Numerik WS 19/20

Zusammenfassung

Jonathan Bernhard

Inhaltsverzeichnis

1 Fehleranalyse

1.1 Konditionierung numerischer Aufgaben

1. Gut konditioniert

Kleine Änderung der Eingabedaten → kleine Änderung des Ergebnisses

2. Numerische Aufgabe

Berechnung endlich vieler Größen y_i $(i = 1...n)$ aus Größen x_j $(j = 1...m)$ mit funktionaler Vorschrift $y_i = f_i(x_1, ..., x_m)$ bzw. $y = f(x), x = (x_1, ...x_m)^\top$, $y = (y_1, ..., y_n)^\top, f = (f_1, ..., f_n)^\top$

3. Abweichung bei Störungen

Gestörtes Ergebnis: $\Delta y + y = f(x + \Delta x)$
Abweichung: $\Delta y_i = f_i(x + \Delta x) - f_i(x) = \sum_{j=1}^{m} \frac{\mathrm{d}f_i}{\mathrm{d}x_j}(x)\Delta x_j + R_i^f(x, \Delta x)$

4. Landau Symbole

$g(t) = O(h(t)) \iff$ für kleine $t \in (0, t_0]$, $c \geq 0$: $|g(t)| \leq c \cdot |h(t)|$
$g(t) = o(h(t)) \iff$ für kleine $t \in (0, t_0]$, $c(t) \to 0$: $|g(t)| \leq c(t)|h(t)|$

5. Relativer Fehler

$$\frac{\Delta y_i}{y_i} = \sum_{j=1}^{m} \frac{\mathrm{d}f_i}{\mathrm{d}x_j}(x)\frac{\Delta x_j}{y_i} = \sum_{j=1}^{m} \underbrace{\frac{\mathrm{d}f_i}{\mathrm{d}x_j}(x)\frac{x_j}{f_i(x)}}_{k_{ij}} \cdot \frac{\Delta x_j}{x_j}$$

6. Relative Konditionszahl

$|k_{ij}| < 1 \to$ Fehlerdämpfung
$|k_{ij}| > 1 \to$ Fehlerverstärkung $\quad |k_{ij}| \gg 1 \to$ schlecht konditioniert

1.2 Zahlendarstellung und Rundungsfehler

1. Normalisierte Gleitkommazahl

Basis: $b \in \mathbb{N}, b \geq 2$
Mantisse: $m = m_1 b^{-1} + \cdots + m_r b^{-r} \quad \in \mathbb{R}, \quad m_i \in \{0, ..., b-1\}$
Exponent: $e = e_{s-1} b^{s-1} + \cdots + e_0 b^0 \quad \in \mathbb{N}_0, \quad e_i \in \{0, ..., b-1\}$

Zahl $x \in \mathbb{R}$: $x = \pm m \cdot b^{\pm e}$

2. Speicherung

Single Precision:

Vorzeichen Exponent Mantisse

Double Precision:

Vorzeichen Exponent Mantisse

Sonderfälle:

$e = 255$ und $M \neq 0 \longrightarrow$ NaN
$e = 0$ und $M = 0 \longrightarrow x = 0$

3. Rundungsfehler

absoluter Fehler: $|x - rd(x)| \leq \frac{1}{2} b^{-r} b^e$
relativer Fehler: $\frac{|x-rd(x)|}{x} \leq \frac{1}{2} \frac{b^{-r} b^e}{|m| b^e} \leq \frac{1}{2} b^{-r+1}$ (Maschinengenauigkeit)

Dabei gilt $rd(x) = x(1 + \epsilon), |\epsilon| \leq eps = \frac{1}{2} b^{-r+1}$, d.h. eps ist die größte Zahl, die bei Addition zu 1 weg fällt, also $1 + eps = 1$.

Bsp. Double: $eps \sim 10^{-16}$

1.3 Stabilität numerischer Aufgaben

1. Verfahren/Algorithmus

Endliche oder abzählbar unendliche Folge von elementaren Abbildungen $\varphi^{(k)}$, die durch sukzessive Anwendung einen Näherungswert $\tilde{y}$ zu y liefern:

$x = x^{(0)} \to \varphi^{(1)}(x^{(0)}) = x^{(1)} \to \varphi^{(2)}(x^{(1)}) = x^{(2)} \to \cdots$

2. Stabile/gutartige Algorithmen

Die im Verlauf der Ausführung akkumulierten Fehler übersteigen den durch die Konditionierung bedingten unvermeidbaren nicht.

3. Horner-Schema

Ein Polynom $p(x) = b_0 + b_1 x + b_2 x^2 + \cdots + b_n x^n$ lässt sich numerisch stabil lösen durch $p(x) = (\cdots (b_n + b_{n-1})x + \cdots)x + b_0$.

2 Interpolation und Approximation

2.1 Grundbegriffe

1. Standardprobleme

Eine Funktion $f(x)$ soll aus gegebenen Argumenten $x_0, ..., x_n$ rekonstruiert werden.

Eine analytisch gegebene Funktion $f(x)$ soll auf einer Rechenanlage effizient und leicht berechnet werden.

2. Approximation

Näherungsweise Funktion, die zu gegebenen Punkten passt. Dabei müssen allerdings die gegebenen Punkte nicht exakt auf der neuen Funktion liegen.

3. Interpolation

Finden einer Funktion, die die gegebenen Punkte beinhaltet durch Fixieren von Funktionswerten $g(x_i) = y_i = f(x_i), \quad i = 0...n.$

4. Extrapolation

Abschätzen einer interpolierten Funktion über die gegebenen Punkte hinaus.

5. Klassen von Funktionen

Polynome:	$p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n$
Rationale Funktionen:	$r(x) = \frac{a_0+a_1x+\cdots+a_nx^n}{b_0+b_1x+\cdots+b_mx^m}$
Trigonometrische Polynome:	$t(x) = a_0 + \sum_{k=1}^{n}(a_k cos(kx) + b_k sin(kx))$
Exponentialsummen:	$e(x) = \sum_{k=1}^{n} a_k exp(b_k x)$

2.2 Polynominterpolation

1. Lagrange-Interpolationsaufgabe

Bestimmen eines Polynoms $p \in P_n$ zu $n+1$ paarweise verschiedenen Stützstellen $x_0, ..., x_n \in \mathbb{R}$ und gegebenen Knotenwerten $y_0, ..., y_n \in \mathbb{R}$ mit der Eigenschaft $p(x_i) = y_i$ für $i = 0...n$.

Die Lagrange-Interpolationsaufgabe ist eindeutig lösbar.

2. Lagrange-Interpolation

Lagrange Basispolynome: $L_i^{(n)}(x) = \prod_{\substack{j=0 \\ i \neq j}}^{n} \frac{x - x_j}{x_i - x_j} \in P_n, \quad i = 0...n$

Lagrange Interpolationspolynom: $p^{(n)}(x) = \sum_{i=0}^{n} y_i L_i^{(n)}(x) \in P_n$

Dabei gilt außerdem: $L_i^{(n)}(x_k) = \left\{ \begin{array}{l} 1 \text{ falls } i = k \\ 0 \text{ falls } i \neq k \end{array} \right\} = \delta_{ik}$ (Kronecker-Symbol)

Problem:
Bei zusätzlichem Punkt (x_{n+1}, y_{n+1}) ist Neuberechnung von $\{L_i^{(n+1)}\}_{i=0}^{n+1}$ notwendig!

3. Newton-Interpolation

Newton Basispolynom: $N_0(x) = 1 \qquad N_i(x) = \prod_{j=0}^{i-1}(x - x_j) = N_{i-1}(x) \cdot (x - x_{i-1})$

Dividierte Differenzen: $y[x_0, ..., x_n] = \frac{y[x_1,...,x_n] - y[x_0,...,x_{n-1}]}{y_n - y_0}$

Newton-Darstellung: $p^{(n)}(x) = \sum_{i=0}^{n} y[x_0, ..., x_i] N_i(x)$

Auswertung von $p(x) = a_0 + a_1(x - x_0) + \cdots + a_n(x - x_0) \cdots (x - x_{n-1})$ im Punkt ξ erfolgt über modifiziertes Horner-Schema:

$b_n = a_n; \quad k = n - 1, ..., 0: \quad b_k \equiv a_k + (\xi - x_k) b_{k+1}; \quad p(\xi) = b_0$

4. Neville-Darstellung

Darstellung:

Zu gegebenen $(x_i, y_i), \quad i = 0...n$ bezeichnet $P_{i_0,...,i_k}$ das Polynom mit $P_{i_0,...,i_k}(x_{ij} = y_{ij}, \quad j = 0...k)$.

Rekursive Berechnung durch:

$P_{i_0}(x) = y_{i_0} \qquad P_{i_0,...,i_k}(x) = \frac{1}{x_{ik} - x_{i_0}} [(x - x_{i_0}) P_{i_1,...,i_k}(x) - (x - x_{ik}) P_{i_0,...,i_{k-1}}(x)]$

Vorteile wie bei Newton, außerdem:
Auswertung an einzelnen Stellen ohne Berechnung der Polynomkeoffizienten.

5. Hermite Interpolation

Verallgemeinerung zu Lagrange Interpolationsaufgabe:

Gegeben:	$x_i,$	$i = 0...m$ (paarweise verschieden)
	$y_i^{(k)}, \quad i = 0...m,$	$k = 0...\mu_i \quad (\mu_i \geq 0)$
Gesucht:	$p \in P_n,$	$n = m + \sum_{i=0}^{m} \mu_i$ sodass gilt
	$p^{(k)}(x_i) = y_i^{(k)}$	für $k = 0...\mu_i$

Die Hermitesche Interpolationsaufgabe bestitz eine eindeutige Lösung.

2.3 Spline Interpolation

1. Stückweise polynomiale Funktionen

Stützstellen $a = x_0 < x_1 < \cdots < x_n = b$ gegeben.

Definiere $I_i = [x_{i-1}, x_i]$, $h_i := x_i - x_{i-1}$, $h := \max\limits_{i=1\ldots n} h_i$.

Für $k, r \in \mathbb{N}_0$ sei $S_h^{k,r}[a,b] := \{p \in C^r[a,b] : p_{|I_{i_{i=1\ldots n}}} \in P_k\}$.

2. Darstellung mit linearer Knotenbasis

Lineare Knotenbasis $\{\varphi_0, ..., \varphi_n\}$ definiert durch

$$\{\varphi_0, ..., \varphi_n\} := \left\{ \begin{array}{l} \varphi_i \overset{!}{\in} S_h^{(1,0)}[a,b] \\ \varphi_i(x_j) = \delta_{i_j} \end{array} \right\}$$

Es gilt $[\varphi_0, ..., \varphi_n] = S_h^{(1,0)}[a,b]$.

$\Longrightarrow \quad p(x) = \sum_{i=0}^n f(x_i)\varphi_i(x)$ (vgl. Lagrange Darstellung)

3. Kubische Splines

Eine Funktion $S[a,b] \to \mathbb{R}$ heißt Kubischer Spline zur Zerlegung $a = x_0 < x_1 < \cdots < x_n = b$ falls gilt:

$$\left. \begin{array}{ll} (i) & S \in C^2[a,b] \\ (ii) & S_{|[x_{i-1},x_i]} \in P_3 \end{array} \right\} \Longleftrightarrow S \in S_h^{(3,2)}[a,b]$$

Gilt weiterhin:

(a)	$s''(a) = s''(b) = 0$		$\longrightarrow$	Natürlicher Spline
(b)	$s'(a) = v_0$,	$s'(b) = v_n$ vorgegeben	$\longrightarrow$	Eingespannter Spline
(c)	$s'(a) = s'(b)$,	$s''(a) = s''(b)$	$\longrightarrow$	Periodische Spline

Der interpolierende kubische Spline existiert und ist eindeutig bestimmt durch zusätzliche Vorgabe von $s_n''(a)$ und $s_n''(b)$.

Ein kubischer Spline ist genau diejenige C^2-Funktion, welche $s(x_i) = y_i, \quad i = 0...n$ erfüllt und dabei minimale Krümmung hat.

Maß für Krümmung:

$$K[s] := \int_a^b \underbrace{(s''(x))}_{\text{Krümmung von s in x}}^2 dx = \|s''\|^2_{C^2[a,b]}$$

Minimale Krümmung:

Sei s kubischer Spline zu $a = x_0 < x_1 < \cdots < x_n = b$ und $f \in C^2[a,b]$ beliebig und $s(x_i) = f(x_i), \quad i = 0...n$.

Falls $s''(a)[f'(a) - s'(a)] = s''(b)[f'(b) - s'(b)]$ gilt, dann ist $K[s] \leq K[f]$.

Explizite Berechnung:

Aufschreiben der Bestandteile $s_{n_{|[x_{i-1},x_i]}} = p_i \in P_3$ in der Form

$p_i(x) = a_0^{(i)} + a_1^{(i)}(x - x_i) + a_2^{(i)}(x - x_i)^2 + a_3^{(i)}(x - x_i)^3 \quad i = 1...n$

und Bestimmen der $4n$ Koeffizienten $a_0^{(i)}, ..., a_3^{(i)}$.

Beispiel natürlicher Spline:

1) Die Interpolationsbedingung $p_i(x_i) = y_i$, $p_i(x_{i-1}) = y_{i-1}$ impliziert:
   $a_0^{(i)} = y_i, \quad i = 1...n$
2) und mit $h_i = x_i - x_{i-1}$:
   $y_{i-1} - y_i = -a_1^{(i)} h_i + a_2^{(i)} h_i^2 - a_3^{(i)} h_i^3 \quad i = 1...n$
3) Die Randbedingungen $p_1''(x_0) = p_n''(x_n) = 0$
   $a_2^{(i)} - 3a_3^{(i)} h_i = 0, \quad a_2^{(n)} = 0$
4) Die Stetigkeit der 1. Ableitung $p_i'(x_i) = p_{i+1}'(x_i)$ impliziert:
   $a_1^{(i)} = a_1^{(i+1)} - 2a_2^{(i+1)} h_{i+1} + 3a_3^{(i+1)} h_{i+1}^2, \quad i = 1...n-1$
5) Stetigkeit $p_i''(x_i) = p_{i+1}''(x_i)$:
   $a_2^{(i)} = a_2^{(i+1)} - 3a_3^{(i+1)} h_{i+1}, \quad i = 1, ..., n-1$

Damit bekommt man den $n-1$-Vektor $(a_2^{(1)}, ..., a_2^{(n-1)})^\top$ ein $(n-1) \times (n-1)$ Gleichungssystem der Form:

$$A = \begin{bmatrix} 2(h_1+h_2) & h_2 & & & 0 \\ h_2 & 2(h_2+h_3) & \ddots & & \\ & h_3 & \ddots & & \\ & & \ddots & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & & & h_{n-1} & 2(h_{n-1}) \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \end{bmatrix}$$

mit $b_i = 3\{\frac{y_{i+1}-y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}\}, \quad i = 1, ..., n-1$

a_1 und a_2 werden dann berechnet durch:

6) $a_3^{(i)} = \frac{a_2^{(i)} - a_2^{(i-1)}}{3h_i}, \quad i = 1, ..., n$
7) $a_1^{(i)} = \frac{y_i - y_{i-1}}{h_1} + \frac{h_i}{3}\{2a_2^{(i)} + a_2^{(i-1)}\}, \quad i = 1, ..., n$

2.4 Trigonometrische Interpolation

1. Trigonometrische Summen

$t_n(x) = \frac{1}{2}a_0 + \sum_{k=1}^{m}\{a_k \cos\left(\frac{kx2\pi}{\omega}\right) + b_k \sin\left(\frac{kx2\pi}{\omega}\right)\} \quad n = 2m$

Diese Summe ist ω-periodisch. Wird o.B.d.A $\omega = 2\pi$ angenommen, so können die Stützstellen äquidistant gewählt werden zu $x_k = k\frac{2\pi}{n+1}, \quad k = 0...n.$

2. Trigonometrische Interpolation

Zu gegebenen $y_0, ..., y_n \in \mathbb{C}$ gibt es genau eine Funktion der Gestalt

$t_n^*(x) = \sum_{k=0}^{n} c_k e^{ikx},$

welche den Interpolationsbedingungnen

$t_n^*(x_j) = y_j \quad (j = 0...n)$ genügt.

Die Koeffizienten sind bestimmt durch

$c_k = \frac{1}{n+1}\sum_{j=0}^{n} y_j e^{-ijx_k}, \quad k = 0...n$

3. Diskrete Fourier Analyse

Für $n \in \mathbb{N}_0$ gibt es zu gegebenen $y_0, ..., y_n \in \mathbb{R}$ genau ein trigonometrisches Polynom der Form

$t_n(x) = \frac{1}{2}a_0 + \sum_{k=1}^{m}\{a_k \cos(kx) + b_k \sin(kx)\} + \frac{\Theta}{2}a_{m+1}\cos((m+1)x)$

mit $t_n(x_j) = y_j, \quad j = 0...n,$

wobei $\begin{cases} \Theta = 0 & m = \frac{1}{2}n \quad \text{falls n gerade} \\ \Theta = 1 & m = \frac{1}{2}(n-1) \quad \text{falls n ungerade} \end{cases}$

Die Koeffiezienten a_k und b_k sind bestimmt durch

$$a_k = \frac{2}{n+1}\sum_{j=0}^{n} y_i \cos(jx_k)$$

$$b_k = \frac{2}{n+1}\sum_{j=0}^{n} y_i \sin(jx_k)$$

Die Abbildung $\{y_j\} \longrightarrow \{a_k, b_k\}$ heißt diskrete Fourier Transformation.

2.5 Interpolationsfehler

1. **Definitionen**
   - für $a < b$ sei $C[a,b] \coloneqq \{f[a,b] \to \mathbb{R} \,|\, f stetig\}$
   - für $k \in \mathbb{N}$ sei $C^k[a,b] \coloneqq \{f \in C[a,b] : f' \in C[a,b], ..., f^{(k)} \in C[a,b]\}$
   - für $f \in C[a,b]$ sei $\|f\|_\infty \coloneqq \max\limits_{x \in [a,b]} |f(x)|$
   - für $s_1, ..., s_n \in \mathbb{R}$ sei $\overline{(s_1, ..., s_n)} \coloneqq [\min s_i, \max s_i]$

2. **Lagrange Interpolationsaufgabe**

   Sei $f \in C^{n+1}[a,b]$ für ein $n \in \mathbb{N}$ und p das Interpolationspolynom bezüglich der Punkte $(x_i, f(x_i))_{i=0...n}$ mit $x_i \in [a,b]$.

   Dann existiert für jedes $x \in [a,b]$ ein $\xi_x \in \overline{(x_0, ..., x_n, x)}$ sodass gilt:

   $$\underbrace{f(x) - p(x)}_{\text{Fehler}} = \underbrace{\frac{f^{(n+1)}(\xi_x)}{(n+1)!}}_{\text{abhängig von Problem}} \underbrace{\prod_{j=0}^{n}(x - x_j)}_{\text{abhängig von den Stützstellen}}$$

3. **Hermitesche Interpolationsaufgabe**

   Sei $f \in C^{n+1}[a,b]$ und $p \in P_n$ sei Hermite Interpolationspolynom.

   Dann existiert für jedes $x \in [a,b]$ ein $\xi_x \in \overline{(x_0, ..., x_m, x)}$ sodass gilt:

   $f(x) - p(x) = \frac{1}{(n+1)!} f^{n+1}(\xi_x) \prod_{j=0}^{m}(x - x_j)^{\mu_j + 1}$

2.6 Numerische Stabilität von Interpolationspolynomen

1. **Satz**

   Sei $p \in P_n$ Interpolationspolynom zu (x_i, y_i), $i = 0...n$ und $\hat{p} \in P_n$ das Interpolationspolynom zu $(x_i, y_i + \Delta_i)$.

   Dann gilt mit $x_i \in [a,b]$:

   $\|p - \hat{p}\|_\infty \leq \Lambda_n \max\limits_{i=0...n} |\Delta_i|$

   $\Lambda_n = \| \sum_{i=0}^{n} |L_i^{(n)}| \|_\infty$ (Lebesgue Konstante)

   Die Lebesgue Konstante gibt an, wie gut die Interpolation einer Funktion im Vergleich zur besten polynomialen Approximation derselben ist.

   Das heißt, die Verteilung der Stützstellen ist optimalerweise so zu wählen, dass Λ_n möglichst klein bleibt.

2.7 Richardson-Extrapolation zum Limes

1. Problemstellung

Ein numerischer Prozess liefert für jeden Wert eines positiven Parameters $h \in \mathbb{R}_+$ $(h \to 0)$ einen Wert $a(h)$.

Gesucht ist die nicht direkt berechenbare Größe $a(0) = \lim_{h \to 0} a(h)$.

Zur Näherung von $a(0)$ berechnet man $a(h_i)$ für Werte $h_i, i = 0, ..., n$ und nimmt den Wert $p_n(0)$ des zugehörigen Interpolationspolynoms zu $(h_i, a(h_i))$ als Schätzung für $a(0)$.

2. Extrapolationsfehler

Für die Funktion $a(h)$, $h \in \mathbb{R}_+$, sei bekannt, dass eine asymptotische Entwicklung der Form

$a(h) = a_0 + \sum_{j=1}^{n} a_j h^{jq} + a_{n+1}(h) h^{(n+1)q}$

gilt, mit einem $q > 0$ und gewissen Koeffizienten $a_j, a_{n+1}(h) = a_{n+1} + o(1)$ für $h \to 0$.

Sei $\{h_k\}_{k=0,1,2,...}$ eine monoton fallende Folge positiver Zahlen mit der Eigenschaft:

$0 < \frac{h_{k+1}}{h_k} \leq \delta < 1$

Für das Interpolationspolynom $p_n^{(k)} \in P_n$ durch $(h_k^q, a(h_k)), ..., (h_{k+n}^q, a(h_{k+n}))$ gilt dann:

$a(0) - p_n^{(k)}(0) = O(h_k^{(n+1)q}) \quad (k \to \infty)$

3 Numerische Integration

3.1 Interpolatorische Quadraturformeln

1. Definition

Mit dem Lagrange-Interpolationspolynom:

$p_n(x) = \sum_{i=0}^{n} f(x_i) L_i^{(n)}(x)$

$I^{(n)}(f) := \int_a^b p_n(x)dx = \sum_{i=0}^{n} f(x_i) \underbrace{\int_a^b L_i^{(n)}(x)dx}_{\text{Gewichte } \alpha_i} = \sum_{i=0}^{n} \alpha_i f(x_i)$

2. Ordnung einer Quadraturformel

Eine Quadraturformel $I^{(n)}(\cdot)$ wird mindestens von der Ordnung m genannt, wenn durch sie alle Polynome P_{m-1} exakt integriert werden.

Die interpolatorischen Quadraturformeln $I^{(n)}(\cdot)$ zu $n+1$ Stützstellen sind also mindestens von der Ordnung $n+1$.

3. Spezialfall äquidistante Stützstellen

(a) abgeschlossene Newton-Cotes-Formeln (a, b sind Stützstellen)

$x_i = a + ih \quad i = 0, .., n \quad h = \frac{b-a}{n}$

(b) offene Newton-Cotes-Formeln (a, b sind keine Stützstellen)

$x_i = a + (i+1)h \quad i = 0, ..., n \quad h = \frac{b-a}{n+2}$

Durch Koordinatentransformation $x \to t = \frac{(x-a)}{h}$ (abgeschlossene Formel) folgt:

$L_i^{(n)}(x) = \prod_{\substack{j=0 \\ i \neq j}}^{n} \frac{x-x_j}{x_i-x_j} = \prod_{\substack{j=0 \\ i \neq j}}^{n} \frac{a+th-a-jh}{a+ih-a-jh} = \prod_{\substack{j=0 \\ i \neq j}}^{n} \frac{t-j}{i-j}$

$\alpha_i = \int_a^b L_i^{(n)}(x)dx = h \int_a^n \prod_{\substack{j=0 \\ i \neq j}}^{n} \frac{t-j}{i-j} dt \quad i = 0, ..., n$

Regeln:

(a) Abgeschlossene Formeln: $(n = 1, 2, 3, 4) \quad H := \frac{b-a}{n}$

$$I^{(1)}(f) = \frac{b-a}{2}\{f(a) + f(b)\} \qquad \text{(Trapezregel)}$$
$$I^{(2)}(f) = \frac{b-a}{6}\{f(a) + 4f(\frac{a+b}{2}) + f(b)\} \qquad \text{(Simpson-Regel)}$$
$$I^{(3)}(f) = \frac{b-a}{8}\{f(a) + 3f(a+H) + 3f(b-H) + f(b)\} \qquad (\tfrac{3}{8}\text{-Regel})$$
$$I^{(4)}(f) = \frac{b-a}{90}\{7f(a) + 32f(a+H) + 12f(\frac{a+b}{2}) + 32f(b-H) + 7f(b)\}$$

(b) Offene Formeln: $(n = 0, 1, 2, 3) \quad H := \frac{b-a}{n}$

$$I^{(0)}(f) = (b-a)f(\frac{a+b}{2}) \qquad \text{(Mittelpunktsregel)}$$
$$I^{(1)}(f) = \frac{b-a}{2}\{f(a+H) + f(b-H)\}$$
$$I^{(2)}(f) = \frac{b-a}{3}\{2f(a+H) - f(\frac{a+b}{2}) + 2f(b-H)\}$$
$$I^{(3)}(f) = \frac{b-a}{24}\{11f(a+H) + f(a+2H) + f(b-2H) + 11f(b-H)\}$$

4. Quadraturrestglieder

Es gelten folgende Restglieddarstellungen:

i) für Trapezregel:
$I(f) - \frac{b-a}{2}\{f(a) + f(b)\} = -\frac{(b-a)^3}{12}f''(\zeta) \quad f \in C^2[a,b]$

ii) für Simpson-Regel:
$I(f) - \frac{b-a}{6}\{f(a) + 4f(\frac{a+b}{2}) + f(b)\} = -\frac{(b-a)^5}{2880}f^{(4)}(\zeta) \quad f \in C^4[a,b]$

5. Summierte Regeln

Summierte Trapezregel $(m = 1)$:

$I_h^{(1)}(f) = \sum_{i=0}^{N-1} \frac{x_{i+1}-x_i}{2}\{f(x_i) + f(x_{i+1})\} = \frac{h}{2}\{f(a) + 2\sum_{i=1}^{N-1} f(x_i) + f(b)\}$

$I(f) - I_h^{(1)}(f) = -\frac{b-a}{12}h^2 f''(\xi), \quad \xi \in [a,b]$

Summierte Simpson-Regel $(m = 3)$:

$$I_h^{(2)}(f) = \sum_{i=0}^{N-1} \frac{x_{i+1}-x_i}{6}\{f(x_i) + 4f(\frac{x_i+x_{i+1}}{2}) + f(x_{i+1})\}$$
$$= \frac{h}{6}\{f(a) + 2\sum_{i=1}^{N-1} f(x_i) + 4\sum_{i=0}^{N-1} f(\frac{x_i+x_{i+1}}{2}) + f(b)\}$$

$I(f) - I_h^{(2)}(f) = -\frac{b-a}{2880}h^4 f^{(4)}(\xi), \quad \xi \in [a,b]$

Summierte Mittelpunktsregel $(m = 1)$:

$I_h^{(0)}(f) = \sum_{i=0}^{N-1}(x_{i+1} - x_i)f(\frac{x_i+x_{i+1}}{2}) = h\sum_{i=0}^{N-1} f(\frac{x_i+x_{i+1}}{2})$

$I(f) - I_h^0(f) = \frac{b-a}{24}h^2 f''(\xi), \quad \xi \in [a,b]$

3.2 Gaußsche Quadraturformeln

1. Definition

Die interpolatorischen Quadraturformeln der Form $I^{(n)}(f) = \sum_{i=0}^{n} \alpha_i f(x_i)$ zu den Stützstellen $x_0, ..., x_n \in [a, b]$ sind nach Konstruktion mindestens von der Ordnung $n+1$, d.h. $R^{(n)}(p) := I(p) - I^{(n)}(p) = 0 \quad p \in P_n$.

Eine obere Grenze für die Ordnung einer Quadraturformel der Art $I^{(n)}(\cdot)$ ist $2n+2$.

Eine Quadraturformel, die diese Eigenschaft besitzt, heißt Gauß-Quadraturformel.

2. Bestimmung optimaler Stützstellen

Verwendet werden das übliche L^2-Skalarprodukt und die zugehörige Norm:

$(f, g) := \int_a^b f(x)g(x)dx, \quad \|f\| := (f, f)^{\frac{1}{2}}$

Da die Bedingung

$\int_a^b \prod_{j=0}^{n}(x - x_j) \cdot q(x)dx = 0 \quad \forall q \in P_n$

besagt, dass das Polynom

$p(x) = \prod_{j=0}^{n}(x - x_j) = x^{n+1} + r(x), \quad r \in P_n$

bezüglich des Skalarprodukts $(\cdot, \cdot)$ orthogonal zum Teilraum $P_n[a, b]$ ist, werden die Nullstellen über Anwenden des Gram-Schmidt-Verfahrens auf die Monombasis $\{1, x, x^2, x..., x^{n+1}\}$ von $P_{n+1}[a, b]$ angewendet:

$p_0(x) = 1$

$k = 1, ..., n+1: \quad p_k(x) := x^k - \sum_{j=1}^{k-1} \frac{(x^k, p_j)}{\|p_j\|^2} p_j(x)$

Damit ergibt sich das Orthogonalsystem $p_0, p_1, ..., p_{n+1}$.

Die $n+1$ Nullstellen von $p_{n+1}(x)$ sind dann mögliche Integrationspunkte.

Die orthogonalen Polynome p_n bezüglich des Skalarproduktes $(\cdot, \cdot)$ auf $[-1, 1]$ sind Vielfache der Legendre-Polynome $L_n(x)$, man kann also die Nullstellen von L_{n+1} als Stützstellen für eine interpolarische Quadraturformel auf $[-1, 1]$ verwenden.

4 Lineare Gleichungssysteme

4.1 Grundbegriffe

1. Problemstellung

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \in \mathbb{R}^{n \times n} \quad b = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \in \mathbb{R}^n$$

Gesucht ist ein Vektor $x \in \mathbb{R}^n$ sodass $Ax = b$.

Das lineare Gleichungssystem ist genau dann lösbar, wenn $\text{Rang}(A) = \text{Rang}([A, b])$,

$$[A, b] = \begin{bmatrix} a_{11} & \cdots & a_{1n} & b_1 \\ \vdots & & \vdots & \vdots \\ a_{n1} & \cdots & a_{nn} & b_n \end{bmatrix}$$

Folgende Aussagen sind äquivalent:

(i) $Ax = b$ ist für jedes b eindeutig lösbar

(ii) $\text{Rang}(A) = n$

(iii) $\det(A) \neq 0$ (d.h. A ist invertierbar)

(iv) Alle Eigenwerte von A sind ungleich 0

2. Eigenwerte und Eigenvektoren

Für $A \in \mathbb{K}^{n \times n}$ ist $\lambda \in \mathbb{K}$ ein Eigenwert, falls ein Eigenvektor $v \in \mathbb{K}^n \setminus \{0\}$ existiert mit:

$Av = \lambda v$

$G(A) = \{\lambda \in \mathbb{K}, \quad \lambda \text{ Eigenvektor von } A\}$ heißt Spektrum von A.

3. Hermitische, symmetrische Matrizen

Eine Matrix $A \in \mathbb{K}^{n \times n}$ heißt hermitesch, falls gilt:

$A = \overline{A}^\top$

Falls $A \in \mathbb{R}^{n \times n}$ und $A = A^\top$, so heißt A symmetrisch.

$B \in \mathbb{K}^{n\times n}$ heißt positiv (semi) definit, falls:

$\overline{x}^\top Bx \in \mathbb{R}$

$\overline{x}^\top Bx > 0$ bzw. $\overline{x}^\top Bx \geq 0 \quad \forall x \in \mathbb{K}^n \setminus \{0\}$

4. Skalarprodukt

Eine Abbildung $(\cdot,\cdot) : \mathbb{K}^n \times \mathbb{K}^n \to \mathbb{K}$ heißt Skalarprodukt, falls:

S1)	$(x,y) = \overline{(x,y)}$	$\forall x, y \in \mathbb{K}^n$	(Symmetrie)
S2)	$(\alpha x + \beta y, z) = \alpha(x,z) + \beta(y,z)$	$\forall x, y, z \in \mathbb{K}^n, \alpha, \beta \in \mathbb{K}$	(Linearität)
S3)	$(x,y) > 0$	$\forall x \in \mathbb{K}^n \setminus \{0\}$	(Definitheit)

Ein Skalarprodukt erzeugt eine Vektornorm durch:

$\|x\| \coloneqq \sqrt{(x,x)}$

4.2 Normen

1. Definition

Eine Abbildung $\|\cdot\| : \mathbb{K} \to \mathbb{R}_+$ heißt Norm, falls

(i)	$\|x\| > 0, \quad x \in \mathbb{K}^n \setminus \{0\}$	(Definitheit)
(ii)	$\|\alpha x\| = \|\alpha\|\|x\|, \quad x \in \mathbb{K}^n, \alpha \in \mathbb{K}$	(absolute Homogenität)
(iii)	$\|x+y\| \leq \|x\| + \|y\|, \quad x, y \in \mathbb{K}^n$	(Subadditivität)

2. Wichtige Normen

$\|x\|_2$	$=$	$(\sum_{k=1}^n \|x_k\|^2)^{\frac{1}{2}}$	Euklidische Norm
$\|x\|_\infty$	$=$	$\max\limits_{i=1\ldots n} \|x_i\|$	Maximum Norm
$\|x\|_1$	$=$	$\sum_{k=1}^n \|x_k\|$	l_1-Norm
$\|x\|_p$	$=$	$\begin{cases} (\sum_{i=1}^n \|x_i\|^p)^{\frac{1}{p}}, & p \in [1,\infty) \\ \max\limits_{1\leq i\leq n} \|x_i\|, & p = \infty \end{cases}$	lp-Norm

3. Normäquivalenz

Auf dem endlich dimensionalen Vektorraum $\mathbb{K}^n$ sind alle Normen äquivalent, d.h. zu je zwei Normen $\|\cdot\|, \|\cdot\|'$ gibt es positive Konstanten m, M, sodass

$m\|x\| \leq \|x\|' \leq M\|x\|, \quad x \in \mathbb{K}^n.$

4. Verträglichkeit und Matrixnormen

Eine Norm $\|\cdot\|$ auf $\mathbb{K}^{n\times n}$ heißt verträglich mit einer Vektornorm $\|\cdot\|$ auf $\mathbb{K}^n$, wenn gilt:

$\|Ax\| \leq \|A\| \cdot \|x\|, \quad x \in \mathbb{K}^n, \quad A \in \mathbb{K}^{n\times n}.$

Sie heißt Matrizennorm, wenn sie submultiplikativ ist:

$\|AB\| \leq \|A\| \cdot \|B\|, \quad A, B \in \mathbb{K}^{n\times n}$

Für eine beliebige Vektornorm $\|\cdot\|$ auf $\mathbb{K}^n$ wird durch

$\|A\| := \sup\limits_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|}{\|x\|} = \sup\limits_{\substack{x\in\mathbb{K}^n \\ \|x\|=1}} \|Ax\|$ (natürliche Norm)

eine mit $\|\cdot\|$ verträgliche Matrixnorm definiert.

5. Wichtige Matrixnormen

$\|A\|_\infty = \max\limits_{1\leq j\leq n} \sum_{k=1}^n |a_{jk}|$ (Maximum Norm)

$\|A\|_1 = \max\limits_{1\leq k\leq n} \sum_{j=1}^n |a_{jk}|$ (l_1-Norm)

$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}$ (Frobenius Norm)

4.3 Störungstheorie

1. Fehleranalyse

Man betrachte $Ax = b$, wobei $A \in \mathbb{K}^{n\times n}$ regulär ist.
$\delta A, \delta b$ bezeichnen Störungen von A und B.

$\Longrightarrow$ gestörtes System $\tilde{A}\tilde{x} = \tilde{b}$, wobei
$\tilde{A} = A + \delta A, \quad \tilde{b} = b + \delta b$

Fehler: $\delta x = \tilde{x} - x$

2. Störungssatz

Sei $A \in \mathbb{K}^{n\times n}$ regulär, $\|\delta A\| < \frac{1}{\|A^{-1}\|}$.

Dann ist $\tilde{A} = A + \delta A$ regulär und es gilt:

$\frac{\|\delta x\|}{\|x\|} \leq \frac{\mathrm{cond}(A)}{1-\mathrm{cond}(A)\frac{\|\delta A\|}{\|A\|}} \left\{ \frac{\|\delta b\|}{\|b\|} + \frac{\|\delta A\|}{\|A\|} \right\}$

mit Konditionszahl $\mathrm{cond}(A) = \|A\| \cdot \left\|A^{-1}\right\|$.

Die Konditionszahl ist abhängig von der verwendeten Matrixnorm. In der Praxis wird sie oft bezüglich der Maximum oder L_2-Norm abgeschätzt.

Für hermitische Matrizen gilt $\mathrm{cond}_2(A) = \frac{|\lambda_{max}|}{|\lambda_{min}|}$ (Spektralkonditionszahl).

Wenn $\text{cond}(A) = 10^s$ und die Elemente von A und b mit einem relativen Fehler der Art

$\frac{\|\delta A\|}{\|A\|} \sim 10^{-k}, \quad \frac{\|\delta b\|}{\|b\|} \sim 10^{-k} \quad (k > s)$

behaftet $\longrightarrow$ relativer Fehler des Ergebnisses der Größenordnung $\frac{\|\delta x\|}{\|x\|} \sim 10^{s-k}$

4.4 Gaußsches Eliminationsverfahren

1. Gestaffelte Systeme

Besonders leich lösbar, z.B. obere Dreiecksmatrix $A = (a_{jk})$ als Koeffizienten-Matrix:

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ & & a_{22}x_2 & + & \cdots & + & a_{2n}x_x & = & b_2 \\ & & & \ddots & & & & & \vdots \\ & & & & & & a_{nn}x_n & = & b_n \end{array}$$

Lösen durch Rückwärtseinsetzen mit $\frac{1}{2}n^2 + O(n)$ arithmetischen Operationen:

$x_n = \frac{b_n}{a_{nn}}, \quad j = n-1, ..., 1, \quad x_j = \frac{1}{a_{ji}}(b_j - \sum_{k=j+1}^{n} a_{jk}x_k)$

2. Gauß Verfahren:

Umformen eines gegebenen Systems $Ax = b$ in ein oberes Dreiecksystem $Rx = c$, welches die selbe Lösung x besitzt und dann durch Rückwärtseinsetzen gelöst werden kann.

Erlaubte Operationen:

1. Vertauschen zweier Gleichungen
2. Addition des Vielfachen einer Gleichung zu einer anderen

Darstellung des Verfahrens mit Permutationsmatrizen:

Permutationsmatrix:

$$P_i = \begin{array}{c} \begin{matrix} & & & i & & & r & & & \end{matrix} \\ \begin{bmatrix} 1 & & & & & & & & & \\ & \ddots & & & & & & & & \\ & & 1 & & & & & & & \\ & & & 0 & \cdots & & 1 & & & \\ & & & & 1 & & & & & \\ & & & \vdots & & \ddots & \vdots & & & \\ & & & & & & 1 & & & \\ & & & 1 & \cdots & & 0 & & & \\ & & & & & & & 1 & & \\ & & & & & & & & \ddots & \\ & & & & & & & & & 1 \end{bmatrix} \end{array} \begin{matrix} \\ \\ \\ i \\ \\ \\ \\ r \\ \\ \\ \end{matrix}$$

Frobenius-Matrix:

$$G_i = \begin{array}{c} \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -q_{i+1,i} & 1 & \\ & & \vdots & & \ddots \\ & & -q_{n,i} & & 1 \end{bmatrix} \end{array} \begin{matrix} \\ \\ i \\ \\ \\ \end{matrix} \quad , \qquad q_{ji} = \frac{a_{ji}}{a_{jj}}$$

Sei $A^{(0)} = A, \quad b^{(0)} = b.$

Dann wird das gestaffelte System $[R, c]$ berechnet durch:

$[R, c] = G_{n-1} P_{n-1} \cdots G_1 P_1 [A, b]$

Algorithmus:

1. Pivotisierung:

- Bestimmen von $a_{ri} \neq 0$ (Pivotelement), meist durch $|a_{ri}| = \max\limits_{1 \leq j \leq n} |a_{ji}|$

- Berechnen von $[\tilde{A}^{(i)}, \tilde{b}^{(i)}] = P_i [A^{(i)}, b^{(i)}]$

2. Elimination:

- Berechnen von $[A^{(i+1)}, b^{(i+1)}] = G_i [A^{(i)}, b^{(i)}]$

3. Determinantenberechnung

Für quadratische Matrizen gilt der Determinantensatz

$\det(AB) = \det(A)\det(B)$.

Für die durch Gauß-Elimination aus der Matrix A gewonnene Dreiecksmatrix

$R = G_{n-1}P_{n-1}\cdots G_1P_1A$

folgt somit unter der Beachtung von

$\det\left(P_i^{-1}\right) = \det(P_1) = -1, \quad \det\left(G_i^{-1}\right) = 1$

die Beziehung

$\det(A) = \det\left(P_1^{-1}G_1^{-1}\cdots P_{n-1}^{-1}G_{n-1}^{-1}R\right) = \pm\det(R) = \pm\prod_{j=1}^{n} r_{jj}$

4.5 LR-Zerlegung

1. Definition

Die Matrizen $L = G_1^{-1}\cdots G_{n-1}^{-1}$ und $R = G_{n-1}\cdots G_1A$ sind in der Form

$$L = \begin{bmatrix} 1 & & & \\ l_{11} & \ddots & & \\ \vdots & & \ddots & \\ l_{n1} & \cdots & \cdots & 1 \end{bmatrix} \text{ und } R = \begin{bmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ & & r_{nn} \end{bmatrix}$$

und bilden eine sogenannte LR-Zerlegung der Matrix PA:

$PA = LR \quad P = P_{n-1}\cdots P_1$

Diese Zerlegung ist im Falle $P = I$ eindeutig bestimmt.

2. Aufwand

Die Lösung der Gleichung $Ax = b$ mithilfe der LR-Zerlegung benötigt

$N_{LR}(n) = \frac{n^3}{3} + O(n^2)$

arithmetische Operationen.

3. Lösungsverfahren

1. Berechne L und R
2. Löse $Ly = b$ durch Vorwärtseinsetzen
3. Löse $Rx = y$ durch Rückwärtseinsetzen

4.6 Cholesky-Zerlegung

1. Definition

Sei $A \in \mathbb{R}^{n\times n}$ symmetrisch, positiv definit. Dann existiert eine weitere Dreiecksmatrix L und eine Diagonalmatrix D, sodass gilt:

$A = LDL^\top = \tilde{L}\tilde{L}^\top$, wobei $\tilde{L} = LD^{\frac{1}{2}}$.

2. Algorithmus

Formeln (ACHTUNG: Nicht so in der Vorlesung aufgeschrieben!):

$$\tilde{l}_{ij} = \begin{cases} 0 & \text{für } i < j \\ \sqrt{a_{ii} - \sum_{k=1}^{i-1} \tilde{l}_{ik}^2} & \text{für } i = j \\ \frac{1}{\tilde{l}_{jj}}(a_{ij} - \sum_{k=1}^{j-1} \tilde{l}_{ik}\tilde{l}_{jk}) & \text{für } i > j \end{cases}$$

$$\tilde{l}_{i1} = \begin{cases} \sqrt{a_{11}} & \text{für } i = 1 \\ \frac{a_{i1}}{\tilde{l}_{11}} = \frac{a_{i1}}{\sqrt{a_{11}}} & \text{für } i > 1 \end{cases}$$

3. Aufwand

Der Rechenaufwand zur Berechnung von L, D ist $N(n) = \frac{1}{6}n^3 + O(n^2)$.

4. Lösungsverfahren

1. Berechne L und $L^\top$

2. Löse $LL^\top = b$ analog zur LR-Zerlegung Einsetzen

4.7 Gleichungssysteme mit spezieller Struktur

1. Bandmatrizen

Eine Matrix $A \in \mathbb{R}^{n\times n}$ heißt Bandmatrix vom Bandtyp m_l, m_r mit $0 \le m_l, m_r \le n-1$ wenn gilt:

$a_{jk} = 0 \quad \text{für} \quad k < j - m_l \quad \text{oder} \quad k > j + m_r \quad (j,k = 1,...,n)$

Das heißt die Elemente von A sind bis auf die Hauptdiagonale und höchstens $m_l + m_r$ Nebendiagonalen gleich Null.

Die Größe $m = m_l + m_r + 1$ heißt Bandbreite.

Ist $A \in \mathbb{R}^{n\times n}$ eine Bandmatrix vom Typ (m_l, m_r), für die die Gauß Elimination ohne Permutation durchführbar ist, dann sind die Matrizen L und R Bandmatrizen vom Typ $(m_l, 0)$ bzw. $(0, m_r)$.

Der Aufwand für die Berechnung der LR-Zerlegung einer Bandmatrix vom Typ m_l, m_r ist:

$N = \frac{1}{3} n \cdot m_l \cdot m_r + O(n(m_l + m_r))$.

Bsp.: LR-Zerlegung von Tridiagonal-Matrix

$$A = \begin{bmatrix} a_1 & b_1 & & 0 \\ c_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & c_n & a_n \end{bmatrix}.$$

Dann ist $A = LR$ mit

$$L = \begin{bmatrix} 1 & & & \\ \gamma_2 & \ddots & & \\ & \ddots & 1 & \\ & & \gamma_n & 1 \end{bmatrix}, \; R = \begin{bmatrix} \alpha_1 & \beta_1 & & \\ & \ddots & \ddots & \\ & & \alpha_{n-1} & \beta_{n-1} \\ & & & \alpha_n \end{bmatrix},$$

$\alpha_1 = a_1, \qquad \beta_1 = b_1 \qquad\qquad i = 2, ..., n-1$

$\gamma_i = \frac{c_i}{\alpha_{i-1}}, \qquad \alpha_i = a_i - \gamma_i \beta_{i-1}, \qquad \beta_i = b_i$

$\gamma_n = \frac{c_n}{\alpha_{n-1}}, \quad \alpha_n = a_n - \gamma_n \beta_{n-1}$

Anzahl an Operationen: $2n - 2$, Anzahl an Speicher: $3n - 2$

2. Diagonaldominante Matrizen

Eine Matrix $A \in \mathbb{R}^{n \times n}, A = (a_{ij})_{i,j=1...n}$ heißt (strikt) diagonaldominant, falls

$\sum_{\substack{k=1 \\ k \neq j}}^{n} |a_{jk}| \leq |a_{jj}| \quad$ bzw. $\quad \sum_{\substack{k=1 \\ k \neq j}}^{n} |a_{jk}| < |a_{jj}|, \quad j = 1...n$

Wenn eine reguläre diagonaldominante Matrix ist, dann existiert eine LR-Zerlegung $A = LR$ ohne Pivotisierung.

3. Symmetrisch positiv definite Matrix

Eine symmetrisch positiv definite Matrix besitzt eine LR-Zerlegung ohne Pivotierung, wobei $a_{ii}^{(i)} > 0, \quad i = 1, ..., n-1$.

4.8 Nicht reguläre Systeme

1. Definition

Mit einer nicht notwendig quadratischen Matrix $A \in \mathbb{R}^{m \times n}$ und einem Vektor $b \in \mathbb{R}^m$ sei das Gleichungssystem $Ax = b$ gegeben.

Es wird auch $\text{Rang}(A) < \text{Rang}[A, b]$ zugelassen, d.h. das System muss nicht unbedingt im eigentlichen Sinne lösbar sein.

2. Least-Squares-Lösung

Es existiert stets eine Lösung mit kleinsten Fehlerquadraten, das heißt

$\|A\overline{x} - b\|_2 = \min\limits_{x \in \mathbb{R}^n} \|Ax - b\|_2.$

Dies ist äquivalent dazu, dass $\overline{x}$ Lösung der Normalgleichung ist:

$A^\top A\overline{x} = A^\top b$

Im Falle $\text{Rang}(A) = n$ ist $\overline{x}$ eindeutig bestimmt, andernfalls ist jede weitere Lösung von der Form $\overline{x} + y$ mit $y \in \text{Kern}(A)$.

3. Gaußsche Ausgleichrechnung

Zu gegebenen Funktionen $u_1, ..., u_n$ und Punkten $(x_j, y_j) \in \mathbb{R}^2, j = 1, ..., m, m > n$ ist eine Linearkombination

$u(x) = \sum_{k=1}^n c_k u_k(x)$

so zu bestimmen, dass die sogenannte mittlere Abweichung

$\Delta_2 \equiv (\sum_{i=1}^n |u(x_i) - y_i|^2)^{\frac{1}{2}}$

minimiert wird.

Lösung:

$y = (y_1, ..., y_m)^\top, \quad c = (c_1, ..., c_n)^\top$

$a_k \coloneqq (u_k(x_1), ..., u_k(x_m))^\top, \quad k = 1, ..., n$

$A = [a_1, ..., a_n]$

Zu minimieren ist dann bezüglich $\mathbb{R}^n$ das Funktional

$F(c) = \|Ac - y\|_2.$

$\Longrightarrow A^\top Ac = A^\top y$ (Normal-Gleichung)

4.9 QR-Zerlegung

1. Definition

Sei $A \in \mathbb{K}^{m \times n}$ eine rechteckige Matrix mit $m > n$ und $\text{Rang}(A) = n$.

Dann existiert eine eindeutig bestimmte Matrix $Q \in \mathbb{K}^{m \times n}$ mit:

$Q^\top Q = I \quad (\mathbb{K} = \mathbb{R}), \quad \overline{Q}^\top Q = I \quad (\mathbb{K} = \mathbb{C})$

und eine eindeutig bestimmte obere Dreiecksmatrix $R \in \mathbb{K}^{n \times n}$ mit reellen Diagonalelementen $r_{ii} > 0, \quad i = 1, ..., n$, sodass:

$A = QR$

2. Berechnung

Die Matrix Q wird bestimmt durch sukzessive Orthogonalisierung der Spaltenvektoren $a_k, k = 1, ..., n$ von A mithilfe des Gram-Schmidt-Verfahrens:

$$\begin{cases} q_1 = \frac{a_1}{\|a_1\|_2} \\ \tilde{q}_k = a_k - \sum_{i=1}^{k-1} (a_k, q_i)_2 q_i, \quad q_k = \frac{\tilde{q}_k}{\|\tilde{q}_k\|_2}, \quad k = 2, ..., n \end{cases}$$

Berechnung von R folgt aus:

$a_k = \sum_{i=1}^{k} r_{ik} q_k, \quad r_{kk} \equiv \|\tilde{q}_k\|_2 \in \mathbb{R}_+, \quad r_{ik} \equiv (a_k, q_i)_2, \quad r_{ik} \equiv 0 \text{ für } i > k$

$\longrightarrow R = (r_{ik}) \in \mathbb{K}^{n \times n}$

3. Lösungsverfahren

1. Bestimme Q und R
2. Berechne $z = Q^\top b$
3. Löse $Rx = z$ durch Rückwärtseinsetzen

5 Nicht lineare Gleichungssysteme

5.1 Iterative Verfahren

1. Definition

Bei einem iterativen Verfahren wird ein Wert x^* durch sukzessive Anwendung einer gegen x^* konvergierende Funktion ϕ berechnet:

$x_0 \to x_1 = \phi(x_0) \to x_2 = \phi(x_1) \cdots \to x_n = \phi(x_{n-1}) \cdots$

Dabei muss $F(x^*) = 0$ sein, damit $x_k \xrightarrow{k\to\infty} x^*$.

2. Banachscher Fixpunktsatz

Sei $M \subset \mathbb{R}$ eine abgeschlossene Teilmenge und die Abbildung $\phi : M \to M$ bezüglich einer Vektornorm $\|\cdot\| : \mathbb{R}^n \to \mathbb{R}$ eine Kontraktion, d.h. für eine Konstante $0 < L < 1$ sei

$\|\phi(x) - \phi(y)\| \leq L\|x - y\|, \quad x, y \in M$

erfüllt.

Dann gelten folgende Aussagen:

a) ϕ besitzt genau einen Fixpunkt $x^* \in M$

b) Für jeden Startwert $x_0 \in M$ liefert die Fixpunktiteration

$x_{k+1} = \phi(x_k), \quad k = 0, 1, ...$

eine gegen x^* konvergierende Folge und es gilt genauer:

$$\|x_k - x^*\| \leq \underbrace{\frac{L}{1-L}\|x_k - x_{k-1}\|}_{\text{a-posteriori}} \leq \underbrace{\frac{L^k}{1-L}\|x_1 - x_0\|}_{a-priori}$$

5.2 Verfahren für lineare Gleichungssysteme

1. **Jacobi-Verfahren**

   Da $A = L + D + U$ existiert, wobei L untere Dreiecks-, D Diagonal- und U obere Dreiecksmatrix, kann man schreiben:

   $Ax - b = 0 \iff (L + D + U)x = b \iff x_{k+1} = -D^{-1}(L + U)x_k + D^{-1}b$

2. **Gauß-Verfahren**

   Wie oben nimmt man an, dass $A = L + D + U$. Dann folgt:

   $(L + D + U)x = b \iff x_{k+1} = -(L + D)^{-1}Ux_k + (L + D)^{-1}b$, falls $(L + D)$ invertierbar.

5.3 Newton-Verfahren in $\mathbb{R}^1$

1. **Definition**

   Gegeben sei $f \in C^2([a, b])$.

   Ziel:

   Bestimmen von $x^* \in [a, b]$ mit $f(x^*) = 0$

   Idee:

   gegeben ist $x_k \in [a, b]$

   Tangente auf f in x_k: $T(x) = f'(x_k)(x - x_k) + f(x_k)$

   x_{n+1} ist Nullstelle von T, d.h. es ist:

   $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$, falls $f'(x_k) \neq 0$

2. **Konvergenz**

   Sei $f \in C^2([a, b])$ und $z \in (a, b)$, $f(z) = 0$,

   $m := \min\limits_{x\in[a,b]} |f'(x)|, \quad M := \max\limits_{x\in[a,b]} |f''(x)|.$

   Sei nun $m > 0$ und $\delta > 0$, sodass

   $q : \frac{M}{2m}\delta < 1$ und $K_\delta(z) = \{x : |x - z| \leq \delta\} \subset [a, b]$

   Dann sind für jedes $x_0 \in K_\delta(z)$ die Newton-Iterierten $x_k \in K_\delta(z)$ definiert und $x_k \xrightarrow{k\to\infty} z$.

   Es gilt:

   $|x_k - z| \leq \frac{2m}{M} q^{(2^k)}$ (a-priori)

   $|x_k - z| \leq \frac{1}{m}|f(x_k)| \leq \frac{M}{2m}|x_k - x_{k-1}|^2$ (a-posteriori)

3. Anmerkungen

- $\min\limits_{x\in[a,b]} |f'(x)| > 0 \Longleftrightarrow$ benötigt für $\begin{cases} \text{Eindeutigkeit} \\ \{x_k\} \text{ ist wohldefiniert} \end{cases}$
- $f \in C^2 \rightarrow$ Taylor
- $|x_0 - z| < \delta$, wobei $\delta < \frac{2m}{M}$ und $K_\delta(z) \subset [a, b]$
- falls m klein und M groß $\Rightarrow \delta$ sehr klein

4. Fazit

+ extrem schnelle Konvergenz
+ Erweiterung auf $\mathbb{R}^n$ möglich

- Startpunkt muss in der Nähe der unbekannten Lösung gewählt werden
- Voraussetzung in Praxis nicht überprüfbar

5.4 Konvergenzverhalten iterativer Verfahren

1. Konvergenz der Ordnung q

Eine Konvergenz der Ordnung $q > 1$ liegt vor, wenn (s_k) konvergiert und ein $c > 0$ existiert mit:

$|s_{k+1} - s| \leq c_k |s_k - s|^q, \quad k = 0, 1, ...$

Test auf maximale Ordnung:

Lineare Konvergenz: $\lim\limits_{k\to\infty} \sup \frac{|x_{k+1}-x^*|}{|x_k-x^*|} = c < 1$

Konvergenz höherer Ordnung: $\lim\limits_{k\to\infty} \sup \frac{|x_{k+1}-x^*|}{|x_k-x^*|^q} = c < 1, \quad q > 1$

2. Anziehend / Abstoßend

Ein Fixpunkt z einer stetig differenzierbaren Abbildung g heißt anziehend, wenn $|g'(z)| < 1$ ist, da dann die Fixpunktiteration für jeden hinreichend nahe bei z gelegenen Startwert gegen ihn konvergiert.

Im Fall $|g'(z)| > 1$ heißt er abstoßend.

3. Satz zur Konvergenzordnung

Die Funktion g sei in einer Umgebung des Fixpunktes z p-mal stetig differenzierbar mit $p \geq 2$.

Genau dann hat die Fixpunktiteration $x_{k+1} = g(x_k)$ die Ordnung p, wenn gilt:

$g'(x) = \cdots = g^{(p-1)}(z) = 0$ und $g^{(p)}(z) \neq 0$.

5.5 Newton-Verfahren in $\mathbb{R}^n$

1. Annahmen

i) $\|\cdot\|$ bezeichne die euklidische Vektornorm mit zugehöriger natürlicher Matrixnorm

ii) $f : G \to \mathbb{R}^n$ sei stetig differenzierbar, d.h. $f \in C^1$

iii) für $y \in G$ ist die Niveaumenge definiert als $D(y) := \{x \in G : \|f(x)\| \leq \|f(y)\|\}$

iv) es existiere $x^* \in G$, sodass gilt:

a) $Df(x)^{-1}$ existiert auf $D_* := D(x^*)$

b) $\left\|Df(x)^{-1}\right\| \leq \beta \quad \forall x \in D_*$

c) $\|Df(x) - Df(y)\| \leq \gamma\|x - y\| \quad \forall x, y \in D_*$

2. Definition

Sei $f : G \subset \mathbb{R}^n \to \mathbb{R}^n$ gegeben.

Newton zur Lösung von $f(x) = 0$:

$$x_{k+1} = x_k - \underbrace{Df(x)^{-1}f(x_k)}_{\delta x_k}, \quad k = 0, 1, 2, ...,$$

mit Startwert $x_0 \in G$, Jakobi-Matrix $Df(x)_{ij} = \frac{df_i}{dx_j} \in \mathbb{R}^{n\times n}$

3. Praktische Umsetzung

i) Löse $Df(x_k)\delta x_k = f(x_k)$

ii) Update: $x_{k+1} = x_k - \delta x_k$

4. Statz (Newton-Kantorovich)

Es gelten die Annahmen von oben, sowie $x_0 \in D_*$ mit

$$\alpha := || \underbrace{Df(x_0)^{-1}f(x_0)}_{\delta x_0, \text{ erstes Update}} ||$$

erfülle:

$q := \frac{1}{2}\alpha\beta\gamma < 1$

Dann erzeugt die Newton-Iteration eine Folge $(x_k)_k < D_*$, welche quadratisch gegen eine Nullstelle $z \in D_*$ von f konvergiert.

Außerdem gilt: $\|x_k - z\| \leq \frac{\alpha}{1-q}q^{(2^k-1)}, \quad k \geq 1.$

Dabei wird die Existenz einer Nullstelle nicht vorausgesetzt, sondern nachgewiesen.

6 Sonstiges

6.1 Taylor-Entwicklung

1. Definition

Sei $I \subset \mathbb{R}$ offenes Intervall, $f : I \to \mathbb{R}$ glatte Funktion und $a \in I$.

Dann heißt die unendliche Reihe

$$\begin{aligned} Tf(x;a) &= \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n \\ &= f(a) + f'(a)(x-a) + \frac{f''(a)}{2}(x-a)^2 + \frac{f'''(a)}{6}(x-a)^3 + \cdots \end{aligned}$$

Taylorreihe von f mit Entwicklungsstelle a.

Man spricht vom Taylor-Polynom, wenn ein Teil der Reihe bis zu einem bestimmten Grad n verwendet wird.

2. Restglieddarstellung

Wird das Taylor-Polynom $T_n f(x;a)$ mit Grad n zur Näherung einer Funktion verwendet, so wird der Fehler bzw. das Restglied angegeben durch:

$$R_n f(x;a) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-a)^{n+1}, \quad \xi \in [a,x] \qquad \text{(Lagrange)}$$

$$R_n f(x;a) = \int_a^x \frac{(x-t)^n}{n!} f^{(n+1)}(t)dt \qquad \text{(Integralrestglied)}$$

6.2 Komplexe Zahlen

1. Komplexe Konjugation

Für eine komplexe Zahl $z = x + yi$ ist die komplex konjugierte $\overline{z} = x - yi$.

Analog nennt man eine Matrix oder einen Vektor komplex konjugiert, wenn alle seine Einträge komplex konjugiert wurden.

2. Rechenregeln

Für zwei komplexe Zahlen

$z_1 = x_1 + y_1 \cdot i$

$z_2 = x_2 + y_2 \cdot i$

gelten folgende Rechenregeln:

1. Addition und Subtraktion:

   $z_1 + z_2 = (x_1 + x_2) + (y_1 + y_2)i$

   $z_1 - z_2 = (x_1 - x_2) + (y_1 - y_2)i$

2. Multiplikation:

   $z_1 \cdot z_2 = (x_1 + y_1 i)(x_2 + y_2 i) = (x_1 x_2 - y_1 y_2) + (x_1 y_2 + x_2 y_1)i$

3. Division:

   $\frac{z_1}{z_2} = \frac{z_1}{z_2} \cdot \frac{\overline{z}_2}{\overline{z}_2}$

3. Einheitswurzeln

Eine komplexe Zahl ζ heißt n-te Einheitswurzel, wenn $\zeta^n = 1$.

Insbesondere ist die Zahl $\zeta_n = e^{\frac{2k\pi i}{n}}$, $i \leq k \leq n$ eine n-te Einheitswurzel.

6.3 Sinus und Cosinus

1. Wichtige Werte

$\text{rad}(0) = 0$	$\text{rad}(30) = \frac{\pi}{6}$	$\text{rad}(45) = \frac{\pi}{4}$	$\text{rad}(60) = \frac{\pi}{3}$	$\text{rad}(90) = \frac{\pi}{2}$
$\sin(0) = 0$	$\sin\left(\frac{\pi}{6}\right) = \frac{1}{2}$	$\sin\left(\frac{\pi}{4}\right) = \frac{1}{2}\sqrt{2}$	$\sin\left(\frac{\pi}{3}\right) = \frac{1}{2}\sqrt{3}$	$\sin\left(\frac{\pi}{2}\right) = 1$
$\cos(0) = 1$	$\cos\left(\frac{\pi}{6}\right) = \frac{1}{2}\sqrt{3}$	$\cos\left(\frac{\pi}{4}\right) = \frac{1}{2}\sqrt{2}$	$\cos\left(\frac{\pi}{3}\right) = \frac{1}{2}$	$\cos\left(\frac{\pi}{2}\right) = 0$

2. Formel von Euler-Moivre

Die Exponentialfunktion mit imaginärem Argument lässt sich mit Hilfe der trigonometrischen Funktionen ausdrücken:

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

Beispiel Einheitswurzeln:

$e^{\frac{2k\pi i}{n}} = \cos\frac{2k\pi}{n} + i\sin\frac{2k\pi}{n}$

6.4 Ableitungsregeln

1. **Produktregel**

   $f(x) = g(x) \cdot h(x) \rightarrow f'(x) = g'(x) \cdot h(x) + g(x) \cdot h'(x)$

2. **Quotientenregel**

   $f(x) = \frac{g(x)}{h(x)} \rightarrow f'(x) = \frac{h(x) \cdot g'(x) - g(x) \cdot h'(x)}{(h(x))^2}$

3. **Kettenregel**

   $f(x) = g(h(x)) \rightarrow f'(x) = g'(h(x)) \cdot h'(x)$

4. **Logarithmus**

   $f(x) = \ln(x) \rightarrow f'(x) = \frac{1}{x}$

   $f(x) = \log_a(x) \rightarrow f'(x) = \frac{1}{\ln(a) \cdot x}$

6.5 Hilfreiche Sätze und Regeln

1. **Mittelwertsatz**

   Sei $f : [a, b] \rightarrow \mathbb{R}$ eine auf $[a, b]$ stetige Funktion, die auf (a, b) stetig differenzierbar ist.

   Dann existiert mindestens ein $\xi \in (a, b)$, sodass:

   $\frac{f(b) - f(a)}{b - a} = f'(\xi)$

2. **Inverse 2×2 Matrix**

   $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \Longrightarrow \quad A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$