$

Wegen

$$T(h) - \int_a^b f(x)\,dx = \frac{h^2(b-a)}{12} f''(\xi)$$

für ein $\xi \in (a, b)$, ist die Trapezregel ein Verfahren 2. Ordnung, d.h., Polynome 1. Grades werden exakt integriert.

Fehlerabschätzung bei der Simpsonregel

Sei f auf $[a, b]$ mindestens 3x stetig differenzierbar und $M := \max_{x \in [x_0, x_0+h]} |f'''(x)|$

$$\begin{aligned} |S_{\text{err}}(h)| &:= \left| \int_{x_0-h}^{x_0+h} f(x)\,dx - \frac{h}{3}(f(x_0-h) + 4f(x_0) + f(x_0+h)) \right| \\ &\leq \frac{1}{36} \cdot M \cdot h^4 \end{aligned}$$

Falls f mind. 4x stetig differenzierbar ist, ist wegen

$$S(h) - \int_a^b f(x)\,dx = \frac{h^4}{180} f^{(iv)}(\xi)$$

für ein $\xi \in (a, b)$, die Simpsonregel ein Verfahren 4. Ordnung, d.h., Polynome 3. Grades werden exakt integriert.

Newton-Cotes-Formeln

Ziel: Suche allgemeine Quadraturregeln der Ordnung p, d.h., Polynome $p-1$-ten Grades sollen exakt integriert werden! Benötige dann mehr Intervalle! Sei n die Anzahl an Teilintervallen und $h = \frac{b-a}{n}$ die Schrittweite. Betrachte ein Polynom P_n vom Grad n mit $\forall_{i=0,\ldots,n}\ P_n(x_i) = f_i = f(x_i)$ und

$$\int_a^b P_n(x)\,dx = h\sum_{i=0}^{n} f_i\alpha_i \approx \int_a^b f(x)\,dx$$

mit Koeffizienten $\alpha_i \in \mathbb{Q},\ i = 0, \ldots, n$.

Korollar 7.2:

Für die Koeffizienten $\alpha_i \in \mathbb{Q}, i = 0, \ldots, n$, gilt $\sum_{i=0}^{n} \alpha_i = n$

Fehlerabschätzung für Newton-Cotes-Formeln

Satz 7.7: (Fehlerabschätzung für Newton-Cotes-Formeln)

Sei f auf $[a, b]$ mindestens k-mal stetig differenzierbar. Dann gilt für den Integrationsfehler der Newton-Cotes-Formeln:

$$\int_a^b P_n \, dx - \int_a^b f(x) \, dx = h^{k+1} \cdot K \cdot f^{(k)}(\xi)$$

für ein $K \in \mathbb{R}$ und ein $\xi \in (a, b)$.

Überblick: Newton-Cotes-Formeln

n	Name der Regel	Koeffizienten	Fehler
1	Trapezregel	1 1	$\frac{h^3}{12} f''(\xi)$
2	Simpsonregel	1 4 1	$\frac{h^5}{90} f^{(iv)}(\xi)$
3	Newton's $\frac{3}{8}$-Regel	1 3 3 1	$\frac{3h^5}{80} f^{(iv)}(\xi)$
4	Milne-Regel	7 32 12 32 7	$\frac{8h^7}{945} f^{(vi)}(\xi)$
5	–	19 75 50 50 75 19	$\frac{275h^5}{12096} f^{(vi)}(\xi)$
6	Weddle-Regel	41 216 27 272 27 216 41	$\frac{9h^9}{1400} f^{(viii)}(\xi)$

höhere Werte für n sind numerisch unbrauchbar, da Koeffizienten negativ werden (numerische Auslöschung möglich!!!)
kleine Änderungen in f, $\tilde{f} = f + \Delta f$, wirken sich je nachdem kaum auf das Integral, aber verheerend auf die Quadraturformel aus!

Romberg-Integration

Problematik:

(i) Genauigkeit: Wann ist das Ergebnis *gut genug*?

(ii) Rechenaufwand: Berechnung der Funktionswerte $f(x)$ kann aufwendig sein!

Kombination mit *Extrapolation* ($h = 0$) mithilfe des sog. *Neville-Aitken-Schemas* (Tafel: am Beispiel der Trapezregel)

Gauß-Quadratur: Motivation

Bisher:

Stützstellen $x_0, ..., x_n$ (äquidistant) sowie Funktionswerte $f(x_0), ..., f(x_n)$ fest vorgegeben.

Motivation:

Verzichte auf Vorgabe der Stützstellen und erhöhe somit die Anzahl an Freiheitsgraden.

Betr. dazu eine Darstellung von f in der Form

$$f(x) := g(x)\rho(x)$$

mit sog. *Gewichtsfunktion* $\rho(x)$ auf einem Intervall $[a, b]$.

Gauß-Quadratur: Problemstellung

Die Gewichtsfunktion $\rho(x)$ sollte mit evtl. Ausnahme von endlich vielen Punkten positiv sein. Berechne

$$\int_a^b g(x)\rho(x)\,dx = \int_a^b f(x)\,dx$$

Problemstellung:

Suche Stützstellen $x_k \in [a,b]$ sowie Gewichte $\alpha_k \in \mathbb{R}$ $(k = 1,...,n)$ so, daß
$\sum_{k=1}^{n} \alpha_k g(x_k)$ das Integral $\int_a^b f(x)\,dx$ möglichst gut approximiert.

Hier andere Numerierung: $x_1,...,x_n$; Gewichte: $\alpha_1,...,\alpha_n$

Gauß-Quadratur: Genauigkeitsanforderung

Genauigkeitsanforderung:

Alle Polynome bis zum Grad $2n-1$ sollen exakt integriert werden, d.h.

$$\forall_{\ell=0,\ldots,2n-1} \sum_{k=1}^{n} \alpha_k x_k^\ell = \int_a^b x^\ell \rho(x)\, dx$$

Dies ist ein nichtlineares Gleichungssystem für x_k und α_k, welches eindeutig lösbar sein muß! Die Gewichte werden sich als positiv herausstellen.

Orthogonale Polynome

Definition 7.6: (Orthogonale Polynome)

Auf dem Vektorraum der Polynome $\mathcal{P}$ sei durch

$$\langle p, q \rangle_\rho := \int_a^b p(x)q(x)\rho(x)\,dx$$

ein Skalarprodukt für $p, q \in \mathcal{P}$ und durch $||p||_\rho := \sqrt{\langle p, p \rangle_\rho} = \left(\int_a^b p^2(x)\rho(x)\,dx \right)^{\frac{1}{2}}$ die entsprechende Norm definiert. Zwei Polynome $p, q \in \mathcal{P}$ heißen *orthogonal*, falls $\langle p, q \rangle_\rho = 0$.

Satz 7.8:

Durch $\langle p, q \rangle_\rho$ ist tatsächlich ein Skalarprodukt definiert.

Konstruktion orthogonaler Polynome

Satz 7.9: (Konstruktion orthogonaler Polynome nach Gram-Schmidt)

Mit der Rekursionsformel

$$\begin{aligned} p_0(x) &:= 1 \\ p_n(x) &:= x^n - \sum_{j=0}^{n-1} \frac{\langle x^n, p_j \rangle_\rho}{\langle p_j, p_j \rangle_\rho} p_j(x), n \in \mathbb{N}, \end{aligned}$$

erhält man paarweise orthogonale Polynome.

Die Nullstellen der Orthogonalpolynome spielen für die Gauß-Quadratur eine essentielle Rolle! Als Integrationsintervall muß stets $[-1, 1]$ betrachtet werden. Im allgemeinen Fall muß das Intervall $[a, b]$ in das Intervall $[-1, 1]$ transformiert werden!

Gauß-Quadraturformel

Definition 7.7: (Gauß-Quadraturformel)

Sind $x_1, ..., x_n$ die Nullstellen des n-ten Orthogonalpolynoms bzgl $\langle \cdot, \cdot \rangle_\rho$, so heißt $\sum_{k=1}^{n} \alpha_k g(x_k)$ mit Gewichten

$\alpha_k = \langle L_k, 1 \rangle_\rho = \int_a^b L_k(x)\rho(x)\,dx$ *Gauß-Quadraturformel n-ter Ordnung zur Gewichtsfunktion* ρ. Dabei sind

$L_k(x) := \prod_{i=1, i\neq k}^{n} \frac{x - x_i}{x_k - x_i}$, $k = 1, ..., n$ die sog. *Lagrange-Polynome*.

Satz 7.10: (Gauß-Quadratur)

Die Gauß-Quadratur existiert und ist eindeutig. Alle Gewichte sind positiv, und die Gauß-Quadratur integriert alle Polynome bis zum Grad $2n - 1$ exakt.

Verschiedene Orthogonalpolynome

Intervall	$\rho(x)$	$p_0, p_1, \ldots$	Bezeichnung
$[-1,1]$	1	$1, x, x^2 - \frac{1}{3}$	Legendre
$[-1,1]$	$\frac{1}{\sqrt{1-x^2}}$	$1, x, x^2 - \frac{1}{2}$	Tschebyscheff
$[-1,1]$	$(1-x)^\alpha(1+x)^\beta$	$1, \frac{1}{2}(n+\alpha+\beta)x$ $+(\alpha-\beta)$	Jacobi $(\alpha, \beta > -1)$
$(-\infty,\infty)$	e^{-x^2}	$1, x, x^2 - \frac{1}{2}, x^3 - \frac{3}{2}x$	Hermite
$[0,\infty)$	$e^{-x}x^\alpha$	$1, x - \alpha - 1,$	Laguerre $(\alpha > -1)$

Die Konstruktion einer Gauß-Quadratur ist also sogar in $(-\infty,\infty)$ möglich (dies hängt von den möglichen Integrationsgrenzen bei der Bildung von Orthogonalpolynomen ab).

Fehler bei der Gauß-Quadratur

Satz 7.11: (Fehler bei der Gauß-Quadratur)

Mit den Stützstellen $x_1, ..., x_n$ und Gewichten $\alpha_1, ..., \alpha_n (n \in \mathbb{N})$ der Gauß-Quadratur gilt für auf einem Intervall $[a, b]$ $2n$-mal stetig differenzierbare Funktionen $g(x)$:

$$\int_a^b g(x)\rho(x)\,dx - \sum_{k=1}^{n} \alpha_k g(x_k) = \frac{||p_n||_\rho^2}{(2n)!} g^{(2n)}(\xi)$$

für ein $\xi \in [a, b]$. p_n ist das n-te Orthogonalpolynom bzgl. $\langle \cdot, \cdot \rangle_\rho$.

Sowohl die Konstante als auch der Grad der Ableitung sind bei der Gauß-Quadratur i.d.R. um eine Größenordnung kleiner bzw. größer im Vergleich zu den Newton-Cotes-Formeln.

Polynominterpolation

Betr. Zahlenpaare $(x_0, y_0), (x_1, y_1), ..., (x_n, y_n) \in \mathbb{R}^2$, $n \in \mathbb{N}$ mit $\forall_{i \neq k}\ x_i \neq x_k$ (sog. *Stützstellen*).
Suche Polynom P vom Grad n mit

$$\forall_{i=0,...,n}\ P(x_i) = y_i$$

(sog. *Interpolationspolynom*).

Satz 8.1: (Eindeutigkeit des Interpolationspolynoms)

Es gibt höchstens ein Interpolationspolynom.

Lagrange-Polynome

Satz 8.2: (Existenz des Interpolationspolynoms)

Es gibt ein Interpolationspolynom.

Dieses Interpolationspolynom ist gegeben durch die Formel

$$P(x) = \sum_{i=0}^{n} L_i(x) y_i$$

wobei

$$L_i(x) := \prod_{k=0, k\neq i}^{n} \frac{x - x_k}{x_i - x_k}, \quad i = 0, ..., n$$

die sog. *Lagrange-Polynome* sind.

Berechnungsschemeta, Dividierte Differenzen

Suche effiziente und schematische Berechnung der Polynomkoeffizienten oder Polynomauswertungen an neuen Stellen, die zwischen den Stützstellen liegen!
Beispiel (vielleicht noch aus der Analysis bekannt): Horner-Schema (halbiert etwa die Anzahl an notwendigen Multiplikationen durch geschicktes Ausklammern)

Definition 8.1: (Dividierte Differenzen)

Betrachte $k+1$ Zahlenpaare $(x_j, f_j), j = i, i+1, ..., i+k, i \geq 0, i+k \leq n$ mit $\forall_{j_1 \neq j_2}\ x_{j_1} \neq x_{j_2}$.
Der höchste Koeffizient a_k des (eindeutig bestimmten) Interpolationspolynoms dieser Zahlenpaare heißt *dividierte Differenz*. Bezeichnung: $[x_i, ..., x_{i+k}]f = a_k$ oder $f_{i,k} = a_k$

Newton-Schema

Satz 8.3: (Newton-Interpolation)

Das Interpolationspolynom kann als sog. *Newton-Polynom* in der Form

$$P_n(x) = \sum_{j=0}^{n} f_{0,j} \prod_{k=0}^{j-1} (x - x_k)$$

dargestellt werden.

Newton-Schema: (zur sukzessiven Berechnung der dividierten Differenzen, also Koeffizienten:)

$$[x_0, ..., x_k]f = \frac{[x_1, ..., x_k]f - [x_0, ..., x_{k-1}]f}{x_k - x_0}$$

Auf der Diagonalen sind die Koeffizienten $f_{0,j}, j = 0, ..., n$, ablesbar!

Neville-Aitken-Schema

Gehe nun von x_i aus um k Stützstellen zurück, betrachte also die Stützstellen $x_{i-k}, ..., x_i$.
Neville-Aitken-Schema: Durch die rekursive Darstellung

$$\begin{aligned} P_{i,0} &= f_i \\ P_{i,k} &= \frac{(x - x_{i-k})P_{i,k-1} - (x - x_i)P_{i-1,k-1}}{x_i - x_{i-k}} \end{aligned}$$

läßt sich das Interpolationspolynom an einer neuen Stelle x effizient auswerten. Es gilt $P_n(x) = P_{k,k}(x)$. Es sind allerdings keine Koeffizienten aus diesem Schema ablesbar!

Satz 8.4: (Neville-Aitken-Rekursion)

Die Neville-Aitken-Rekursion gilt tatsächlich!

Rekursionsformel der dividierten Differenzen

Satz 8.5: (Rekursion der dividierten Differenzen)

Die Rekursionsformel der dividierten Differenzen

$$\begin{aligned} [x_0]f &= f_0 \\ [x_0, ..., x_k]f &= \frac{[x_1, ..., x_k]f - [x_0, ..., x_{k-1}]f}{x_k - x_0} \end{aligned}$$

gilt tatsächlich!

Interpolationsfehler

Sei $\varepsilon(x) := f(x) - P_n(x)$ der Fehler zwischen Originalfunktion f (mind. $n+1$-mal differenzierbar) und P_n bei $x \in [a,b]$.
Mithilfe der Newton-Darstellung läßt sich leicht zeigen:

$$\varepsilon(x) = f_{0,n+1} \prod_{j=0}^{n} (x - x_j)$$

Mit dem Mittelwertsatz der Differentialrechnung folgt:

$$\begin{aligned}
\exists_{\xi \in [a,b]}\ \varepsilon(x) &= \prod_{j=0}^{n} (x - x_j) \frac{f^{(n+1)}(\xi)}{(n+1)!} \\
\Rightarrow\ |\varepsilon(x)| &\leq \left| \prod_{j=0}^{n} (x - x_j) \right| \cdot \max_{t \in [a,b]} \frac{|f^{n+1}(t)|}{(n+1)!} \\
&\leq (b-a)^{n+1} \cdot \max_{t \in [a,b]} \frac{|f^{n+1}(t)|}{(n+1)!}
\end{aligned}$$

Glatte Interpolation, Knotenwahl

Je *zerklüfteter f* ist, desto schwieriger ist die Interpolation. Die Interpolationsfunktion sollte möglichst glatt sein.

Definition 8.2: (Knotenpolynom)

Für beliebige Knoten $x_j, j = 0, ..., n$, heißt

$$\omega(x) := \prod_{j=0}^{n} (x - x_j)$$

Knotenpolynom.

Knotenwahl ist für gute Interpolation ausschlaggebend!

Tschebyscheff-Knoten und Tschebyscheff-Polynome

Definition 8.3: (Tschebyscheff-Knoten)

Die Knoten

$$x_j := \frac{1}{2}\left(a + b + (b-a)\cos\left(\frac{2(n-j)+1}{2n+2}\pi\right)\right),\ j = 0, ..., n,$$

heißen *Tschebyscheff-Knoten*.

Mit Tschebyscheff-Knoten kann eine viel glattere Interpolation erzielt werden!

Definition 8.4: (Tschebyscheff-Polynome)

Die Polynome $T_n(x) := \cos(n \arccos(x)), x \in [-1, 1]$, heißen *Tschebyscheff-Polynome*.

Optimalität der Tschebyscheff-Knoten

Satz 8.6: (Tschebyscheff-Polynome)

Die Funktionen $T_n(x) = \cos(n \arccos(x)), x \in [-1, 1]$, sind tatsächlich Polynome vom Grad n. Die Tschebyscheff-Knoten sind genau die Nullstellen von $T_{n+1}(x)$.

Man kann zeigen, daß für Knotenfunktionen ω und Tschebyscheff-Knoten gilt

$$|\omega(x)| \leq 2 \left(\frac{b-a}{4} \right)^{n+1}, x \in [a, b],$$

und mithilfe von $|T_n(x)| \leq 1$ läßt sich zeigen, daß die Wahl der Tschebyscheff-Knoten den Interpolationsfehler minimiert.

Hermite-Interpolation

Bei der Hermite-Interpolation sollen nicht nur die Funktionswerte, sondern auch deren Ableitungswerte bis zu einer gewissen Ordnung wiedergegeben werden. Bestimme also ein Polynom P_n mit $\forall_{j=0,\ldots,n} \forall_{i=0,\ldots,i_j} P_n^{(i)}(x_j) = f^{(i)}(x_j)$.
Dies kann dadurch gelöst werden, daß manche Knoten *mehrfach* gewählt werden, mit dividierten Differenzen

$$[x_j, x_j]f := \lim_{h\to 0}[x_j, x_j + h]f := \lim_{h\to 0} \frac{f(x_j + h) - f(x_j)}{h} = f'(x_j)$$

Neville-Hermitesche Formel

Satz 8.7: (Neville-Hermitesche Formel)

Die rekursiv definierten Polynome ($0 \leq i \leq k \leq n$)

$$
\begin{aligned}
P_{i,0}(x) &:= f_i \\
P_{i,k}(x) &:= \begin{cases} \frac{(x-\xi_i)P_{i+1,k-1}(x)-(x-\xi_{i+k})P_{i,k-1}(x)}{(\xi_{i+k}-\xi_i)}, & \xi_i \neq \xi_{i+k} \\ P_{i,k-1} + (x-\xi_i)^k \frac{f^{(k)}(\xi_i)}{k!}, & \xi_i = \xi_{i+k} \end{cases}
\end{aligned}
$$

realisieren eine Hermite-Interpolation der Funktion f an den gegebenen nicht notwendigerweise verschiedenen Stellen $\xi_i, \xi_{i+1}, ..., \xi_{i+k}, i + k \leq n$.

Existenz des Hermite-Interpolationspolynoms

Satz 8.8: (Hermite-Interpolation)

Es existiert ein (eindeutiges) Hermite-Interpolationspolynom. Es ist gegeben durch

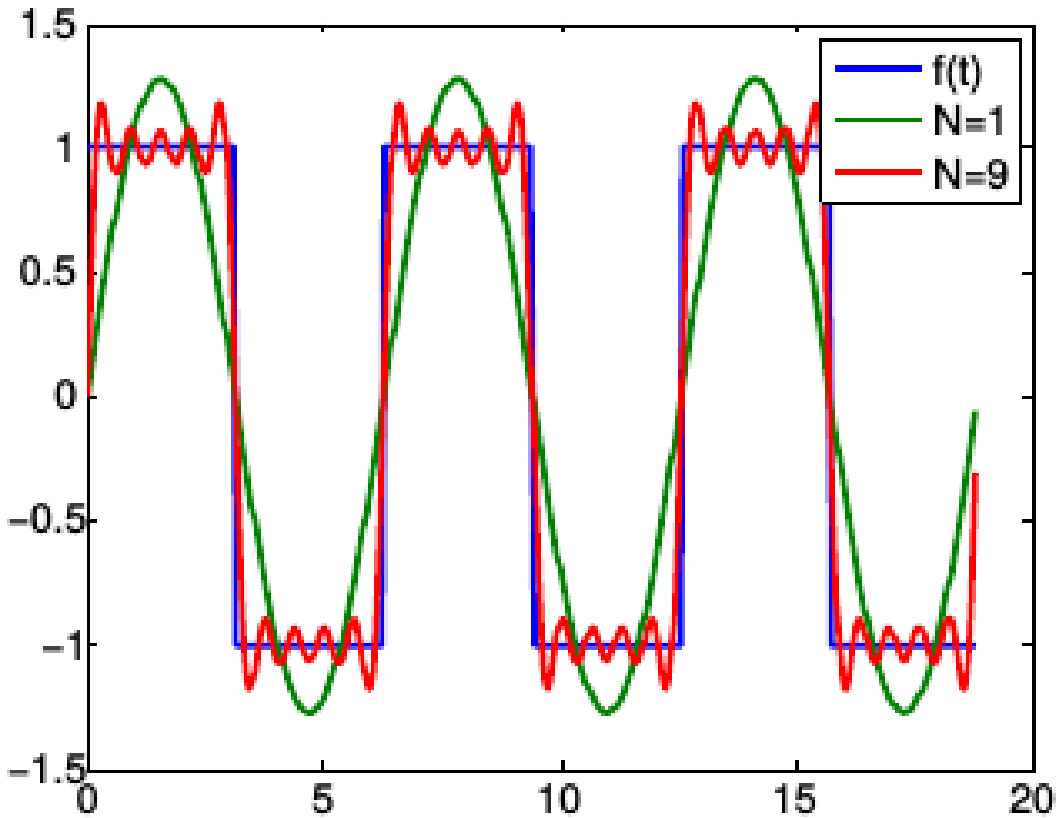
$$P_n(x) = \sum_{j=0}^{n} [\xi_0, \xi_1, ..., \xi_j] f \cdot \prod_{k=0}^{j-1} (x - \xi_k)$$

Der Interpolationsfehler ist (für f mind. $n+1$-mal differenzierbar) gegeben durch

$$\begin{aligned} \varepsilon(x) &= f(x) - P_n(x) = \prod_{j=0}^{n} (x - \xi) \frac{f^{(n+1)}(\xi)}{(n+1)!}, x, \xi \in [a, b] \\ &= (x-a)^{\kappa} (b-x)^{k} \frac{f^{(\kappa+k)}(\xi)}{(\kappa+k)!}, \kappa, k \in \mathbb{N}, \kappa + k \leq n. \end{aligned}$$

Diskrete und schnelle Fouriertransformation

Physikalische Motivation: Zerlegung eines Signals in Signale mit verschiedenen Frequenzen



Fourier-Reihe

Definition 8.5: (Fourier-Reihe)

Sei $\omega \in \mathbb{Z}$ eine Frequenz und $f : \left[0, \frac{2\pi}{\omega}\right] \to \mathbb{R}$ stückweise stetig. Dann heißen

$$a_n := \frac{\omega}{\pi} \int_0^{\frac{2\pi}{\omega}} f(t)\cos(n\omega t)\,dt, b_n := \frac{\omega}{\pi} \int_0^{\frac{2\pi}{\omega}} f(t)\sin(n\omega t)\,dt,\ n \in \mathbb{N},$$

Fourier-Koeffizienten von f und

$$\psi(x) := \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(n\omega x) + b_n \sin(n\omega x)$$

die *Fourier-Reihe* von f.

Wie funktioniert mp3?

Satz 8.9:

Jede periodische Funktion kann durch eine Fourier-Reihe dargestellt werden.

Funktionsweise von mp3:

Lasse bei Audio-Aufnahmen die höheren Frequenzen weg, die sowieso für das menschlische Gehör nicht wahrnehmbar sind, weg, d.h., berechne nur Fourier-Summen bis zu gewissem $N \in \mathbb{N}$:

$$\frac{a_0}{2} + \sum_{n=1}^{N} a_n \cos(n\omega x) + b_n \sin(n\omega x)$$

($\rightarrow$ Datenkompression, signifikant weniger Speicherverbrauch!)

Wir brauchen jetzt *Komplexe Zahlen:* Siehe Folien zur Analysis bzw. kurze Wiederholung an der Tafel!!!

Diskrete Fouriertransformation

Satz 8.10:

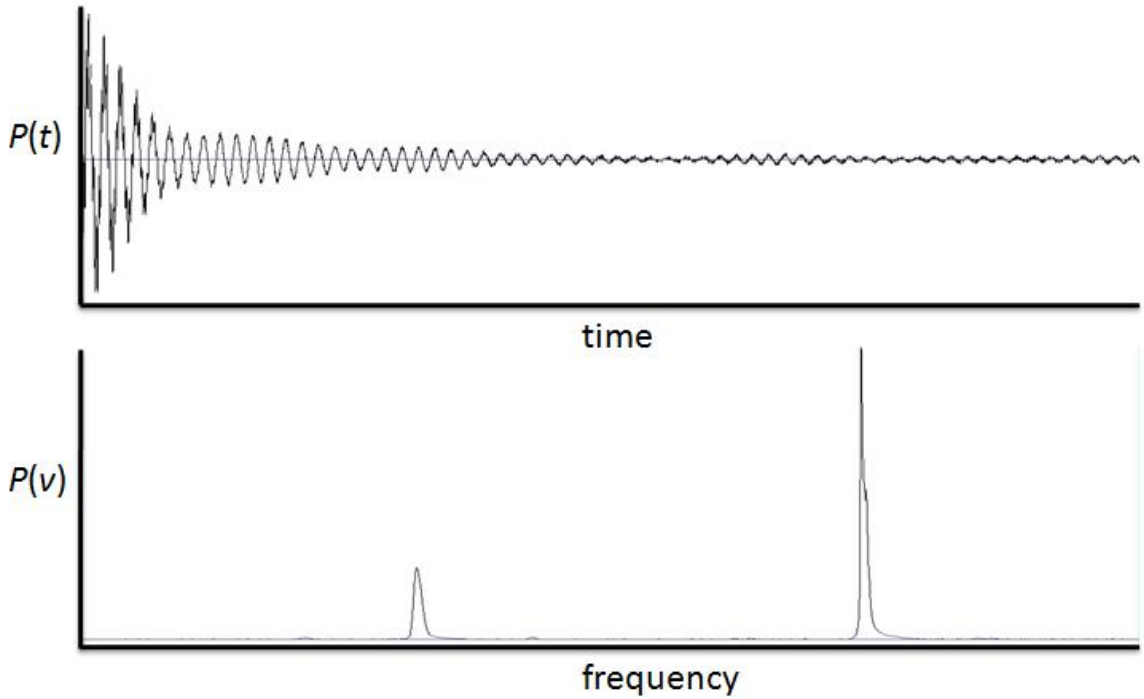
Zu beliebigen Stützpunkten $(x_k, f_k) \in \mathbb{R} \times \mathbb{C}, k = 0, ..., n-1, n \in \mathbb{N}$, mit äquidistanten Stützstellen $x_k = \frac{2\pi k}{n}$ und $f : [0, 2\pi] \to \mathbb{C}$, gibt es genau ein Interpolationspolynom $P : [0, 2\pi] \to \mathbb{C}$ der Form

$P(x) = \sum_{j=0}^{n} \beta_j e^{ix}, \ \beta_j \in \mathbb{C}, j = 0, ..., n-1$, mit $P(x_k) = f_k$ für alle $k = 0, ..., n-1$. Die Koeffizienten $\beta_j, j = 0, ..., n-1$, sind gegeben durch

$$\beta_j = \sum_{k=0}^{n-1} f_k \omega_k^{-j} =: \tilde{\psi}(j)$$

mit $\omega_k = e^{i\frac{2\pi k}{n}}, k = 0, ..., n-1$. $\tilde{\psi}$ beschreibt das Signal jetzt *frequenzabhängig*, nicht mehr *zeitabhängig*.

Anwendung: Signalverarbeitung



- Messe Signal zu bestimmten Zeitpunkten
- Diskrete Fouriertransformation liefert das Ausmaß gewisser Frequenzen j, also eine Abbildung $\tilde{\psi} : \mathbb{Z} \to \mathbb{R}$ mit $\tilde{\psi}(j) = |\beta_j|, j = 0, ..., n-1$ (Beispiele in den Übungen).

Cooley-Tukey-Methode

Problem:

Berechnung der $\beta_j = \sum_{k=0}^{n-1} f_k \left(e^{-\frac{2\pi i}{n}}\right)^{jk}, j = 0, ..., n-1$, erfodert $\mathcal{O}(n^2)$ Multiplikationen.

Cooley-Tukey-Methode:

Reduktion des großen Problems auf n kleinere Probleme der Größe $\ell = \log_2(n) \Leftrightarrow 2^\ell = n$ und damit des Rechenaufwands zu $\mathcal{O}(n \log(n))$.

Sei i.f. n eine Zweiterpotenz. Schreibe stattdessen 2^n statt n.

Schnelle Fouriertransformation

Satz 8.11: (Schnelle Fouriertransformation)

Seien $\beta_j, j = 0, ..., 2^n - 1$, die Koeffizienten der Diskreten Fouriertransformation, $\varepsilon_n := e^{-\frac{2\pi i}{2^n}}, m = \frac{2^n}{2}$ und $(x_k, f_k).k = 0, ..., 2^n - 1$, die Stützpunkte der Diskreten Fouriertransformation. Dann gilt für $\kappa = 0, ..., m - 1$:

(i)

$$\beta_{2\kappa} = \sum_{k=0}^{2^n-1} f_k \varepsilon_n^{2\kappa k} = \sum_{k=0}^{m-1} (f_k + f_{k+m}) \varepsilon_{n-1}^{\kappa k}$$

(ii)

$$\beta_{2\kappa+1} = \sum_{k=0}^{2^n-1} f_k \varepsilon_n^{(2\kappa+1)k} = \sum_{k=0}^{m-1} ((f_k - f_{k+m}) \varepsilon_n^k) \varepsilon_{n-1}^{\kappa k}$$

Schnelle Fouriertransformation

Es seien $f'_k := f_k + f_{k+m}$ und $f''_k := (f_k - f_{k+m})\varepsilon_n^k$ für $k = 0, ..., n-1$ für $n = 2^\ell$.
Die Summen der Länge $m = \frac{n}{2}$ lassen sich auf dieselbe Weise weiter reduzieren! Insgesamt gibt es

$$\left|\left\{\frac{n}{2}, \frac{n}{4}, \frac{n}{8}, ..., 1\right\}\right| = \ell \ (1 = \frac{n}{n} = \frac{n}{2^\ell})$$

Summen der Länge m zu berechnen! Die detaillierten Rekursionsformeln sind im Stoer-Bulirsch 1, auf S. 82, zu finden!

Radiale Basisfunktionen

Radial bedeutet, daß die Approximations-/Interpolationsfunktionen in Abhängigkeit des *Abstandes* zweier benachbarter Punkte angegeben werden.
Approximation/Interpolation einer Funktion $f : \mathbb{R}^n \to \mathbb{R}$ erfolgt gemäß

$$f(X) \approx \sum_{i=1}^{k} \beta_i h_i(||X - X_i||)$$

für $X \in \mathbb{R}^n$. Die $X_1, ..., X_K \in \mathbb{R}^n, K \in \mathbb{N}$, heißen *Zentroide* und die $h_i, i = 1, ..., K$, *radiale Basisfunktionen (RBFs)*.

Beispiele von RBFs

- $||X - X_i||$ (linear)
- $\exp(-\gamma||X - X_i||^2)$ (Gauß)
- $\sqrt{||X - X_i||^2 + c^2}$ (multiquadrisch)
- $\left(\sqrt{||X - X_i||^2 + c^2}\right)^{-1}$ (invers multiquadrisch)
- $||X - X_i||^3$ (kubisch)
- $||X - X_i||^2 \log ||X - X_i||$ (Thin-Plate Splines)

Bestimmung der Parameter c, γ so, daß Interpolations-/Approximationsfehler minimal wird.

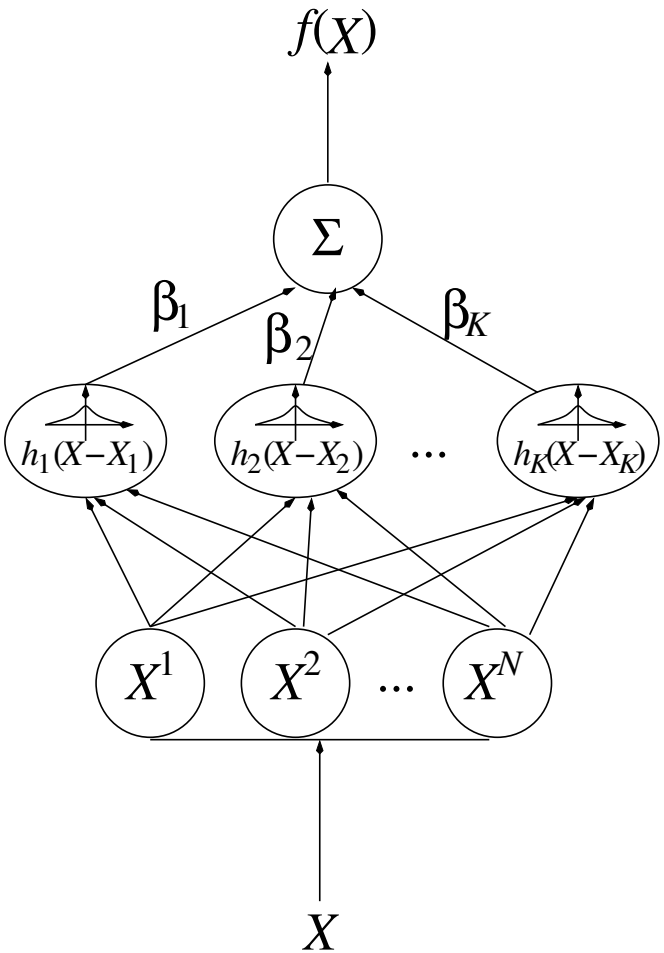
Approximation/Interpolation mit RBFs

Bestimmung der Koeffizienten $\beta_1, ..., \beta_K$ durch Lösen des LGS $H\beta = Y$ mit

$$H = \begin{pmatrix} h_1(||X_1 - X_1||) & h_2(||X_1 - X_2||) & \cdots & h_K(||X_1 - X_K||) \\ h_1(||X_2 - X_1||) & h_2(||X_2 - X_2||) & \cdots & h_K(||X_2 - X_K||) \\ \vdots & \vdots & \ddots & \vdots \\ h_1(||X_m - X_1||) & h_2(||X_m - X_2||) & \cdots & h_K(||X_m - X_K||) \end{pmatrix}$$

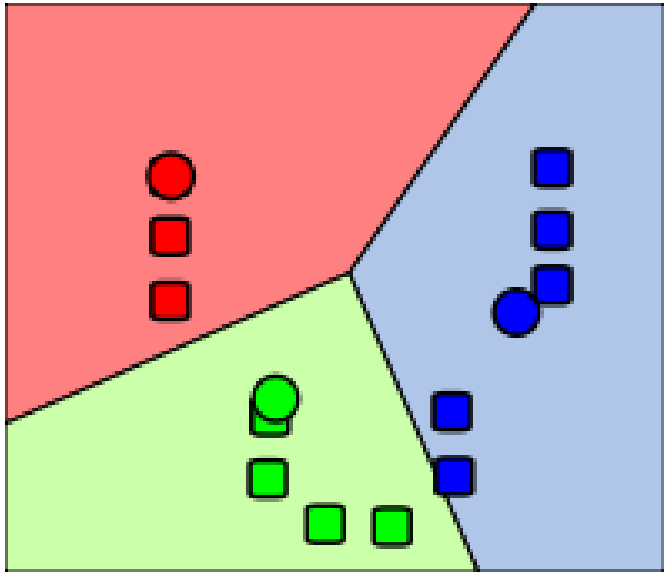
und $\beta = (\beta_1, ..., \beta_K)^T$, $Y = (f(X_1), ..., f(X_m))^T$.
Die $X_1, ..., X_m \in \mathbb{R}^n, m \in \mathbb{N}$, sind die Stützpunkte. Für $m = K$ ist das LGS quadratisch (Lösungsverfahren: Kapitel 2, 3, 4.3) und für $m > K$ überbestimmt (Lösungsverfahren: Kapitel 5).

RBF-Netze



Clustering

Bestimmung der Zentroide über sog. *Clustering*, z.B. *K-means*:



Es gehören diejenigen Punkte zu einem der K Cluster, die zum Mittelpunkt des Clusters im Gegensatz zu allen anderen Mittelpunkten den kleinsten Abstand haben.
RBFs spielen eine sehr große Rolle bei Data-Mining-Methoden!

Splines

Definition 8.6: (Spline-Funktion)

Eine auf einem Gitter $\Omega := \{a = x_0 < x_1 < ... < x_n = b\}$ auf einem Intervall $[a, b] \subseteq \mathbb{R}$ definierte reelle Funktion $s_\Omega : [a, b] \to \mathbb{R}$ heißt *Spline-Funktion* vom (polynomiellen) Grad $n \in \mathbb{N}_0$ und der (Differenzierbarkeits-)ordnung $k \in \mathbb{N}_0$, falls

(i) s_Ω ist auf $[a, b]$ k-mal stetig differenzierbar.

(ii) Auf jedem Teilintervall $[x_i, x_{i+1}]$, $i = 0, ..., n-1$, stimmt s_Ω mit einem Polynom vom Grad n überein.

Die x_i heißen auch *Stützstellen*, und die Menge aller Splines auf Ω, die Grad n und Ordnung k haben, wird mit $\mathcal{S}_{n,k,\Omega}$ bezeichnet.

Kubische Splines

Kubische Splines $s_\Omega \in \mathcal{S}_{3,2,\Omega}$ werden durch die Lösung eines LGS bestimmt, wobei eine der folgenden Nebenbedingungen verlangt wird:

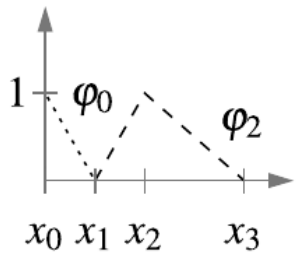
(i) $s_\Omega''(a) = s_\Omega''(b) = 0$

(ii) $s_\Omega^{(k)}(a) = s_\Omega^{(k)}(b) = 0$ für $k = 0, 1, 2$, d.h., $s_\Omega^{(k)}$ ist periodisch

Aufgrund der zweimaligen stetigen Differenzierbarkeit erscheinen kubische Splines stets schön glatt.

Lineare Splines

Lineare Splines $s_\Omega \in \mathcal{S}_{1,0,\Omega}$ sind stückweise linear, aber nicht überall differenzierbar. Ihre Basis besteht aus sog. *Hutfunktionen* $\varphi_0, ..., \varphi_n$ zu gegebenen Stützstellen $x_0, ..., x_n$, welche gegeben sind durch $\varphi_i(x_j) = \delta_{ij},\ 0 \leq i, j \leq n$, definiert sind:

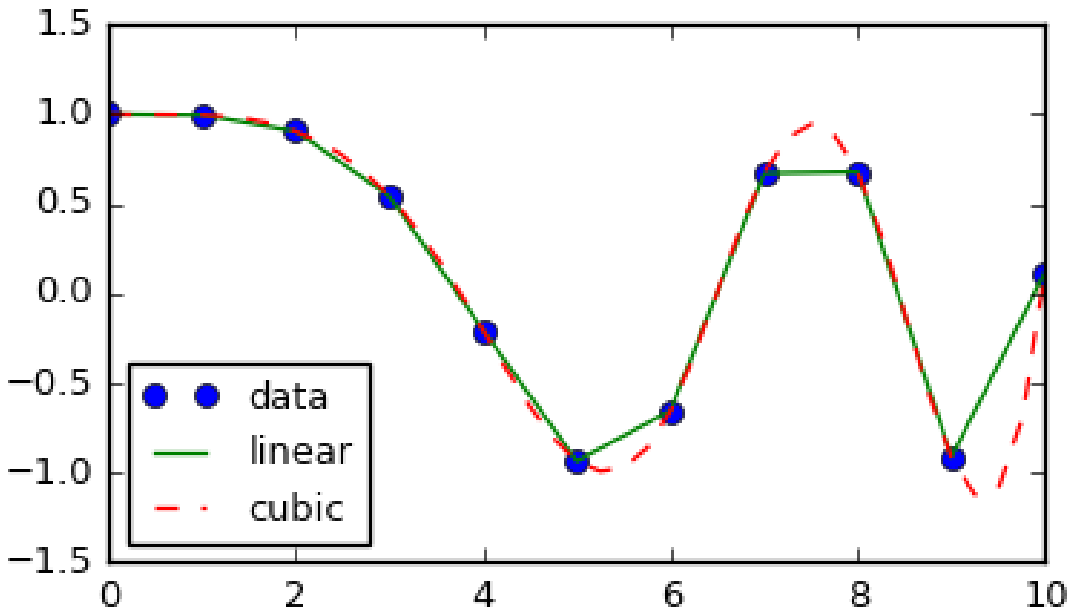


Satz 8.12: (Existenz und Eindeutigkeit linearer Splines)

Für gegebene Stützstellen $x_0, ..., x_n$ mit Funktionswerten $y_0, ..., y_n$ existiert genau ein linearer Spline mit $s(x_i) = y_i$ für alle $i = 0, ..., n$. Dieser ergibt sich als Linearkombination von Hutfunktionen:

$$s = \sum_{i=0}^{n} y_i \varphi_i.$$

Lineare vs. kubische Splines



Minimaleigenschaft von Splines

Sei $f : [a, b] \to \mathbb{R}$ zweimal stetig diffbar und $|f''|$ auf $[a, b]$ integrierbar. Betrachte die Seminorm $\left(\int_a^b |f''(x)|^2 \, dx \right)^{\frac{1}{2}}$, welche die Gesamtkrümmung von f beschreibt.

Satz 8.13: (Minimaleigenschaft und Eindeutigkeit von Splines)

Seien $y_0, ..., y_n \in \mathbb{R}$ gegebene Funktionswerte auf einem Gitter $\Omega = \{a = x_0 < x_1 < ... < x_n = b\}$ und $f : [a, b] \to \mathbb{R}$ zweimal stetig diffbar und $|f''|$ auf $[a, b]$ integrierbar mit $\forall_{i=0,...,n} \; f(x_i) = y_i$, d.h., f ist eine Interpolationsfunktion auf dem Gitter Ω. Sei weiterhin $s_\Omega : [a, b] \to \mathbb{R}$ eine kubische Spline-Interpolationsfunktion auf Ω. Dann gilt $||f|| \geq ||s_\Omega||$, insbesondere $||f - s_\Omega||^2 = ||f||^2 - ||s_\Omega||^2 \geq 0$, d.h. Splines haben minimale Gesamtkrümmung. Durch eine der o.g. Nebenbedingungen sind Splines eindeutig bestimmt.

B-Splines

Definition 8.7: (B-Splines)

Die *B-Splines k-ter Ordnung* $N_j^k, j = 1, ..., n-k$, sind rekursiv definiert durch

$$\begin{aligned} N_j^0(x) &:= \chi_{[x_j, x_{j+1})} \\ N_j^k(x) &:= \frac{x - x_j}{x_{j+k} - x_j} N_j^{k-1}(x) + \frac{x_{j+k+1} - x}{x_{j+k+1} - x_{j+1}} N_{j+1}^{k-1}(x) \end{aligned}$$

B-Splines sind positiv, haben kompakten Träger: $N_j^k(x) = 0$ für $x \notin [x_j, x_{j+k+1}]$ und sind stückweise Polynome vom Höchstgrad k auf den Knotenintervallen.

Kontrollpunkte

Eine Splinekurve im $\mathbb{R}^N$ der Ordnung k ergibt sich dann gemäß

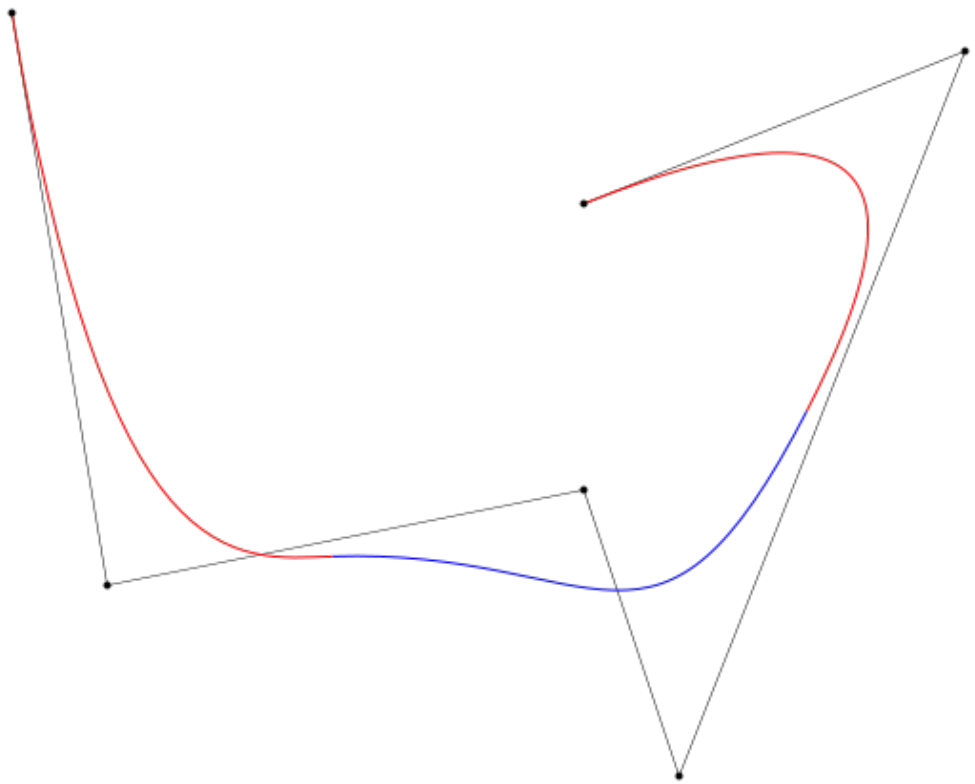
$$\mathcal{S}_k(d) = \sum_{j=1}^{n} d_j^k N_j^k(x)$$

Dabei heißen die $d_j \in \mathbb{R}^N, j = 1, ..., n$, *Kontrollpunkte*. Diese lassen sich mit dem *Algorithmus von de Boor* mithilfe der Konvexkombination

$$\begin{aligned} d_j^k(x) &= \alpha(x) d_{j-1}^{k-1}(x) + (1 - \alpha(x)) d_j^{k-1}(x), \\ \alpha(x) &= \alpha_j(x) = \frac{x_{j+k+1} - x}{x_{j+k+1} - x_j} \end{aligned}$$

bestimmen.

B-Spline-Kurven



Die B-Spline-Kurven liegen in der konvexen Hülle der Kontrollpunkte.

Bernstein-Polynome

Definition 8.8: (Bernstein-Polynome)

Die Polynome $B_{k,n}(x) := \binom{n}{k} \left(\frac{x-a}{b-a}\right)^k \left(\frac{b-x}{b-a}\right)^{n-k}$,
$x \in [a,b] \subseteq \mathbb{R}$, heißen *Bernstein-Polynome.*

Satz 8.14: (Approximation mit der Bernstein-Formel)

Sei f auf dem Intervall $[a,b]$ Lipschitz-stetig mit der Lipschitzkonstanten $L > 0$ sowie eine Genauigkeit $\varepsilon > 0$ gegeben. Dann gilt für $n > \frac{L^2}{\varepsilon}$ die Abschätzung

$$|f(x) - b_n(x)| := \left| f(x) - \sum_{k=0}^{n} f\left(a + \frac{k}{n}(b-a)\right) B_{k,n}(x) \right| < \varepsilon.$$

Bézier-Kurven

Das bedeutet, mithilfe der Bernstein-Polynome läßt sich f beliebig genau approximieren. Führt man für $F : \mathbb{R} \to \mathbb{R}^N$ für jede Komponente von $F(t)$ eine Bernstein-Approximation durch, so entstehen sog. *Bézier-Kurven*. Die $n+1$, auf denen F gegeben sein muß bilden die Kontrollpunkte der Bézier-Kurven, welche in deren konvexen Hülle liegt (sog. *Stützpolygon*).

Motivation: Gewöhnliche Differentialgleichungen

Problem in der Informatik: Bildrestauration (engl. *Inpainting*)

Kann man daraus ein Interpolationsproblem machen?

Ja, aber am Rand können Probleme auftreten (Kanten/Texturen können vollkommen zerstört werden)!

Ausweg:

Differentialgleichungen (DGLs)

Sei Ω ein Bild, das in $D \subset \Omega$ restauriert werden soll und in $\Omega \backslash D$ bekannt ist.
DGL: $-\Delta u + \lambda \chi_{\Omega \backslash D}(u - f) = 0$.
Dabei ist f das Originalbild und u das restaurierte Bild. Δ ist der sog. *Laplace-Operator* $\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$.
Ableitungen, d.h. Veränderungen von u spielen hier eine wesentliche Rolle!

Gewöhnliche Differentialgleichungen, Analytische Lösung

Betrachte zunächst nur Funktionen $u : \mathbb{R} \to \mathbb{R}$ und *gewöhnliche* Ableitungen $u', u'', u''', ..., u^{(n)}$, $n \in \mathbb{N}$.

Definition 9.1: (Gewöhnliche Differentialgleichung)

Eine Gleichung der Form

$$F(x, u, u', u'', ..., u^{(n)}) = 0$$

in den Ableitungen einer unbekannten Funktion $u : \mathbb{R} \to \mathbb{R}, x \mapsto u(x)$, bis zur Ordnung $n \in \mathbb{N}$, heißt *gewöhnliche Differentialgleichung (GDGL) n-ter Ordnung*. Falls nach $u^{(n)}$ aufgelöst werden kann, $u^{(n)} = f(x, u, u', u'', ..., u^{(n-1)})$, so heißt die GDGL *explizit*, ansonsten *implizit*. $u : \mathbb{R} \to \mathbb{R}$ heißt *Lösung* der GDGL, falls u eine der beiden obigen Gleichungen erfüllt.

Anfangswertproblem

Oft hängt u nur von der Zeit t ab. Man schreibt dann $\dot{u} := u'(t), \ddot{u} := u''(t)$ usw. und hat u oftmals zur Anfangszeit $t = 0$ gegeben.

Definition 9.2: (Anfangswertproblem)

Bei einem sog. *Anfangswertproblem (AP)* werden der Lösung $u(x)$ folgende n Anfangswerte (AWe) an der Stelle $x_0 \in \mathbb{R}$ vorgeschrieben:

$$u(x_0) = u_0,\ u'(x_0) = u_1, ...,\ u^{(n-1)}(x_0) = u_{n-1};\ u_1, ..., u_{n-1} \in \mathbb{R}$$

Damit werden Parameter, die durch unbestimmte Integrale entstehen, eindeutig.

Wichtigste lineare Differentialgleichung

Satz 9.1:

Die lineare GDGL $u' = \lambda u, \lambda \in \mathbb{R}$, mit AW $u(t_0) = u_0; t_0, u_0 \in \mathbb{R}$, hat die Lösung

$$u(t) = u_0 e^{\lambda(t-t_0)}$$

Analytische Lösung

1. Homogene GDGLs, deren Variablen getrennt werden können ($u'(x) = f(x)g(u) + 0$) können durch **Separation der Variablen** gelöst werden.
2. Inhomogene GDGLs ($u'(x) = f(x)u(x) + r(x)$) können durch **Variation der Konstanten** gelöst werden.

Verfahren an der Tafel und Beispiele in den Übungen!

Taylorreihe

Definition 7.6: (Taylorreihe)

Sei $f : [a, b] \to \mathbb{R}$ beliebig oft differenzierbar, $x_0 \in [a, b]$. Dann heißt die Potenzreihe

$$T_f(x) := \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k$$

Taylorreihe von f mit *Entwicklungspunkt* x_0. Das Polynom

$$T_n(x) := \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k$$

heißt *Taylorpolynom* vom Grad n. Falls $x_0 = 0$, so heißt die Taylorreihe auch *Mac-Laurin-Reihe*.

Taylorformel

Satz 7.15: (Taylorformel)

Sei $f : [a, b] \to \mathbb{R}$ $(n+1)$-mal stetig differenzierbar und $x, x_0 \in [a, b]$. Dann gilt:

$$\exists_{\xi \in (x_0, x)} \; f(x) = T_n(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)^{n+1}$$

Wichtige Taylorreihe:

$$\log(1 + x) = \sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k} x^k$$

Wichtige Folgerungen aus der Taylorformel

- Glatte, d.h. genügend oft differenzierbare Funktionen lassen sich durch Taylorpolynom + Restglied darstellen.
- Falls das Restglied für $n \to \infty$ gegen 0 geht, so stimmt die Taylorreihe mit der Funktion überein, es gilt also $f(x) = T_f(x)$. Die Approximation ist dann beliebig genau.
- Falls f durch eine Potenzreihe mit Konvergenzradius $R > 0$ darstellbar ist, so ist diese Potenzreihe bereits die Taylorreihe.

Einschrittverfahren

Definition 9.3: (Einschrittverfahren)

Sei $x_0 \in \mathbb{R}$ Startwert für das Anfangswertproblem $u'(x) = F(x,u),\ u(x_0) = u_0$. Dann ist durch

$$\begin{aligned} \eta_0 &:= u_0, \\ \eta_{i+1} &:= \eta_i + h\Phi(x_i, \eta_i; h; F), \\ x_{i+1} &:= x_i + h, i \in \mathbb{N}_0 \end{aligned}$$

ein sog. *Einschrittverfahren* definiert. Im Spezialfall $\Phi(x,u;h;F) = F(x,u)$ erhält man das sog. *Euler-Verfahren*.

Konsistenz

Definition 9.4: (Lokaler Diskretisierungsfehler/Konsistenz)

Sei $z(x)$ die exakte Lösung des Anfangswertproblems $u'(x) = F(x,u),\ u(x_0) = u_0$ und

$$\Delta(x,u;h;F) := \begin{cases} \frac{z(x+h)-u(x)}{h}, & h \neq 0 \\ \Phi(x,u;h;F), & h = 0 \end{cases}$$

der Differenzenquotient von z. Die Differenz $\tau(x,u;h;F) := \Delta(x,u;h;F) - \frac{\eta(x+h,h)-\eta(x,h)}{h}$ heißt *lokaler Diskretisierungsfehler*. Falls $\lim_{h\to 0} \tau(x,u;h;F) = 0$, so heißt das Verfahren *konsistent*, d.h., die exakte Lösung erfüllt die Gleichung des Einschrittverfahrens. Ein Verfahren hat *Konsistenzordnung* p, falls $\tau(x,u;h;F) = \mathcal{O}(h^p)$.

Weitere Einschrittverfahren

Das einfach Euler-Verfahren hat Konsistenzordnung 1.
Verfahren höherer Ordnung erhält man durch eine andere Wahl von $\Phi(x, u; h; F)$, z.B.:

- $\Phi(x, u; h; F) := \frac{1}{2}\left(F(x, u) + F(x + h, u + hF(x, u))\right)$
  (Verfahren von Heun)
- $\Phi(x, u; h; F) := F\left(x + \frac{h}{2}, u + \frac{h}{2}F(x, u)\right)$
  (Verbessertes Euler-Verfahren nach Collatz)
- allgemeiner Ansatz: Runge-Kutta-Verfahren
  (siehe Abschnitt 9.4)

Näheres zur Konsistenz und Konvergenz in Abschnitt 9.3!

Implizites Euler-Verfahren

Betrachte wieder das Anfangswertproblem
$u'(x) = F(x, u),\ u(x_0) = u_0$. Sei $i \in \mathbb{N}_0$:

- *Explizites Euler-Verfahren:* $\eta_{i+1} = \eta_i + hF(x_i, \eta_i)$
- *Implizites Euler-Verfahren:* $\eta_{i+1} = \eta_i + hF(x_{i+1}, \eta_{i+1})$

D.h., zum Erhalt von η_{i+1} muß in jedem Schritt eine nichtlineare Gleichung gelöst werden (z.B. mit dem Newton-Verfahren).
Das implizite Euler-Verfahren hat ebenfalls Konsistenzordnung 1.
Es hat allerdings im Gegensatz zum expliziten Euler-Verfahren gute Stabilitätseigenschaften (näheres zur Stabilität in Abschnitt 9.3).

Mehrschrittverfahren

Definition 9.5: (Mehrschrittverfahren)

Es seien $\eta_0, ..., \eta_{r-1}$ $r > 1$ Startwerte für das Anfangswertproblem $u'(x) = F(x, u),\ u(x_0) = u_0$, also Näherungswerte für $u(x_0), ..., u(x_{r-1})$, wobei $x_0, ..., x_{r-1}$ äquidistante Stützstellen sind. Sei $j \in \mathbb{N}_0$. Ein Verfahren zur Lösung des APs, für welches man für die Berechnung von η_{j+r} nicht nur η_{j+r-1}, sondern η_k für $k = j, j+1, ..., j+r-1$, benötigt, heißt *Mehrschrittverfahren* oder auch *r-Schrittverfahren*.

Die Startwerte $\eta_{j+r}, ..., \eta_{j+r-1}$ kann man mithilfe eines vorangeschalteten Einschrittverfahrens erhalten.

Verfahren von Adams-Bashforth und Adams-Moulton

Wie bei der Berechnung der Newton-Cotes-Formeln mithilfe von Lagrange-Polynomen (Herleitung an der Tafel!) erhält man ein allgemeines Mehrschrittverfahren der Form

$$\eta_{p+k} = \eta_{p-j} + h \sum_{i=0}^{q} \beta_{qi} F_{p-i}, \quad k, j, q \in \mathbb{N}_0$$

Spezialfälle:

- *Adams-Bashforth:* $k = 1, j = 0$: $\eta_{p+1} = \eta_p + h \sum_{i=0}^{q} \beta_{qi} F_{p-i}$,
$$\beta_{qi} := \int_0^1 \prod_{\ell=0, \ell \neq i}^{q} \frac{s+\ell}{-i+\ell} \, ds$$
- *Adams-Moulton:* $k = 0, j = 1$: $\eta_p = \eta_{p-1} + h \sum_{i=0}^{q} \beta_{qi} F_{p-i}$,
$$\beta_{qi} := \int_{-1}^0 \prod_{\ell=0, \ell \neq i}^{q} \frac{s+\ell}{-i+\ell} \, ds$$

Koeffizienten

i	0	1	2	3	4
β_{0i}	1				
$2\beta_{1i}$	3	-1			
$12\beta_{2i}$	23	-16	5		
$24\beta_{3i}$	55	-59	37	-9	
$720\beta_{4i}$	1901	-2774	2616	-1274	251

(Adams-Bashforth)

i	0	1	2	3	4
β_{0i}	1				
$2\beta_{1i}$	1	1			
$12\beta_{2i}$	5	8	-1		
$24\beta_{3i}$	9	19	-5	1	
$720\beta_{4i}$	251	646	-264	106	-19

(Adams-Moulton)

Prädiktor-Korrektor-Verfahren

Ersetze beim Adams-Moulton-Verfahren p durch $p+1$:
$\eta_{p+1} = \eta_p + h\sum_{i=0}^{q} \beta_{qi} F_{p+1-i}$.
Da $p+1$ sowohl links als auch rechts vorkommt, handelt es sich um ein implizites Verfahren, d.h. η_{p+1} wird über eine nichtlineare Gleichung bestimmt, also iterativ:

$$\eta_{p+1}^{(i+1)} = \eta_p + h\left(\beta_{q0} F(x_{p+1}, \eta_{p+1}^{(i)}) \sum_{i=1}^{q} \beta_{qi} F_{p+1-i}\right), \quad i = 0, 1, 2, \ldots$$

Es handelt sich dabei um eine Fixpunktiteration.

1. Adams-Bashforth als explizites Verfahren zum Erhalt einer guten Startnäherung $\eta_{p+1}^{(0)}$ (sog. *Prädiktor-Verfahren*)
2. Adams-Moulton als implizites Verfahren zum Erhalt einer besseren Näherung $\eta_{p+1}^{(1)}$ (sog. *Korrektor-Verfahren*)

Allgemeine Mehrschrittverfahren

Ein allgemeines Mehrschrittverfahren (r-Schrittverfahren) hat die Form

$$\eta_{j+r} + a_{r-1}\eta_{j+r-1} + \ldots + a_0\eta_j = h\Phi(x_j; \eta_{j+r}, \eta_{j+r-1}, \ldots, \eta_j; h; F)$$

mit Koeffizienten $a_0, \ldots, a_{r-1}$.
Die Funktion Φ hängt dabei linear von F ab:

$$\Phi(x_j; \eta_{j+r}, \eta_{j+r-1}, \ldots, \eta_j; h; F) = \sum_{i=0}^{r} b_i F(x_{j+i}, \eta_{j+i})$$

mit Koeffizienten $b_0, \ldots, b_r$.

Konsistenz bei Mehrschrittverfahren

Definition 9.4: (Lokaler Diskretisierungsfehler/Konsistenz MSV)

Sei $z(x)$ die exakte Lösung des Anfangswertproblems $u'(x) = F(x,u),\ u(x_0) = u_0$. Die Differenz

$$\begin{aligned}\tau(x,u;h;F) \quad := \quad & \frac{1}{h}\left(z(x+rh) + \sum_{i=0}^{r-1} a_i z(x+ih) \right. \\ & \left. -h\Phi(x; z(x+rh), ..., z(x); h; F) \right)\end{aligned}$$

heißt *lokaler Diskretisierungsfehler* des Mehrschrittverfahrens. Falls es für jedes F eine Funktion $\sigma(h)$ gibt mit $\lim_{h\to 0} \sigma(h) = 0$, so daß $|\tau(x,u;h;F)| \leq \sigma(h)$, so heißt das Mehrschrittverfahren *konsistent*. Ein Mehrschrittverfahren hat *Konsistenzordnung* p, falls $\sigma(h) = \mathcal{O}(h^p)$.

Konsistenz, Stabilität und Konvergenz

Betrachte in diesem Abschnitt wieder das Anfangswertproblem

$$u' = F(x, u);\ u(x_0) = u_0,\ x \in \Omega \subseteq \mathbb{R}$$

- $\eta(x)$: Näherung für $u(x)$
- $\eta_i, i \in \mathbb{N}$: Näherungen innerhalb eines Ein- oder Mehrschrittverfahrens
- $\Phi(x, u; h; F)$: Verfahrensfunktion, $h > 0$: Schrittweite, $x \in \Omega_h$ (diskretisiertes Gebiet)

Globaler Diskretisierungsfehler und Konvergenz

Definition 9.7: (Globaler Diskretisierungsfehler/Konvergenz)

$e(x, u; h; F) := u(x) - \eta(x)$ heißt *globaler Diskretisierungsfehler (Konvergenzfehler)* der Lösung u an der Stelle x. Weiterhin wird definiert: $e^*(x, u; h; F) := \sup_{x \in \Omega_h} |e(x, u; h; F)|$. Das Verfahren heißt *konvergent*, falls $\lim\limits_{h \to 0} e^*(x, u; h; F) = 0$. Es heißt *konvergent von der Ordnung p*, falls $e^*(x, u; h; F) = \mathcal{O}(h^p)$.

Satz 9.2: (Konvergenz von Einschrittverfahren)

Sei $F \in \mathcal{F}_\infty([a, b])$ und $x_0 = a, u_0 = u(x_0) \in \mathbb{R}$. z sei die exakte Lösung des Anfangswertproblems auf $[a, b]$. Φ erfülle die globale Lipschitz-Bedingung bzgl. η:

Die Menge $\mathcal{F}_N([a, b])$ beinhaltet alle Funktionen, deren partielle Ableitungen bis zur Ordnung N auf dem Intervall $[a, b]$ existieren, stetig und beschränkt sind.

Konvergenz von Einschrittverfahren

Satz 9.2: (Konvergenz von Einschrittverfahren), Forts.

$$\exists_{L>0} \; \forall_{x\in[a,b]} \; \forall_{\eta,\tilde{\eta}\in\mathbb{R}} \; \forall_{h>0} \; |\Phi(x,\eta;h;F) - \Phi(x,\tilde{\eta};h;F)| \leq L|\eta - \tilde{\eta}|$$

Für den lokalen Diskretisierungfehler τ gelte

$$\exists_{K=K(z)\in\mathbb{R}_+} \; \forall_{x\in[a,b-h]} \; \forall_{h>0,h<b-a} \; |\tau(x,u;h;F)| \leq K \cdot h^p$$

Dann gilt für den globalen Diskretisierungsfehler $e(x,\eta;h;F)$ bei Anwendung eines Einschrittverfahrens mit konstanter Schrittweite $h > 0$:

$$\forall_{x\in[a,b]} \; |e(x,\eta;h;F)| \leq h^p \cdot K \cdot \frac{e^{L(x-x_0)} - 1}{L} =: C \cdot h^p$$

Globaler Gesamtfehler

Der sog. *globale Gesamtfehler* ergibt sich aus Diskretisierungs- und Rundungsfehler:

$$E_j(x_j) := z(x_j) - \tilde{\eta}_j(x_j, h),$$

wobei $\tilde{\eta}(x_j, h) = \eta(x_j, h) + \delta_{j+1}$ ein mit Rundungsfehlern behafteter Näherungswert ist, wobei $|\delta_{j+1}| \leq \delta = \delta(\mathrm{eps})$. Es gilt die Abschätzung (Tafel!):

$$|E(x, u; h; F)| \leq \left(K \cdot h^p + \frac{\delta}{h} \right) \cdot \frac{e^{L(x-x_0)} - 1}{L}$$

Asymptotische Entwicklung des globalen Diskretisierungsfehlers

Satz 9.3: (Asymptotische Entwicklung des globalen Diskretisierungsfehlers)

Sei $F \in \mathcal{F}_{N+2}([a,b])$, alles andere sei so wie in Satz 9.2. Dann gilt für $x \in [a,b], h > 0$:

$$\begin{aligned} e(x,u;h;F) &= u(x) - \eta(x,h) \\ &= h^p e_p(x) + h^{p+1} e_{p+1}(x) + \ldots \\ &+ h^N e_N(x) + h^{N+1} \tilde{e}_{N+1}(x,h), \end{aligned}$$

wobei $e_k = e_k(F, u_0, \Phi), k = p, \ldots, N+1$, differenzierbar mit $e_k(x_0) = 0$ und

$$\sup_{h>0} |\tilde{e}_{N+1}(x,h)| < \infty$$

Richardson-Korrektur: Erhöhung der Verfahrensordnung

Satz 9.4: (Richardson-Korrektur)

Es gilt die Schätzformel

$$e(x,u;h;F)\doteq\frac{\eta\left(x,\frac{h}{2}\right)-\eta(x,h)}{2^p-1}$$

für den globalen Diskretisierungsfehler sowie

$$u(x)\doteq\eta\left(x,\frac{h}{2}\right)+\frac{\eta\left(x,\frac{h}{2}\right)-\eta(x,h)}{2^p-1}=:\hat{\eta}(x,h)$$

$\hat{\eta}$ ist eine exptrapolierte Näherung mit Fehlerordnung $p+1$.

Schrittweitensteuerung

Asymptotische Betrachtungen fordern *h hinreichend klein*. Aber wie wählt man h optimal?

Sog. *adaptive Schrittweitensteuerung:* Wähle h nicht a priori, sondern adaptiv im Laufe des Verfahrens, und zwar so, daß

$$e(x, u; h; F) \approx \varepsilon$$

mit vorgegebener Genauigkeit ε. (Herleitung an der Tafel!)
Kleinere Schrittweite bedeutet stets höhere Genauigkeit!

Adaptive Schrittweitensteuerung

Bei geeigneter Testschrittweiter $H_0 > 0$ und vorgegebener Genauigkeit ε ergibt sich als erste adaptive Schrittweite:

$$h_0 = H_0 \cdot \sqrt[p+1]{\frac{2^p - 1}{2^p} \cdot \frac{\varepsilon}{\left|\eta\left(x_0 + H_0, \frac{H_0}{2}\right) - \eta(x_0 + H_0, H_0)\right|}}$$

Wahl von H_0 durch folgende Überlegung: Nach Satz 9.2 gilt unter Vernachlässigung der Terme höherer Ordnung:

$$\frac{e(x, u; qh; F)}{e(x, u; h; F)} \doteq \frac{(qh)^p e_p(x)}{h^p e_p(x)} = q^p$$

Bei Änderung von h um Faktor q sollte sich der Fehler um Faktor q^p ändern. Für $q = \frac{1}{2}$ gilt: Falls obiges Fehlerverhältnis $\approx \frac{1}{2^p}$, dann ist h okay. Ansonsten halbiere h!

Lineare Differenzengleichungen

Definition 9.8: (Lineare Differenzengleichung)

Eine Gleichung der Form

$$\eta_{j+r} + a_{r-1}\eta_{j+r-1} + a_{r-2}\eta_{j+r-2} + ... + a_1\eta_{j+1} + a_0\eta_j = 0,\ j \in \mathbb{N}_0,$$

mit Koeffizienten $a_0, ..., a_r \in \mathbb{R}, \alpha_r = 1$ heißt *homogene lineare Differenzengleichung r-ter Ordnung*.

Die Lösungen $\eta_{j+r}, \eta_{j+r-1}, ..., \eta_j$, können nach dem Fundamentalsatz der Algebra komplex sein!
Genauer: Zu jedem Satz von komplexen Startwerten $\eta_0, ..., \eta_{r-1}$ gibt es genau eine Folge $(\eta_j)_{j\in\mathbb{N}}$, die die lineare Differenzengleichung erfüllt.
Bei Mehrschrittverfahren sollte das Wachstumsverhalten den η_j *stabil* sein, d.h., es sollte gelten: $\lim\limits_{j\to\infty} \frac{\eta_j}{j} = 0$

Erstes charakteristisches Polynom von Mehrschrittverfahren

Definition 9.9: (Erstes charakteristisches Polynom)

Das *erste charakteristische Polynom* $\chi : \mathbb{C} \to \mathbb{C}$ einer linearen Differenzengleichung

$$\sum_{i=0}^{r} a_i \eta_{j+i} = 0$$

bzw. eines linearen Mehrschrittverfahrens

$$\sum_{i=0}^{r} a_i \eta_{j+i} = \Phi(x_j; \eta_{j+r}, ...\eta_j; h; F)$$

ist definiert durch

$$\chi(\lambda) := \sum_{i=0}^{r} a_i \lambda^i$$

Stabilität von Mehrschrittverfahren

Definition 9.10: (Stabilität von Mehrschrittverfahren)

Ein Mehrschrittverfahren heißt *stabil*, falls alle (komplexen) Nullstellen seines zugehörigen charakteristischen Polynoms $\lambda_i, i = 0, ..., r$, mit $|\lambda_i| = 1$ Vielfachheit 1 haben und ansonsten $|\lambda_i| \leq 1$ gilt. Gibt es nur ein $\tilde{i} \in \{0, ..., r\}$ mit $|\lambda_{\tilde{i}}| = 1$ und gilt $\forall_{i \neq \tilde{i}} |\lambda_i| < 1$, so heißt das Verfahren *stark stabil*.

Satz 9.5: (Stabilität von Mehrschrittverfahren)

Ein Mehrschrittverfahren ist genau dann stabil, falls $\lim_{j \to \infty} \frac{\eta_j}{j} = 0$ für alle Startwerte $\eta_0, ..., \eta_{r-1} \in \mathbb{C}$, d.h. Definition 9.10 macht tatsächlich Sinn!

Zusammenhang: Konsistenz, Stabilität, Konvergenz

Satz 9.6: (Konvergenz $\Rightarrow$ Stabilität)

Falls ein Mehrschrittverfahren konvergent ist, und falls $\Phi(x, u; h; 0) \equiv 0$ (also für $F \equiv 0$), dann ist es auch stabil.

Satz 9.7: (Konsistenz + Stabilität $\Leftrightarrow$ Konvergenz)

Ein konsistentes Mehrschrittverfahren, wobei $\Phi(x, u; h; 0) \equiv 0$ und Φ der Lipschitz-Bedingung

$$\exists_{L>0} \ \forall_{x\in\Omega, h>0} \qquad |\Phi(x, \eta_r, ..., \eta_0; h; F) - \Phi(x, \tilde{\eta}_r, ..., \tilde{\eta}_0; h; F)|$$
$$\leq \quad L \sum_{i=0}^{r} |\eta_i - \tilde{\eta}_i|$$

genügt, ist genau dann für alle F konvergent, wenn es stabil ist.

Zweites charakteristisches Polynom von Mehrschrittverfahren

Definition 9.11: (Zweites charakteristisches Polynom)

Das *zweite charakteristische Polynom* $\xi : \mathbb{C} \to \mathbb{C}$ eines linearen Mehrschrittverfahrens

$$\sum_{i=0}^{r} a_i \eta_{j+i} = h \sum_{i=0}^{r} b_i F_{j+i}, \ j \in \mathbb{N}_0$$

ist definiert durch

$$\xi(\lambda) := \sum_{i=0}^{r} b_i \lambda^i$$

Konsistenzordnung bei Mehrschrittverfahren

Satz 9.8: (Konsistenz von Mehrschrittverfahren)

Ein Mehrschrittverfahren ist genau dann konsistent, wenn $\chi(1) = 0 \ \wedge \ \chi'(1) = \xi(1)$.

Satz 9.9: (Konsistenzordnung bei Mehrschrittverfahren)

Folgende Aussagen sind äquivalent:

(i) Ein Mehrschrittverfahren hat Konsistenzordnung p.

(ii) Es gilt $\sum_{i=0}^{r-1} \left(i^\ell a_i - \ell i^{\ell-1} b_i \right) = 0, \ \ell = 0, ..., p.$

(iii) Das Mehrschrittverfahren liefert für $F(x, u) := \ell x^{\ell-1}$ bei exakten Startwerden $\eta_i := x_i^\ell$ die exakte Lösung $u(x) = x^\ell$ für $\ell = 0, ..., p$.

Konsistenzordnung bei Mehrschrittverfahren

Satz 9.9: (Konsistenzordnung bei Mehrschrittverfahren, Forts.)

Folgende Aussagen sind äquivalent:

(iv) Für die spezielle GDGL $u'(x) = F(x,u) := u(x)$ gilt $\tau(x, \tilde{u}; h; F) = \mathcal{O}(h^p)$ für alle Lösungen $\tilde{u}$.

(v) Die Funktion $\varphi(x) := \chi(e^x) - x\xi(e^x)$ hat $x = 0$ als $p+1$-fache Nullstelle.

(vi) Die Funktion $Q(\lambda) := \frac{\chi(\lambda)}{\ln(\lambda)} - \xi(\lambda)$ hat $\lambda = 1$ als p-fache Nullstelle (beachte: $\frac{\chi(\lambda)}{\ln(\lambda)}$ ist stetig hebbar an der Stelle λ).

Satz 9.10: (Konvergenz $\Rightarrow$ Konsistenz)

Jedes konvergente Mehrschrittverfahren ist auch konsistent.

Fazit

1. Für **Einschrittverfahren** gilt:

   Konsistenz p-ter Ordnung + Lipschitz-Bedingung für Φ $\Leftrightarrow$ Konvergenz p-ter Ordnung

2. Für **Mehrschrittverfahren** gilt:

   Konsistenz p-ter Ordnung + Lipschitz-Bedingung für Φ + Stabilität $\Leftrightarrow$ Konvergenz p-ter Ordnung

Konsistenz und Stabilität sind mit einfachen Rechnungen mithilfe der charakteristischen Polynome nachweisbar!

Runge-Kutta-Verfahren

Beachte:

Runge-Kutta-Verfahren sind Einschrittverfahren !!!

Motivation: Entwickle Verfahren p-ter Ordnung mittels Taylorentwicklung von Δ und Φ sowie Koeffizientenvergleich!

Allgemeine Runge-Kutta-Verfahren

Definition 9.12: (Runge-Kutta-Verfahren)

Ein sog. *Runge-Kutta-Verfahrens* zur Lösung des Anfangswertproblems $u' = F(x,u), u(x_0) = u_0$, ist gegeben durch

$$\begin{aligned}
\eta_0 &:= u_0 \\
\eta_{i+1} &:= \eta_i + h\Phi(x,\eta;h;F), i \in \mathbb{N}_0, \\
\Phi(x,\eta;h;F) &:= \sum_{i=0}^{n} \gamma_i k_i(x,\eta;h;F), n \in \mathbb{N}, \\
k_0(x,\eta;h;F) &:= F(x,\eta) \\
k_1(x,\eta;h;F) &:= F(x+\alpha_1 h, \eta + h\beta_{10} k_0(x,\eta;h;F)) \\
&\dots \\
k_n(x,\eta;h;F) &:= F(x+\alpha_n h, \eta + \sum_{j=0}^{n-1} \beta_{nj} k_j(x,\eta;h;F))
\end{aligned}$$

Koeffizienten der Runge-Kutta-Verfahren

Die Koeffizienten
$\gamma_0, ..., \gamma_n; \alpha_1, ..., \alpha_n; \beta_{ij}, i = 0, ..., n; j = 0, ..., n-1$, ergeben sich aus einem unterbestimmten Gleichungssystem, d.h. es gibt unendlich viele Möglichkeiten, die Koeffizienten zu wählen. Aus der Konsistenzbedingung folgt als notwendige Bedingung $\sum_{i=0}^{n} \gamma_i = 1$.

Klassisches Runge-Kutta-Verfahren

Definition 9.13: (Klassisches Runge-Kutta-Verfahren)

Das sog. *klassische Runge-Kutta-Verfahren* zur Lösung des Anfangswertproblems $u' = F(x,u), u(x_0) = u_0$, ist gegeben durch

$$\begin{aligned}
\Phi(x,\eta;h;F) &:= \frac{1}{6}(k_0 + 2k_1 + 2k_2 + k_3) \\
k_0 &:= F(x,\eta) \\
k_1 &:= F\left(x + \frac{h}{2}, \eta + \frac{h}{2}k_0\right) \\
k_2 &:= F\left(x + \frac{h}{2}, \eta + \frac{h}{2}k_1\right) \\
k_3 &:= F\left(x + h, \eta + \frac{h}{2}k_2\right)
\end{aligned}$$

Klassisches Runge-Kutta-Verfahren

Es gilt

$$k_3 = F\left(x+h, \eta + hF\left(x+\frac{h}{2}, \eta + \frac{h}{2}\left(F(x+\frac{h}{2}, \eta + \frac{h}{2}F(x,\eta))\right)\right)\right)$$

Satz 9.11: (Klassisches Runge-Kutta-Verfahren)

Das klassische Runge-Kutta-Verfahren hat Konsistenzordnung 4.

Runge-Kutta-Koeffizientenschema

$$
\begin{array}{c|ccccc}
\alpha_1 & \beta_{10} & & & & \\
\alpha_2 & \beta_{20} & \beta_{21} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
\alpha_n & \beta_{n0} & \beta_{n1} & \cdots & \beta_{n,n-1} & \\
\hline
 & \gamma_0 & \gamma_1 & \cdots & \gamma_{n-1} & \gamma_n
\end{array}
$$

Differentialgleichungen höherer Ordnung

Betr. GDGL n-ter Ordnung der Form

$$u^{(n)} + a_{n-1}(x)u^{(n-1)} + \ldots + a_0(x)u = b(x)$$

(inhomogene GDGL mit nicht-konstanten Koeffizienten)
Anfangswertproblem: $u(x_0) = u_0$, $u'(x_0) = u_1, \ldots, u^{(n-1)}(x_0) = u_{n-1}$
Der Ansatz $y_1(x) := u(x), y_2(x) := u'(x), \ldots, y_n(x) := u^{(n-1)}(x)$
führt zu einem System von GDGLn erster Ordnung:

$$\begin{aligned}
y_1' &= y_2,\ y_1(x_0) = u_0 \\
y_2' &= y_3,\ y_2(x_0) = u_1 \\
&\vdots \\
y_n' &= -a_{n-1}(x)u^{(n-1)} - \ldots - a_0(x)u + b(x),\ y_n(x_0) = u_{n-1}
\end{aligned}$$

Fundamentalsystem von Lösungen

Seien nun die Koeffizienten a_i konstant und $b(x) = 0$ (homogenes System). Das Polynom

$$\chi(\lambda) := \lambda^n + a_{n-1}\lambda^{n-1} + ... + a_1\lambda + a_0$$

ist das sog. *charakteristische Polynom* der GDGL mit Nullstellen $\lambda_1, ..., \lambda_n \in \mathbb{C}$.
$u_i(x) := e^{\lambda_i x}, i = 1, ..., n$, sind Lösungen und bilden ein sog. *Fundamentalsystem*.

Fundamentalsystem von Lösungen

Nach dem Fundamentalsatz der Algebra ist, falls λ Nullstelle von χ, auch $\bar{\lambda}$ Nullstelle. Somit sind
$\mathrm{Re}(e^{\lambda_i x}), \mathrm{Im}(e^{\lambda_i x}), \mathrm{Re}(e^{\bar{\lambda}_i x}), \mathrm{Im}(e^{\bar{\lambda}_i x})$ Lösungen.
Falls λ_i mehrfache Nullstelle ist, so sind

$$\begin{aligned} u_i(x) &= e^{\lambda_i x} \\ u_{i+1} &= x e^{\lambda_i x} \\ u_{i+2} &= x^2 e^{\lambda_i x} \end{aligned}$$

usw. Lösungen.

Fundamentalsystem von Lösungen

Fundamentalsystem von Lösungen einer homogenen GDLG n-ter Ordnung mit konstanten Koeffizienten:

$$\begin{aligned} & e^{\lambda_1 x}, && xe^{\lambda_1 x}, \ldots, x^{n_1-1}e^{\lambda_1 x} \\ & e^{\lambda_2 x}, && xe^{\lambda_2 x}, \ldots, x^{n_2-1}e^{\lambda_2 x} \\ & \qquad \vdots \\ & e^{\lambda_n x}, && xe^{\lambda_n x}, \ldots, x^{n_n-1}e^{\lambda_n x} \end{aligned}$$

Dabei sind $n_i, i = 1, \ldots, n$, die Vielfachheiten der Nullstelle λ_i. Im Falle komplexer λ_i kommen Real- und Imaginärteile hinzu.

Praktische Vorgehensweise

Problem:

Nullstellenbestimmung von χ problematisch (Newton-Verfahren stark startwertabhängig, findet oftmals nicht alle Nullstellen, gerade nicht im Komplexen!)

Also:

Mache aus GDGL n-ter Ordnung ein System aus GDGLn erster Ordnung und löse diese von n bis 1 mit bekannten numerischen Verfahren!

Systeme gewöhnlicher Differentialgleichungen

Man ist in der Praxis auch oft an der Lösungen allgemeiner Systeme gewöhnlicher Differentialgleichungen erster Ordnung zu lösen.
Hierzu braucht man Verfahren zur Eigenwert- und Eigenvektorbestimmung (z.B. QR-Verfahren, siehe Abschnitt 6.4).

Randwertprobleme

Definition 9.14: (Randwertproblem)

Bei einem sog. *Randwertproblem (RP)* werden der Lösung $u : [a, b] \to \mathbb{R}$ an den Intervallrändern Werte vorgeschrieben:

$$u(a) = \alpha, \ u(b) = \beta \in \mathbb{R}$$

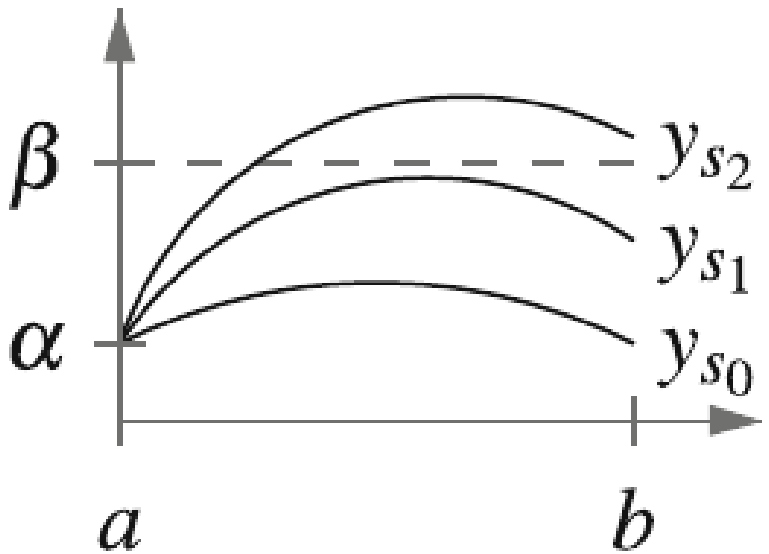
Dies ist selbstverständlich auch für Ableitungen möglich.

Wir betrachten hier nur das RP

$$u''(x) = F(x, u, u'), x \in (a, b), \ u(a) = \alpha, \ u(b) = \beta$$

Schießverfahren

Motivation: Beschreibe die Flugbahn eines Balles, der bei a auf der Höhe α so geworfen wird, daß er bei b die Höhe β hat. Die Steigung der Tangente an u an der Stelle a sei durch einen zusätzlichen Parameter $s \in \mathbb{R}$ gegeben.



Lösung des RPs durch mehrere APe

Betrachte für verschiedene $s \in \mathbb{R}$ das AP

$$y''(x) = F(x, y, y'), x \in (a, b)\ y(a) = \alpha,\ y'(a) = s$$

welches mit bekannten Methoden gelöst werden kann. s muß so gewählt werden, daß die Funktion

$$f : \mathbb{R}^2 \to \mathbb{R},\ y(s, b) - \beta$$

bei s eine Nullstelle hat (z.B. mit dem Bisektionsverfahren).
$y(s, b)$ ist die Lösung des obigen von s abhängigen APs.

Partielle Differentialgleichungen

Im Gegensatz zu gewöhnlichen Differentialgleichungen: Betrachte mehrdimensionale Funktion $u : \mathbb{R}^n \to \mathbb{R}$ mit partiellen Ableitungen $D^k u$ vom Grad k, $k \in \mathbb{N}_0$.
Betrachte hier nur Funktionen $u : \mathbb{R}^2 \to \mathbb{R}, (x, y) \mapsto z = u(x, y)$, und schreibe verkürzt

$$\frac{\partial^{i+j} u}{\partial^i x \partial^j y} =: u_{\underbrace{x \ldots x}_{i-\text{mal}} \underbrace{y \ldots y}_{j-\text{mal}}}$$

also z.B. $\frac{\partial^2 u}{\partial x^2} = u_{xx}$, $\frac{\partial^2 u}{\partial x \partial y} = u_{xy}$, $\frac{\partial^2 u}{\partial y^2} = u_{yy}$

Partielle Differentialgleichung zweiter Ordnung

Definition 10.1: (Partielle Differentialgleichung zweiter Ordnung)

Sei $u : \mathbb{R}^n \to \mathbb{R}$ mindestens k-mal partiell differenzierbar. Ein Ausdruck der Form

$$F(x, u, Du, D^2u, ..., D^k u) = 0$$

heißt *partielle Differentialgleichung zweiter Ordnung*. Für $u : \mathbb{R}^2 \to \mathbb{R}$ gilt

$$F(x, u, u_x, u_y, u_{xx}, u_{xy}, u_{yy}, ...) = 0$$

Partielle Differentialgleichungen (PDGLn) 1. Ordnung

Betrachte eine sog. *quasi-lineare PDGL 1. Ordnung:*

$$a(x,y,u)u_x + b(x,y,u)u_y = c(x,y,u)$$

(quasi-linear, da Koeffizienten nicht konstant!) Insbesondere:

$$a(x,y)u_x + b(x,y)u_y = c_0(x,y)u + c_1(x,y)$$

bzw.

$$\left\langle \begin{pmatrix} a \\ b \\ c_0 u + c_1 \end{pmatrix}, \begin{pmatrix} u_x \\ u_y \\ -1 \end{pmatrix} \right\rangle = 0$$

Charakteristische Kurven

Der Vektor $(u_x, u_y, -1)^T$ ist ein Normalenvektor auf die Fläche $(x, y, u(x, y))$. Also liegt der Vektor $(a, b, c_0 u + c_1)$ in der Tangentialebene.
Betrachte die parametrisierte Kurve $(x(t), y(t), u(t))$. Es gilt dann

$$\begin{aligned}(x'(t), y'(t), u'(t)) &= (a(x(t), y(t)), b(x(t), y(t)), \\ &\quad c_0(x(t), y(t))u(t) + c_1(x(t), y(t)))\end{aligned}$$

wobei $u(t) = u(x(t), y(t))$, und löse die gewöhnlichen Differentialgleichungen

$$\begin{aligned}x'(t) &= a(x(t), y(t)) \\ y'(t) &= b(x(t), y(t)) \\ u'(t) &= c_0(x(t), y(t))u(t) + c_1(x(t), y(t))\end{aligned}$$

(sog. *charakteristische Gleichungen* der PDGL). Lösungen: *charakteristische Kurven*.

Methode der Charakteristiken

- Brauche Anfangsbedingung für die Kurven: Alle Anfangspunkte der charakteristischen Kurven sollen auf einer Anfangskurve $\Gamma(s)$ liegen!
- Jede Kurve $(x(t), y(t), u(t))$ entwickelt sich dann aus einem anderen Anfangspunkt von $\Gamma(s)$.
- Betr. weitere Parameterdarstellung $(x(t,s), y(t,s), u(t,s))$ mit Anfangsbedingungen $x(0,s) = x_0(s), y(0,s) = y_0(s), u(0,s) = u_0(s)$, d.h., jeder Anfangspunkt liegt auf der von s beschriebenen Kurve!
- Für jeden Anfangspunkt aus $\Gamma(s)$ erhält man eine Anfangskurve. Alle diese Anfangskurven (sog. *Charakteristiken*) bilden zusammengeklebt die Lösung des AWPs, d.h. die Lösung der PDGL (sog. *Integralkurve*).

Prominentes Beispiel: Transportgleichung

$$u_t + cu_x = 0$$

(t: Zeit, c : Geschwindigkeit, x : *Ort*), z.B. Transport eines im Wasser gelösten Stoffs mit Wasserströmung (ohne Diffusion des Stoffs an sich).
Die Lösung u ist die Konzentration des Stoffs in x-Richtung zu bestimmtem Zeitpunkt $t = t_0$ (sog. *Snapshot*). Somit erhält man eine Kurve $u(x,t)|_{t=t_0}$ (Charakteristik). Alle diese Charakteristiken ergeben die Lösung der Transportgleichung.

Partielle Differentialgleichungen 2. Ordnung

Betrachte hier nur quasi-lineare PDGLn 2. Ordnung:

Definition 10.2: (Quasi-lineare partielle Differentialgleichung zweiter Ordnung)

Sei $u : \mathbb{R} \to \mathbb{R}$ mindestens zweimal partiell differenzierbar. Eine *quasi-lineare partielle Differentialgleichung zweiter Ordnung* ist gegeben durch

$$-\sum_{i,j=1}^{k} a_{ij}(x) u_{x_i x_j}(x) + \sum_{i=1}^{k} b_i(x) u_{x_i}(x) + c(x)u(x) = f(x)$$

Sind a_{ij}, bi und c unabhängig von x, so spricht man von einer *PDGL mit konstanten Koeffizienten*. Ist $f \equiv 0$, so ist die PDGL *homogen*, ansonsten *inhomogen*.

Elliptische, parabolische und hyperbolische PDGLn

Nach dem Satz von Schwarz gilt $\forall_{i,j}\ u_{x_i x_j} = u_{x_j x_i}$. Gehe also von $a_{ij}(x) = a_{ji}(x)$ aus!

Definition 10.3: (Elliptische/Parabolische/Hyperbolische PDGLn)

Sei $\Omega \subseteq \mathbb{R}^n$ ein zulässiges Gebiet. Sei $A(x) := (a_{ij}(x))_{i,j=1,\dots,k}$ die Koeffizientenmatrix aus Def. 10.2.

(i) Die PDGL aus Def. 10.2 heißt *elliptisch*, falls $A(x)$ für alle $x \in \Omega$ positiv definit ist.

(ii) Sie heißt *parabolisch*, falls $A(x)$ für alle $x \in \Omega$ positiv semidefinit, aber nicht definit ist ist und $rg\left(A(x), (b_1(x), \dots, b_k(x))^T\right) = n$.

(iii) Sie heißt *hyperbolisch*, falls $A(x)$ für alle $x \in \Omega$ genau einen negativen und sonst nur positive Eigenwerte hat.

Woher kommen diese Bezeichnungen?

Betr. wichtigsten Spezialfall: PDGL 2. Ordnung:

$$a(x,y)u_{xx}+b(x,y)u_{xy}+c(x,y)u_{yy}+d(x,y)u_x+e(x,y)u_y+f(x,y)u=g$$

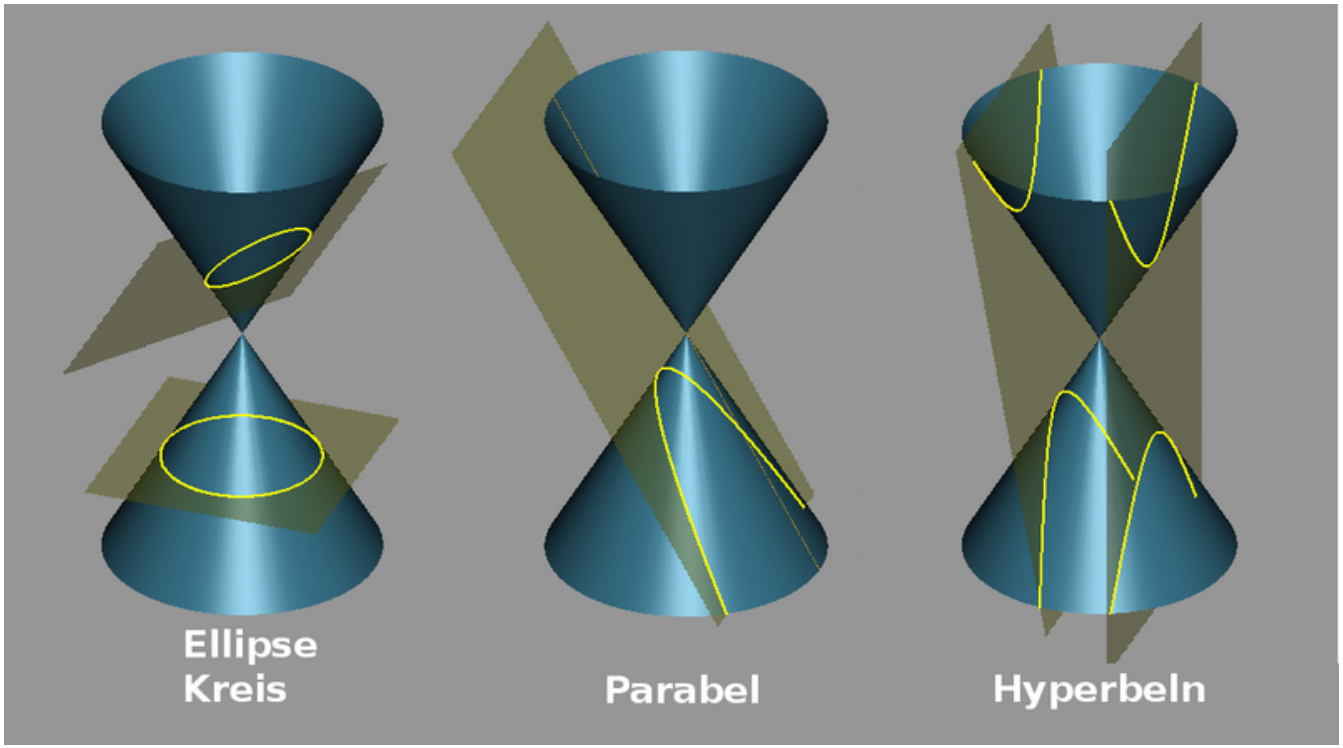
und die Determinante von $A(x,y)$: $D(x,y) := \det(A(x,y)) =$

$$\det\begin{pmatrix} a(x,y) & \frac{b(x,y)}{2} \\ \frac{b(x,y)}{2} & c(x,y) \end{pmatrix} = a(x,y)c(x,y) - \frac{b^2(x,y)}{4}.$$

Die PDGL ist

- elliptisch, falls $D(x,y) > 0$
- parabolisch, falls $D(x,y) = 0$
- hyperbolisch, falls $D(x,y) < 0$

Kegelschnitte



Kegelschnittgleichung

Die Kegelschnittgleichung ist ähnlich der PDGL 2. Ordnung und lautet:

$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

Die Gleichung beschreibt den Schnitt eines Doppelkegels mit einer Ebene. Der Schnitt ist

- eine Ellipse (bzw. ein Kreis), falls $ac - \frac{b^2}{4} > 0$
- eine Parabel, falls $ac - \frac{b^2}{4} = 0$
- eine Hyperbel, falls $ac - \frac{b^2}{4} < 0$

Laplace-Gleichung als elliptische PDGL

Definition 10.4: (Laplace-Operator)

Sei $u : \mathbb{R}^n \to \mathbb{R}$ mindestens zweimal partiell differenzierbar. Dann ist der *Laplace-Operator* gegen durch

$$\Delta u(x) := \sum_{i=1}^{n} \frac{\partial^2 u}{\partial x_i^2}$$

Die sog. *Laplace-Gleichung* $\Delta u = 0$ ist elliptisch. Mithilfe der Fourier-Reihe ergibt sich die analytische (exakte) Lösung

$$u(x, y) = a_0 + \sum_{k=1}^{\infty} r^k (a_k \cos(k\varphi) + b_k \sin(k\varphi))$$

mit **Randbedingung** $x^2 + y^2 \leq 1$ ($x = r\cos(\varphi), y = r\sin(\varphi)$ Polarkoordinaten, $a_k, k = 0, .., n$; $b_k, k = 1, .., n$, Fourierkoeff.)

Poisson-Gleichung als elliptische PDGL

Inhomogene Variante der Laplace-Gleichung:

$$-\Delta u(x) = f(x)$$

Anwendung in der Physik: f Ladungsdichte, u Spannung (negative Änderung der Spannung führt zu gewisser Ladungsdichte)

Wärmeleitungsgleichung als parabolische PDGL

Zeitabhängigkeit:

$$u_t = u_{xx}$$

u ist die Temperatur z.B. auf einem Stab. Benötige sowohl Anfangstemperatur als auch die Temperaturen auf den Rändern des Stabs (sog. *Anfangsrandwertproblem (ARWP)*).
Analytische Lösung:

$$u(x,t) = \sum_{k=1}^{\infty} a_k e^{-k^2 t} \sin(kx)$$

Unendlich langer Stab, d.h. Randwerte entfallen:

$$u(x,t) = \frac{1}{2\sqrt{\pi t}} \int_{-\infty}^{\infty} e^{-\frac{\xi^2}{4t}} f(x-\xi)\, d\xi, \;\; f(x) = u(x,0)$$

Wellengleichung als hyperbolische PDGL

Allgemeine Form:

$$u_{tt} = c^2 \Delta u$$

c Konstante, u Druck (2D: Schwingung einer Membran)
Eindimensionaler Fall (Schwingung eines Stabs):

$$u_{tt} = u_{xx}$$

Analytische Lösung:

$$u(x,t) = \frac{1}{2}(f(x+t) + f(x-t)) + \frac{1}{2}\int_{x-t}^{x+t} g(\xi)\, d\xi$$

Anfangswertproblem: $u(x,0) = f(x), u(x,0) = g(x)$

Korrekt gestellte Probleme

Man sieht: Ja nach Typ der PDGL muß die Problemstellung anders gewählt werden:

- elliptisch: RWP
- parabolisch: ARWP
- hyperbolisch: AWP

Definition 10.4: (korrekt gestellt)

Ein Problem heißt *korrekt* oder auch *sachgemäß* gestellt (engl. *well-posed*), falls eine Lösung existiert, die eindeutig ist und stetig von den vorgegebenen Daten abhängt. Ansonsten heißt das Problem *schlecht gestellt* (engl. *ill-posed* oder *improperly posed*).

Diskretisierung und Finite Differenzen

Betr. die Randwertaufgabe $-u''(x) = f(x), x \in [0,1]$
(GDLG 2. Ordnung), mit f auf $[0,1]$ stetig und Lösung u auf $[0,1]$ mindestens zweimal stetig differenzierbar,
Randbedingungen: $u(0) = g_0,\ u(1) = g_1,\ g_0, g_1 \in \mathbb{R}$.

Numerische Lösung u_h auf äquidistantem Gitter Ω_h (*Diskretisierung*):

$x_0 = 0,\ x_i = x_0 + ih,\ i = 0, ..., n,\ h = \frac{1}{n},\ x_n = 1$

$$-u''(x_i) = \frac{1}{h^2}\left(-u_h(x_{i-1}) + 2u_h(x) - u_h(x_{i+1})\right) = f(x_i),\ i = 1, ..., n-1$$

$$u_h(x_0) = g_0,\ u_h(x_n) = g_1$$

(sog. *Finite Differenzen*)

Eindimensionale Differenzensterne

Der diskrete Vektor

$$(u_h(x_0), u_h(x_1), ..., u_h(x_n))^T$$

ist Lösung eines $(n+1) \times (n+1)$-Linearen Gleichungssystems, bzw. nach Elimination der Randbedingungen ist der Vektor

$$(u_h(x_1), u_h(x_2), ..., u_h(x_{n-1}))^T$$

Lösung eines $(n+1) \times (n+1)$-Linearen Gleichungssystems.
Andere Schreibweise:

$$\begin{bmatrix} -1 & 2 & -1 \end{bmatrix}_h \cdot u_h = r_h$$

(sog. *Differenzenstern*, engl *stencil*)

Zweidimensionale Poissongleichung

Betrachte Poissiongleichung mit sog. *Dirichlet-Randbedingungen*:

$$\begin{aligned} -\Delta u(x,y) &= f(x,y),\ (x,y) \in \Omega := (0,1)^2 \subseteq \mathbb{R}^2 \\ u(x,y) &= g(x,y),\ (x,y) \in \partial\Omega \end{aligned}$$

mit $\partial\Omega := \{(0,y)|y \in [0,1]\} \cup \{(x,0)|x \in [0,1]\}$
$\cup\{(1,y)|y \in [0,1]\} \cup \{(x,1)|x \in [0,1]\}$ (Rand von Ω),
f stetig auf $\bar{\Omega} := \Omega \cup \partial\Omega$, g stetig auf $\partial\Omega$, Lösung u mind. zweimal stetig differenzierbar auf Ω und stetig auf $\bar{\Omega}$

Zweidimensionale Differenzensterne

Die Diskretisierung des Laplace-Operators liefert:

$$\begin{aligned}\Delta_h u_h(x,y) &\approx \frac{1}{h^2}(u_h(x-h,y)+u_h(x+h,y)+u_h(x,y-h)\\ &+ u_h(x,y+h)-4u(x,y))\end{aligned}$$

Dies liefert den zweidimensionalen Differenzenstern

$$\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}_h$$

Die Gestalt der Matrix des zugehörigen Linearen Gleichungssystems hängt von der Numerierung der Gitterpunkte ab.

Finite Elemente und Finite Volumina

Brauche Erweiterung des Lösungsbegriffs von PDGLn (sog. *schwache Lösungen*, im Gegensatz zu *klassischen Lösungen*), da man i.a. Differenzierbarket der Lösung nicht voraussetzen kann! Auch kann die Glattheit des Gebiets i.a. nicht vorausgesetzt werden (Gebiete sollten z.B. auch Ecken haben dürfen)!

In praktischen Anwendungen gibt es dennoch Lösungen!

Finite-Elemente-Methode basiert auf erweitertem Lösungsbegriff!

Partielle Integration

Partielle Integration liefert für RWP
$u'' = f, x \in (0,1), u(0) = u(1) = 0$ für alle Funktionen v, die auf $(0,1)$ integrierbar sind mit $v(0) = v(1) = 0$:

$$\int_0^1 u'(x)v'(x)\,dx = \int_0^1 f(x)v(x)\,dx$$

d.h., man braucht nur noch die Voraussetzung, daß sowohl $u'v'$ als auch fv auf $(0,1)$ integrierbar sind. Die Existenz von u'' wird also gar nicht mehr verlangt!

Frage:

Welche Eigenschaften muß (die sog. *Testfunktion*) v erfüllen?

Antwort:

Brauche dazu Theorien aus der Funktionalanalysis!

Lebesgue-Integrierbarkeit

- beruht auf sog. *einfachen Funktionen*, d.h. nichtnegativen Funktionen, die nur endliche Funktionswerte annehmen dürfen
- Treppenfunktionen auf einem Intervall $[a, b]$ sind Teilmenge der Menge der einfachen Funktionen; das sog. *Lebesgue-Integral* ist eine Verallgemeinerung des Riemann-Integrals (Integral ist Grenzwert einer Folge von Integralen einfacher Funktionen)
- Die sog. *Dirichletsche Funktion*

$$g : [0,1] \to \{0,1\},\ g(x) = \begin{cases} 1, & x \in \mathbb{Q} \\ 0, & \text{sonst.} \end{cases}$$

ist Lebesgue-integrierbar mit Integralwert 0, aber nicht Riemann-integrierbar. Jede Riemann-integrierbare Funktion ist auch Lebesgue-integrierbar mit demselben Integralwert.

L^p-Räume

$$
\begin{aligned}
v \in L^p(a,b) \quad &:\Leftrightarrow \quad \int_a^b |v(x)|^p \, dx < \infty, \;\; p \in [1,\infty) \\
v \in L^\infty(a,b) \quad &:\Leftrightarrow \quad \operatorname{ess} \sup_{x \in (a,b)} |v(x)| := \inf_{\mu(N)} \sup_{(a,b)\setminus N} |v(x)| < \infty \\
\mu(N) = 0 \quad &:\Leftrightarrow \quad \int_a^b v(x) \, dx = 0
\end{aligned}
$$

d.h. N ist eine sog. *Nullmenge* bzgl. des *Lebesgue-Maßes* μ. Man sagt: $v_1(x) = v_2(x)$ *fast überall*, falls $v_1(x) = v_2(x)$ außer auf einer Nullmenge. Für die Dirichletsche Funktion gilt $g(x) = 0$ fast überall, denn einzelne Punkte sind Nullmengen.

L^p-Normen

$$||v||_p := \left(\int_a^b ||v(x)||^p \, dx \right)^{\frac{1}{p}}$$

$$||v||_\infty = \operatorname{ess} \sup_{x \in (a,b)} |v(x)|$$

Falls es ein stetiges $\tilde{v}$ gibt mit $\tilde{v} = v$ fast überall, gilt nach dem Satz von Weierstraß (Max/Min von stetigen Funktionene werden auf kompakten Mengen angenommen):

$$||v||_\infty = \max_{x \in (a,b)} |\tilde{v}(x)|$$

Skalarprodukt: $(u, v) := \int_a^b u(x)v(x) \, dx$

Die Räume $L^1_{\text{loc}}(a,b)$ und $C_0^\infty(a,b)$

Definition 10.6: $(L^1_{\text{loc}}(a,b))$

$L^1_{\text{loc}}(a,b)$ ist der Raum der auf jeder kompakten Teilmenge von (a,b) Lebesgue-integrierbaren Funktionen. Es gilt
$v \in L^1_{\text{loc}}(a,b) \;\Leftrightarrow\; \forall_{[a',b']\subseteq[a,b]}\; v \in L^1(a,b)$

Definition 10.7: (Träger einer Funktion)

Der *Träger* einer Funktion $f : (a,b) \to \mathbb{R}$ ist definiert als

$$\operatorname{supp}(f) := \overline{\{x \in (a,b) \;\mid\; f(x) \neq 0\}}$$

Definition 10.8: $(C_0^\infty(a,b))$

$C_0^\infty(a,b)$ ist der Raum der auf (a,b) unendlich oft differenzierbaren Funktionen mit kompaktem Träger.

Schwache Ableitungen

Definition 10.9: (Schwache Ableitung)

(i) Seien $u, v \in L^1_{\text{loc}}(a,b)$. Es gelte für $\varphi \in C_0^\infty(a,b)$

$$\int_a^b u(x)\varphi'(x)\,dx = -\int_a^b v(x)\varphi(x)\,dx$$

Dann heißt $v(x)$ *schwache Ableitung* von u. Man schreibt dann auch wie gewohnt $u' = v$.

(ii) Sei $\alpha = (\alpha_1, ..., \alpha_n) \in \mathbb{N}_0^n$ ein Multiindex. Seien $u, v \in L^1_{\text{loc}}(\Omega)$. Es gelte für alle $\varphi \in C_0^\infty(\Omega)$

$$\int_\Omega u D^\alpha \varphi = (-1)^{|\alpha|} \int_\Omega v\varphi$$

Dann heißt v *mehrdimensionale schwache Ableitung* von u.

Sobolev-Räume

Definition 10.10: (Sobolev-Raum)

(i) Der Raum
$W^{k,p}(a,b) := \{u \in L^1_{\text{loc}}(a,b) \mid u^{(i)} \in L^p(a,b),\ p \in [1,\infty]\}$,
wobei die $u^{(i)}$ die schwachen Ableitungen von u sind, heißt *Sobolev-Raum*. Schreibe kurz $H^k(a,b)$ für $W^{k,2}(a,b)$.

(ii) Der Raum $W^{k,p}(\Omega) := \{u \in L^1_{\text{loc}}(a,b) \mid ||u||_{W^{k,p}(\Omega)} < \infty\}$ mit der *Sobolev-Norm*

$$||u||_{W^{k,p}(\Omega)} := \left(\sum_{|\alpha| \le k} ||D^\alpha u||_p^p \right)^{\frac{1}{p}},\ p \in [1,\infty)$$

(p-Normen der schwachen Ableitungen $D^\alpha u$) bzw.
$||u||_{W^{k,\infty}(\Omega)} := \max\limits_{|\alpha| \le k} ||D^\alpha u||_\infty$, heißt *mehrdimensionaler Sobolev-Raum*. Schreibe kurz $H^k(\Omega)$ für $W^{k,2}(\Omega)$.

Sobolev-Räume für partielle Differentialgleichungen

Also:

Was ist ein Sobolev-Raum?

Wichtig hier:

Ein Sobolev-Raum ist ein Raum, in dem alle schwachen Ableitungen bis zur Ordnung k existieren und in der p-Norm beschränkt sind. Schwache Lösungen von partiellen Differentialgleichungen leben in Sobolev-Räumen!

Schwache Formulierung einer Randwertaufgabe

Für das Randwertproblem

$$-au''(x)+b(x)u'(x)+c(x)u(x) = f(x),\ x \in (0,1),\ u(0) = u(1) = 0$$

nennt man die aus partieller Integration entstehende Integralformulierung

$$\int_0^1 (-au''(x) + b(x)u'(x) + c(x)u(x))v(x)\,dx = \int_0^1 f(x)v(x)\,dx$$

auch *schwache Formulierung*.

Variationsproblem einer Randwertaufgabe

Satz 10.1: (Variationsproblem einer Randwertaufgabe)

Jede klassische Lösung der mehrdimensionalen Randwertaufgabe

$$-\sum_{i,k} \partial_i(a_{ik}\partial_k u) + a_0 u = f,\ x \in \Omega,\ u = 0,\ x \in \partial\Omega,$$

ist Lösung des Variationsproblems

$$J(v) := \min \int_\Omega \frac{1}{2}\sum_{i,k} a_{ik}\partial_i v \partial_k v + \frac{1}{2}a_0 v^2 - fv\, dx$$

Existenzsatz von Lax-Milgram

Betrachte die Bilinearform

$$a(u,v) := \int_\Omega \sum_{i,k} a_{ik}\partial_i u \partial_k v + a_0 uv \, dx$$

sowie das lineare Funktional

$$\langle \ell, v\rangle := \int_\Omega fv \, dx$$

Satz 10.2: (Lax-Milgram)

Das Variationsproblem min $J(v) = \frac{1}{2}a(v,v) - \langle \ell, v\rangle$ ist lösbar.

Also ist die RWA zumindest schwach lösbar, und die schwache Lösung ergibt sich als Lösung der schwachen Formulierung, also des Variationsproblems.

Diskretisierung des Sobolev-Raums

Betrachte statt der klassischen RWA die entsprechende schwache Formulierung und minimiere $J(v)$ in einem Sobolev-Raum.
Sei S_h eine Diskretisierung des Sobolev-Raums. Nach dem Charakterisierungssatz der Variationsrechnung ist u_h genau dann eine Lösung in S_h, wenn $a(u_h, v) = \langle \ell, v \rangle$ für alle $v \in S_h$.
Sei $\{\Psi_1, ..., \Psi_N\}$ eine Basis von S_h. Betrachte das Problem

$$a(u_h, \Psi_i) = \langle \ell, \Psi_i \rangle, \ i = 1, ..., N$$

Die Lösung u ist eine Linearkombination der Basisfunktionen:

$$u_h = \sum_{k=1}^{N} z_k \Psi_k$$

Finite Elemente

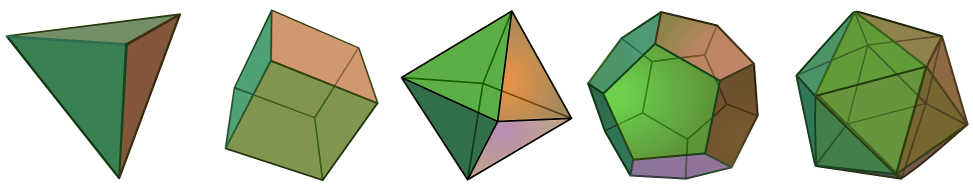
Der Ansatz $u_h = \sum_{k=1}^{N} z_k \Psi_k$ führt zu dem LGS

$$\sum_{k=1}^{N} a(\Psi_k, \Psi_i) z_k = \langle \ell, \Psi_i \rangle = \int_\Omega f \Psi_i \, dx, i = 1, ..., N$$

Kurzschreibweise: $Az = b, A = (a_{ki}) = a(\Psi_k, \Psi_i), b_i = \langle \ell, \Psi_i \rangle$.
Die Ψ_i werden als stückweise Polynome gewählt, d.h., das Gebiet Ω wird in Teilgebiete (sog. *Finite Elemente*) zerlegt, auf denen die Basisfunktionen Ψ_i Polynome sind.

Finite Elemente in der Praxis

- Zerlege das Gebiet Ω und endlich viele Teilgebiete (Finite Elemente)
- Basisfunktionen sind Polynome auf jedem Teilgebiet
- Teilgebiete:
  - 2D: Dreiecke, Vierecke, Vielecke, Polygone, ...
  - 3D: Tetraeder, Würfel, Quader, Oktaeder, Ikosaeder, ...



Zulässige Zerlegungen

Definition 10.11: (Zulässige Zerlegung)

Eine Zerlegung $\mathcal{T} = \{T_1, ..., T_M\}, M \in \mathbb{N}$, von Ω heißt *zulässig*, wenn die folgenden Eigenschaften erfüllt sind:

(i) $\bar{\Omega} = \bigcup_{i=1}^{M} T_i$

(ii) $T_i \cap T_j = \{x\} \Rightarrow$ x ist Eckpunkt sowohl von T_i als auch von T_j

(iii) $T_i \cap T_j = \mathcal{M}, \ |\mathcal{M}| > 1, \Rightarrow \mathcal{M}$ ist Kante sowohl von T_i als auch von T_j

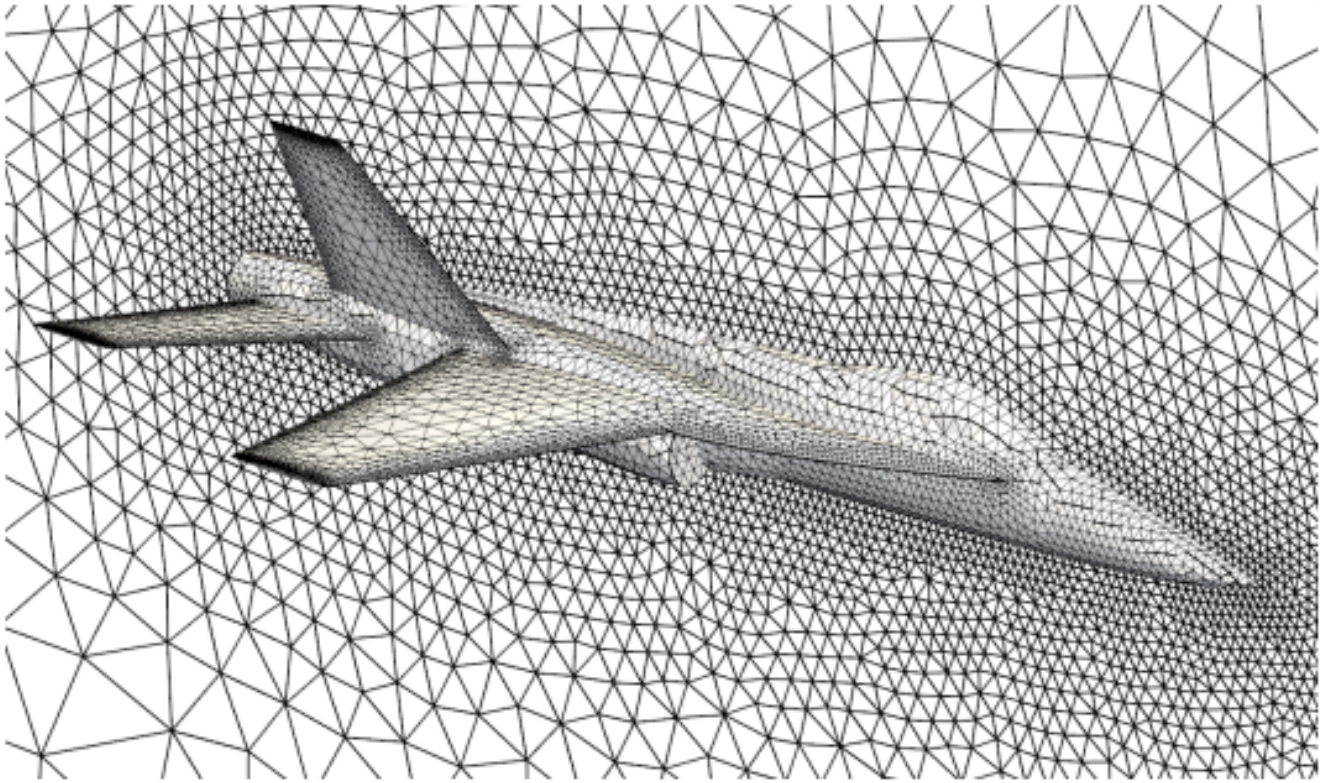
Nodale Basis

Definition 10.12: (Nodale Basis)

Sei $\mathcal{M}$ eine Menge von Punkten so, daß die Werte der Finiten Elementfunktionen auf $\mathcal{M}$ bekannt sind. Diejenigen Funktionen, die an genau einem Punkt von $\mathcal{M}$ einen von Null verschiedenen Wert annehmen, bilden die sog. *nodale Basis*.

Anwendung der Finite-Elemente-Methode

Simulation der Luftströmung eines Flugzeugs:



Zugrundeliegende partielle Differentialgleichungen:
Navier-Stokes-Gleichungen

Finite Volumina

- Basisfunktionen leben nicht auf den Randpunkten einer Zelle, sondern auf den Mittelpunkten (sog. *Zellkernen*)
- Methode der Finiten Volumina wird insbesondere für partielle Differentialgleichungen angewandt, denen ein Erhaltungssatz zugrundeliegt

Typeneinteilung bzgl. numerischer Verfahren

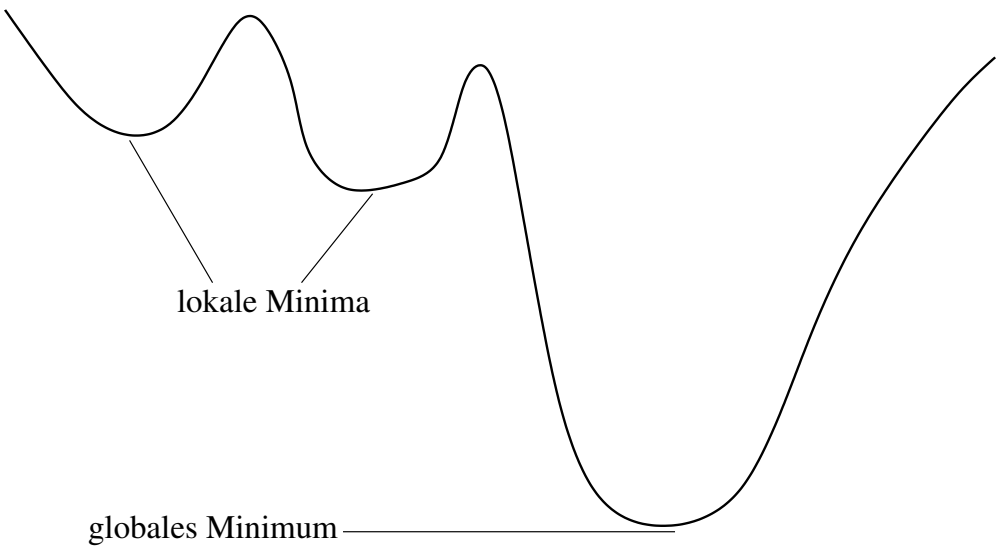
Bei einfachen PDGLn wird oftmals die Finite-Differenzen-Methode angewandt, ansonsten gilt:

- elliptisch: Finite Elemente/Finite Volumina
- parabolisch: Liniensuchverfahren
- hyperbolisch: Methode der Charakteristiken (Rückführung auf GDLGn)

Numerische Optimierung

Problemstellung:

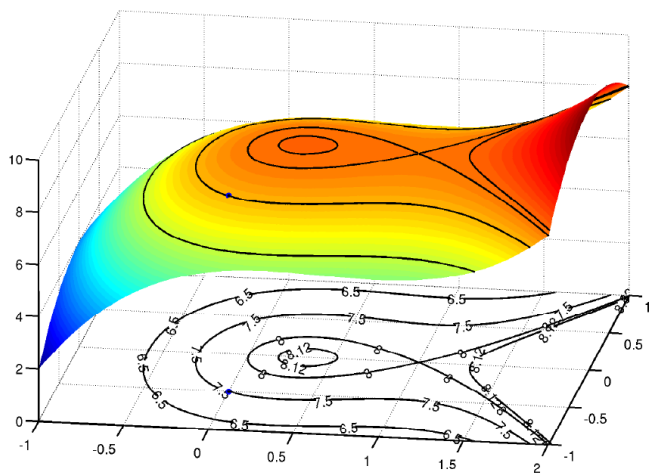
Minimiere eine Funktion $f : \mathbb{R}^n \to \mathbb{R}$ und finde möglichst das **globale** Minimum.

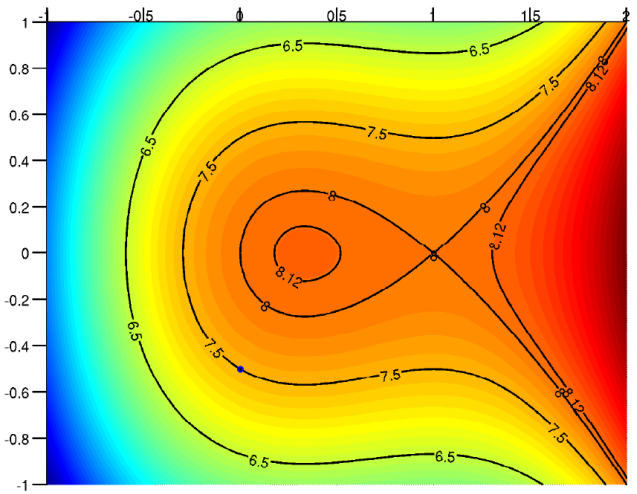


Extrema und Sattelpunkte im $\mathbb{R}^n$

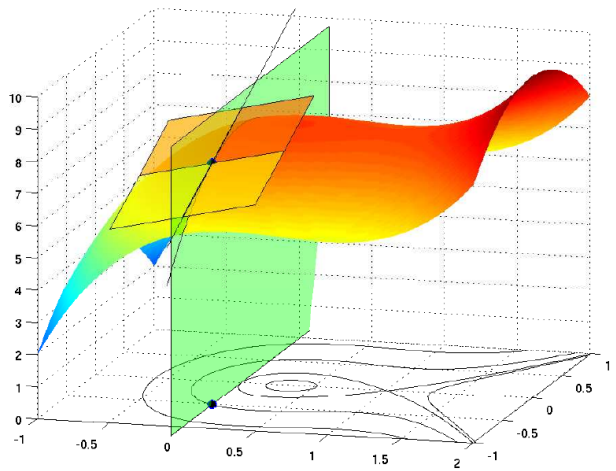
Ein *stationärer Punkt* ist ein Punkt $x \in \mathbb{R}^n$ mit $\nabla f(x) = 0$. Analog zu $\mathbb{R}$ ist die notwendige Bedingung für die Existenz eines lokalen Extremums am Punkt x, daß x ein stationärer Punkt ist.
Für die Entscheidung, ob es sich dabei um ein Minimum, ein Maximum oder einen Sattelpunkt handelt, braucht man wieder die 2. Ableitung (Hesse-Matrix).

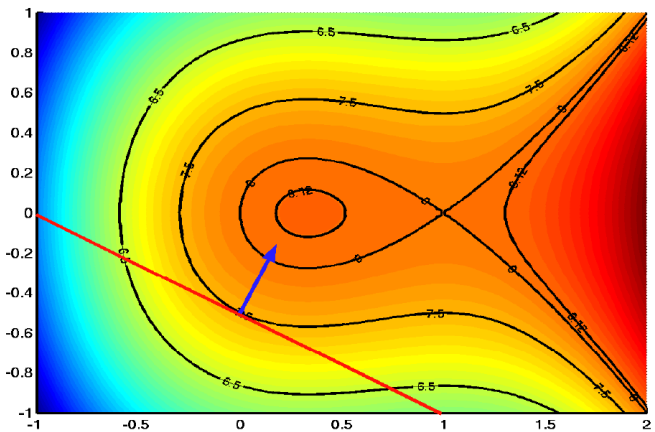
Funktionen im $\mathbb{R}^n$: Höhenlinien



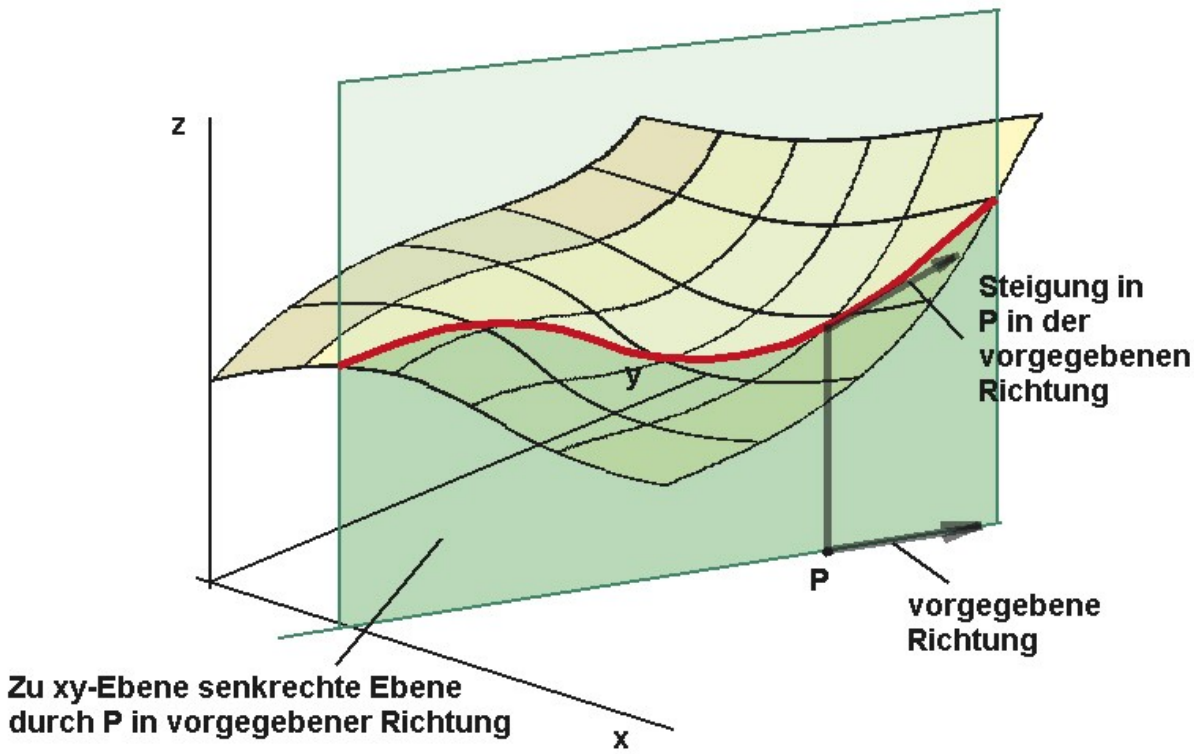


Gradient


10
9
8
7
6
5
4
3
2
1
0
-1
-0.5
0
0.5
1
1.5
2
-1
-0.5
0
0.5
1

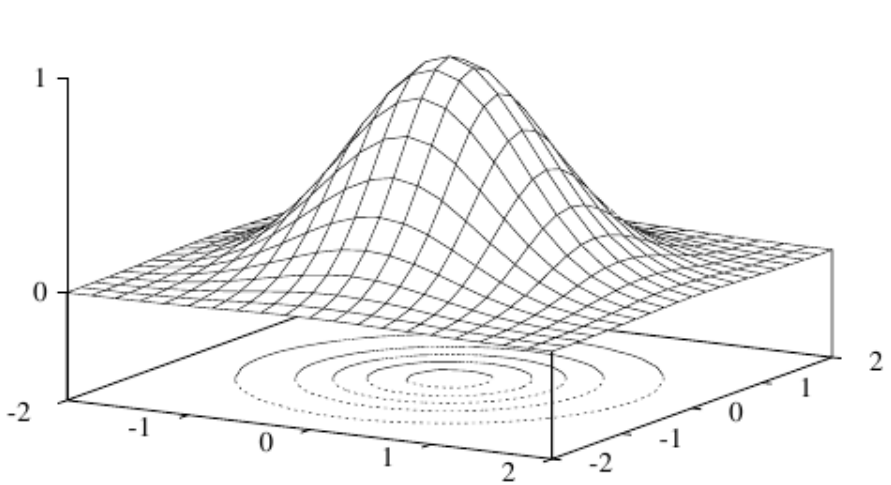

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
1.5
2

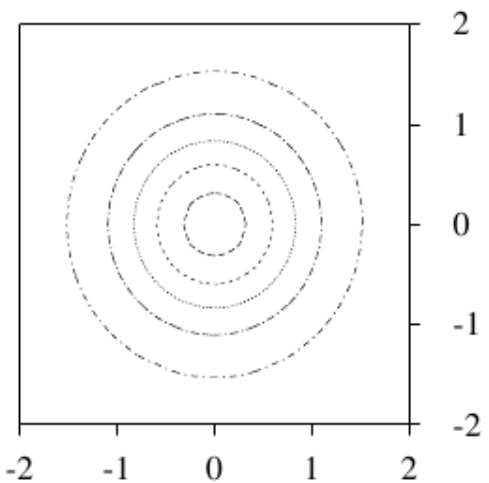
Richtungsableitung



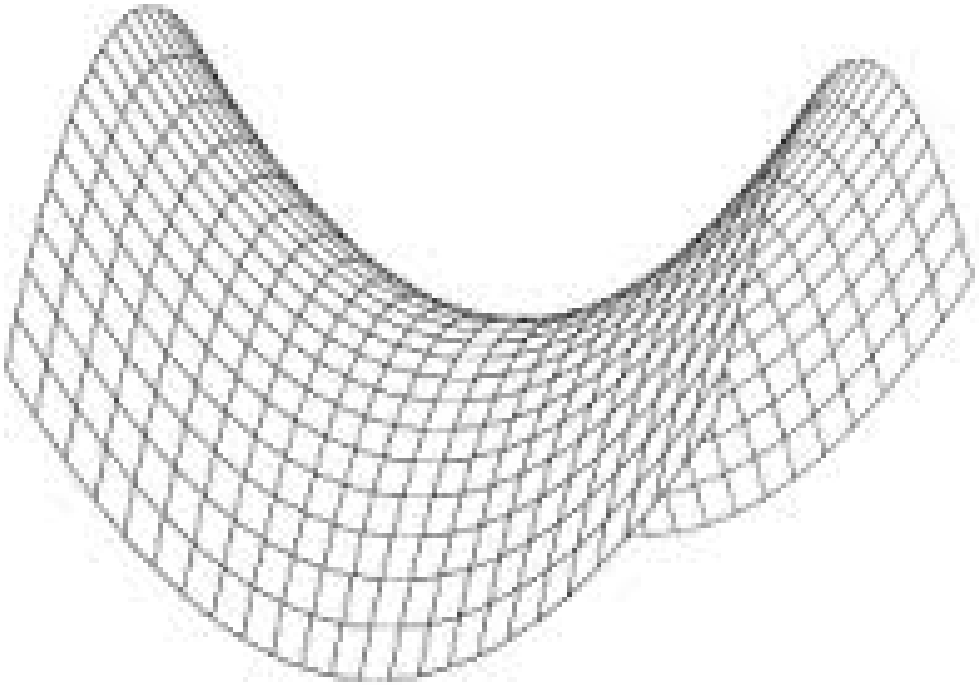
Der Gradient zeigt in Richtung des **steilsten Anstiegs**.

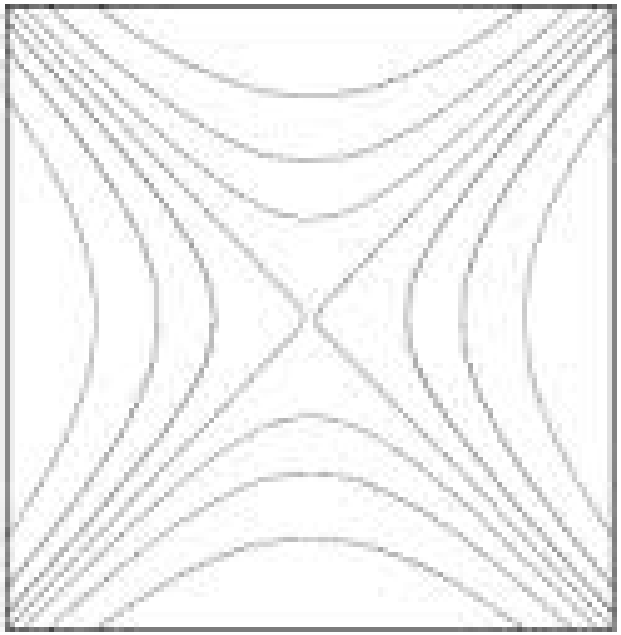
Lokales Maximum im $\mathbb{R}^n$





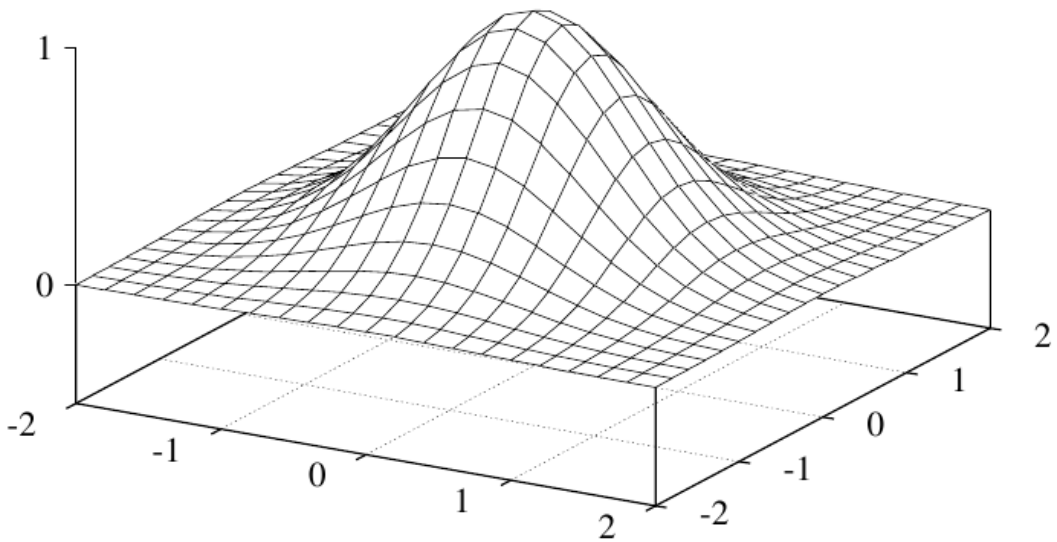
Sattelpunkt im $\mathbb{R}^n$



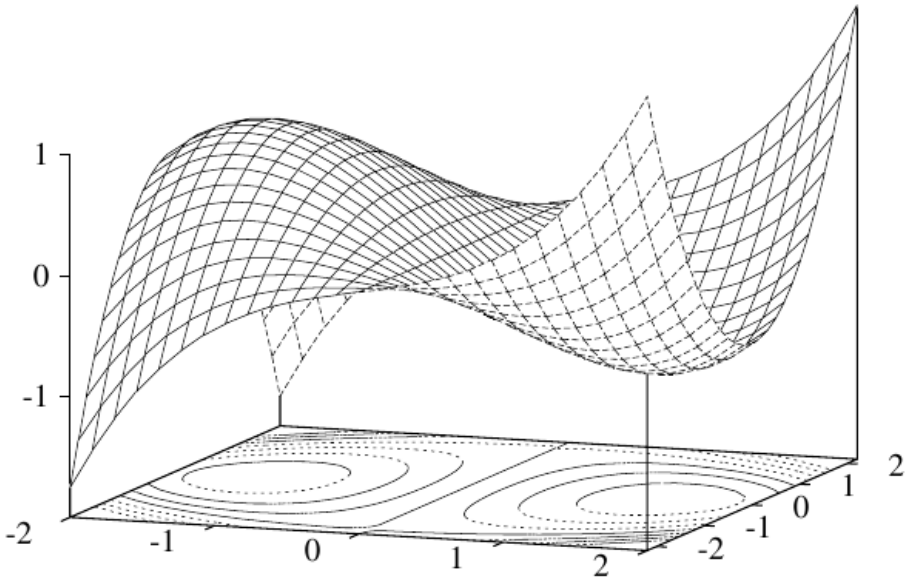


Beispiel 1: 1 Maximum

$$f: \mathbb{R}^2 \to \mathbb{R}, (x,y) \mapsto f(x,y) = \exp(-x^2 - y^2)$$



Beispiel 2: Min/Max, 2 Sattelpunkte



$$f(x,y) = \frac{1}{6}x^3 - x + \frac{1}{4}xy^2$$

Hauptminoren

Definition 11.1: (Hauptminor)

Die Determinante der linken oberen $k \times k$-Untermatrix einer Matrix heißt der k-te *Hauptminor*, Bezeichnung: M_k.

Satz 11.1: (Sylvester)

Es gilt:

(i) $D^2 f(x)$ pd $\Leftrightarrow \forall_{k=1,\ldots,n}\ M_k > 0$

(ii) $D^2 f(x)$ nd $\Leftrightarrow \forall_{k=1,\ldots,n}\ (-1)^k M_k > 0$

(iii) $\forall_{k=1,\ldots,n-1}\ M_k > 0\ \wedge\ M_n = \det(D^2 f(x)) = 0\ \Rightarrow\ D^2 f(x)$ psd

(iv) $\forall_{k=1,\ldots,n-1}\ (-1)^k M_k > 0\ \wedge\ M_n = \det(D^2 f(x)) = 0\ \Rightarrow$ $D^2 f(x)$ nsd

Hauptminoren und Definitheit

Satz 11.1: (Sylvester, Forsetzung)

(v) $M_n \neq 0$, aber weder (i) noch (ii) trifft zu $\Rightarrow$ $D^2 f(x)$ indefinit

(vi) $M_k > 0$ für ein gerades k $\Rightarrow$ $D^2 f(x)$ indefinit

(vii) $M_k > 0 \ \wedge \ M_\ell < 0$ für k, ℓ ungerade $\Rightarrow$ $D^2 f(x)$ indefinit

(viii) $D^2 f$ hat mindestens ein echt positives und ein echt negatives Diagonalelement $\Rightarrow$ $D^2 f(x)$ indefinit

Vorgehensweise zur Definitheitsprüfung

1. Hat $D^2 f(x)$ Diagonalgestalt? Falls ja, lese Eigenwerte auf Diagonalen ab und entscheide die Definitheit direkt.
2. Falls nein, bestimme die Hauptminoren von $D^2 f(x)$.
3. Sind alle Hauptminoren echt positiv, so ist $D^2 f(x)$ pd.
4. Ist der erste Hauptminor echt negativ und danach liegen wechselnde Vorzeichen vor (keine 0 erlaubt!), so ist $D^2 f(x)$ nd.
5. Ist der letzte Hauptminor, also die Determinante, gleich 0? Falls ja, so liegt Semidefinitheit vor.
6. Falls nein, aber es konnte bislange noch keine Entscheidung getroffen werden, so liegt Indefinitheit vor.

Hinweis: Aussage (viii) von Satz 11.1 ist auch sehr nützlich, um direkt auf Indefinitheit zu entscheiden.

Hinreichende Bedingung für lokale Extrema im $\mathbb{R}^n$

Satz 11.2: (Hinr. Bedingung für lokale Extrema im $\mathbb{R}^n$)

Sei $f : U \to \mathbb{R}$ zweimal stetig partiell differenzierbar, und $x_0 \in U$ sei stationärer Punkt von f, also $\nabla f(x_0) = 0$. Dann gilt:

(i) $D^2 f(x_0)$ pd $\Rightarrow$ x_0 lokales Minimum

(ii) $D^2 f(x_0)$ nd $\Rightarrow$ x_0 lokales Maximum

(iii) $D^2 f(x_0)$ indefinit $\Rightarrow$ x_0 Sattelpunkt

Bei Semidefinitheit läßt sich keine Aussage treffen.

Extrema mit Nebenbedingungen

Betrachte das folgende Optimierungsproblem $(*)$:
Maximiere/Minimiere eine Funktion $f : U \to \mathbb{R}$ unter der Nebenbedingung $g(x) = 0,\ x \in U$ und $g : U \to \mathbb{R}$. Dabei sollen f und g zweimal stetig partiell differenzierbar sein.

Definition 11.2: (Lagrange-Funktion)

Die Funktion

$$\mathcal{L}(x; \lambda) := f(x) - \lambda g(x),\ \lambda \in \mathbb{R},$$

heißt die zu Optimierungsproblem $(*)$ gehörige *Lagrange-Funktion*.

Falls $x^* \in U$ das Optimierungsproblem $(*)$ löst, so gilt $\mathcal{L}(x^*; \lambda) = f(x^*)$, da $g(x^*) = 0$.

Notw. und hinr. Bedingung für Extrema mit Nebenbedingungen

Satz 11.3: (Notw. Bedingung für Extrema mit Nebenbedingungen)

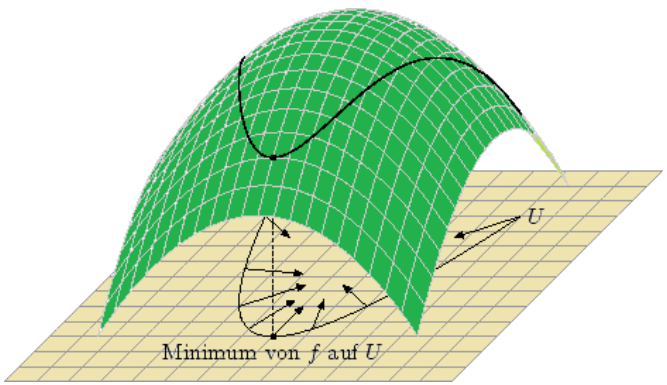
Falls $x^* \in U$ Lösung von Optimierungsproblem $(*)$, so existiert ein $\lambda \in \mathbb{R}$, so daß $\nabla f(x^*) = \lambda \nabla g(x^*)$

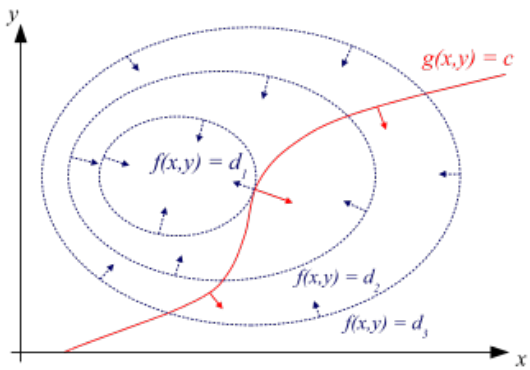
Satz 11.4: (Hinr. Bedingung für Extrema mit Nebenbedingungen)

Falls $\nabla \mathcal{L}(x^*, \lambda) = (0; 0)$ und $D^2 \mathcal{L}_x(x^*; \lambda)$ pd (nd), dann ist x^* lokales Minimum (Maximum) von f mit Nebenbedingung $g = 0$.

$D^2 \mathcal{L}_x(x^*)$ bedeutet, daß bei der Bildung der Hesse-Matrix von $\mathcal{L}$ nur nach den x_i, $i = 1, ..., n$, und nicht nach λ differenziert wird. Bei der Auswertung jedoch wird auch λ eingesetzt.

Veranschaulichung: Extrema mit Nebenbedingungen

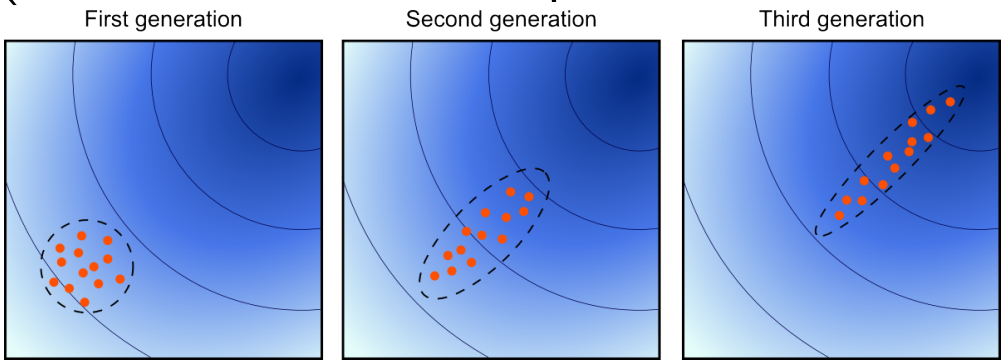




Die Pfeile veranschaulichen die jeweiligen Gradienten.

Globale Optimierungsverfahren

- **Evolutionäre Algorithmen:** (Prinzip *der Fitteste überlebt*)
  - Sampling/Abtasten des Parameterraums (oftmals – gerade am Anfang – per Zufallsprinzip)
  - CMA-ES (Covariance Matrix Adaptation Evolution Strategy):



- **Tabusuche:** Vermeide bereits besuchte Regionen, in denen das globale Minimum nicht liegen kann

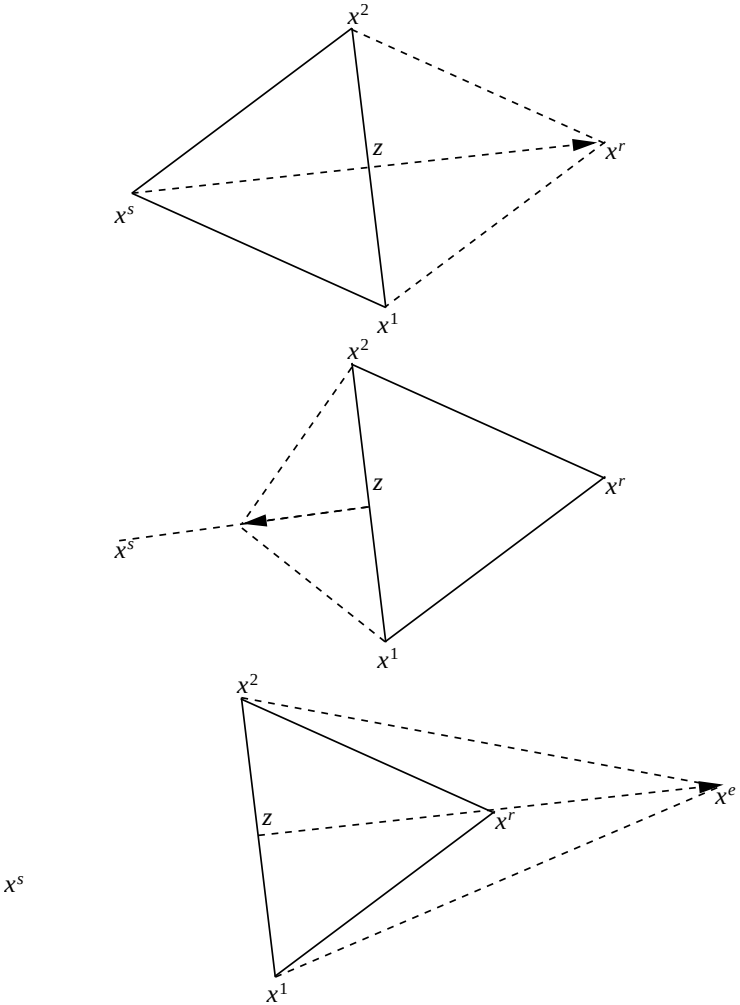
Nachteil:

Globale Optimierer sind meist sehr ineffizient in einer Umgebung des Minimums!

Lokale Optimierungsverfahren

- Effizienter/schneller im Einzugsbereich eines lokalen Minimums (vgl. Newton-Verfahren zur Nullstellenbestimmung)
- Betr. sog. *Abstiegsverfahren* (Gradient!), gradientenbasierte Verfahren unterteilen sich in zwei große Klassen:
  1. Liniensuchverfahren
  2. Trust-Region-Verfahren
- Es gibt auch ableitungsfreie Verfahren, die auf Interpolation/Approximation beruhen und das Minimum diskret auf einem Gitter bestimmen oder auf geometrischen Überlegungen beruhen, wie z.B. das **Simplex-Verfahren nach Nelder und Mead**

Das Simplex-Verfahren nach Nelder und Mead



- *Reflektion:* ersetze schlechtesten Punkt (d.h. den mit dem größten Funktionswert) durch reflektierten Punkt
- *Kontraktion:* falls reflektierter Punkt noch schlechter ist, kontrahiere den Simplex in Richtung des besten Punktes
- *Expansion:* falls reflektierter Punkt besser, expandiere den Simplex

Liniensuchverfahren

Eine wichtige Klasse numerischer Verfahren zur Bestimmung lokaler Extrema im $\mathbb{R}^n$ bilden die sogenannten **Liniensuchverfahren** mit der Iterationsvorschrift

$$x^{(k+1)} = x^{(k)} + t_k d^{(k)}, \ k \in \mathbb{N}_0$$

Dabei ist $x^{(0)}$ ein geeigneter Startvektor aus dem $\mathbb{R}^n$, $t_k \in \mathbb{R}$ eine effiziente Schrittweite und $d^{(k)} \in \mathbb{R}^n$ eine Suchrichtung. Handelt es sich bei $d^{(k)}$ um eine Abstiegsrichtung der zu minimierenden Funktion $f : \mathbb{R}^n \to \mathbb{R}$, so spricht man von einem *Abstiegsverfahren*.

Abstiegsverfahren

Definition 11.3: (Abstiegsverfahren)

Sei $f : \mathbb{R}^n \to \mathbb{R}$ mindestens zweimal stetig partiell differenzierbar, und sei $d^{(k)}$ die Suchrichtung innerhalb des Liniensuchverfahrens

$$x^{(k+1)} = x^{(k)} + t_k d^{(k)}, \ k \in \mathbb{N}_0$$

Falls $\left\langle \nabla f(x^{(k)}), d^{(k)} \right\rangle < 0$, so heißt $d^{(k)}$ *Abstiegsrichtung* bezüglich f. Falls $d^{(k)}$ für alle $k \in \mathbb{N}_0$ eine Abstiegsrichtung ist, so heißt das Verfahren *Abstiegsverfahren*.

Die zwei bekanntesten Abstiegsverfahren:

- $d^{(k)} = -\nabla f(x^{(k)})$: Methode des steilsten Abstiegs (engl. *steepest descent*)
- $d^{(k)} = -\left(D^2 f(x^{(k)})\right)^{-1} \nabla f(x^{(k)})$: Newton-Raphson-Verfahren

Abstiegsverfahren für Extrema mit Nebenbedingungen

Bei Extema mit Nebenbedingungen verwendet man das sog. *Newton-Lagrange-Verfahren*, das ist nichts anderes als das Newton-Raphson-Verfahren angewandt auf die Lagrange-Funktion.

Beachte:

Ein lokales Minimum der Zielfunktion unter einer Nebenbedingung ist ein Sattelpunkt der zughörigen Lagrange-Funktion. Daher handelt es sich hierbei um ein sog. *Sattelpunktproblem*.

Überlineare Konvergenz

Definition 11.4: (q-überlineare Konvergenz)

Ein iteratives Verfahren konvergiert *q-überlinear* gegen ein $x^* \in \mathbb{R}^n$, falls

$$\forall_{k \in \mathbb{N}_0} \; ||x^{(k+1)} - x^*|| \leq c_k ||x^{(k)} - x^*||$$

wobei $(c_k)_{k \in \mathbb{N}_0}$ eine Nullfolge ist.

Methode des steilsten Abstiegs

$$x^{(k+1)} = x^{(k)} - t_k \nabla f(x^{(k)})$$

Heuristisches Verfahren: Konvergenz nur dann garantiert, wenn die Folge $(f(x^{(k)}))_{k\in\mathbb{N}_0}$ streng monoton fallend ist (durch Schrittweitensteuerung erzielbar). Dann ist jeder Häufungspunkt von $(x^{(k)})_{k\in\mathbb{N}_0}$ ein stationärer Punkt von f.
Das Verfahren eignet sich sehr gut zu Beginn der Minimierung, um schnell in eine Umgebung des Minimums zu gelangen.

Newton-Raphson-Verfahren

$$x^{(k+1)} = x^{(k)} - t_k \left(D^2 f(x^{(k)}) \right)^{-1} \nabla f(x^{(k)})$$

mehrdimensionale Erweiterung des 1D-Newton-Verfahrens zur Nullstellenbestimmung
zusätzlich zur Steigung werden auch Krümmungseigenschaften von f mitberücksichtigt

Voraussetzungen an die Hesse-Matrix:

- muß invertierbar sein
- muß positiv definit sein, damit $-\left(D^2 f(x^{(k)})\right)^{-1} \nabla f(x^{(k)})$ eine Abstiegsrichtung ist

Konvergenz: Falls $D^2 f$ in einer Umgebung des Minimums Lipschitz-stetig ist, ist die Konvergenz quadratisch.

Quasi-Newton-Verfahren

$$x^{(k+1)} = x^{(k)} - t_k H_k^{-1} \nabla f(x^{(k)})$$

Dabei ist H_k eine Approximation der Hesse-Matrix, $H_0 = D^2 f(x^{(0)})$. Bestimmung von H_k erfolgt iterativ. Falls H_k spd, dann auch H_{k+1}, falls die sog. *Sekantenbedingung*

$$H_{k+1} \underbrace{\left(x^{(k+1)} - x^{(k)}\right)}_{=:s_k} = \underbrace{\nabla f(x^{(k+1)}) - \nabla f(x^{(k)})}_{=:y_k}$$

erfüllt ist (Theorem von Dennis und Moré)

PSB-Verfahren

Powell Symmetric Broyden (PSB):

$$\begin{aligned} H_{k+1}^{\mathrm{PSB}} &:= H_k + \frac{1}{\langle s_k, s_k \rangle} \left((y_k - H_k s_k) s_k^T + s_k (y_k - H_k s_k)^T \right) \\ &- \frac{(y_k - H_k s_k)^T s_k}{\langle s_k, s_k \rangle^2} s_k s_k^T \end{aligned}$$

Mit dieser Aufdatierung ist die Sekantenbedingung $H_{k+1}^{\mathrm{PSB}} s_k = y_k$ automatisch erfüllt.
Falls f in einer Umgebung des Minimums Lipschitz-stetig ist und $\sum_{k=0}^{\infty} ||x^{(k)} - x^*|| < \infty$, so ist die Konvergenz q-überlinear.

DFP-Verfahren

Davidon Fletcher Powell (DFP):

$$\begin{aligned} H_{k+1}^{\mathrm{DFP}} &:= H_k + \frac{1}{\langle y_k, s_k \rangle} \left((y_k - H_k s_k) y_k^T + y_k (y_k - H_k s_k)^T \right) \\ &- \frac{(y_k - H_k s_k)^T s_k}{\langle y_k, s_k \rangle^2} y_k y_k^T \end{aligned}$$

Mit dieser Aufdatierung ist die Sekantenbedingung $H_{k+1}^{\mathrm{DFP}} s_k = y_k$ automatisch erfüllt.
Die Konvergenz ist unter denselben Voraussetzungen wie beim PSB-Verfahren q-überlinear.

BFGS-Verfahren

Broyden Fletcher Goldfarb Shanno (BFGS): Hierbei wird eine Approximation der inversen Hesse-Matrix $B_k \to B_{k+1}$ aufdatiert, wobei $B_0 = (D^2 f(x^{(0)}))^{-1}$:

$$\begin{aligned} B_{k+1}^{\mathrm{BFGS}} &:= B_k + \frac{1}{\langle y_k, s_k \rangle} \left((s_k - B_k y_k) s_k^T + s_k (s_k - B_k y_k)^T \right) \\ &- \frac{(s_k - B_k y_k)^T y_k}{\langle y_k, s_k \rangle^2} s_k s_k^T . \end{aligned}$$

Sekantenbedingung $B_{k+1}^{\mathrm{BFGS}} y_k = s_k$ wieder automatisch erfüllt. Dies hat zwar den Nachteil, daß im ersten Schritt die Inverse der Hesse-Matrix bestimmt werden muß, die nächsten Schritte jedoch nur noch aus Matrix-Vektor-Multiplikationen erhalten werden und nicht mehr aus der Lösung eines LGS. Falls f in einer Umgebung des Minimums Lipschitz-stetig ist und für alle $k \geq 0$ die Frobenius-Norm der Hesse-Matrix beschränkt ist, konvergiert das BFGS-Verfahren ebenfalls q-überlinear.

Verfahren der Konjugierten Gradienten

Abstiegsrichtung:

$$d^{(0)} = -\nabla f(x^{(0)}),\ d^{(k+1)} = -\nabla f(x^{(k+1)}) + \alpha^{(k)} d^{(k)}$$

Transfer der Methode der Konjugierten Gradienten für Lineare Gleichungssysteme (vgl. Abschnitt 4.3) in den $\mathbb{R}^n$, allerdings mit neuen Konvergenzbeweisen. Ohne Rundungsfehler handelt es sich dabei um ein direktes Verfahren. Die Konjugiertheit der Gradienten

$$\langle \nabla f(x^{(k+1)}), \nabla f(x^{(k)}) \rangle = 0$$

ist im Mehrdimensionalen allerdings nicht mehr gegeben!

Fletcher-Reeves-Verfahren

$$\alpha_{\mathrm{FR}}^{(k)} := \frac{\langle \nabla f(x^{(k+1)}), \nabla f(x^{(k+1)}) \rangle}{\langle \nabla f(x^{(k)}), \nabla f(x^{(k)}) \rangle}$$

Ansatz: Betrachte Gradienten zunächst trotzdem als konjugiert!
Falls sie sog. *Levelmenge* $\mathcal{L}(x^{(0)}) := \{x \mid f(x) < f(x^{(0)})\}$ kompakt ist und f gleichmäßig konvex auf der Levelmenge ist, konvergiert das Fletcher-Reeves-Verfahren gegen das eindeutig bestimmte globale Minimum von f. Zur lokalen Konvergenz ist der zulässige Bereich für x einzuschränken.

Polak-Ribière-Verfahren

$$\alpha_{\mathrm{PR}}^{(k)} := \frac{\langle \nabla f(x^{(k+1)}) - \nabla f(x^{(k)}), \nabla f(x^{(k+1)}) \rangle}{\langle \nabla f(x^{(k)}), \nabla f(x^{(k)}) \rangle}$$

Ansatz: Gradienten nicht-konjugiert, ansonsten gilt $\alpha_{\mathrm{PR}}^{(k)} = \alpha_{\mathrm{FR}}^{(k)}$ (sowieso nicht erfüllt!).
Ist die Levelmenge kompakt und ∇f in einer Kugel, welche die Levelmenge enthält, Lipschitz-stetig, konvergiert das Polak-Ribière-Verfahren gegen das eindeutig bestimmte globale Minimum von f. Zur lokalen Konvergenz ist der zulässige Bereich für x einzuschränken.

Konjugierte-Gradienten-Verfahren in der Praxis

- Konvergenztheorie nur bei Verwendung einer bestimmten Schrittweite (vgl. Abschnitt 11.2), was in der Praxis jedoch nur mit hohem Rechenaufwand oder gar nicht realisierbar ist.
- Bei praktischen Anwendungen hat sich herausgestellt, daß das Polak-Ribière-Verfahren besser geeignet ist als das Fletcher-Reeves-Verfahren. Einen theoretischen Beweis hierfür gibt es jedoch nicht.
- Effizient ist auch die Kombination:
  1. Methode des steilsten Abstiegs
  2. Konjugierte-Gradienten-Verfahren

Schrittweitensteuerung

- muß funktionsangepaßt, also adaptiv sein
- steile Bereiche: große Schritte, flache Bereiche: kleine Schritte
- Schrittweiten müssen monoton fallend sein, um zu vermeiden, daß das Verfahren um das Minimum hin- und herspringt
- auch Norm der Abstiegsrichtung bestimmt die Schrittweite, daher zur besseren Kontrolle:

$$x^{(k+1)} = x^{(k)} + t_k \frac{d^{(k)}}{||d^{(k}||}$$

- in jedem Fall muß die Monotonie $f(x^{(k+1)}) < f(x^{(k)})$ gewährleistet sein

Effiziente Schrittweite

Definition 11.5: (Effiziente Schrittweite)

Eine Schrittweite t_k heißt *effizient*, falls

$$\exists_{\theta, \theta \neq \theta(x^{(k)}, d^{(k)})} \; f(x^{(k)} + t_k d^{(k)}) \leq f(x^{(k)}) - \theta \left(\frac{\langle \nabla f(x^{(k)}), d^{(k)} \rangle}{||x^{(k)}||} \right)^2$$

Die Schrittweite soll also vom Abstieg $\langle \nabla f(x^{(k)}), d^{(k)} \rangle$ abhängen und für signifikant kleinere Funktionswerte sorgen. Die Konstante θ ist eine globale Konstante.

Armijo-Schrittweitensteuerung

Definition 11.6: (Skalierte Armijo-Schrittweite)

Eine Schrittweite t_A heißt *skalierte Armijo-Schrittweite*, falls

$$t_A = \max\{s\beta_A^\ell | \ell = 0, 1, ..., f(x+s\beta_A^\ell d) \leq f(x)+\zeta_A s\beta_A^\ell \langle \nabla f(x), d\rangle\},$$

wobei $\beta_A, \zeta_A \in (0, 1)$, d eine Abstiegsrichtung und $s > 0$.

Satz 11.5: (Effizienz der skalierten Armijo-Schrittweite)

Falls $f : \mathbb{R}^n \to \mathbb{R}$ mind. einmal stetig differenzierbar und ∇f Lipschitz-stetig auf der Levelmenge $\mathcal{L}(x^{(0)})$ ist, und falls $s \geq -c\frac{\langle \nabla f, d\rangle}{||d||^2}$ bei fest vorgegebenem $c > 0$, so ist die skalierte Armijo-Schrittweite für alle $x \in \mathcal{L}(x^0)$ effizient. Weiterhin ist t_A wohldefiniert, falls d eine Abstiegsrichtung ist.

Wolfe-Powell-Schrittweitensteuerung

Definition 11.7: (Wolfe-Powell-Schrittweite)

Eine *Wolfe-Powell-Schrittweite* t_{WP} ist eine Armijo-Schrittweite mit $\zeta_A \in (0, \frac{1}{2})$ und der Zusatzbedingung

$$\langle \nabla f(x + t_{WP} d), d \rangle \geq \rho \langle \nabla f(x), d \rangle, \ \rho \in [\zeta_A, 1)$$

Satz 11.6: (Effizienz der Wolfe-Powell-Schrittweite)

Falls f nach unten beschränkt ist, so ist die Wolfe-Powell-Schrittweite t_{WP} wohldefiniert. Falls ∇f auf der Levelmenge $\mathcal{L}(x^{(0)})$ Lipschitz-stetig ist, so ist t_{WP} für alle $x \in \mathcal{L}(x^0)$ effizient.

Konvergenz von Abstiegsverfahren

Satz 11.7: (Konvergenz von Abstiegsverfahren)

Sei durch

$$x^{(k+1)} = x^{(k)} + t_k d^{(k)}$$

ein Liniensuchverfahren gegeben, wobei $d^{(k)}$ eine Abstiegsrichtung und t_k eine Wolfe-Powell-Schrittweite ist. Sei $f : \mathbb{R}^n \to \mathbb{R}$ nach unten beschränkt und auf einer offenen Menge, welche die Levelmenge $\mathcal{L}(x^{(0)})$ enthält, mind. einmal stetig differenzierbar. Falls ∇f auf der Levelmenge $\mathcal{L}(x^{(0)})$ Lipschitz-stetig ist, so konvergiert das Liniensuchverfahren, d.h.

$$\lim_{k\to\infty} ||\nabla f(x^{(k)})|| = 0$$

Trust-Region-Verfahren

- Schrittweitensteuerung ist im Verfahren selbst enthalten: In jeder Iteration k wird eine Schrittweite Δ_k bestimmt, so daß f in dem Ball $B_{\Delta_k}(x^{(k)})$ (sog. *Vertrauensgebiet*) durch eine quadratische Form approximiert werden kann
- Bestimme Abstiegsrichtung für jede Schrittweite Δ_k sowie die neue Iteration $x^{(k+1)} = x^{(k)} + d^{(k)}$
- **Trust-Region-Idee:** Falls $d^{(k)}$ als nicht geeignet abgestuft wird, wird Δ_k verkleinert, ansonsten wird Δ_k (vorsichtshalber, d.h., um den Suchraum zu vergößern) erhöht, aber was bedeutet *geeeignet?*

Trust-Region-Teilproblem:

Wie bestimmt man eine geeignete Abstiegsrichtung $d^{(k)}$?

Trust-Region-Teilproblem

Taylorentwicklung liefert die folgende quadratische Form:

$$\begin{aligned} q_{x^{(k)}}(d^{(k)}) := f(x^{(k)} + d^{(k)}) &\doteq f(x^{(k)}) + \langle \nabla f(x^{(k)}), d^{(k)} \rangle \\ &+ \frac{1}{2}(d^{(k)})^T D^2 f(x^{(k)}) d^{(k)} \end{aligned}$$

Bestimme

$$\min_{||d^{(k)}|| \leq \Delta_k} q_{x^{(k)}}(d^{(k)})$$

Güte der Approximation wird durch folgenden Verhältnis geschätzt:

$$r^{(k)} := \frac{f(x^{(k+1)}) - f(x^{(k)})}{q_{x^{(k)}}(d^{(k)}) - q_{x^{(k)}}(0)} = \frac{f(x^{(k+1)}) - f(x^{(k)})}{q_{x^{(k)}}(d^{(k)}) - f(x^{(k)})}$$

Änderungen des Vertrauensgebiets

- 1. Fall: $r^{(k)} < 1$: Dann stimmt das quadratische Modell schlecht mit f überein, da es den Abstieg von f überschätzt. Verkleinere in diesem Fall das Vertrauensgebiet.
- 2. Fall: $r^{(k)} \geq 1$: Dann stimmt das quadratische Modell gut mit f überein. Vergrößere in diesem Fall das Vertrauensgebiet.

Trust-Region-Teilproblem kann auf zwei Arten gelöst werden:

1. approximativ durch die *Methode des Doppelten Hundebeins*
2. exakt

Cauchy-Punkt

Definition 11.8: (Cauchy-Punkt)

Es seien $\Delta > 0$ und

$$\phi : [0,1] \to \mathbb{R},\ \phi(t) := q_x\left(-t\frac{\Delta\nabla f(x)}{||\nabla f(x)||}\right).$$

Dann ist der $\mathrm{Cauchy\text{-}Punkt}$ d_C gegeben als das Minimum von ϕ auf $[0,1]$, also

$$d_C = -t_C\frac{\Delta\nabla f(x)}{||f(x)||},\ \text{wobei } \phi'(t_C) = 0 \vee t_C = 1.$$

Es ist

$$t_C = \min\left(\frac{||\nabla f(x)||^3}{\Delta\nabla f(x)^T D^2 f(x)\nabla f(x)}, 1\right).$$

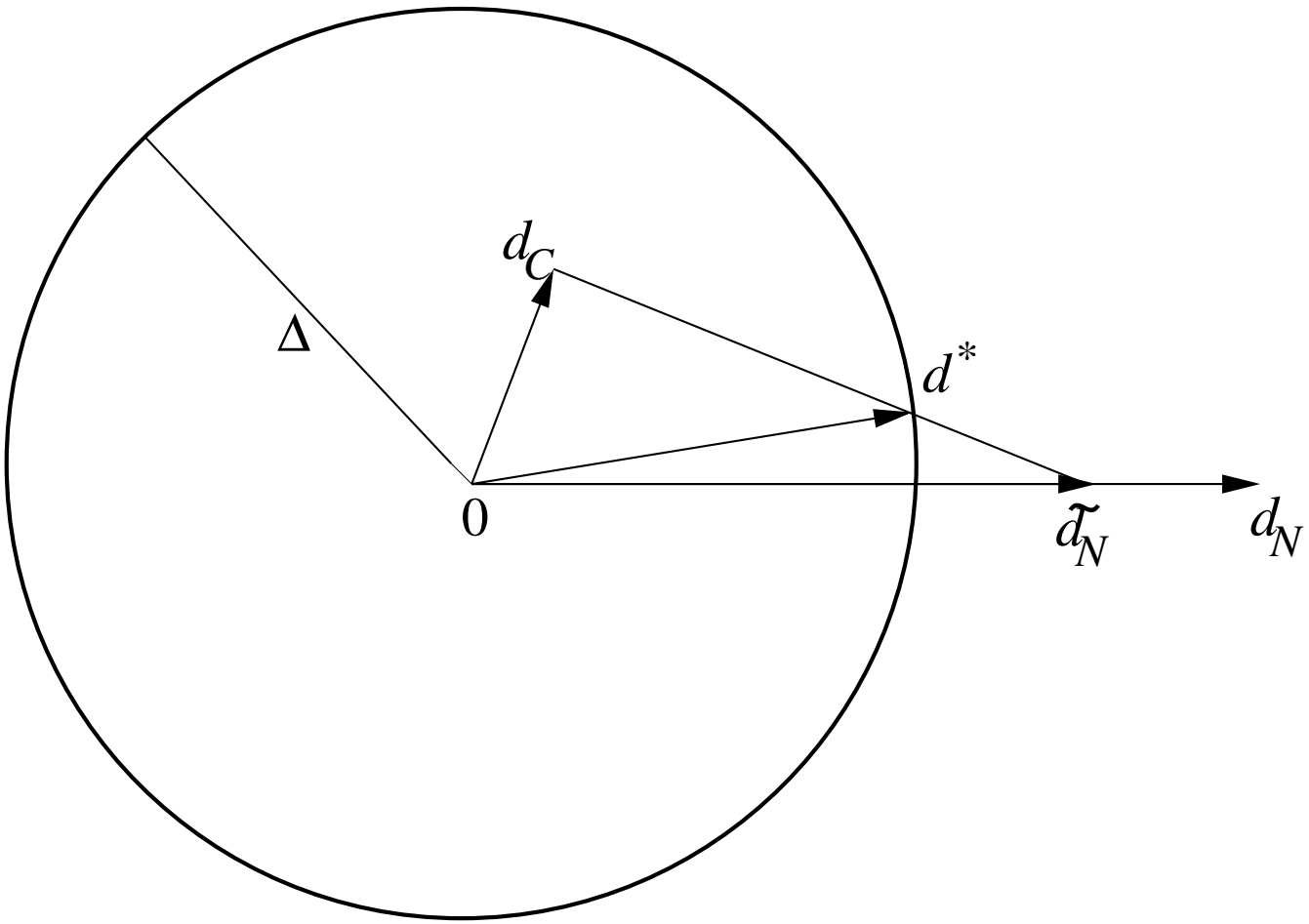
Methode des Doppelten Hundebeins

Der Cauchy-Punkt zwar in Richtung des negativen Gradienten und garantiert daher einen Abstieg, man ist aber an einer besseren Lösung d^* interessiert.
Sei $d_N := -(D^2 f(x))^{-1} \nabla f(x)$ die Newton-Richtung. Das quadratische Modell wird entlang des folgenden Pfades monton fallen:

$$\tilde{d}(t) := \begin{cases} t d_C, & 0 \leq t \leq 1 \\ d_C + (t-1)(d_N - d_C), & 1 \leq t \leq 2. \end{cases}$$

1. $D^2 f(x)$ nicht positiv definit $\Rightarrow d^* := d_C$.
2. $D^2 f(x)$ positiv definit, $||d_N|| < \Delta \Rightarrow d^* := d_N$.
3. $D^2 f(x)$ positiv definit, $||d_N|| \geq \Delta \Rightarrow$ bestimme d^* aus Gleichung $||d_C + (t_0 - 1)(d_N - d_C)||^2 = \Delta^2$.

Veranschaulichung des Doppelten Hundebeins



Exakte Lösung des Teilproblems

Problem:

Doppeltes Hundebein funktioniert nur im Falle der positiven Definitheit von $D^2 f$ gut!

Bestimme nahezu exakte Lösung durch Karush-Kuhn-Tucker-(KKT-)Formulierung: Es seien $g := \nabla f(x)$, $B := D^2 f(x)$ und $\lambda \geq 0$. d^* ist genau dann eine Lösung des Teilproblems, wenn gilt:

$$\begin{aligned} (B + \lambda I) d^* &= -g \\ \lambda(\Delta - ||d^*||) &= 0. \end{aligned}$$

Idee: Ist B nicht positiv definit, dann macht man B positiv definit!

Iteratives Lösungsverfahren

Satz 11.8: (Nahezu Exakte Lösung des Trust-Region-Teilproblems)

Es sei $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, ..., \lambda_n)$ mit $B = Q\Lambda Q^T$, wobei $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ die Eigenwerte von B sind. Die Matrix Q enthält als Spalten die Eigenvektoren $q_1, ..., q_n$. Es sei $\Delta > 0$, und $\lambda^{(0)} > 0$ sei so gewählt, daß $B + \lambda^{(0)} I$ positiv definit (in jedem Fall $\lambda > -\lambda_1$). Falls B indefinit, so sei $\langle q_1, g \rangle \neq 0$. Dann konvergiert die Folge $\left(d(\lambda^{(\ell)})\right)_{\ell \in \mathbb{N}}$, wobei

$$\lambda^{(\ell+1)} = \lambda^{(\ell)} + \frac{||p_\ell|| - \Delta}{\Delta} \left(\frac{||p_\ell||}{||q_\ell||} \right)^2,$$

gegen die eindeutig bestimmte Lösung des TR-Teilproblems. Dabei ist $B + \lambda^{(\ell)} I$ positiv definit und somit Cholesky-zerlegbar in $R^T R$, wobei R eine obere Dreiecksmatrix ist. Weiterhin $R^T R p_\ell = -g$ und $R^T q_\ell = p_\ell$.

Literatur Numerische Mathematik 1

[1] Stoer/Bulirsch: Numerische Mathematik 1, Springer (2007)

[2] A. Quarteroni, R. Sacco, F. Saleri: Numerische Mathematik 1 – Springer (2002)

[3] R. Schaback, H. Wendland: Numerische Mathematik, 5. Auflage; Springer (2005).

[4] G. Bärwolff: Numerik für Ingenieure, Physiker und Informatiker; Springer (2015).

[5] G. Opfer: Numerische Mathematik für Anfänger – Eine Einführung für Mathematiker, Ingenieure und Informatiker, 4. Auflage; Vieweg (2002).

[6] M. Knorrenschild: Mathematik-Studienhilfen, Numerische Mathematik, Eine beispielorientierte Einführung, 6. Auflage; Fachbuchverlag Leipzig im Carl Hanser Verlag (2017)

[7] G. Fischer: Lineare Algebra, Eine Einführung für Studienanfänger, 18. Auflage, Springer Spektrum Wiesbaden (2014)

[8] K.-U. Witt: Lineare Algebra für die Informatik – Vektorräume, Gleichungssysteme, Codierung, Quantenalgorithmen, Springer Vieweg Wiesbaden (2013)

Literatur Numerische Mathematik 2

[1] J. Stoer, R. Bulirsch: Numerische Mathematik 2, 5. Auflage, Springer (2005)

[2] A. Quarteroni, R. Sacco, F. Saleri: Numerische Mathematik 2 – Springer (2002)

[3] J. Nocedal, S. J. Wright: Numerical Optimization – Springer (1999)

[4] D. Braess: Finite Elemente – Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie, 3. Auflage; Springer (2003).

[5] D. Grieser: Analysis 1 – Eine Einführung in die Mathematik des Kontinuums, Springer-Verlag (2015)

[6] O. Forster: Analysis 2 – Differentialrechnung im $\mathbb{R}^n$, gewöhnliche Differentialgleichungen, Springer-Verlag (2010), 8. Auflage