

Gleitpunktzahlen

$$\text{FL} = \left\{ + - \sum_{l=1}^{L_m} a_l B^{-l} B^e : e = e_{\min} + \sum_{l=0}^{L_e-1} c_l B^l, a_l, c_l \in \{0, \dots, B-1\}, a_1 \neq 0 \right\} \cup \{0\}$$

$$e_{\max} = e_{\min} + B^{L_e} - 1$$

$$\max \text{FL} = -\min \text{FL} = B^{e_{\max}}(1 - B^{-L_m})$$

$$\min \text{FL}_+ = B^{e_{\min}-1}$$

$$B^{-1} \leq |m| < 1$$

Relative Maschinengenauigkeit

$$\frac{|x - \text{fl}(x)|}{|x|} \leq \frac{B^{1-L_m}}{2} =: \text{eps}$$

$$\text{fl}(x) = x(1 + \varepsilon) \quad \text{mit } |\varepsilon| \leq \text{eps}$$

$$x \hat{\circ} y := \text{fl}(x \circ y) = (x \circ y)(1 + \varepsilon) \quad \text{mit } |\varepsilon| \leq \text{eps}$$

Stabilität eines Algorithmus: Fehlerverstärkung des Verfahrens ist “moderat” groß (im vergleich zum unvermeidbaren Fehler).

Cholesky-Zerlegung

Ist eine $A \in \mathbb{R}^{N \times N}$ symmetrisch und positiv definit, existiert eine Cholesky-Zerlegung $A = LL^T$

1. Berechne Cholesky-Zerlegung $A = LL^T$
2. Löse $Ly = b$ durch Vorwärtssubstitution
3. Löse $L^T x = y$ durch Rückwärtssubstitution

$l_{nn} > 0$ macht die Zerlegung eindeutig.

$$Q(\lambda m) = \lambda Qm = -\lambda w \quad \text{für alle } \lambda \in \mathbb{R}$$

$$Qy = y \quad \text{für alle } y \in \mathbb{R}^N \quad \text{mit } w^T y = 0$$

$$\text{Spiegelebene } E = \{y \in \mathbb{R}^M : w^T y = 0\}$$

LR-Zerlegung

Ist jeder Hauptminor von $A \in \mathbb{R}^{N \times N}$ regulär, existiert eine LR-Zerlegung $A = LR$. Ist $A \in \mathbb{R}^{N \times N}$ regulär existiert eine LR-Zerlegung mit Spaltenpivotwahl $PA = LR$.

1. Berechne Zerlegung $PA = LR$ durch Gauß-Elimination
2. Löse $Ly = Pb$ durch Vorwärtssubstitution in $\frac{1}{2}N^2$ Ops
3. Löse $Rx = y$ durch Rückwärtssubstitution $x_n = \frac{y_n - \sum_{j=n+1}^N r_{nj}x_j}{r_{nn}}$ in $O(\frac{1}{2}N^2)$ Ops

$l_{nn} = 1$ liefert Eindeutigkeit der Zerlegung.

Kondition und Normen

Problem: Wie wirken sich Störungen der Eingabegrößen (hier A und b) auf die Lösung unabhängig vom gewählten Algorithmus aus?

Zugehörige Matrixnorm

$$\|A\| := \sup_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \sup_{x \neq 0} \left\| A \frac{x}{\|x\|} \right\| = \sup_{\|x\|=1} \|Ax\| = \max_{\|x\|=1} \|Ax\|$$

$$\|A^{-1}\| = \sup_{x \neq 0} \frac{\|A^{-1}x\|}{\|x\|} \stackrel{A^{-1}x=1}{=} \sup_{z \neq 0} \frac{\|z\|}{\|Az\|} = \max_{\|z\|=1} \frac{1}{\|Az\|} = \frac{1}{\min_{\|z\|=1} \|Az\|}$$

Spaltensummennorm	$\ A\ _1 = \max_{m=1,\dots,N} \sum_{n=1}^N a_{nm} $	$\ x\ _1 = \sum_{n=1}^N x_n $	1-Norm
Zeilensummennorm	$\ A\ _\infty = \max_{n=1,\dots,N} \sum_{m=1}^N a_{nm} $	$\ x\ _\infty = \max_{n=1,\dots,N} x_n $	Maximumsnorm
Spektralnrm	$\ A\ _2 = \sqrt{\text{größter EW von } A^T A}$	$\ x\ _2 = \sqrt{\sum_{n=1}^N x_n^2}$	euklidische Norm

Eigenschaften: positive Definitheit $\|x\| \geq 0$, Dreiecksungleichung $\|x + y\| \leq \|x\| + \|y\|$, Homogenität $\|\lambda x\| = |\lambda| \|x\|$
 $\|Ax\| \leq \|A\| \|x\|$, $\|I_N\| = 1$, Submultiplikativität $\|AB\| \leq \|A\| \|B\|$, $\|x\| = \sqrt{\langle x, x \rangle}$

Störungen

$$x - \tilde{x} = A^{-1}b - A^{-1}\tilde{b} = A^{-1}(b - \tilde{b})$$

$$\Leftrightarrow \|x - \tilde{x}\| = \|A^{-1}(b - \tilde{b})\| \leq \|A^{-1}\| \|b - \tilde{b}\| \quad (\text{absolute Abweichung})$$

$$\Leftrightarrow \frac{\|x - \tilde{x}\|}{\|x\|} = \frac{\|A^{-1}(b - \tilde{b})\|}{\|x\|} \leq \frac{\|b\| \|A^{-1}\| \|b - \tilde{b}\|}{\|x\| \|b\|} \quad (\text{relative Abweichung})$$

$$\Leftrightarrow \frac{\|x - \tilde{x}\|}{\|x\|} \stackrel{\|b\|=\|Ax\| \leq \|A\| \|x\|}{\leq} \underbrace{\|A\| \|A^{-1}\|}_{=: \text{cond}(A)} \frac{\|b - \tilde{b}\|}{\|b\|}$$

Konditionszahl einer Matrix

$$\boxed{\text{cond}(A) := \|A\| \|A^{-1}\|} \stackrel{\text{Submultiplikativit}}{\geq} \|AA^{-1}\| = \|I_N\| = 1$$

$$\text{cond}_2(A) = \frac{\max\{|\lambda| : \lambda \text{ EW von } A\}}{\max\{|\lambda| : \lambda \text{ EW von } A\}}, \text{cond}(A) = \frac{\max_{\|y\|=1} \|Ay\|}{\min_{\|z\|=1} \|Az\|}, \text{cond}(A) = \text{cond}(\alpha A), \alpha \in \mathbb{R} \setminus \{0\}$$

Seien relative Abweichung beschränkt

$$\frac{\|A - \tilde{A}\|}{\|A\|} \leq \varepsilon_A, \quad \frac{\|b - \tilde{b}\|}{\|b\|} \leq \varepsilon_b$$

Dann gilt im Fall $\varepsilon_A \text{ cond}(A) < 1$ die Abschätzung

$$\frac{\|x - \tilde{x}\|}{\|x\|} \leq \frac{\text{cond}(A)}{1 - \varepsilon_A \text{ cond}(A)} (\varepsilon_A + \varepsilon_b)$$

$\text{cond}(\alpha I_N) = 1$, für orthogonale Matrizen Q: $\|Qx\|_2^2 = x^T Q^T T x = x^T x = \|x\|_2^2$, $\|Q\|_2 = 1 \Rightarrow \text{cond}_2(Q) = 10 = \|Q\| \|Q^{-1}\| = \|Q\|^2$

QR-Zerlegung

Geg. $A \in \mathbb{R}^{M \times N}$, $M \geq N$, voller Rang ($= N$)

Ges. orthogonales $Q \in \mathbb{R}^{M \times N}$ und $R = \begin{pmatrix} \bar{R} \\ 0 \end{pmatrix} \in \mathbb{R}^{M \times N}$, rechte obere Dreiecksmatrix $\bar{R} \in \mathbb{R}^{N \times N}$

Householder-Transformation

Householdermatrix Q ist symmetrisch, orthogonal und eine Spiegelung eines Vektors v auf ein Vielfaches des Vektors $u (= e^1)$ mit Spiegelachse $\perp w$ (Householdervektor).

$$Q = I_N - 2ww^T \Rightarrow Qv = \sigma e^1 = -\text{sign}(v_1)\|v\|_2 e^1 = v - 2w^T v w \quad (2M \text{ Ops})$$

Q wird eigl. nie ausgerechnet, sondern nur w gespeichert und QA wird berechnet aus $(Qa^1, Qa^2, \dots)$
Spaltenvektoren von A . ($2M(N-1)$) Ops

$$(\text{Householdervektor}) \quad w = \frac{v - u}{\|v - u\|_2} = \frac{v - \sigma e^1}{\|v - \sigma e^1\|_2} \quad \sigma = -\text{sign}(v_1)\|v\|_2 = \begin{cases} -\|v\|_2, & \text{falls } v_1 > 0 \\ \|v\|_2, & \text{falls } v_1 \leq 0 \end{cases}$$

Berechne Householder-Transformation des ersten Spaltenvektors a^1 von $A^{(n)}$

$$w = \frac{a^1 - \sigma e^1}{\|a^1 - \sigma e^1\|_2}, \quad Qa^1 = r^1 = \sigma e^1 = -\text{sign}(a^1)\|a^1\|_2 e^1, \quad Q^{(n)}A^{(n)} = \begin{pmatrix} r_{11} & r_{12}^T \\ & A^{(n+1)} \end{pmatrix} = \begin{pmatrix} \sigma & r_{12}^T \\ 0 & A^{(n+1)} \end{pmatrix}$$

Durch die Householder-Transformation bekommen wir die Matrix für die nächste Iteration.

Householdervektor $\tilde{w}^{(n+1)}$ ist eine Dimension kleiner. Um diesen an das vorherige Ergebnis zu multiplizieren $w^{(n+1)} = \begin{pmatrix} 0 \\ \tilde{w}^{(n+1)} \end{pmatrix}$ sodass $Q^{(n+1)} = \begin{pmatrix} 1 & 0 \\ 0 & \tilde{Q}^{(n+1)} \end{pmatrix}$ (alternativ: es wird auf e^n -Vektor gespiegelt)

Insgesamt: in $MN^2 - \frac{1}{3}N^3$ Ops $Q_N \dots Q_2 Q_1 A =: R \quad Q_1 \dots Q_N =: Q$

Lineares Ausgleichsproblem $\|Ax - b\|_2 = \min!$

Lösen von LGS $Ax = b$ (falls $M = N$)

Berechne $A = QR$

$$Ax = b$$

Berechne $Q^T b = \begin{pmatrix} c \\ d \end{pmatrix}$

$$\Leftrightarrow QRx = b$$

Löse $\bar{R}x = c$ (Rückwärtssubstitution)

$$\Leftrightarrow Rx = Q^T b$$

QR-Algorithmus (Eigenwertberechnung)

$$\Leftrightarrow Rx = c \quad (\text{Rückwärtssubstitution})$$

1. setze $A_0 = A$ und $k = 0$

2. zerlege $A_k = Q_k R_k$

3. berechne $A_{k+1} = R_k Q_k$, erhöhe k um 1 und gehe zu 2

$O(N^3)$ Ops pro Iteration

$$k \rightarrow \infty : A_k \rightarrow R = \begin{pmatrix} \lambda_1 & * & * & * \\ & \lambda_2 & * & * \\ & & \ddots & * \\ 0 & & & \lambda_N \end{pmatrix}, \quad Q_k \rightarrow I_N$$

$$\text{Konvergenz: } \frac{|\lambda_N|}{|\lambda_{N-1}|}$$

Benutze **Hessenbergform** für weniger Ops pro Iteration $O(N^2)$. Kann durch $N-2$ Householder-Transformation berechnet werden $Q^T A Q = H$

Householder-Transformation $\tilde{Q}_2 = I_{N-1} - 2\tilde{w}^2(\tilde{w}^2)^T \in \mathbb{R}^{(N-1) \times (N-1)}$ mit $\tilde{Q}_2 \begin{pmatrix} a_{21} \\ a_{31} \\ \vdots \\ a_{N1} \end{pmatrix} = \begin{pmatrix} * \\ 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^{N-1}$, $Q_2 = I_N -$

$$2ww^T = \begin{pmatrix} 1 & \\ & \tilde{Q}_2 \end{pmatrix}, \quad w^2 = \begin{pmatrix} 0 \\ \tilde{w}^2 \end{pmatrix}$$

$$A^{(1)} = Q_2 A Q_2$$

$$Q_{N-1} A^{(N-3)} Q_{N-1} = Q_{N-1} \dots Q_2 A Q_2 \dots Q_{N-1} = Q^T A Q = H, \quad Q = Q_2 \dots Q_{N-1}$$

Wenn wir eine Hessenbergmatrix QR-Zerlegung $H = QR$, dann ist die Matrix $\bar{H} = RQ$ ebenfalls Hessenbergmatrix, wegen $\bar{H} = RQ = Q^T Q R Q = Q^T H Q$

QR-Algorithmus mit Shift $\mu_k = h_{NN}^{(k)}$ für schnellere Konvergenz $|\tilde{h}_{n+1,n}^{(k)}| = O\left(\left|\frac{\lambda_{n+1} - \mu}{\lambda_n - \mu}\right|\right)$

1. setze $H_0 = H$, $k = 0$ und wähle Toleranz ε

2. zerlege $H_k - h_{NN}^{(k')} I_N = Q_k R_k$

3. berechne $H_{k+1} = R_k Q_k + h_{NN}^{(k)} I_N$

4. ist $|h_{N,N-1}^{(k+1)}| \leq \varepsilon \left(|h_{NN}^{(k+1)}| + |h_{N-1,N-1}^{(k+1)}| \right)$ so akzeptiere $h_{NN}^{(k+1)}$ als EW. Sonst erhöhe k um 1 und gehe zu 2.

Newton-Verfahren

Problem: Für eine vorgegebene (nichtlineare) Funktion $f : D \subset \mathbb{R}^N \rightarrow \mathbb{R}^N$ finde $x^* \in D$ mit $f(x^*) = 0_N \in \mathbb{R}^N$

Sei x^0 in der Nähe der Lösung x^* . Eine Taylor-Entwicklung liefert

$$\begin{aligned}
 0 &= f(x^*) = f(x^0) + f'(x^0)(x^* - x^0) + \overbrace{O(\|x^0 - x^*\|^2)}^{\text{vernachlässigen} \approx 0} \\
 &\leadsto 0 = f(x^0) + f'(x^0)(x^* - x^0) \\
 &= f(x^0) + f'(x^0)x^* - f'(x^0)x^0 \\
 &\Leftrightarrow f'(x^0)x^0 - f(x^0) = f'(x^0)x^* \\
 &\Leftrightarrow f'(x^0)^{-1}(f'(x^0)x^0 - f(x^0)) = x^* \\
 &\Leftrightarrow x^* \approx x^0 - f'(x^0)^{-1}f(x^0) \\
 &\Rightarrow x^{k+1} = x^k - f'(x^k)^{-1}f(x^k) \\
 &\quad \text{Berechnung der inversen Jacobi-Matrix vermeiden} \\
 &\Leftrightarrow x^{k+1} = -f'(x^k)^{-1}f(x^k) \\
 &\Leftrightarrow f'(x^k) \underbrace{(x^{k+1} - x^k)}_{=d^k} = -f(x^k)
 \end{aligned}$$

1. Wähle Startwert $x^0 \in D$ und Toleranz ε . Setze $k = 0$
2. Löse $f'(x^k)d^k = -f(x^k)$ (LGS durch LR-Zerlegung)
 $x^{k+1} = x^k + d^k$
3. Falls $(\|d^k\| < \varepsilon)$: STOP
 Sonst erhöhe k um 1 und gehe zu 2.

Bem.: Man wählt nicht $\|f(x^k)\| < \varepsilon$ als STOP-Kriterium, weil das Problem $Af(x) = 0$ (A regulär) dann eine andere Lösung hätte. $Af(x) = 0$ ändert Iterierten nicht, deshalb ist $\|d^k\| < \varepsilon$ besser, weil so bei $Af(x) = 0$ und $f(x) = 0$ die Lösung und Iterierten gleich bleiben.

lokale quadratische Konvergenz

Es gibt eine Kugel $\kappa := \kappa_p(x^*) = \{x \in \mathbb{R}^N : \|x - x^*\| \leq p\} \subset D$ sodass x^* die einzige Nullstelle von f in κ ist. Zudem liegen die Folgenglieder $x^{k+1} = x^k - f'(x^k)^{-1}f(x^k)$ für jeden Startwert $x^0 \in \kappa$ ebenfalls in κ und es gilt $\lim_{k \rightarrow \infty} x^k = x^*$. Weiter existiert eine Konstante $c > 0$ mit $\|x^* - x^k\| \leq c\|x^* - x^{k-1}\|^2$ für $k \in \mathbb{N}$

Homotopie-Verfahren: guten Startwert suchen

$g : \mathbb{R}^N \rightarrow \mathbb{R}^N$ eine (einfache) Funktion mit bekannter Nullstelle x^0 . Definiere $f_s(x) = (1-s)g(x) + sf(x)$, wählen $0 < s_1 < s_2 < \dots < s_M = 1$ und lösen $f_{s_n}(x) = 0$ mit Newton mit Startvektor x^{m-1} mit $f_{s_{m-1}}(x^{m-1}) = 0$ für $m = 1, \dots, M$

Dämpfungsstrategie: Verbesserung der Konvergenz

Bestimme $t_k \in (0, 1]$ sodass für $x^{k+1} = x^k - t_k f'(x^k)^{-1}f(x^k)$ das Residuum kleiner wird $\|f(x^{k+1})\| < \|f(x^k)\|$

Differenzenquotient

Wenn $f'(x^k)$ aufwendig zu berechnen, wähle $A_k \approx f'(x^k)$ durch $(A_k)_{nm} = \frac{1}{h}(f_n(x^k + h^m) - f_n(x^k))$

Vereinfachtes Newton-Verfahren

pro Iteration f' auswerten und LR-Zerlegen ist teuer. Stattdessen $A \approx f'(x^0)$ also einmal f' auswerten und LR-Zerlegen. $F(x) = x - A^{-1}f(x)$ Nurnoch lineare Konvergenz.

Globale Minimierung

Wir suchen $x^* \in \mathbb{R}^N$ mit $f(x^*) \leq f(x)$ für alle $x \in \mathbb{R}^N$. dh insbesondere grad $f(x^*) = 0_N$. Wir berechnen Nullstelle von $\nabla f(x) = (\text{grad } f(x))^T$ mit Newton $x^{k+1} = x^k - H_f(x^k)^{-1}\nabla f(x^k)$. H_f Hessematrix.

Nichtlineare Ausgleichsrechnung $\|f(x)\|_2 = \min!$

Wähle Startwert $x^0 \in \mathbb{R}^N$ für $k = 0, 1, \dots$

1. löse lineares Ausgleichsproblem $\|f'(x^k)d^k + f(x^k)\|_2 = \min$
2. setze $x^{k+1} = x^k + d^k$

Interpolation

Problem: Suche zu den Stützstellen $(x_0, f_0), \dots, (x_N, f_N)$ eine Funktion p mit $p(x_n) = f_n$ für alle $n = 0, \dots, N$

Polynom-Interpolation

Problem: Suche zu $N+1$ Stützstellen $(x_0, f_0), \dots, (x_N, f_N)$ ein Polynom p vom Grad $\leq N$ ($p \in \mathbb{P}_N$), welches das Interpolationsproblem (s.o.) löst: $p(x_n) = f_n$ für alle $n = 0, \dots, N$,

Eindeutigkeit des Interpolationspolynoms zu einem Interpolationsproblem:

Nehmen wir an es gäbe zwei Polynome $p, q \in \mathbb{P}_N$ vom Grad $\leq N$ welche das gleiche Interpolationsproblem lösen. $p(x_n) = q(x_n) = f_n$ für $n = 0, \dots, N$. Dann wäre $p - q \in \mathbb{P}_N$ ein Polynom vom Grad $\leq N$, welches $N + 1$ Nullstellen an den Stützstellen x_n hat: $p(x_n) - q(x_n) = f_n - f_n = 0$ für alle $n = 0, \dots, N$.

Nach dem Fundamentalsatz der Algebra kann ein Polynom vom Grad $= N$ nur maximal N Nullstellen haben außer es ist das konstante 0-Polynom $f = 0$.

Damit das Polynom $p - q$ vom Grad $\leq N$ also $N + 1$ Nullstellen haben kann, muss nach dem Fundamentalsatz der Algebra $p - q = 0 \Leftrightarrow p = q$ das konstante 0-Polynom sein, was bedeutet, dass p und q identisch sein müssen.

Lagrange-Basis

Das n -te Lagrange-Polynom L_n zu den Stützstellen $x_0, \dots, x_N$ ist definiert als:

$$p(x) = \sum_{n=0}^N f_n L_n(x) \quad , \quad L_n(x) = \prod_{m=0, m \neq n}^N \frac{x - x_m}{x_n - x_m} = \sigma_{nm} = \begin{cases} 1, & \text{falls } x_m = x_n \\ 0, & \text{falls } x_m \neq x_n \end{cases}$$

Lagrange-Polynome L_n bilden aufgrund von Skalarprodukt $\langle p, q \rangle = \sum_{n=0}^N p(x_n)q(x_n)$ eine orthonormalbasis und Linearkombination ($\mathbb{P}_N$).

Kondition

Frage: Wie wirken sich Störungen der Eingabegrößen (Stützwerte f_n) auf die Lösung der Interpolation aus?

$$p(x) = \sum_{n=0}^N f_n L_n(x) \quad \tilde{p}(x) = \sum_{n=0}^N \tilde{f}_n L_n(x)$$

$$|p(x) - \tilde{p}(x)| = \left| \sum_{n=0}^N (f_n - \tilde{f}_n) L_n(x) \right| \leq \sum_{n=0}^N |f_n - \tilde{f}_n| |L_n(x)| \leq \max_{n=0, \dots, N} |f_n - \tilde{f}_n| \sum_{n=0}^N |L_n(x)|$$

$$\Leftrightarrow \max_{x \in [a, b]} |p(x) - \tilde{p}(x)| \leq \underbrace{\max_{x \in [a, b]} \sum_{n=0}^N |L_n(x)|}_{=: \Lambda_N \geq 1} \max_{n=0, \dots, N} |f_n - \tilde{f}_n|$$

Lebesgue-Konstante Λ_N ist invariant unter Affinen Transformationen daher nur von relativer Lage der Stützstellen x_n zueinander abhängig.

	Leitkoeffizient 	
Newton-Darstellung	$p(x) = a_0 + a_1 w_1(x) + \dots + a_N w_N(x)$	$w_n(x) = \prod_{m=0}^{n-1} (x - x_m)$
Aus Darstellung folgt	$p(x) = p_{0, N-1}(x) + a_N w_N(x)$	$p_{n, n}(x) = f_n = f_{n, n}$
	$= p_{0, N}(x) = f_{0, 0} + f_{0, 1} w_1(x) + \dots + f_{0, N} w_N(x)$	mit $a_n = f_{0, n}$

Lemma von Aitken	$p_{n, k}(x) = \frac{(x_n - x)p_{n+1, k}(x) - (x_k - x)p_{n, k-1}}{x_n - x_k}$	$f_{n, k-1} \searrow$	$2(N-1)$ Additionen $N-1$ Divisionen
$\Rightarrow$ Für Koeffizienten	$f_{n, k} = \frac{f_{n, k-1} - f_{n+1, k}}{x_n - x_k}, \quad (\text{Dividierte Differenzen})$	$f_{n+1, k} \rightarrow f_{n, k}$	

Interpolationsfehler $f(x) - p(x) = w_{N+1}(x) \frac{f^{N+1}(\xi)}{(N+1)!}, \quad \xi = \xi(x) \in (a, b) \text{ Zwischenstelle}$

Maximaler Interpolationsfehler/Approximationsgüte $\max_{x \in [a, b]} |f(x) - p(x)| \leq \max_{x \in [a, b]} |w_{N+1}(x)| \max_{x \in [a, b]} \frac{|f^{N+1}(x)|}{(N+1)!}$

Tschebyscheff-Interpolation

Tschebyscheff-Interpolationsformel

$$p(x) = \frac{1}{2}c_0 + c_1T_1(x) + \dots + c_NT_N(x)$$

mit Koeffizienten $c_m = \frac{2}{N+1} \sum_{n=0}^N f_n \cos\left(m \frac{2n+1}{2N+2}\pi\right)$ mit $(N+1)^2$ Multiplikationen

Tschebyscheff-Polynom vom Grad N ist auf $[-1, 1]$ gegeben durch

$$T_0(x) = 1$$

$$T_1(x) = x \quad \text{oder} \quad T_N(x) = \cos(N \arccos(x))$$

$$T_{N+1}(x) = 2 \cdot T_N(x) - T_{N-1}(x)$$

Tschebyscheff-Stützstellen: $x_n = \cos\left(\frac{2n+1}{2N+2}\pi\right)$ für alle $n = 0, \dots, N$

Min-Max-Eigenschaft: $\max_{x \in [-1, 1]} |w_{N+1}(x)|$ wird genau für Tschebyscheff-Stützstellen minimal mit 2^{-N}

mit FFT lässt sich Koeffizient c_m in $O(N \log N)$ Multiplikationen berechnen, wenn $N+1 = 2^k$ eine 2-er Potenz ist.

Clenshaw-Algorithmus zur Auswertung $p(x)$ mit $N+2$ Multiplikationen und $2N$ Additionen

$$d_{N+2} = d_{N+1} = 0$$

$$d_n = c_n + 2x \cdot d_{n+1} - d_{n+2} \quad \text{für } n = N, N-1, \dots, 0$$

$$\text{Dann gilt } p(x) = \frac{1}{2}(d_0 - d_2)$$

Zusammenfassung	Bestimmung	Auswertung
Monom	LGS mit $\frac{1}{3}N^3$ Ops	Horner-Schema mit N Ops
Lagrange	direkt gegeben	zu aufwendig
Newton	Dividierte Differenzen mit N^2 Ops	Horner-Schema mit $2N$ Ops
Tschebyscheff	FFT mit $O(N \log N)$ Ops	Clenshaw-Algorithmus mit $2N$ Ops

nur in Newton-Darstellung können zusätzliche Stützstellen hinzugefügt werden, wobei nicht alle Koeffizienten neu berechnet werden müssen.

Spline-Interpolation

Problem: Suche zeimal stetig differenzierbare Funktion $s \in C^2([a, b], \mathbb{R})$ mit $s(x_n) = y_n$ für $n = 0, \dots, N$, sodass folgendes Integral minimal ist:

$$\int_a^b |s''(x)|^2 dx \leq \int_a^b |s''(x) + \varepsilon h''(x)|^2 dx$$

$$\Leftrightarrow \int_a^b s''(x)h''(x)dx = \sum_{n=1}^N \int_{x_{n-1}}^{x_n} s''(x)h''(x)dx = s''(b)h'(b) - s''(a)h'(a) \stackrel{!}{=} 0$$

für alle $\varepsilon \in \mathbb{R}$ und C^2 -Funktionen $h : [a, b] \rightarrow \mathbb{R}$ mit $h(x_n) = 0$ für alle $n = 0, \dots, N$

Aus dieser Gleichung ergeben sich drei Lösungen und damit drei Typen von Splines:

$$s'(a) = v_0 \quad \text{und} \quad s'(b) = v_N \quad \text{eingespannt / hermitisch}$$

$$s''(a) = 0 = s''(b) \quad \text{natürlich}$$

$$s'(a) = s'(b) \quad \text{und} \quad s''(a) = s''(b) \quad \text{periodisch (sinnvoll für } \overbrace{s(a) = s(b)}^{y_0=y_N})$$

not-a-knot-Bedingung falls Ableitung an Randpunkten unbekannt: $s_1'''(x_1) = s_2'''(x_1)$ und $s_{N-1}''' = s_N'''(x_{N-1})$

kubischer interpolierender Spline

Eine C^2 -Funktion $s : [a, b] \rightarrow \mathbb{R}$ heißt kubischer interpolierender Spline zu den Stützpunkten $(x_0, y_0), \dots, (x_N, y_N)$ mit der Eigenschaft $a = x_0 < \dots < x_N = b$, falls

1. $s|_{[x_{n-1}, x_n]} \in \mathbb{P}_3$ für alle $n = 1, \dots, N$
2. $s(x_n) = y_n$ für alle $n = 0, \dots, N$

$$s|_{[x_{n-1}, x_n]} = s_n(x) = \underbrace{y_{n-1} + (x - x_{n-1}) \overbrace{y_{n-1,n}^{= \frac{y_{n-1} - y_n}{x_{n-1} - x_n}}}}_{\text{interpolierender Anteil}} + \underbrace{(x - x_{n-1})(x - x_n)[\alpha_n(x - x_{n-1}) + \beta_n(x - x_n)]}_{\text{glättender Anteil}}$$

Daraus erbeben sich $2N$ Unbekannte α_n, β_n . Diese erhält man durch die $2N - 2$ Glattheitsbedingungen + 2 Randbedingungen (s.o.). Also erhält man α_n, β_n aus LGS mit dim $2N$.

$$\left. \begin{aligned} s'_n(x_n) &= s'_{n+1}(x_n) \\ s''_n(x_n) &= s''_{n+1}(x_n) \end{aligned} \right\} \quad \text{für } n = 1, \dots, N-1 \Rightarrow 2N-2 \text{ Bedingungen}$$

Die Berechnung der Unbekannten können wir vereinfachen, indem wir die α_n, β_n durch $N+1$ Unbekannte Momente γ_n darstellen:

$$\left. \begin{aligned} s''_n(x_{n-1}) &= \gamma_{n-1} \\ s''_n(x_n) &= \gamma_n \end{aligned} \right\} \quad \text{damit } s''_n(x_n) = s''_{n+1}(x_n) = \gamma_n$$

Dies sind $2N$ Gleichungen für die $N+1$ Unbekannten $\gamma_0, \dots, \gamma_N$. Durch den Ansatz sind $N-1$ Glattheitsbedingungen an die zweite Ableitung automatisch erfüllt.

Mit den Gitterweiten $h_n := x_n - x_{n-1}$ erhalten wir $\alpha_n = \frac{1}{6h_n}(\gamma_{n-1} + 2\gamma_n) \quad \beta_n = -\frac{1}{6h_n}(2\gamma_{n-1} + \gamma_n)$

Eingesetzt in s' in der Glattheitsbedingung $s'_n(x_n) = s'_{n+1}(x_n)$ folgt

$$\frac{h_n}{6}(\gamma_{n-1} + 2\gamma_n) + \frac{h_{n+1}}{6}(2\gamma_n + \gamma_{n+1}) = \gamma_{n,n+1} - \gamma_{n-1,n} =: d$$

mit Einschränkung auf eingespannte kubische Splines $s'_1(x_0) = 0_0, s'_N(x_N) = v_N$ erhalten wir außerdem

$$\frac{h_1}{6}(2\gamma_0 + \gamma_1) = y_{0,1} - v_0 =: d_0 \quad \frac{h_N}{6}(\gamma_{N-1} + 2\gamma_N) = v_N - y_{N-1,N} =: d_N$$

Insgesamt ein LGS:
$$\frac{1}{6} \underbrace{\begin{pmatrix} 2h_1 & h_1 & & & \\ h_1 & 2(h_1 + h_2) & h_2 & & \\ & \ddots & \ddots & \ddots & \\ & & h_{N-1} & 2(h_{N-1} + h_N) & h_N \\ & & & h_N & 2h_N \end{pmatrix}}_{\text{strikt diagonaldominante symmetrische Tridiagonalmatrix (spd, regulär)}} \begin{pmatrix} \gamma_0 \\ \gamma_1 \\ \vdots \\ \gamma_{N-1} \\ \gamma_N \end{pmatrix} = \begin{pmatrix} d_0 \\ d_1 \\ \vdots \\ d_{N-1} \\ d_N \end{pmatrix}$$

$\Rightarrow$ mit Cholesky in $O(N)$ Ops lösbar.

Kondition

$\tilde{s}(x)$ der Spline zu gestörten Daten $(x_n, \tilde{y}_n)$, Lagrange-Spline $l_n(x_m) = \begin{cases} 1, & \text{falls } n=m \\ 0, & \text{falls } n \neq m \end{cases}, \quad l'_n(a) = 0, l'_n(b) = 0$

$$|s(x) - \tilde{s}(x)| \leq \sum_{n=0}^N |y_n - \tilde{y}_n| \|l_n(x)\| \leq \sum_{n=0}^N |l_n(x)| \max_{m=0, \dots, N} |y_m - \tilde{y}_m| \leq \Lambda_N \max_{n=0, \dots, N} |x_m - \tilde{y}_m|$$

$$\text{mit } \Lambda_N := \max_{x \in [a, b]} \sum_{n=0}^N |l_n(x)| \quad \text{also} \quad \max_{x \in [a, b]} |s(x) - \tilde{s}(x)| \leq \Lambda_N \max_{m=0, \dots, N} |y_m - \tilde{y}_m|$$

Bei äquidisanten Unterteilung gilt $\Lambda_N \leq 2$ für alle $N \in \mathbb{N}$

Numerische Integration

Problem: Berechne für eine gegebene Funktion $f : [a, b] \rightarrow \mathbb{R}$ das Riemann-Integral.

Eigenschaften von Integralen

Additiv: $\int_a^b f(x)dx = \int_a^c f(x)dx + \int_c^b f(x)dx$

Linear: $I(\lambda f + \mu g) = \lambda I(f) + \mu I(g)$

Monoton: falls $f \geq g$ auf $[a, b]$ dann $\int_a^b f(x)dx \geq \int_a^b g(x)dx$

$$|I(f)| \leq I(|f|) \Leftrightarrow \text{cond}_1(f) = \frac{I(|f|)}{|I(f)|} \geq 1$$

Norm: $\|f\|_1 = I(|f|)$

Quadraturformeln

$$\int_a^b f(x)dx \approx \int_a^b p(x)dx = \int_a^b \sum_{n=0}^N f(x_n) L_n(x) = \sum_{n=0}^N \overbrace{\int_a^b L_n(x)dx}^{=(b-a)b_{n+1}} f(x_n) = (b-a) \sum_{k=1}^s b_k f(a + c_k(b-a))$$

Rechteckregel	$s = 1, b_1 = 1, c_1 = 0, p = 1$	$I(f) \approx (b-a)f(a)$
Mittelpunktregel	$s = 1, b_1 = 1, c_1 = \frac{1}{2}, p = 2$	$I(f) \approx (b-a)f(\frac{a+b}{2})$
Trapezregel	$s = 2, b_1 = b_2 = \frac{1}{2}, c_1 = 0, c_2 = 1, p = 2$	$I(f) \approx (b-a) \frac{f(a)+f(b)}{2}$
Simpsonregel	$s = 3, b_1 = b_3 = \frac{1}{6}, b_2 = \frac{4}{6}, c_1 = 0, c_2 = \frac{1}{2}, c_3 = 1, p = 4$	$I(f) \approx (b-a) \frac{1}{6} (f(a) + 4f(\frac{a+b}{2}) + f(b))$

Eine QF $(b_k, c_k)_{k=1, \dots, s}$ besitzt die Ordnung p , falls sie für alle Polynome vom Grad $\leq p-1$ das Integral exakt berechnet, wobei p maximal ist.

TODO: BEWEIS ZU ORDNUNGSBEDINGUNGEN!!!!

Ordnungsbedingungen: Eine QF besitzt genau dann die Ordnung p , falls

$$\frac{1}{q} = \sum_{k=1}^s b_k c_k^{q-1} \text{ für alle } q = 1, \dots, p \text{ aber nicht mehr für } q = p+1 \text{ gilt}$$

Für eine QF mit vorgegebenen Knoten $c_1 < \dots < c_s$ können die Gewichte genau dann mit $b_k = \int_0^1 \underbrace{L_k(x)}_{\prod_{j=1, j \neq k}^s \frac{x-c_j}{c_k-c_j}} dx$ eindeutig bestimmt werden wenn $p \geq s$.

Symmetrische QF:

$$b_k = b_{s+1-k}, \quad c_k = 1 - c_{s+1-k} \text{ also symmetrisch um } \frac{1}{2} \text{ verteilt}$$

Die Ordnung p einer symmetrischen QF ist gerade. **TODO: Beweis**

Für symmetrische Knoten c_k und $p \geq s$ sind die eindeutigen Gewichte b_k ebenfalls symmetrisch.

QF mit erhöhter Ordnung

Die maximale Ordnung einer QF ist $2s$ (insb. bei Gauß-QF)

Bew.: $\langle M, M \rangle = \int_0^1 M(x)^2 dx > 0, \deg(M) = s$

Gauß-QF: Ordnung $2s$ gegeben durch $c_k = \frac{1}{2}(1 + \gamma_k)$ mit $\gamma_1, \dots, \gamma_s$ die Nullstellen des Legendre-Polynoms vom Grad s . Die Gewichte b_k einer Gauß-QF sind positiv. Gauß-QF sind symmetrisch, weil Legendre-Nullstellen $\gamma_1, \dots, \gamma_s$ symmetrisch zum Punkt 0 im Intervall $(-1, 1)$ und so Knoten symmetrisch zu Punkt $\frac{1}{2}$ im Intervall $(0, 1)$.

Summierte QF: Um Fehler zu verkleinern zerlege $[a, b]$ in Teilintervalle $[x_{n-1}, x_n], n = 1, \dots, N$ mit Intervalllängen $x_n - x_{n-1} = h_n$

$$\int_a^b f(x)dx \approx \sum_{n=1}^N h_n \sum_{k=1}^s b_k (x_{n-1} + c_k h_n)$$