Numerische Mathematik I

Manfred Dobrowolski*

17. Oktober 2014

Inhaltsverzeichnis

*Institut für Mathematik, Universität Würzburg, Am Hubland, 97047 Würzburg

1 Grundlagen

1.1 Gleitpunktarithmetik und Rundungsfehler Eine *Gleitpunktzahl*, auch *Gleitkommazahl* genannt, im Dezimalsystem ist von der Form

$$x = 0.d_1 d_2 \dots d_t \times 10^k, \quad d_i \in \{0, 1, \dots, 9\}, \quad (= t \text{ gültige Stellen})$$

mit

$$d_1 \neq 0 \quad \text{für } x \neq 0 \quad \text{und} \quad -m \leq k \leq m.$$

Es gibt also nur endlich viele Gleitkommazahlen! Da wir uns um Exponentenunter- und überlauf nicht kümmern wollen, setzen wir im Folgenden $m = \infty$ voraus. Es sei

$$M = \text{ Menge der Gleitpunktzahlen.}$$

Die *Maschinengenauigkeit* `eps` ist die kleinste positive Zahl in M mit

$$1. + \texttt{eps} > 1. \ == \ \texttt{true}.$$

In die Maschinengenauigkeit geht neben der Zahl der gültigen Stelle auch die Art der Rundung (normales Auf- und Abrunden bzw. Abschneiden) ein. Da das Rechenwerk meist sehr viel genauer arbeitet als die Speicherung, wird die Maschinengenauigkeit in der Regel nur vom verwendeten Speicherplatz und der Art der Rundung der Gleitpunktzahl abhängen. Wir verwenden daher folgendes Programm für die Bestimmung von `eps`:

```
  program
  eps=1
1 eps=0.99*eps
  if(masch(1.+eps)==1) goto 1
  write eps
  end
```

Das aufgerufene Funktionsunterprogramm ist

```
  function masch(q)
  masch=0
  if(q>1.) masch=1
  end
```

Durch den Aufruf von `masch` muss `1.+eps` das Rechenwerk verlassen.

Für den fortran90 ifort-Compiler erhalten wir mit diesem Programm

Bytelänge	`eps`
4	$0.6 \cdot 10^{-7}$
8	$1.0 \cdot 10^{-16}$
16	$1.0 \cdot 10^{-34}$

Diese Werte stimmen genau mit der üblichen Belegung der Speicherplätze überein, wonach ein Byte für den Exponenten und die übrigen Bytes für die Mantisse verwendet wird.

Folgendes Modell für die Beschreibung von Rundungsfehlern hat sich in der Numerischen Mathematik durchgesetzt:
Es gibt eine Abbildung $\mathrm{rd} : \mathbb{R} \to M$ mit

$$\mathrm{rd}\,(x) = x(1 + \varepsilon) \quad \text{mit } |\varepsilon| \leq \texttt{eps}.$$

Nach dem oben Gesagten sind Rundungsfehler relative Fehler, was in diesem Modell berücksichtigt wird.

Die Gleitpunktoperationen $\circ^* : M \times M \to M$ sind dann definiert durch

$$x \pm^* y = \mathrm{rd}\,(x \pm y), \quad x \cdot^* y = \mathrm{rd}\,(x \cdot y), \quad x/^* y = \mathrm{rd}\,(\frac{x}{y}).$$

Im Einklang mit dem oben Angeführten wird also angenommen, dass in M exakt gerechnet und anschließend gerundet wird.

Die Gleitpunktoperationen sind weder assoziativ noch distributiv. Als einfaches Beispiel betrachten wir die Addition $+^*$ bei $t = 1$ und normaler Auf- und Abrundung:

$$(0.8 +^* 0.6) +^* 0.3 = 0.1 \times 10^1$$

$$0.8 +^* (0.6 +^* 0.3) = 0.2 \times 10^1$$

Wir kommen nun zur Auswertung von Funktionen. Die Funktion $f : \mathbb{R}^n \to \mathbb{R}$ sei stetig differenzierbar. Die Auswertung von f heißt *numerisch stabil*, wenn es eine Konstante K gibt mit

$$\Big|\frac{f(x) - f(x + \Delta x)}{f(x)}\Big| \leq K \frac{|\Delta x|}{|x|}$$

für alle $\Delta x \in \mathbb{R}^n$ mit Δx genügend klein. $|x| = \sqrt{x_1^2 + \ldots + x_n^2}$ ist hier die euklidische Norm des Vektors x (siehe später). In der Regel hängt die Konstante K auch von x selber ab.

Anschaulich bedeutet die numerische Stabilität, dass die Rundungsfehler bei der Auswertung von f nur kontrolliert verstärkt werden: Die *Konditionszahl* K sollte möglichst klein sein. Da Rundungsfehler relative Fehler sind, müssen auch in der Definiton der numerischen Stabilität relative Fehler stehen.

Beispiel Für die Addition $f(x_1, x_2) = x_1 + x_2$ erhalten wir

$$\Big|\frac{f(x) - f(x + \Delta x)}{f(x)}\Big| = \Big|\frac{\Delta x_1 + \Delta x_2}{x_1 + x_2}\Big| \leq ?$$

Es gilt

$$|\Delta x_1 + \Delta x_2| \leq \sqrt{2}(\Delta x_1^2 + \Delta x_2^2)^{1/2} = \sqrt{2}|\Delta x|$$

und

$$(x_1^2 + x_2^2)^{1/2} \leq |x_1| + |x_2|,$$

also, falls $x_1, x_2 > 0$,

$$\Big|\frac{f(x) - f(x + \Delta x)}{f(x)}\Big| = \Big|\frac{\Delta x_1 + \Delta x_2}{x_1 + x_2}\Big| \leq \sqrt{2}\frac{|\Delta x|}{|x|}.$$

Die Addition zweier positiver Zahlen ist also numerisch stabil und die Subtraktion zweier ungefähr gleich großer Zahlen ist instabil.

Auf die gleiche Weise leitet man her, dass Multiplikation und Division numerisch stabile Operationen sind. Die Subtraktion zweier etwa gleich großer Zahlen kommt allerdings häufig vor. Die Auswertung eines Polynoms in der Nähe einer Nullstelle und die Bestimmung des Skalarprodukts zweier fast orthogonaler Vektoren sind die wichtigsten Beispiele.

Fazit: Die Akkumulation von Rundungsfehlern bei der Addition nichtnegativer Zahlen verläuft linear in der Anzahl der Summanden und kann durch die Verwendung einer höheren Genauigkeit ausgeglichen werden. Numerische Instabilität aufgrund Auslöschung kann nur durch modifizierte Algorithmen behoben werden.

1.2 Ein Beispiel Für die reellen Zahlen $p, q > 0$ mit $p >> q > 0$ soll der Ausdruck

$$y = -p + \sqrt{p^2 + q}$$

in Gleitpunktarithmetik näherungsweise bestimmt werden. y ist die kleinere der beiden Nullstellen von

$$y^2 + 2py - q = 0.$$

Das Standardverfahren (=Algorithmus 1) zur Bestimmung von y ist sicherlich die direkte Auswertung der obigen Formel

$$\begin{aligned} t &= p^2 + q \\ u &= \sqrt{t} \\ y &= -p + u \end{aligned}$$

Unter obigen Voraussetzungen ist $p \sim u$, der Algorithmus daher instabil.

Man kann die Auswertung verbessern durch (=Algorithmus 2)

$$\begin{aligned} t &= p^2 + q \\ u &= \sqrt{t} \\ v &= p + u \\ y &= q/v \end{aligned}$$

was wegen

$$y = \frac{q}{p + \sqrt{p^2 + q}} = \frac{q}{p + \sqrt{p^2 + q}} \cdot \frac{p - \sqrt{p^2 + q}}{p - \sqrt{p^2 + q}} = \frac{q(p - \sqrt{p^2 + q})}{p^2 - (p^2 + q)} = -p + \sqrt{p^2 + q}$$

das gleiche Ergebnis liefert. In $v = p + u$ haben wir nun zwei positive Zahlen addiert. Da die Auswertung der Wurzel ebenfalls numerisch stabil ist, ist der gesamte Algorithmus numerisch stabil. In der Tat erhalten wir bei zwölfstelliger Rechnung für

$$p = 1000, \quad q = 0.018\,000\,000\,081$$

die Ergebnisse

$$\begin{aligned} \text{Alg. 1}: &\quad 0.900\,030\ldots \times 10^{-5} \\ \text{Alg. 2}: &\quad 0.899\,999\,999\,999\,999 \times 10^{-5}, \end{aligned}$$

der exakte Wert ist 0.9×10^{-5}.

1.3 Vektor- und Matrizennormen Sei X ein nicht notwendig endlich dimensionaler linearer Vektorraum über $\mathbb{K} = \mathbb{R}$ oder $\mathbb{K} = \mathbb{C}$. Eine Abbildung $\|\cdot\| : X \to \mathbb{R}$ heißt *Norm*, wenn sie den folgenden Bedingungen genügt:

(i) $\|x\| \geq 0$ und $\|x\| = 0 \Leftrightarrow x = 0$

(ii) $\|\alpha x\| = |\alpha| \, \|x\|$ für alle $\alpha \in \mathbb{K}$ und $x \in X$.

(iii) $\|x + y\| \leq \|x\| + \|y\|$.

Anschaulich können wir uns vorstellen, dass die Norm von x die Länge des Vektors x angibt oder, falls wir x als Punkt ansehen, die Entfernung dieses Punktes zum Nullpunkt. In dieser Interpretation sind alle Axiome sinnvoll: (i) besagt, dass Abstände positiv sind, sofern es sich nicht um den Nullvektor handelt. (ii) drückt aus, dass die Länge eines Vielfachen das Vielfache der Länge ist. Und schließlich besagt (iii), dass die Strecke die kürzeste Verbindung zweier Punkte ist.

Für $x, y \in \mathbb{C}^n$ ist das Skalarprodukt definiert durch

$$(x, y) = \sum_{j=1}^{n} x_j \overline{y}_j.$$

Die *euklidische Norm* ist für Vektoren $x \in \mathbb{C}^n$ (oder $x \in \mathbb{R}^n$) definiert durch

$$|x| = \Big(\sum_{j=1}^{n} |x_j|^2 \Big)^{1/2} = (x, x)^{1/2}.$$

Aufgrund des Satzes von Pythagoras handelt es sich hierbei in der Tat um die Entfernung des Punktes x zum Nullpunkt. Die Normaxiome (i) und (ii) sind klar, für die Dreiecksungleichung benötigen wir ein Hilfsmittel:

Lemma 1.1 (Cauchy-Ungleichung) *Für $x, y \in \mathbb{C}^n$ gilt*

$$|(x, y)| \le |x|\,|y|.$$

Beweis: Für $a, b \ge 0$ gilt die *Youngsche Ungleichung*

$$ab \le \frac{1}{2}a^2 + \frac{1}{2}b^2,$$

die man aus der binomischen Formel beweist. Mit ihr erhalten wir

$$|(x, y)| = \Big| \sum_{j=1}^{n} x_j \overline{y}_j \Big| \le \sum_{j=1}^{n} \big(\frac{1}{2}|x_j|^2 + \frac{1}{2}|y_j|^2 \big) \le \frac{1}{2}|x|^2 + \frac{1}{2}|y|^2.$$

Für $|x| = |y| = 1$ ist die Ungleichung damit bewiesen. Für $x, y \neq 0$ schreibe wieder

$$x = \alpha \tilde{x}, \quad y = \beta \tilde{y} \quad \text{mit } |\tilde{x}| = |\tilde{y}| = 1$$

und erhalte die Cauchy-Ungleichung. □

Nun zum Beweis der Dreicksungleichung für die Euklidische Norm:

$$\begin{aligned} |x + y|^2 = (x + y, x + y) &= |x|^2 + 2(x, y) + |y|^2 \\ &\le |x|^2 + 2|x|\,|y| + |y|^2 = (|x| + |y|)^2. \end{aligned}$$

Für die euklidische Matrixnorm, auch Frobenius-Norm genannt, schreiben wir entsprechend

$$|A| = \Big(\sum_{j=1}^{m} \sum_{k=1}^{n} |a_{jk}|^2 \Big)^{1/2}, \quad A \in \mathbb{C}^{m \times n} \text{ oder } A \in \mathbb{R}^{m \times n}.$$

Für Matrizen $A \in \mathbb{C}^{m \times n}$, $A = (a_{jk})_{j=1,\ldots,m,\, k=1,\ldots,n}$ setze

$$A^T = (a_{kj})_{k=1,\ldots,n,\, j=1,\ldots,m} = \text{transponierte Matrix}$$

$$A^H = (\overline{a}_{kj})_{k=1,\ldots,n,\, j=1,\ldots,m} = \text{adjungierte Matrix}$$

Ist $A \in \mathbb{C}^{n \times n}$ regulär, so gilt

$$(A^H)^{-1} = (A^{-1})^H,$$

weil aus $AA^{-1} = I$ folgt, dass $(A^{-1})^H A^H = I^H = I$. Damit ist die Inverse von A^H gerade die Matrix $(A^{-1})^H$. Diese Beziehung rechtfertigt die Schreibweise $A^{-H} = (A^{-1})^H$ sowie $A^{-T} = (A^{-1})^T$ für reelle Matrizen.

Vektoren werden auch als Spaltenmatrizen aufgefasst, was zur alternativen Schreibweise $y^H x$ für das Skalarprodukt (x, y) führt.

$A \in \mathbb{R}^{n \times n}$ heißt *symmetrisch*, wenn $A = A^T$, also $a_{ij} = a_{ji}$ für alle $1 \le i, j \le n$. $A \in \mathbb{C}^{n \times n}$ heißt *hermitesch*, wenn $A = A^H$, also $a_{ij} = \overline{a}_{ji}$ für alle $1 \le i, j \le n$, insbesondere $a_{ii} \in \mathbb{R}$.

Satz 1.2 *Auf endlich dimensionalen Räumen sind alle Normen äquivalent, d.h. zu zwei Normen* $\|\cdot\|_1$ *und* $\|\cdot\|_2$ *auf einem endlich dimensionalen Raum* V *gibt es Konstanten* $m, M > 0$ *mit*

$$m\|x\|_2 \le \|x\|_1 \le M\|x\|_2 \quad \textit{für alle } x \in V.$$

Zum Beweis verwenden wir folgendes

Lemma 1.3 *Für jede Norm gilt die* inverse Dreiecksungleichung

$$\|x - y\| \ge \big|\, \|x\| - \|y\| \,\big| \quad \textit{für alle } x, y \in V.$$

Beweis: Mit der normalen Dreiecksungleichung folgt

$$\|x\| = \|(x - y) + y\| \le \|x - y\| + \|y\|,$$

also

$$\|x - y\| \ge \|x\| - \|y\|.$$

Vertauschen wir hier die Rollen von x und y, so ist das Lemma bewiesen. □

Nun zum Beweis des letzten Satzes. Es genügt, den Fall $V = \mathbb{C}^n$ (oder $V = \mathbb{R}^n$) zu betrachten. Jede Norm $\|\cdot\| : \mathbb{C}^n \to \mathbb{R}$, $x \mapsto \|x\|$ ist stetig, denn wenn $x_k \to x$, so folgt aus dem letzten Lemma

$$\big|\, \|x_k\| - \|x\| \,\big| \le \|x_k - x\| \ \to\ 0.$$

Die Einheitssphäre

$$S = \{x \in \mathbb{C}^n : |x| = 1\}$$

ist kompakt. Da die Normen $\|\cdot\|_j$ stetig sind, nehmen sie ihr Minimum und Maximum auf S an, also

$$m_j \le \|x\|_j \le M_j \quad \text{für alle } x \text{ mit } |x| = 1.$$

Wegen $0 \notin S$, sind die $m_j > 0$. Mit $x = \alpha\tilde{x}$, $|\tilde{x}| = 1$, folgt hieraus

$$m_j|x| \le \|x\|_j \le M_j|x| \quad \forall x \in \mathbb{C}^n.$$

Damit ist der Satz bewiesen.

Die *p-Normen* auf $\mathbb{C}^n$ (oder $\mathbb{R}^n$) sind definiert durch

$$\|x\|_p = \Big(\sum_{j=1}^{n} |x_j|^p\Big)^{1/p}, \quad 1 \le p < \infty,$$

$$\|x\|_\infty = \max_{1\le j\le n} |x_j|.$$

Sei $\|\cdot\|_1$ eine Norm auf dem $\mathbb{C}^n$ und $\|\cdot\|_2$ eine Norm auf dem $\mathbb{C}^m$. Der Ausdruck

$$\|A\|_{1\to 2} = \sup_{x\in\mathbb{C}^n\setminus\{0\}} \frac{\|Ax\|_2}{\|x\|_1}, \quad A \in \mathbb{C}^{m\times n},$$

ist eine Norm auf dem Matrizenraum $\mathbb{C}^{m\times n}$ und heißt *(von* $\|\cdot\|_1\|$ *und* $\|\cdot\|_2$*) induzierte Matrixnorm.* Die induzierte Matrixnorm auf dem $\mathbb{R}^{m\times n}$ ist analog definiert.

Bevor wir die Norm-Eigenschaft der induzierten Matrixnorm explizit beweisen, sei auf zwei wichtige Tatsachen hingewiesen. Offenbar gilt

$$\|A\|_{1\to 2} = \sup_{\|x\|_1=1} \|Ax\|_2.$$

Des Weiteren ist die Menge $\{x : \|x\|_1 = 1\}$ kompakt, so dass die stetige Funktion $x \mapsto \|Ax\|_2$ das Maximum annimmt. Wir können daher auch gleich

$$\|A\|_{1\to 2} = \max_{x\in\mathbb{C}^n\setminus\{0\}} \frac{\|Ax\|_2}{\|x\|_1}$$

schreiben.

Aufgrund der letzten Darstellung existiert $\|A\|_{1\to 2}$. $\|A\|_{1\to 2} = 0$ gilt genau dann, wenn $Ax = 0$ für alle x, also $A = 0$. Die positive Homogenität folgt aus der positiven Homogenität von $\|\cdot\|_2$

$$\|\alpha A\|_{1\to 2} = \sup_{\|x\|_1=1} \|\alpha Ax\|_2 = |\alpha| \sup_{\|x\|_1=1} \|Ax\|_2 = |\alpha|\,\|A\|_{1\to 2},$$

und die Dreiecksungleichung folgt analog aus der Dreiecksungleichung für $\|\dot{\|}_2$

$$\|A+B\|_{1\to 2} = \sup_{\|x\|_1=1} \|Ax+Bx\|_2 \le \sup_{\|x\|_1=1} \|Ax\|_2 + \sup_{\|x\|_1=1} \|Bx\|_2 = \|A\|_{1\to 2} + \|B\|_{1\to 2}.$$

Sei $\|\cdot\|_M$ eine Matrixnorm auf $\mathbb{C}^{n\times n}$ und $\|\cdot\|_V$ eine Vektornorm auf $\mathbb{C}^n$.
(i) $\|\cdot\|_M$ heißt *verträglich* mit $\|\cdot\|_V$, wenn

$$\|Ax\|_V \le \|A\|_M \|x\|_V \quad \forall A \in \mathbb{C}^{n\times n}\ \forall x \in \mathbb{C}^n.$$

(ii) $\|\cdot\|_M$ heißt *submultiplikativ*, wenn

$$\|AB\|_M \le \|A\|_M \|B\|_M \quad \forall A, B \in \mathbb{C}^{n\times n}.$$

Beispiele (i) Euklidische Normen sind verträglich,

$$|Ax| \le |A|\,|x|$$

was man mit Hilfe der Cauchy-Ungleichung beweist,

$$|Ax|^2 = \sum_j \Big|\sum_k a_{jk}x_k\Big|^2 \le \sum_j \Big(\sum_k |a_{jk}|^2 \times \sum_k |x_k|^2\Big) = \sum_{j,k} |a_{jk}|^2 \sum_k |x_k|^2 = |Ax|^2|x|^2.$$

Ferner ist $|\cdot|$ auch submultiplikativ,

$$|AB| \le |A|\,|B|$$

wegen

$$|AB|^2 = \sum_{j,l} \Big|\sum_k a_{jk}b_{kl}\Big|^2 \le \sum_{j,l} \Big(\sum_k |a_{jk}|^2 \times \sum_k |b_{kl}|^2)\Big) = |A|^2|B|^2.$$

(ii) Die von einer Vektornorm $\|\cdot\|_V$ induzierte Matrixnorm $\|\cdot\|_{V\to V}$ ist mit dieser Vektornorm verträglich und submultiplikativ

$$\|Ax\|_V \le \|A\|_{V\to V}\|x\|_V, \quad \|AB\|_{V\to V} \le \|A\|_{V\to V}\|B\|_{V\to V}.$$

Die erste Abschätzung folgt aus der Definition der induzierten Norm: $\|A\|_{V\to V}$ ist nämlich die kleinste Konstante c, so dass $\|Ax\|_V \le c\|x\|_V$ für alle x richtig ist. Die zweite Abschätzung folgt aus der ersten

$$\begin{aligned}\|AB\|_{V\to V} &= \sup_{\|x\|_V=1} \|ABx\|_V \le \sup_{\|x\|_V=1} \|A\|_{V\to V}\|Bx\|_V \\ &\le \sup_{\|x\|_V=1} \|A\|_{V\to V}\|B\|_{V\to V}\|x\|_V = \|A\|_{V\to V}\|B\|_{V\to V}\end{aligned}$$

Der Begriff der Submultiplikativität lässt sich noch etwas verallgemeinern. Haben wir drei Räume $\mathbb{C}^l, \mathbb{C}^m, \mathbb{C}^n$, so heißen die drei zugehörigen Normen *submultiplikativ*, wenn

$$\|AB\|_{\mathbb{C}^n \to \mathbb{C}^l} \leq \|A\|_{\mathbb{C}^n \to C^m} \|B\|_{C^n \to \mathbb{C}^m} \quad \forall A \in \mathbb{C}^{l \times m} \; \forall B \in \mathbb{C}^{m \times n}.$$

Auch hier sind die euklidischen Matrixnormen und die induzierten Matrixnormen (nachdem man auf jedem der drei Räume eine Vektornorm festgelegt hat) submultiplikativ.

Notation: Sei $A \in \mathbb{C}^{m \times n}$. Wenn nichts anderes gesagt wird, ist

$$\|A\| = \sup_{x \neq 0} \frac{|Ax|}{|x|}.$$

Lemma 1.4 *Für $A \in \mathbb{C}^{n \times n}$ gilt*

$$\|A\| = \rho(AA^H)^{1/2} = \rho(A^H A)^{1/2},$$

wobei $\rho(B)$ den betragsmäßig größten Eigenwert von B bezeichnet.

Ist $A \in \mathbb{C}^{n \times n}$ hermitesch, so gilt

$$\|A\| = \rho(A).$$

Beweis: Es gilt

$$\|A\|^2 = \sup_{x \neq 0} \frac{(Ax, Ax)}{(x, x)} = \sup_{x \neq 0} \frac{(A^H Ax, x)}{(x, x)}.$$

Da die Matrix $A^H A$ hermitesch und positiv semidefinit ist, besitzt sie einen vollständigen Satz von Eigenvektoren $v_1, \ldots, v_n$ mit $A^H A v_j = \lambda_j v_j$ und nichtnegativen λ_j. In die letzte Identität setzen wir $x = \sum_j c_j v_j$ ein und erhalten wegen der Orthogonalität der v_j

$$\|A\|^2 = \sup_{c \neq 0} \frac{(A^H A \sum_j c_j v_j, \sum_j c_j v_j)}{(\sum_j c_j v_j, \sum_j c_j v_j)} = \sup_{c \neq 0} \frac{\sum_j \lambda_j |c_j|^2}{\sum_j |c_j|^2}.$$

Dieses Optimierungsproblem wird natürlich durch $c_n = 1$, $c_j = 0$ für $1 \leq j \leq n-1$ gelöst, wenn λ_n der größte Eigenwert von $A^H A$ ist.

Ist A hermitesch, so erhalten wir die Eigenwerte und Eigenvektoren von $A^H A = A^2$ aus den Eigenwerten und Eigenvektoren von A. □

Einige weitere Beispiele von Matrizennormen auf dem $\mathbb{C}^{m \times n}$ sind

(i) $\|A\|_Z = \max_j \sum_{k=1}^n |a_{jk}|$ (=Zeilensummennorm),

(ii) $\|A\|_S = \max_k \sum_{j=1}^m |a_{jk}|$ (=Spaltensummennorm),

(iii) $\|A\|_\infty = \max_{j,k} |a_{jk}|$.

(i) und (ii) sind submultiplikativ, (iii) nicht wegen

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix}.$$

1.4 Stabilität linearer Gleichungssysteme Sei $A \in \mathbb{C}^{n \times n}$ (oder $A \in \mathbb{R}^{n \times n}$). Wir wollen das lineare Gleichungssystem

$$Ax = b \quad \text{für } b \in \mathbb{C}^n$$

lösen.

Satz 1.5 *Sei A regulär. Dann gilt für die Lösung $x + \Delta x$ des gestörten Problems*

$$A(x + \Delta x) = b + \Delta b$$

die Abschätzung

$$\frac{|\Delta x|}{|x|} \leq \|A\| \, \|A^{-1}\| \frac{|\Delta b|}{|b|}.$$

Die Zahl

$$cond\,(A) = \|A\| \, \|A^{-1}\|$$

gibt die Verstärkung des relativen Fehlers bei ungenauer Auswertung der rechten Seite an und heißt deshalb Konditionszahl *der Matrix A.*

Beweis: Aus $A\Delta x = \Delta b$ folgt

$$\Delta x = A^{-1}\Delta b \;\Rightarrow\; |\Delta x| \leq \|A^{-1}\| \, |\Delta b|.$$

Aus $Ax = b$ erhalten wir entsprechend die Abschätzung $|b| \leq \|A\| \, |x|$ und damit die Behauptung. □

Anwendung Bei iterativen Verfahren zur Lösung von $Ax = b$ erhalten wir Näherungen $\tilde{x}$, aber kennen den Fehler $|x - \tilde{x}|$ nicht. Wann sollen wir abbrechen? Einfachste Lösung dieses Problems besteht in der Bestimmung des Residuums $r = b - A\tilde{x} = A(x - \tilde{x})$. $\tilde{x}$ ist dann die exakte Lösung von $A\tilde{x} = b - r$ und für den relativen Fehler gilt dann

$$\frac{|x - \tilde{x}|}{|x|} \leq \text{cond}\,(A) \frac{|r|}{|b|}.$$

Mit $\tilde{x}$ sind r und b bekannt, die Kondition von A muss allerdings geschätzt werden.

2 Direkte Lösung linearer Gleichungssysteme

2.1 Gauß Elimination Zu $b \in \mathbb{R}^n$ und $A \in \mathbb{R}^{n\times n}$ ist ein $x \in \mathbb{R}^n$ gesucht mit

$$Ax = b.$$

Die Lösung eines solchen $n \times n$-Gleichungssystems ist sicherlich das am häufigsten in der Praxis auftretende Problem. Die Idee des Gaußschen Eliminationsverfahrens besteht darin, durch Vertauschung von Zeilen und Addition des Vielfachen einer Zeile auf eine andere Zeile das System auf Dreiecksgestalt zu bringen

$$Rx = c$$

mit

$$R = \begin{bmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \end{bmatrix}$$

Dieses System wird nun von unten nach oben gelöst: Aus der letzten Zeile bestimmen wir x_n, setzen diesen Wert in die vorletzte Zeile ein und erhalten so x_{n-1} usw.

Dreiecksmatrizen Rechte obere Dreiecksmatrizen R mit $r_{ij} = 0$ für $i > j$ kommen häufig in der Numerik vor. Wir nennen R *normiert*, wenn zusätzlich $r_{ii} = 1$ für $1 \le i \le n$.

Lemma 2.1 (a) *Für eine rechte obere Dreiecksmatrix R gilt* $\det R = \prod_{i=1}^n r_{ii}$. *Insbesondere ist R genau dann regulär, wenn $r_{ii} \neq 0$ für $1 \le i \le n$.*

(b) *Produkte und Inverse von (normierten) rechten oberen Dreiecksmatrizen sind (normierte) rechte obere Dreiecksmatrizen.*

Beweis: (a) Wir entwickeln $\det R$ nach der ersten Spalte, in der nur das Element $r_{11} \neq 0$ vorkommt. Nach Streichen der ersten Zeile und ersten Spalte verbleibt eine rechte obere Dreiecksmatrix, deren Determinante wieder nach der ersten Spalte entwickelt wird.

(b) Seien R, S rechte obere Dreiecksmatrizen. Für $i > j$ gilt

$$(RS)_{ij} = \sum_{k=1}^{n} r_{ik}s_{kj} = \sum_{k=1}^{i-1} r_{ik}s_{kj} + \sum_{k=i}^{n} r_{ik}s_{kj}$$

und $r_{ik} = 0$ in der ersten Summe und $s_{kj} = 0$ in der zweiten. Sind die Dreiecksmatrizen normiert, so folgt ganz analog

$$(RS)_{ii} = \sum_{k=1}^{i-1} r_{ik}s_{ki} + \sum_{k=i}^{n} r_{ik}s_{ki} = r_{ii}s_{ii} = 1.$$

Wir bekommen die j-te Spalte x_j der Inversen einer regulären Matrix A, indem wir das lineare Gleichungssystem $Ax_j = e_j$ lösen, wobei e_j der j-te Einheitsvektor ist. Bei einer regulären rechten oberen Dreiecksmatrix R lösen wir dieses Gleichungssystem von unten nach oben. Für die Komponenten $x_{j,n}, \ldots, x_{j,j+1}$ bekommen wir lauter Nullen. Erst in der j-ten Gleichung steht $r_{jj}x_{j,j} = 1$, insbesondere ist $x_{j,j} = 1$, falls R eine normierte obere Dreiecksmatrix ist. Man beachte, dass $(A^{-1})_{ij} = x_{j,i}$. Damit ist R^{-1} eine (normierte) rechte obere Dreiecksmatrix. □

Natürlich gilt dieses Lemma für linke untere Dreiecksmatrizen völlig analog.

Gauß-Algorithmus Die Elimination geschieht spaltenweise beginnend mit der 1. Spalte. Mit $(A^{(0)}, b^{(0)}) = (A, b)$ konstruieren wir eine Folge $(A^{(j)}, b^{(j)})$, so dass $A^{(n-1)} = R$ und $b^{(n-1)} = c$.

Sei

$$(A^{(j)}, b^{(j)}) = \left[\begin{array}{ccc|ccc|c} * & \cdots & * & * & \cdots & * & * \\ & \ddots & \vdots & \vdots & & \vdots & \vdots \\ 0 & & * & * & \cdots & * & * \\ \hline & & & * & \cdots & * & * \\ & 0 & & \vdots & & \vdots & \vdots \\ & & & * & \cdots & * & * \end{array}\right]$$

die Matrix und die rechte Seite, die wir erhalten, nachdem die ersten j Spalten eliminiert worden sind. Es gibt nun zwei Möglichkeiten, das Element der Spalte $j+1$ zu finden, mit der die Elimination der Spalte $j+1$ durchgeführt wird.

a) (Spaltenpivotsuche) Bestimme r mit

$$|a_{r\,j+1}| = \max_{i \geq j+1} |a_{i\,j+1}|.$$

Vertausche die Zeile r mit der Zeile $j+1$. Wenn $a_{i\,j+1} = 0$ für alle $i \geq j+1$, so ist die Matrix singulär.

b) (Totale Pivotsuche) Bestimme r und s mit

$$|a_{rs}| = \max_{i,k \geq j+1} |a_{ik}|.$$

Vertausche die Zeile r mit Zeile $j+1$ und die Spalte s mit Spalte $j+1$. Wenn $a_{rs} = 0$, so ist die Matrix singulär.

Hat man ein nichtverschwindendes Pivot-Element gefunden, so kann man nun den unteren Teil der Spalte $j+1$ mit Hilfe von $a_{j+1\,j+1}$ eliminieren. Wir bezeichnen die Elemente nach erfolgter Vertauschung mit $a_{ik}^{(j)}$ und $b_i^{(j)}$. Setze nun

$$a_{ik}^{(j+1)} = a_{ik}^{(j)} - a_{j+1\,k}^{(j)} \frac{a_{i\,j+1}^{(j)}}{a_{j+1\,j+1}^{(j)}}, \quad i > j+1,\ k \geq j+1$$

$$b_i^{(j+1)} = b_i^{(j)} - b_{j+1}^{(j)} \frac{a_{i\,j+1}^{(j)}}{a_{j+1\,j+1}^{(j)}}, \quad i > j+1,$$

d.h. wir addieren das $-a_{i\,j+1}^{(j)}/a_{j+1\,j+1}^{(j)}$-fache der $j+1$-ten Zeile auf die Zeile i.

Matrizielle Darstellung (bei Spaltenpivotsuche) Für $1 \leq j < k \leq n$ heißt

$$P_{jk} = \begin{bmatrix} 1 & & & & & & & & & & \\ & \ddots & & & & & & & & & \\ & & 1 & & & & & & & & \\ & & & 0 & \cdots & \cdots & \cdots & 1 & & & \\ & & & \vdots & 1 & & & \vdots & & & \\ & & & \vdots & & \ddots & & \vdots & & 0 & \\ & 0 & & \vdots & & & 1 & \vdots & & & \\ & & & 1 & \cdots & \cdots & \cdots & 0 & & & \\ & & & & & & & & 1 & & \\ & & & & & & & & & \ddots & \\ & & & & & & & & & & 1 \end{bmatrix}$$

Permutationsmatrix. Die Multiplikation mit einer Matrix A hat die folgende Wirkung

$$P_{jk}A \quad \text{wie } A\text{, aber Zeile } j \text{ wird mit Zeile } k \text{ vertauscht,}$$

$$AP_{jk} \quad \text{wie } A\text{, aber Spalte } j \text{ wird mit Spalte } k \text{ vertauscht.}$$

Demnach ist $P_{jk}P_{jk}A = A$, also $P_{jk}^{-1} = P_{jk}$ Für Zahlen $l_{j+1\,j}, \ldots, l_{n\,j}$ ist die *Frobenius-Matrix* definiert durch

$$G_j = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & 0 & \\ & & \ddots & & & & \\ & & & 1 & & & \\ & & & -l_{j+1\,j} & \ddots & & \\ & 0 & & \vdots & & \ddots & \\ & & & -l_{n\,j} & & & 1 \end{bmatrix}, \quad G_j^{-1} = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & 0 & \\ & & \ddots & & & & \\ & & & 1 & & & \\ & & & l_{j+1\,j} & \ddots & & \\ & 0 & & \vdots & & \ddots & \\ & & & l_{n\,j} & & & 1 \end{bmatrix}.$$

Multiplikation von links an die Matrix A hat die gewünschte Eliminationswirkung

$$G_jA \quad \text{addiert das } -l_{k\,j}\text{-fache der Zeile } j \text{ auf die Zeile } k \text{ von } A \text{ für } k = j+1, \ldots, n$$

Der im vorigen Abschnitt dargestellte Gauß-Algorithmus besteht daher bei der Spalten-Pivotsuche aus der abwechselnden Multiplikation von (A, b) mit einer Permutationsmatrix (=Pivotisierung) und einer Frobenius-Matrix (=Eliminationsschritt),

$$(R, c) = G_{n-1}P_{n-1}G_{n-2}P_{n-2}\ldots G_1P_1(A, b). \tag{2.1}$$

Für $i > j$ gilt

$$P_iG_j = \tilde{G}_jP_i, \tag{2.2}$$

wobei $\tilde{G}_j$ aus G_j durch Vertauschen von $-l_{k,j}$ mit $-l_{l,j}$ hervorgeht. $\tilde{G}_j$ ist also eine Frobenius-Matrix vom gleichen Typ wie G_j. Wir wenden diese Beziehung auf (2.1) an und bringen damit die Frobenius-Matrizen nach links

$$(R, c) = G_{n-1}\tilde{G}_{n-2}\ldots\tilde{G}_1P_{n-1}P_{n-2}\ldots P_1(A, b).$$

Damit ist

$$\tilde{G}_1^{-1}\ldots\tilde{G}_{n-2}^{-1}G_{n-1}^{-1}R = P_{n-1}\ldots P_1A$$

also

$$LR = PA$$

mit

$$L = \text{ untere Dreiecksmatrix mit } l_{ii} = 1$$

$$R = \text{ obere Dreiecksmatrix}$$

$$P = \text{ Permutationsmatrix}$$

Satz 2.2 (Existenz der LR-Zerlegung) *Zu jeder regulären Matrix A gibt es eine Permutationsmatrix P, so dass*

$$PA = LR.$$

Praktische Durchführung Wir beschreiben die Umsetzung des Gaußschen Eliminationsverfahren mit Spalten-Pivotsuche (siehe File `gaus.f90`).

Üblicherweise konstruiert man zuerst die LR-Zerlegung und speichert sie auf der Eingangsmatrix ab, die dabei zerstört wird. Die Matrix L kommt auf die untere Dreiecksmatrix, wobei die Einser in der Hauptdiagonalen nicht abgespeichert werden. Dadurch kann die Hauptdiagonale für die Dreiecksmatrix R genutzt werden. In `sr gausz` wird zuerst die Spaltenpivotsuche durchgeführt. Wenn kein nichtverschwindendes Pivotelement gefunden wurde, wird mit einer Fehlermeldung abgebrochen. Anschließend werden die entsprechenden Zeilen der gesamten Matrix vertauscht, also auch die im vorderen Teil des Schemas bereits konstruierten Elemente der Matrix L. Dies folgt aus (2.2), wonach die Vertauschung von P_iG_j genau so eine Vertauschung in G_j bewirkt. Anschließend werden die neuen Elemente von G_k^{-1} bestimmt und abgespeichert und die Elimination durchgeführt. Zu bemerken bleibt noch, dass für das Produkt von Frobenius-Matrizen G_i^{-1} mit i-ter Spalte g_i gilt

$$G_1^{-1} \dots G_{n-1}^{-1} = [g_1| \dots |g_{n-1}|e_n].$$

Abgesehen von der Zeilenvertauschung bleiben also einmal konstruierte Elemente l_{ij} der Frobenius-Matrizen erhalten. Der Algorithmus ist also sehr viel einfacher, als die obigen Formeln zunächst nahe legen.

Die Zeilenvertauschungen werden in der `sr gausz` im Vektor `irow` abgespeichert. Neben der LR-Zerlegung wird dieser Vektor gebraucht, um für eine konkrete rechte Seite b die Lösung zu bestimmen, was in der `sr gauszloe` geschieht. Dort werden zuerst die Zeilenvertauschungen durchgeführt, anschließend müssen für $LR = Pb$ zwei Gleichungssysteme mit Dreiecksmatrizen gelöst werden.

Die Konstruktion der Zerlegung und die Bestimmung der Lösung des linearen Gleichungssystems geschieht separat, weil man häufig mehrere Gleichungssysteme mit derselben Matrix lösen muss. Offenbar benötigt man zur Elimination der j-ten Spalte $O((n+1-j)^2)$ Operationen, für die gesamte LR-Zerlegung demnach $O(n^3)$ Operationen. Dagegen müssen für die Lösung des Gleichungssystems $LRx = c$ nur $O(n^2)$ Operationen aufgewendet werden.

Das vorliegende Programm ist in dieser Form noch unbefriedigend, weil es nur für reguläre Matrizen halbwegs sicher funktioniert. Bei einer singulären Matrix wird in der Regel aufgrund von Rundungsfehlern ein nichtverschwindendes Pivotelement gefunden, so dass die Fehlermeldung nicht greift.

Der Sinn des Pivotisierens In unzähligen Prüfungen im Fach Numerische Mathematik spielt sich folgendes Gespräch ab:

P: Was ist der Sinn des Pivotisierens? S: Durch das Pivotisieren teile ich durch relativ große Zahlen und halte die Rundungsfehler klein. P: Die Division ist immer eine numerisch stabile Operation unabhängig davon, ob der Nenner groß oder klein ist. S: ??

Der Mechanismus ist anders als in diesem Gespräch vom Studenten angenommen. Wird ein kleines Pivot-Element genommen, so wird die Zeile, mit dem die Elimination vorgenommen wird, mit einer großen Zahl multipliziert. Dadurch werden große Zahlen auf die unteren Zeilen der Matrix addiert, was zum völligen oder teilweisen Verschlucken der Information in den unteren Zeilen führt.

Beispiel 2.3 Wir betrachten das Gleichungssystem

$$\begin{bmatrix} 0.005 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}$$

mit exakter Lösung

$$x = \frac{5000}{9950} = 0.503, \quad y = \frac{4950}{9950} = 0.497.$$

Bei zweistelliger Rechnung ohne Pivotisieren ergibt sich

$$\left[\begin{array}{cc|c} 0.005 & 1 & 0.5 \\ 0 & -200 & -99 \end{array}\right].$$

Bei dieser Elimination wurde die erste Zeile mit 200 multipliziert und dies von der zweiten Zeile abgezogen. Für das Element $a_{22}^{(1)}$ gilt dann $a_{22}^{(1)} = \mathrm{rd}\,(1-200) = -200$, womit die Originalinformation der Matrix komplett verschwunden ist. Als Lösung erhalten wir nun

$$\tilde{y} = \mathrm{rd}\left(\frac{99}{200}\right) = 0.50, \quad \tilde{x} = 0.00.$$

Mit Pivotisieren wird das 0.005-fache der (neuen) ersten Zeile von der zweiten abgezogen,

$$\left[\begin{array}{cc|c} 1 & 1 & 1 \\ 0 & 1 & 0.5 \end{array}\right]$$

mit Lösung

$$\tilde{y} = 0.50, \quad \tilde{x} = 0.50.$$

Hätten wir das äquivalente System

$$\begin{bmatrix} 1 & 200 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 100 \\ 1 \end{bmatrix}$$

zu lösen gehabt, so würden wir keine Pivotisierung durchführen und kämen auf die gleiche falsche Lösung wie oben. □

Das Beispiel zeigt deutlich, dass Pivotisieren ohne *Equilibrieren*, also die Matrixelemente auf die gleiche Größenordnung zu bringen, häufig nicht viel bringt. In der *idealen Equilibrierung* sucht man Diagonalmatrizen D_l und D_r, so dass für die Matrix $\tilde{A} = D_l A D_r$ gilt

$$\sum_{k=1}^{n} |\tilde{a}_{ik}| \approx \sum_{j=1}^{n} |\tilde{a}_{jl}|, \quad 1 \le i, l \le n,$$

dass also alle Zeilen- und Spaltensummen von der gleichen Größenordnung sind. Dafür gibt es keinen guten Algrithmus, in manchen Programmen wird immerhin

$$D_l = I, \quad D_r = \mathrm{diag}\,(s_1, \ldots, s_n) \;\text{ mit }\; s_i = \frac{1}{\sum_k |a_{ik}|},$$

verwendet, also die Normierung der Zeilensummen zu 1.

Bestimmung der Determinante und der Inversen einer Matrix Für $PA = LR$ gilt nach dem Determinantenmultiplikationssatz wegen $l_{ii} = 1$

$$\det A = \det P \prod_{i=1}^{n} r_{ii},$$

wobei $\det P$ nur am Vorzeichen etwas ändert, je nachdem, ob die Anzahl der Permutationen gerade (Vorzeichen +) oder ungerade (Vorzeichen −) ist.

Die Bestimmung der Inversen einer Matrix ist in der Praxis nur selten erforderlich. Man löst dazu die Gleichungssyteme (e_i=i-ter Einheitsvektor)

$$Ax_i = e_i, \quad i = 1, \ldots, n$$

und erhält für die Inverse $A^{-1} = [x_1 | \ldots | x_n]$. Da die Lösung von $LR = c$ nur $O(n^2)$ Operationen erfordert, kann die Inverse in $O(n^3)$ Operationen berechnet werden.

2.2 Die Cholesky-Zerlegung Eine Matrix $A \in \mathbb{C}^{n\times n}$ heißt *positiv definit*, wenn sie hermitesch ist, also $A = A^H$ erfüllt ist, und wenn gilt

$$(Ax, x) > 0 \quad \forall x \in \mathbb{C}^n \setminus \{0\}.$$

Bei dieser Definition ist zu beachten, dass bei hermiteschen Matrizen die zugehörige quadratische Form reellwertig ist wegen

$$q(x) := (Ax, x) = (x, A^H x) = (x, Ax) = \overline{(Ax, x)} \;\Rightarrow\; q(x) \in \mathbb{R}.$$

Satz 2.4 *Sei $A \in \mathbb{C}^{n\times n}$ positiv definit.*

(a) *A ist regulär und A^{-1} ist wieder positiv definit.*

(b) *Alle Hauptdiagonalelemente von A sind positiv.*

(c) *Alle* Hauptuntermatrizen

$$A[k] = (a_{ij})_{i,j=1,\ldots,k}$$

sind positiv definit und alle Hauptunterdeterminanten

$$\Delta_k = \det A[k]$$

sind positiv.

Beweis: (a) Wäre A nicht regulär, so existiert $x \in \mathbb{C}^n \setminus \{0\}$ mit $Ax = 0$. Dann ist auch $(Ax, x) = 0$, was einen Widerspruch zur positiven Definitheit bedeutet. A^{-1} ist hermitesch wegen $(A^{-1})^H = (A^H)^{-1} = A^{-1}$ und A^{-1} ist positiv definit wegen

$$(A^{-1}y, y) = (A^{-1}Ax, Ax) = (x, Ax) > 0 \quad \text{für } y \neq 0.$$

(b) Mit dem k-ten Einheitsvektor e_k folgt $0 < (Ae_k, e_k) = a_{kk}$.

(c) Jeder Vektor $x \in \mathbb{C}^k$ lässt sich durch Ergänzen von Nullen zu einem Vektor $\tilde{x} \in \mathbb{C}^n$ machen. Für $x \neq 0$ ist auch $\tilde{x} \neq 0$ und daher

$$(A[k]x, x)_{\mathbb{C}^k} = (A\tilde{x}, \tilde{x})_{\mathbb{C}^n} > 0.$$

Damit ist auch $A[k]$ positiv definit.

Zu zeigen bleibt, dass die Hauptunterdeterminanten positiv sind, was wir durch Induktion über n zeigen. Für $n = 1$ ist die Behauptung richtig. Sei $\Delta_{n-1} > 0$ für alle $A \in \mathbb{C}^{(n-1)\times(n-1)}$. Für positiv definites $A \in \mathbb{C}^{n\times n}$ ist nach dem oben Gesagten die Matrix $A[n-1]$ positiv definit. Mit $A^{-1} = (\alpha_{ij})$ gilt nach der Cramerschen Regel

$$0 < \alpha_{nn} = \frac{\det \tilde{A}}{\det A}.$$

wobei $\tilde{A}$ aus A entsteht, indem die letzte Spalte von A durch den Einheitsvektor e_n ersetzt wird. Entwicklung nach der letzten Spalte liefert $\det \tilde{A} = \det A[n-1] > 0$. Damit ist auch $\det A > 0$. □

Satz 2.5 (Cholesky-Zerlegung) *Die hermitesche Matrix A ist genau dann positiv definit, wenn es eine linke untere Dreiecksmatrix L mit l_{ii} reell und positiv gibt, so dass $A = LL^H$. Für reelles A ist auch L reell. Wenn die Matrix L existiert, so ist sie in der Klasse der unteren Dreiecksmatrizen mit reeller und positiver Hauptdiagonale eindeutig bestimmt.*

Beweis: Die Richtung „⇒“ zeigen wir durch Induktion über n. Für $n = 1$ ist $A = a_{11} > 0$ und daher ist $L = \sqrt{a_{11}}$ die gesuchte Matrix. Sei die Behauptung für alle $(n-1)\times(n-1)$-Matrizen erfüllt. Für $A \in \mathbb{C}^{n\times n}$ positiv definit schreibe

$$A = \begin{bmatrix} A_{n-1} & b \\ b^H & a_{nn} \end{bmatrix}, \quad A_{n-1} \in \mathbb{C}^{(n-1)\times(n-1)},\ b \in \mathbb{C}^{n-1},\ a_{nn} \in \mathbb{R}.$$

Mit einer linken unteren Dreiecksmatrix $L_{n-1} \in \mathbb{C}^{(n-1)\times(n-1)}$, einem $c \in \mathbb{C}^{n-1}$ und $\alpha \in \mathbb{C}$ setzen wir an

$$\begin{bmatrix} A_{n-1} & b \\ b^H & a_{nn} \end{bmatrix} = \begin{bmatrix} L_{n-1} & 0 \\ c^H & \alpha \end{bmatrix} \begin{bmatrix} L_{n-1}^H & c \\ 0 & \alpha \end{bmatrix},$$

also

$$A_{n-1} = L_{n-1}L_{n-1}^H, \quad L_{n-1}c = b, \quad c^H c + \alpha^2 = a_{nn}.$$

Aufgrund der Induktionsvoraussetzung ist $A_{n-1} = L_{n-1}L_{n-1}^H$ die eindeutig bestimmte Cholesky-Zerlegung. $c = L_{n-1}^{-1}b$ existiert, weil L_{n-1} regulär ist. Ebenso gibt es ein $\alpha \in \mathbb{C}$, das die letzte Gleichung erfüllt. Wir müssen nun noch zeigen, dass α reell und positiv gewählt werden kann und diese Wahl dann eindeutig ist. Nach dem Determinantenmultiplikationssatz gilt

$$\det A = \det L_{n-1}\alpha \det L_{n-1}^H\alpha = |\det L_{n-1}|^2\alpha^2$$

Damit muss $\alpha^2 > 0$ sein. Als α wählen wir die positive Lösung von $\alpha^2 = a_{nn} - c^H c$. Damit ist auch $\alpha > 0$ eindeutig bestimmt.

Gilt nun umgekehrt $A = LL^H$, so ist A wegen $(LL^H)^H = LL^H$ hermitesch und wegen

$$(Ax, x) = (LL^H x, x) = |L^H x|^2 > 0 \quad \text{für } x \neq 0$$

auch positiv definit. □

In der praktischen Durchführung der Cholesky-Zerlegung wird nur die obere Dreiecksmatrix einschließlich der Hauptdiagonalen für die Speicherung der Matrix A benötigt. Die untere Dreicksmatrix steht daher für die Speicherung von L zur Verfügung, man benötigt noch einen Vektor d für die Speicherung der Hauptdiagonalen von L.

Wir beschreiben den Algorithmus der Einfacheit halber nur für reelle symmetrische Matrizen A. Von der Matrix L werden sukzessive die Hauptuntermatrizen aufgebaut. Für $k = 1$ ist $l_{11} = \sqrt{a_{11}}$. Sei $L[k-1]$ bekannt. Dann sieht $A[k] = L[k]L[k]^T$ folgendermaßen aus

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kk} \end{bmatrix} = \begin{bmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{k1} & l_{k2} & \cdots & l_{kk} \end{bmatrix} \begin{bmatrix} l_{11} & l_{21} & \cdots & l_{k1} \\ & l_{22} & \cdots & l_{k2} \\ & & \ddots & \vdots \\ & & & l_{kk} \end{bmatrix}$$

Die Auswertung der k-ten Zeile liefert die Berechnungsvorschrift für die l_{ki}

$$\begin{array}{lll} a_{k1} = l_{k1}l_{11} & \Rightarrow & l_{k1} = a_{k1}/l_{11} \\ a_{k2} = l_{k1}l_{21} + l_{k2}l_{22} & \Rightarrow & l_{k2} = (a_{k2} - l_{k1}l_{21})/l_{22} \\ \vdots & \vdots & \vdots \\ a_{ki} = l_{k1}l_{i1} + \ldots + l_{ki}l_{ii} & \Rightarrow & l_{ki} = (a_{ki} - l_{k1}l_{i1} - \ldots - l_{k\,i-1}l_{i\,i-1})/l_{ii} \\ \vdots & \vdots & \vdots \\ a_{kk} = l_{k1}^2 + l_{k2}^2 + \ldots + l_{kk}^2 & \Rightarrow & l_{kk} = \sqrt{a_{kk} - l_{k1}^2 - l_{k2}^2 - \ldots - l_{k\,k-1}^2} \end{array} \tag{2.3}$$

Das Ziehen der Wurzel in der letzten Formel macht die Cholesky-Zerlegung sehr stabil, wie man auch an folgendem Beispiel sieht.

Beispiel 2.6 Für $a > 1$ gilt $A = LL^T$ mit

$$A = \begin{bmatrix} 1 & 1 \\ 1 & a \end{bmatrix}, \quad L = \begin{bmatrix} 1 & 0 \\ 1 & \sqrt{a-1} \end{bmatrix}.$$

Für $a > 1$ nahe bei 1 gilt für die Eigenwerte von A, dass $\lambda_1 \approx 1$ und $\lambda_2 \approx a - 1$. Für $a \approx 1$ ist die Matrix sehr schlecht konditioniert. Weiter ist in der LR-Zerlegung von A $r_{22} = a - 1$, durch das im Eliminationsschritt geteilt wird. Dagegen wird im Cholesky-Verfahren durch $\sqrt{a-1}$ geteilt. □

2.3 Orthogonalisierungsverfahren $U \in \mathbb{C}^{n\times n}$ ($U \in \mathbb{R}^{n\times n}$) heißt *unitär* (*orthogonal*), wenn

$$U^H = U^{-1} \quad (U^T = U^{-1}).$$

Wegen $U^H U = UU^H = I$ bilden die Zeilen- und Spaltenvektoren jeweils eine Orthonormalbasis des $\mathbb{C}^n$. Ferner gilt für unitäres (oder orthogonales) U

$$|Ux|^2 = (Ux, Ux) = (U^H Ux, x) = |x|^2,$$

daher

$$\|U\| = 1, \quad \|U^{-1}\| = 1, \quad \text{cond}\,(U) = 1.$$

Für eine unitäre Matrix $U \in \mathbb{C}^{n\times n}$ und eine allgemeine Matrix $A \in \mathbb{C}^{n\times n}$ gilt wegen $|Ux| = |x|$

$$\|A\| = \sup_{x\neq 0} \frac{|Ax|}{|x|} = \sup_{x\neq 0} \frac{|UAx|}{|x|} = \|UA\|,$$

insbesondere ändert sich die Kondition einer regulären Matrix nicht, wenn man eine unitäre Matrix heranmultipliziert, $\text{cond}\,(A) = \text{cond}\,(UA)$.

Produkte unitärer Matrizen U, V sind unitär wegen $(UV)^H(UV) = V^H U^H UV = I$.

Sei $A \in \mathbb{C}^{n\times n}$. Gilt für eine unitäre Matrix $Q \in \mathbb{C}^{n\times n}$ und eine rechte obere Dreiecksmatrix $R \in \mathbb{C}^{n\times n}$ die Beziehung $A = QR$, so sprechen wir von einer *QR-Zerlegung der Matrix A*.

Lemma 2.7 (a) *Für eine reguläre Matrix $A \in \mathbb{C}^{n\times n}$ existiert eine eindeutige unitäre Matrix $Q \in \mathbb{C}^{n\times n}$ und eine eindeutige rechte obere Dreiecksmatrix $R \in \mathbb{C}^{n\times n}$ mit reellwertigen positiven Diagonalelementen $r_{ii} > 0$, so dass $A = QR$. Die Matrizen Q, R hängend dabei stetig von den Koeffizienten a_{ij} ab.*

(b) Ist $A = Q_1 R_1$ eine weitere QR-Zerlegung der regulären Matrix A, so gibt es eine Diagonalmatrix D, $D = \text{diag}\,(d_{11}, \ldots, d_{nn})$, mit $|d_{ii}| = 1$ für $1 \leq i \leq n$ und

$$Q_1 = QD^H, \quad R_1 = DR,$$

wobei Q, R die Matrizen aus (a) sind.

Beweis: Die Matrix $A^H A$ ist positiv definit und besitzt daher eine eindeutige Cholesky-Zerlegung $A^H A = R^H R$ mit einer rechten oberen Dreiecksmatrix R mit $r_{ii} > 0$. Mit $Q = AR^{-1}$ gilt

$$Q^H Q = R^{-H} A^H A R^{-1} = R^{-H} R^H R R^{-1} = I,$$

wobei hier $B^{-H} = (B^{-1})^H = (B^H)^{-1}$ verwendet wurde. Die Matrix Q ist damit unitär und definitionsgemäß gilt $A = QR$.

Gäbe es eine weitere Zerlegung $A = \hat{Q}\hat{R}$ mit den angegebenen Eigenschaften, so

$$A^H A = \hat{R}^H \hat{Q}^H \hat{Q} \hat{R} = \hat{R}^H \hat{R},$$

also $\hat{R} = R$ wegen der Eindeutigkeit der Cholesky-Zerlegung. Damit gilt aber auch $\hat{Q} = A\hat{R}^{-1} = AR^{-1} = Q$.

Die Cholesky-Zerlegung setzt sich – als Funktion der Matrixelemente – aus den elementaren Operationen und der Wurzel zusammen, ist somit eine stetige Funktion der Matrixelemente. In $Q = AR^{-1}$ ist sowohl die Inverse (Cramersche Regel!) als auch die Matrizenmultiplikation eine stetige Operation.

(b) Aus $A = QR = Q_1 R_1$ folgt $Q_1^H Q R R_1^{-1} = I$. Als Inverse von $Q_1^H Q$ ist RR_1^{-1} selber unitär und muss als Inverse eine linke untere Dreiecksmatrix besitzen. Andererseits besitzt die rechte obere Dreiecksmatrix RR_1^{-1} nach Lemma 2.1 eine rechte obere Dreiecksmatrix als Inverse. Beides ist nur

möglich, wenn RR_1^{-1} bereits eine Diagonalmatrix ist, sagen wir $RR_1^{-1} = D^{-1}$ oder $R_1 = DR$. Aus $QR = Q_1 DR$ folgt dann $Q_1 = QD^{-1}$. Da Q_1 als unitär vorausgesetzt wurde, gilt $I = Q_1^H Q_1 = D^{-H} Q^H Q D^{-1} = D^{-H} D^{-1}$, also $|d_{ii}| = 1$ und $D^{-1} = D^H$. □

Man kann sich die Aussage (b) in der Form $A = (QD^H)(DR)$ veranschaulichen, in der die Diagonalmatrix D mit $|d_{ii}| = 1$ frei gewählt werden kann. Obwohl die QR-Zerlegung damit nahezu eindeutig bestimmt ist, gibt es zu ihrer Realisierung eine ganze Reihe von numerischen Verfahren, die auf den ersten Blick nichts miteinander zu tun haben.

Aus der linearen Algebra dürfte das Schmidtsche Orthonormalisierungsverfahren bekannt sein. Sei $\{a_1, \ldots, a_n\}$ eine Basis des $\mathbb{C}^n$. Das Verfahren wird gestartet, indem der erste Vektor zu 1 normiert wird,

$$r_{11} = |a_1|, \quad q_1 = \frac{a_1}{r_{11}}.$$

Sei eine Orthonormalbasis $q_1, \ldots, q_{k-1}$ von $\operatorname{span}\{a_1, \ldots, a_{k-1}\}$ bereits konstruiert. Dann setze

$$b_k = a_k - r_{1k} q_1 - \ldots - r_{k-1\,k} q_{k-1} \quad \text{mit } r_{ik} = (a_k, q_i),\ 1 \le i \le k-1,$$

$$r_{kk} = |b_k|, \quad q_k = \frac{b_k}{r_{kk}}.$$

Die Vorschrift stellt sicher, dass b_k (und damit auch q_k) auf den bereits konstruierten q_j senkrecht steht,

$$(b_k, q_j) = (a_k, q_j) - \sum_{i=1}^{k-1} (a_k, q_i)(q_i, q_j) = (a_k, q_j) - (a_k, q_j) = 0.$$

Wir stellen die Spaltenvektoren a_i, q_i zu Matrizen zusammen,

$$A = (a_1 | \ldots | a_n), \quad Q = (q_1 | \ldots | q_n).$$

Da die Vektoren $q_1, \ldots, q_n$ ein Orthonormalsystem bilden, ist die Matrix Q unitär, $Q^H Q = I$. Die r_{ij} bilden eine rechte obere Dreiecksmatrix R, es gilt

$$A = QR \ \text{ oder } \ a_k = \sum_{i=1}^{k} r_{ik} q_i.$$

Wegen r_{ii} reell und positiv, hat diese QR-Zerlegung gerade die Form wie in Lemma 2.7(a). Das Schmidt-Verfahren ist rundungsfehleranfällig und sollte in der gesamten numerischen Mathematik unbedingt vermieden werden.

Eine numerisch sehr stabile QR-Zerlegung bekommt man durch das *Householder-Verfahren*, das im Folgenden beschrieben wird.

Notation Vektoren sind bei uns immer Spaltenvektoren. So kann das Matrix-Vektor-Produkt Ax auch als Matrizenprodukt der $n \times n$-Matrix A mit der $n \times 1$-Matrix x interpretiert werden. Für Vektoren $x, y \in \mathbb{C}^n$ ist $y^H x$ dann ein Skalar wegen „Zeile mal Spalte“

$$y^H x = \sum_{j=1}^{n} \overline{y}_j x_j = (x, y), \quad x^H x = |x|^2.$$

Entsprechend multiplizieren wir bei xy^H die $n \times 1$-Matrix x mit der $1 \times n$-Matrix y^H, es kommt daher die Matrix

$$xy^H = \begin{bmatrix} x_1\overline{y}_1 & \cdots & x_1\overline{y}_n \\ \vdots & & \vdots \\ x_n\overline{y}_1 & \cdots & x_n\overline{y}_n \end{bmatrix}$$

heraus. Das Assoziativgesetz für die Matrizenmultiplikation $(AB)C = A(BC)$ gilt auch, wenn unter den Matrizen Vektoren sind, die dann als $n \times 1$-Matrizen interpretiert werden.

Lemma 2.8 *Sei $w \in \mathbb{C}^n$, $|w| = 1$, und*

$$E = \{x \in \mathbb{C}^n : (x, w) = 0\}$$

sei die Hyperebene der Vektoren, die auf w senkrecht stehen. Dann beschreibt die Matrix

$$P = I - 2ww^H$$

die Spiegelung an E. P ist hermitesch und unitär, daher involutorisch

$$P^2 = PP = P^H P = I.$$

Beweis: Nach dem Projektionssatz können wir eine beliebiges $x \in \mathbb{C}^n$ in der Form $x = z + \lambda w$ zerlegen mit einem $z \in E$. Dann folgt aus $w^H z = 0$ und $w^H w = 1$

$$Px = z + \lambda w - 2w(w^H z) - 2\lambda w(w^H w) = z - \lambda w.$$

Dies ist gerade die behauptete Spiegelungseigenschaft von P. $P = P^H$ folgt sofort aus obiger Darstellung von P. $P^2 = I$ folgt aus der gerade bewiesenen Spiegelungseigenschaft von P, lässt sich aber auch direkt durch

$$\begin{aligned} P^H P = P^2 &= (I - 2ww^H)(I - 2ww^H) \\ &= I - 4ww^H + 4w(w^H w)w^H = I \end{aligned}$$

beweisen. □

Beim Householder-Verfahren wird die Matrix $A \in \mathbb{C}^{n\times n}$ durch Multiplikation mit Spiegelungsmatrizen P auf Dreiecksgestalt gebracht. Für $x \in \mathbb{C}^n$ bestimmen wir daher ein w mit

$$Px = (I - 2ww^H)x = ke_1, \quad |w| = 1,$$

wobei $k \in \mathbb{C}$ und e_1 den ersten kanonischen Einheitsvektor bezeichnet. Dann ist

$$|x|^2 = |Px|^2 = |k|^2$$

und weil $(Px, x) = x^H(x - 2ww^H x) = |x|^2 - 2|w^H x|^2$ reell ist,

$$(Px, x) = (ke_1, x) = k\overline{x}_1 \text{ reell.}$$

Also ist

$$k = \mp e^{i\alpha}\sigma \text{ mit } \sigma = |x| \text{ und } x_1 = e^{i\alpha}|x_1|.$$

Um $Px = ke_1$ zu erreichen müssen wir an der Ebene mit Normaler $x - ke_1$ spiegeln, daher

$$w = \frac{x - ke_1}{|x - ke_1|}.$$

Da durch $|x - ke_1|$ geteilt wird, versuchen wir, das Vorzeichen von k so zu wählen, dass $|x - ke_1|$ möglichst groß wird

$$\begin{aligned} |x - ke_1| &= |x \pm e^{i\alpha}\sigma e_1| \\ &= \sqrt{|x_1 \pm e^{i\alpha}\sigma|^2 + |x_2|^2 + \ldots + |x_n|^2} \\ &\overset{x_1 = e^{i\alpha}|x_1|}{=} \sqrt{||x_1| \pm \sigma|^2 + |x_2|^2 + \ldots + |x_n|^2} \end{aligned}$$

Wir wählen daher $k = -e^{i\alpha}\sigma$. Dann ist

$$|x_1 - k|^2 = \big||x_1| + \sigma\big|^2 = \sigma^2 + 2\sigma|x_1| + |x_1|^2,$$
$$|x - ke_1|^2 = |x_1 - k|^2 + \sigma^2 - |x_1|^2 = 2\sigma^2 + 2\sigma|x_1|,$$
$$2ww^H = 2\frac{(x - ke_1)(x - ke_1)^H}{|x - ke_1|^2}.$$

Zusammengefasst verwenden wir die folgenden Formeln

$$P = I - \beta uu^H, \quad u \in \mathbb{C}^n, \quad \beta \in \mathbb{R},$$

mit

$$\sigma = |x|, \quad x_1 = e^{i\alpha}|x_1|, \quad k = -\sigma e^{i\alpha},$$
$$u = x - ke_1 = \begin{pmatrix} e^{i\alpha}(|x_1| + \sigma) \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad \beta = (\sigma(\sigma + |x_1|))^{-1}.$$

Wir konstruieren schrittweise solche Householdermatrizen P_j mit

$$A^{(j)} = P_j A^{(j-1)},$$

so dass

$$A^{(n-1)} = R = \begin{bmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \end{bmatrix}.$$

Sei $a_1^{(0)}$ die erste Spalte von $A = A^{(0)}$. Konstruiere P_1 mit

$$P_1 a_1^{(0)} = ke_1.$$

Nach $j - 1$ Schritten erhalte

$$A^{(j-1)} = \left[\begin{array}{ccc|ccc} * & \cdots & * & * & \cdots & * \\ & \ddots & \vdots & \vdots & & \vdots \\ 0 & & * & * & \cdots & * \\ \hline & & & a_{jj}^{(j-1)} & \cdots & a_{jn}^{(j-1)} \\ & 0 & & \vdots & & \vdots \\ & & & a_{jn}^{(j-1)} & \cdots & a_{nn}^{(j-1)} \end{array}\right]$$

Bestimme zunächst die Householder-Matrix $\tilde{P}_j \in \mathbb{C}^{(n-j+1)\times(n-j+1)}$ mit

$$\tilde{P}_j \begin{pmatrix} a_{jj}^{(j-1)} \\ \vdots \\ a_{nj}^{(j-1)} \end{pmatrix} = k \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{C}^{n-j+1}.$$

Erweitere $\tilde{P}_j$ zu einer $n \times n$-Matrix durch

$$P_j = \begin{bmatrix} I_{j-1} & 0 \\ 0 & \tilde{P}_j \end{bmatrix}.$$

Mit $P = P_{n-1} \dots P_1$ gilt dann

$$PA = R \quad \Rightarrow \quad A = P^{-1}R =: QR \text{ mit } Q = P^{-1} = P^H.$$

Ist A reell, so ist $k = \sigma$ für $x_1 \leq 0$ und $k = -\sigma$ für $x_1 > 0$. In diesem Fall ist die gesamte Zerlegung reell.

Da die die Komponenten $u_{j,i}$ der u_j für $i = 1, \dots, j-1$ verschwinden, können wir sie auf die linke untere Dreiecksmatrix von A schreiben, die Diagonale von R wird separat gespeichert, ansonsten kommt R auf die rechte obere Dreiecksmatrix von A. Für die Speicherung von β_j benötigen wir einen weiteren Hilfsvektor. $P_1 A$ berechnet man durch

$$P_1 A = A - u_j y_j^H \quad \text{mit } y_j^H = \beta_j u_j^H A,$$

für $P_j A$ geht man analog vor. An dieser Darstellung sieht man schon, dass das Householder-Verfahren teurer ist als die Gauß-Elimination. Für $P_1 A$ benötigt man im Wesentlichen zwei Matrix-Vektor Multiplikationen, bei der Gauß-Elimination kommt man mit einer aus. Im j-ten Schritt brauchen wir für die Elimination $O((n+1-j)^2)$ Operationen, asymptotisch erhalten wir den gleichen Gesamtaufwand wie bei der Gauß-Elimination, nämlich $O(n^3)$ Operationen.

Das lineare Gleichungssystem $Ax = b$ löst man durch

$$Rx = P_{n-1} \dots P_1 b$$

mit

$$P_1 b = b - \beta_1 u_1 u_1^H b = b - \beta_1 u_1 (u^H b).$$

2.4 Welches Verfahren ist vorzuziehen? Wenn das Gleichungssystem positiv-definit ist, wird man das Cholesky-Verfahren allein schon wegen seinen guten Stabilitätseigenschaften vorziehen. Da es außerdem die Symmetrie der Matrix ausnutzt, kommt es mit $\frac{1}{3}n^3 + O(n^2)$ Gleitpunktoperationen aus im Gegensatz zur Gauß-Elimination, bei der $\frac{2}{3}n^3 + O(n^2)$ Operationen anfallen. In dieser Hinsicht ist das Householder-Verfahren mit $\frac{4}{3}n^3 + O(n^2)$ am schlechtesten.

Für die Kondition

$$\operatorname{cond}(A) = \|A\| \, \|A^{-1}\|$$

gilt im Householder-Verfahren wegen $A = QR$

$$\operatorname{cond}(A) = \operatorname{cond}(QR) = \operatorname{cond}(R).$$

Bei exakter Ausführung des Householder-Verfahrens ändert sich im Gegensatz zur Gauß-Zerlegung die Kondition nicht. Dem steht die doppelte Rechenzeit für das Householder-Verfahren gegenüber. In der Praxis wird daher die Gauß-Elimination vorgezogen.

3 Metrische und normierte Räume

3.1 Metrische Räume Mit der metrischen Struktur wird der aus dem $\mathbb{R}^n$ bekannte Abstandsbegriff abstrahiert. Wir können uns einen metrischen Raum als eine Punktmenge vorstellen, in der Entfernungen zwischen den Punkten definiert sind, die den folgenden plausiblen Bedingungen genügen müssen.

Sei X eine Menge. Eine Abbildung $d : X \times X :\to [0, \infty)$ heißt *Metrik* auf X, wenn die folgenden Bedingungen erfüllt sind.

(i) $d(x, y) = 0 \;\Leftrightarrow\; x = y$,

(ii) $d(x, y) = d(y, x)$ (Symmetrie),

(iii) $d(x, z) \le d(x, y) + d(y, z)$ (Dreiecksungleichung).

Das Paar (X, d) heißt dann *metrischer Raum*.

Aus der Definition der Metrik folgt die *Viereckungleichung*

$$|d(x, y) - d(x', y')| \le d(x, x') + d(y, y'), \tag{3.1}$$

denn die Dreiecksungleichung liefert

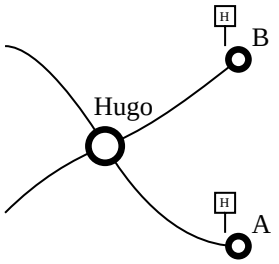
$$d(x, y) \le d(x, x') + d(x', y') + d(y, y'), \quad d(x', y') \le d(x, x') + d(x, y) + d(y, y'),$$

womit (3.1) gezeigt ist.

Beispiele (i) Der $\mathbb{K}^n$ mit $\mathbb{K} = \mathbb{R}$ oder $\mathbb{K} = \mathbb{C}$ mit der Metrik $d(x, y) = |x - y|$ ist metrischer Raum.

(ii) Jede Teilmenge eines metrischen Raumes ist mit der gleichen Abstandsfunktion selber ein metrischer Raum.

(iii) (Erlanger Metrik) Um in Erlangen mit dem Bus von einem Ortsteil in den benachbarten zu kommen (Fußweg 5'), muss man zuerst zum zentralen Busbahnhof fahren, dort umsteigen und dann im wesentlichen die gleiche Strecke wieder zurückfahren. Diese Metrik, die nicht nur bei den Mathematikern, sondern auch bei den Benutzern des öffentlichen Nahverkehrs immer wieder neu auf große Begeisterung stößt, kann folgendermaßen abstrakt definiert werden: Grundraum ist der $\mathbb{R}^2$ mit dem Ursprung als ausgezeichneten Punkt, die Metrik ist



$$d(x, y) = \begin{cases} |x - y| & \text{wenn } x = \lambda y \text{ für ein } \lambda \in \mathbb{R}, \\ |x| + |y| & \text{sonst.} \end{cases}$$

Der Beweis der Dreiecksungleichung macht einige Fallunterscheidungen notwendig, ist ansonsten trivial. Ich habe übrigens diese Metrik unter dem Namen „Französische Eisenbahn Metrik“ kennengelernt (fährt immer über Paris), in einem amerikanischen Buch wird sie „Washington D.C. Metrik“ genannt. Die Probleme sind also überall die gleichen.

Eine Folge (x_k) heißt *konvergent*, wenn es zu jedem $\varepsilon > 0$ ein $K \in \mathbb{N}$ gibt mit

$$d(x_k, x) < \varepsilon \quad \text{für alle } k \ge K.$$

Im Falle $X = \mathbb{R}^n$ bekommen wir unseren wohlbekannten Konvergenzbegriff zurück.

Eine Folge (x_k) im metrischen Raum X heißt *Cauchy-Folge*, wenn es zu jedem $\varepsilon > 0$ ein $K \in \mathbb{N}$ gibt mit

$$d(x_k, x_l) < \varepsilon \quad \text{für alle } k, l \ge K.$$

X heißt *vollständig*, wenn jede Cauchy-Folge gegen ein $x \in X$ konvergiert.

Jede konvergente Folge ist Cauchy-Folge, denn aus $d(x_k, x) < \varepsilon$, $d(x_l, x) < \varepsilon$ für alle $k, l \geq K$ folgt mit der Dreiecksungleichung $d(x_k, x_l) < 2\varepsilon$.

Beispiele (i) Der metrische Raum $\mathbb{Q}$ mit der Standardmetrik $d(x, y) = |x - y|$ ist unvollständig. Als (x_k) wählen wir eine beliebige Folge in $\mathbb{Q}$, die gegen ein Element $x \in \mathbb{R} \setminus \mathbb{Q}$ konvergiert. Diese Folge ist Cauchy-Folge in $\mathbb{R}$, weil sie dort konvergiert, somit auch eine Cauchy-Folge in $\mathbb{Q}$, die aber in $\mathbb{Q}$ keinen Grenzwert besitzt.

(ii) $\mathbb{R}$ mit der Standardmetrik $d(x, y) = |x - y|$ ist ein vollständiger metrischer Raum. Denn wenn (x_k) eine Cauchy-Folge in $\mathbb{R}$ ist, so ist diese beschränkt und enthält eine konvergente Teilfolge $x_{k_m} \to x$. Da der Betrag stetig ist, können wir in $|x_{k_m} - x_l| < \varepsilon$ zum Grenzwert übergehen und erhalten $|x - x_l| \leq \varepsilon$ für alle $l \geq K$. Damit ist die ganze Folge gegen x konvergent.

(iii) Der $\mathbb{R}^n$ ist ebenfalls vollständig unter der Standardmetrik. Denn die Komponenten einer Cauchy-Folge des $\mathbb{R}^n$ bilden selber eine Cauchy-Folge in $\mathbb{R}$. Mit $\mathbb{R}$ ist daher auch der $\mathbb{R}^n$ vollständig.

3.2 Der Banachsche Fixpunktsatz Für eine Abbildung $T : X \to X$ in einem metrischen Raum X möchten wir die *Fixpunktgleichung*

$$T\bar{x} = \bar{x} \tag{3.2}$$

mit Hilfe des einfachsten Verfahrens, der *sukzessiven Approximation*

$$x_{k+1} = Tx_k, \quad x_0 \in X \text{ vorgegeben}, \tag{3.3}$$

lösen. Da schon einfachste Beispiele im $\mathbb{R}^1$ zeigen, dass dieses Verfahren auch bei Existenz eines Fixpunktes nicht konvergieren muss, benötigen wir einschränkende Voraussetzungen an die Abbildung T.

Seien (X, d_x) und (Y, d_y) metrische Räume. $T : X \to Y$ heißt *lipschitzstetig*, wenn es ein $L \in \mathbb{R}_+$ gibt mit

$$d_y(Tx_1, Tx_2) \leq L d_x(x_1, x_2) \quad \text{für alle } x_1, x_2 \in X.$$

L heißt dann *Lipschitzkonstante*. Wenn $X = Y$ und $L < 1$ in dieser Abschätzung gewählt werden kann, so heißt T *Kontraktion*.

Satz [Banachscher Fixpunktsatz] Sei (X, d) ein vollständiger metrischer Raum und $T : X \to X$ eine Kontraktion. Dann besitzt T genau einen Fixpunkt $\bar{x}$ und die Folge der sukzessiven Approximation (3.3) konvergiert für alle Startwerte $x_0 \in X$ gegen $\bar{x}$. Weiter gilt die Fehlerabschätzung

$$d(x_k, \bar{x}) \leq \frac{L^k}{1 - L} d(x_0, Tx_0).$$

Beweis: Es gibt höchstens einen Fixpunkt, denn für Fixpunkte $\bar{x}, \bar{y} \in X$ folgt

$$d(\bar{x}, \bar{y}) = d(T\bar{x}, T\bar{y}) \leq L d(\bar{x}, \bar{y})$$

und damit $d(\bar{x}, \bar{y}) = 0$ und $\bar{x} = \bar{y}$.

Aus (3.3) erhalten wir

$$d(x_k, x_{k+1}) = d(Tx_{k-1}, Tx_k) \leq L d(x_{k-1}, x_k) \leq L^k d(x_0, Tx_0)$$

und aus der Dreiecksungleichung

$$d(x_k, x_{k+l}) \leq \sum_{i=0}^{l-1} d(x_{k+i}, x_{k+i+1}) \leq \sum_{i=0}^{l-1} L^{k+i} d(x_0, Tx_0) = \frac{L^k - L^{k+l}}{1 - L} d(x_0, Tx_0).$$

Damit ist (x_k) Cauchy-Folge und konvergiert wegen der Vollständigkeit von X gegen ein $x \in X$. Aufgrund der Stetigkeit von T kann man in der Gleichung (3.3) zum Grenzwert $k \to \infty$ gehen und erhält $x = Tx$. Die Fehlerabschätzung ergibt sich aus der letzten Ungleichung für $l \to \infty$. □

3.3 Der Banachsche Fixpunktsatz im $\mathbb{R}^n$ In diesem Abschnitt betrachten wir $X = D \subset \mathbb{R}^n$ versehen mit der Standardmetrik $d(x, y) = |x - y|$. Damit ist (X, d) ein metrischer Raum, die Frage ist aber, ob $(\mathbb{R}^n, d)$ seine Vollständigkeit auf (D, d) vererbt.

Satz (D, d) ist genau dann vollständig, wenn D abgeschlossen ist.

Beweis: Sei (x_k) eine Cauchy-Folge in D. Diese ist auch eine Cauchy-Folge im $\mathbb{R}^n$ und hat daher einen Grenzwert $x \in \mathbb{R}^n$. Ist D abgeschlossen, so gilt $x \in D$, denn andernfalls wäre x innerer Punkt von D^c und $x_k \to x$ nicht möglich. Für die andere Richtung nimmt man eine Cauchy-Folge, die gegen einen Berührpunkt $x \notin D$ konvergiert. □

Wir wollen uns nun der Frage zuwenden, wann im Falle $X = D \subset \mathbb{R}^n$ eine Funktion einer Lipschitzbedingung genügt. Ein Gebiet D des $\mathbb{R}^n$ heißt *konvex*, wenn zu je zwei Punkten $x, y \in D$ auch die Verbindungsstrecke $\overline{xy}$ zu D gehört.

Lemma Sei $D \subset \mathbb{R}^n$ ein konvexes Gebiet. Dann ist jede in D stetig differenzierbare Funktion $f : D \to \mathbb{R}^m$ mit beschränkter Ableitung $\sup_{x\in\Omega} |f'(x)| = M$ lipschitzstetig in D mit Lipschitzkonstante M,

$$|f(x) - f(y)| \le M|x - y| \quad \text{für alle } x, y \in D. \tag{3.4}$$

Beweis: Für $x, y \in D$ erhalten wir aus dem Mittelwertsatz

$$f(x) - f(y) = \int_0^1 f'(tx + (1 - t)y))(x - y)\, dt.$$

In dieser Gleichung setzen wir Beträge, verwenden $|Ax| \le |A|\,|x|$ und schätzen $|f'|$ durch M ab. □

Eine differenzierbare Funktion mit unbeschränkter Ableitung ist nicht lipschitzstetig, wie z.B. die eindimensionale Funktion $f(x) = x^2$ mit $f(x) - f(y) = (x + y)(x - y)$ zeigt.

Beispiel Sei

$$f(x) = \frac{1}{2}x + \frac{1}{8}(x - 1)^2 + \frac{1}{2}.$$

Wir zeigen, dass f den vollständigen metrischen Raum $[0, 2]$ auf sich abbildet und dort eine Kontraktion ist. Es gilt $f(0) = \frac{5}{8}$ und $f(2) = 1 + \frac{1}{8} + \frac{1}{2} < 2$. Ferner ist $f'(x) = \frac{1}{2} + \frac{1}{4}(x - 1)$ mit Nullstelle $x = 3$. Damit besitzt f im Intervall $[0, 2]$ keine Extremwerte und bildet das Intervall $[0, 2]$ auf sich ab. Für die Ableitung gilt $f'(0) = \frac{1}{2}$, $f'(2) = \frac{3}{4}$. Ferner ist $f'' = \frac{1}{4}$. Damit besitzt auch f' keine Extremwerte im Intervall $[0, 2]$ und es gilt $|f'(x)| \le \frac{3}{4}$. Damit ist f eine Kontraktion. Der vom Banachschen Fixpunktsatz garantierte Fixpunkt ist $\overline{x} = 1$.

3.4 Banach Räume Sei X ein nicht notwendig endlich dimensionaler linearer Vektorraum über $\mathbb{K} = \mathbb{R}$ oder $\mathbb{K} = \mathbb{C}$. Eine Abbildung $\|\cdot\| : X \to \mathbb{R}$ heißt *Norm*, wenn sie den folgenden Bedingungen genügt:

(i) $\|x\| \ge 0$ und $\|x\| = 0 \Leftrightarrow x = 0$

(ii) $\|\alpha x\| = |\alpha|\,\|x\|$ für alle $\alpha \in \mathbb{K}$ und $x \in X$.

(iii) $\|x + y\| \le \|x\| + \|y\|$.

Da wir diese Axiome für den $\mathbb{R}^n$ mit $\|\cdot\| = |\cdot|$ nachgewiesen haben, ist er ein normierter Raum. Auch die Schlußfolgerungen, die wir aus diesen Axiomen gezogen haben, bleiben mit gleichem Beweis für allgemeine normierte Räume gültig, nämlich die umgekehrte Dreiecksgleichung

$$\big|\|x\| - \|y\|\big| \le \|x - y\|.$$

sowie die eigentliche Dreiecksungleichung

$$\|x - z\| \le \|x - y\| + \|y - z\|.$$

Aus den Normaxiomen folgt sofort, dass

$$d(x, y) = \|x - y\|_X$$

eine Metrik auf X ist. Jeder normierte Raum ist damit auch ein metrischer Raum, sodass alle Begriffsbildungen aus den letzten Abschnitten verwendet werden können. Insbesondere heißt eine Folge (x_k) in X *Cauchy-Folge*, wenn es zu jedem $\varepsilon > 0$ ein $K \in \mathbb{N}$ gibt mit

$$\|x_k - x_l\| < \varepsilon \quad \text{für alle } k, l \geq K.$$

Der normierte Raum heißt *vollständig*, wenn jede Cauchy-Folge einen Grenzwert besitzt, wenn also ein $x \in X$ existiert mit $\|x_k - x\| < \varepsilon$ für alle $k \geq K(\varepsilon)$. Ein vollständiger normierter Raum heißt *Banach Raum*. Die Räume $\mathbb{R}^n$ und $\mathbb{C}^n$ mit der Norm $\|x\|_X = |x|$ sind demnach nicht nur vollständige metrische Räume, sondern auch Banach Räume.

Interessanter als die genannten endlich dimensionalen Räume sind die Funktionenräume, von denen einer vorgestellt werden soll.

Satz Sei $D \subset \mathbb{R}^n$ eine kompakte Menge. Dann ist der Raum

$$C(D)^m = \{f : D \to \mathbb{R}^m : f \text{ stetig auf } D\}$$

mit der Norm

$$\|f\|_\infty = \max_{x \in D} |f(x)|$$

ein Banach Raum.

Beweis: Ist f stetig, so ist auch die reellwertige Funktion $|f|$ stetig und nimmt nach einem bekannten Satz das Maximum an. Die Norm ist daher auf $C(D)^m$ wohldefiniert. Die Normaxiome lassen sich leicht überprüfen, die Dreiecksungleichung folgt aus

$$\|f + g\|_\infty = \max_{x \in D} |f(x) + g(x)| \leq \max_{x \in D} \big(|f(x)| + |g(x)|\big)$$

$$\leq \max_{x \in D} |f(x)| + \max_{x \in D} |g(x)| = \|f\|_\infty + \|g\|_\infty.$$

Nun zeigen wir die Vollständigkeit des Raumes. Sei (f_k) eine Cauchy-Folge in $C(D)^m$, also

$$|f_k(x) - f_l(x)| < \varepsilon \quad \text{für alle } k, l \geq K \text{ und für alle } x \in D. \tag{3.5}$$

Insbesondere gilt $|f_k(x) - f_l(x)| < \varepsilon$ für jedes $x \in D$. Damit sind auch die Folgen $(f_k(x))$ Cauchy-Folgen des $\mathbb{R}^m$ und besitzen einen Grenzwert, den wir $f(x)$ nennen. Da der Absolutbetrag stetig ist, können wir in (3.5) zum Grenzwert $l \to \infty$ übergehen und erhalten

$$|f_k(x) - f(x)| \leq \varepsilon \quad \text{für alle } k \geq K \text{ und für alle } x \in D.$$

Damit konvergiert (f_k) gegen f in der Norm von $C(D)^m$, insbesondere ist f als *gleichmäßiger* Grenzwert der Folge (f_k) stetig. Damit ist $f \in C(D)^m$ gezeigt. □

4 Nichtlineare Gleichungssysteme

4.1 Nichtlineare Gleichungssysteme und Optimierungsprobleme Sei $f : D \to \mathbb{R}^n$ mit $D \subset \mathbb{R}^n$ ein Gebiet. Gesucht ist dann ein $x \in D$ mit

$$f(x) = 0 \quad \text{oder} \quad f^i(x_1, \ldots, x_n) = 0 \ \text{ für } i = 1, \ldots, n.$$

Solche nichtlinearen Gleichungssysteme entstehen häufig aus Optimierungsproblemen. Im einfachsten Fall ist eine Funktion $f : D \to \mathbb{R}$ in $D \subset \mathbb{R}^n$ zu minimieren. Statt die Minimumaufgabe direkt anzugehen, sucht man oft besser Lösungen der notwendigen Bedingungen, also

$$\nabla f(x) = 0,$$

was wiederum die Lösung eines $n \times n$-Gleichungssystems bedeutet. Ist beispielsweise f strikt konvex, so besitzt das System höchstens eine Lösung, die dann auch Lösung des Optimierungsproblems ist.

Optimierung mit Gleichungsrestriktionen Seien $f : D \to \mathbb{R}$ und $g : D \to \mathbb{R}^m$, $m \leq n$, mit $D \subset \mathbb{R}^n$ ein Gebiet gegeben. Gesucht ist dann ein $x \in D$ mit

$$f(x) \to \text{Min} \quad \text{in } D \text{ unter der Bedingung } g(x) = 0. \tag{4.1}$$

Auch dies lässt sich in die Form eines Gleichungssystems bringen.

Satz Seien f, g einmal stetig differenzierbar und x_0 eine Lösung von (4.1) mit $\operatorname{rang} \nabla g(x_0) = m$. Dann gibt es *Lagrange-Multiplikatoren* $\lambda_1, \ldots \lambda_m \in \mathbb{R}$, die das *Lagrange-Funktional*

$$\mathcal{L}(x_0, \lambda) = f(x_0) + \sum_{i=1}^{m} \lambda_i g^i(x_0), \quad \mathcal{L} : \mathbb{R}^{n+m} \to \mathbb{R},$$

stationär machen, also $\nabla_x \mathcal{L}(x_0, \lambda) = 0$, $\nabla_\lambda \mathcal{L}(x_0, \lambda) = 0$, oder

$$\begin{aligned} \nabla_x f(x_0) + \sum_{i=1}^{m} \lambda_i \nabla_x g^i(x_0) &= 0, \\ g^i(x_0) &= 0, \quad i = 1, \ldots, m. \end{aligned}$$

Dies ist ein Gleichungssystem der Dimension $m + n$.

Beweis: Wegen $\operatorname{rang} \nabla_x g(x_0) = m$ können wir ohne Beschränkung der Allgemeinheit annehmen, dass die ersten m Spalten von

$$\nabla g = \begin{bmatrix} \partial_1 g^1 & \cdots & \partial_n g^1 \\ \vdots & & \vdots \\ \partial_1 g^m & \cdots & \partial_n g^m \end{bmatrix}$$

für $x = x_0$ linear unabhängig sind. Schreibe daher

$$g(y_1, \ldots, y_m, \tilde{x}_1, \ldots, \tilde{x}_{n-m}),$$

so dass $\nabla_y g(y_0, \tilde{x}_0)$ regulär ist. Nach dem Satz über implizite Funktionen ist $y(\tilde{x})$ in Umgebung von $\tilde{x}_0$ definiert mit

$$g(y(\tilde{x}), \tilde{x}) = 0$$

und

$$\nabla_{\tilde{x}} y = -\nabla_y g^{-1} \nabla_{\tilde{x}} g. \tag{4.2}$$

Die Funktion

$$F(\tilde{x}) = f(y(\tilde{x}), \tilde{x})$$

besitzt in $\tilde{x}_0$ ein lokales Minimum, also

$$0 = \nabla_{\tilde{x}} F = \nabla_y f \nabla_{\tilde{x}} y + \nabla_{\tilde{x}} f \quad \text{für } \tilde{x} = \tilde{x}_0.$$

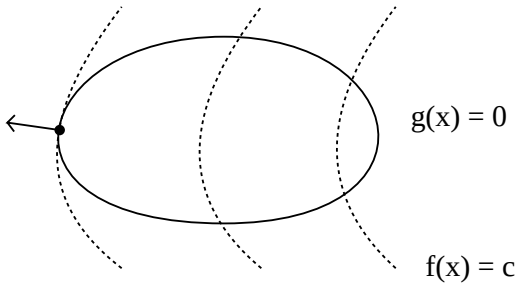
Mit (4.2) folgt

$$-\nabla_y f \nabla_y g^{-1} \nabla_{\tilde{x}} g + \nabla_{\tilde{x}} f = 0 \quad \text{für } \tilde{x} = \tilde{x}_0.$$

Mit $(\lambda_1, \ldots, \lambda_m) = \lambda^T = -\nabla_y f \nabla_y g^{-1}$ erhalten wir die Behauptung

$$\lambda^T \nabla_y g = -\nabla_y f, \quad \lambda^T \nabla_{\tilde{x}} g + \nabla_{\tilde{x}} f = 0.$$

□



Der Fall $n = 2$ und $m = 1$ lässt sich schön an obiger Skizze veranschaulichen. Das Minimum befindet sich offenbar an der Stelle, wo eine Höhenlinie von f die Kurve $g(x) = 0$ berührt. Da der Gradient senkrecht zur Höhenlinie gerichtet ist, muss der Gradient von f ein Vielfaches des Gradienten von g sein.

Beispiel Sei $n = 1$ und $f(x) = x$, $g(x) = x^2$. Die Lösung von (4.1) ist dann offenbar $x_0 = 0$, aber es gibt kein λ, das die notwendige Bedingung

$$f'(0) + \lambda g'(0) = 1 \neq 0$$

löst. Die Bedingung $\operatorname{rang} g'(x_0) = 1$ ist hier verletzt. Wir können $g(x) = 0$ durch die hier äquivalente Bedingung $\tilde{g}(x) = x = 0$ ersetzen, wodurch $\operatorname{rang} g'(x) = 1$ erfüllt ist. Wir sagen dazu: x_0 ist regulär parametrisiert.

4.2 Allgemeine Iterationsverfahren Sei $f : \mathbb{R}^n \to \mathbb{R}^n$ und $f(\xi) = 0$. Zur Bestimmung von ξ verwende eine Iterationsvorschrift der Form

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

mit der *Iterationsfunktion* $\phi : \mathbb{R}^n \to \mathbb{R}^n$. Unter der Voraussetzung, dass ϕ stetig ist, folgt aus $x^k \to \xi$, dass

$$\xi = \lim x^{k+1} = \lim \phi(x^k) = \phi(\xi),$$

also muss ξ ein Fixpunkt von ϕ sein. Die Idee solcher Iterationsverfahren besteht nun darin, die Funktion f zu ersetzen durch Funktionen, deren Nullstellen sich leicht bestimmen lassen.

Beispiele (i) Für eine Funktion $f \in C^2$ können wir für einen Start x die Funktion $f(y)$ in Umgebung von x ersetzen durch die Tangentialebene $T(y)$ im Punkt $(x, f(x))$. In erster Näherung ist dann

$$T(y) = f(x) + \nabla f(x)(y - x) \stackrel{!}{=} 0$$

zu lösen mit Nullstelle

$$y = x - \nabla f(x)^{-1} f(x) =: \phi(x).$$

Damit erhalten wir das *Newton-Verfahren*

$$x^{k+1} = x^k - \nabla f(x^k)^{-1} f(x^k).$$

(ii) Für $n = 1$ kann man die gleiche Idee mit einer genaueren quadratischen Approximation durchführen

$$T_2(y) = f(x) + f'(x)(y-x) + \frac{1}{2} f''(x)(y-x)^2 \stackrel{!}{=} 0.$$

Bei jedem Iterationsschritt muss eine quadratische Gleichung gelöst werden.

(iii) Wenn x_0 sich schon in einer Umgebung der Nullstelle ξ befindet, so ist auch

$$\tilde{T}(y) = f(x) + \nabla f(x_0)(y-x)$$

eine gute Näherung für $f(y)$. Wir erhalten damit das *vereinfachte Newton-Verfahren*

$$x^{k+1} = x^k - \nabla f(x_0)^{-1} f(x^k),$$

wobei x_0 eine alte Iterierte ist. In diesem Fall kann die LR-Zerlegung von $\nabla f(x_0)$ öfter verwendet werden.

Sei $\phi : \mathbb{R}^n \to \mathbb{R}^n$ eine stetige Iterationsfunktion mit $\phi(\xi) = \xi$. Das Verfahren $x^{k+1} = \phi(x^k)$ heißt *von der Ordnung* $p \geq 1$, wenn in einer Umgebung $U(\xi)$ gilt

$$|\phi(x) - \xi| \leq \overline{c} |x - \xi|^p \quad \forall x \in U(\xi)$$

mit

$$\overline{c} \begin{cases} \text{beliebig} & \text{für } p > 1 \\ < 1 & \text{für } p = 1 \end{cases}.$$

Satz Jedes Verfahren der Ordnung $p \geq 1$ ist lokal konvergent, d.h. es gibt eine Umgebung $V(\xi)$, so dass für

$$x^{k+1} = \phi(x^k), \quad x^0 \in V(\xi)$$

gilt $x^k \in V(\xi)$ für alle k und $x^k \to \xi$.

Beweis: Für $p = 1$ wähle $\varepsilon > 0$ mit $B_\varepsilon(\xi) \subset U(\xi)$. Dann

$$|x^{k+1} - \xi| \leq \overline{c} |x^k - \xi| \leq \overline{c}^{k+1} |x^0 - \xi|.$$

Wegen $\overline{c} < 1$ folgt daraus sowohl $x^k \in B_\varepsilon(\xi)$ als auch $x^k \to \xi$.

Für $p > 1$ wähle zusätzlich ε so klein, dass

$$\overline{c} \varepsilon^{p-1} < 1.$$

Für $x_0 \in B_\varepsilon(\xi)$ folgt dann

$$|x^1 - \xi| \leq \overline{c} |x^0 - \xi|^p \leq \overline{c} \varepsilon^p < \varepsilon$$

daher $x^1 \in B_\varepsilon(\xi)$ sowie $x^k \to \xi$. □

Beispiel Sei $n = 1$. Für $\phi \in C^p$ folgt

$$x^{k+1} = \phi(x^k) = \phi(\xi) + \phi'(\xi)(x^k - \xi) + \ldots + \frac{1}{p!} \phi^{(p)}(\xi)(x^k - \xi)^p + o(|x^k - \xi|^p).$$

Wegen $\phi(\xi) = \xi$ ist das Verfahren genau dann von der Ordnung p, wenn

$$\phi'(\xi) = \ldots = \phi^{(p-1)}(\xi) = 0 \quad \text{für } p > 1$$

beziehungsweise

$$|\phi'(\xi)| < 1 \quad \text{für } p = 1.$$

Für das einfache Verfahren

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

liegt daher nur dann lokale Konvergenz erster Ordnung vor, wenn die sehr einschränkende Bedingung

$$|1 - f'(\xi)| < 1$$

erfüllt ist.

4.3 Das Newton-Verfahren Sei $f : \mathbb{R}^n \to \mathbb{R}^n$, $f \in C^2(\mathbb{R}^n)^n$, mit $f(\xi) = 0$. Das Newton-Verfahren ist dann definiert durch die Vorschrift

$$x^{k+1} = x^k - \nabla f(x^k)^{-1} f(x^k).$$

Bei jeder Iteration muss ∇f bestimmt und ein lineares Gleichungssystem für die *Korrektur* $y = x^{k+1} - x^k$ gelöst werden, nämlich $\nabla f(x^k)y = -f(x^k)$. Die Inverse von $\nabla f(x^k)$ wird natürlich *nicht* berechnet.

Mit der Iterationsfunktion $\phi(x) = x - \nabla f(x)^{-1} f(x)$ gilt dann $\phi(\xi) = \xi$. Wir wollen zeigen, dass das Newton-Verfahren lokal von zweiter Ordnung konvergent ist, sofern $\nabla f(\xi)$ regulär ist. Das im $\mathbb{R}^1$ formulierte Beispiel aus dem letzten Beispiel gilt sinngemäß auch in höheren Dimensionen n, dass also ein Verfahren quadratisch konvergent ist, wenn die erste Ableitung von ϕ in ξ verschwindet. Da die erste Ableitung von ϕ recht mühsam zu bestimmen ist, gehen wir im Beweis des folgenden Satzes etwas anders vor.

Satz Sei $f : \mathbb{R}^n \to \mathbb{R}^n$ mit $f(\xi) = 0$.

(a) Wenn $f \in C^2$ und $\nabla f(\xi)$ regulär ist, so ist das Newton-Verfahren lokal quadratisch konvergent.

(b) Wenn in (a) statt $f \in C^2$ nur $f \in C^1$ erfüllt ist, so ist das Newton-Verfahren lokal linear konvergent.

(c) Wenn f genügend oft differenzierbar ist und $\nabla f(x)$ regulär für $x \in U(\xi) \setminus \{\xi\}$ ist, so konvergiert das Newton-Verfahren lokal ebenfalls.

Beweis: (a) Taylorentwicklung liefert

$$0 = f(\xi) = f(x^k) + \nabla f(x^k)(\xi - x^k) + O(|\xi - x^k|^2).$$

Wir lösen dies nach $x^k - \xi$ auf

$$x^k - \xi = \nabla f(x^k)^{-1} f(x^k) + \nabla f(x^k)^{-1} O(|\xi - x^k|^2)$$

und setzen für x^k die Newton-Gleichung $x^k = x^{k+1} + \nabla f(x^k)^{-1} f(x^k)$ ein

$$x^{k+1} - \xi = \nabla f(x^k)^{-1} O(|\xi - x^k|^2).$$

Ist $|\nabla f^{-1}(\xi)| \le M$, so gibt es ein $B_r(\xi)$, $r > 0$, mit $|\nabla f(x)^{-1}| \le 2M$ für alle $x \in B_r(\xi)$. In $\overline{B_r(\xi)}$ sind die zweiten Ableitungen beschränkt, $|f_{xx}(x)| \le K$. Mit einer Konstanten $c(n)$, die nur von der Raumdimension abhängt, folgt dann

$$|x^{k+1} - \xi| \le 2Mc(n)K|x^k - \xi|^2 = c|x^k - \xi|^2.$$

Bei (b) geht man genauso vor.

Teil (c) beweisen wir nur für $n = 1$. Sei ξ eine l-fache isolierte Nullstelle von f, also

$$f(\xi) = f'(\xi) = \ldots = f^{(l-1)}(\xi) = 0, \quad f^{(l)}(\xi) \neq 0.$$

Setze

$$\phi(x) = \begin{cases} x - \frac{f(x)}{f'(x)}, & x \neq \xi \\ \xi, & x = \xi \end{cases}.$$

Mit

$$f(x) = \frac{1}{l!} f^{(l)}(\xi)(x-\xi)^l + R(x), \quad |R(x)| \leq c|x-\xi|^{l+1}$$

folgt

$$\frac{f(x)}{f'(x)} = \frac{\frac{1}{l!} f^{(l)}(\xi)(x-\xi)^l + R(x)}{\frac{1}{(l-1)!} f^{(l)}(\xi)(x-\xi)^{(l-1)} + R'(x)} = \frac{1}{l}(x-\xi) + T(x), \quad |T(x)| \leq c|x-\xi|^2.$$

Damit ist ϕ stetig und wegen

$$\frac{d}{dx}\Big(\frac{f(x)}{f'(x)}\Big)\Big|_{x=\xi} = \frac{1}{l}$$

gilt

$$\phi'(\xi) = 1 - \frac{1}{l}.$$

Das Newton-Verfahren ist also linear konvergent, der Konvergenzfaktor wird allerdings mit zunehmender Vielfachheit der Nullstelle immer schlechter. □

Bemerkung Man erhält die quadratische Konvergenz des Newton-Verfahrens auch unter der schwächeren Bedingung $f \in C^1(\mathbb{R}^n)^n$ mit lokal lipschitzstetiger Ableitung, d.h. es gibt eine Umgebung U von ξ und eine Konstante L mit

$$|\nabla f(x) - \nabla f(y)| \leq L|x-y| \quad \text{für alle } x, y \in U.$$

4.4 Das Newton-Verfahren im Komplexen Sei $f : \mathbb{C} \to \mathbb{C}$ komplex differenzierbar, d.h. die Komponenten in $f(z) = f^1(x,y) + if^2(x,y)$, $z = x + iy$, sind reell differenzierbar und für sie sind die *Cauchy-Riemannschen Differentialgleichungen*

$$f^1_x = f^2_y, \quad f^1_y = -f^2_x$$

erfüllt. Die reelle Funktionalmatrix ist daher von der speziellen Form

$$\nabla f = \begin{bmatrix} a & b \\ -b & a \end{bmatrix}, \quad a = f^1_x = f^2_y, \quad b = f^1_y = -f^2_x$$

Das reelle Newton-Verfahren lautet

$$(x,y)^{k+1} = (x,y)^k - \nabla f(x^k, y^k)^{-1} f(x^k, y^k),$$

was wegen der Cauchy-Riemannschen Differentialgleichungen äquivalent ist zu

$$z^{k+1} = z^k - \frac{f(z^k)}{f'(z^k)}, \quad f'(z) = f^1_x(x,y) + if^2_x(x,y) \text{ mit } z = x + iy.$$

Verfügt der Rechner über eine komplexe Arithmetik, kann man das Newton-Verfahren auch direkt im Komplexen durchführen.

4.5 Das gedämpfte Newton-Verfahren Beispiel Wir betrachten den eindimensionalen Fall mit $f(x) = \arctan(x)$. Offenbar gibt es eine kritische Konstante c, so dass für $|x^0| > c$ das Newton-Verfahren divergiert und für $|x^0| < c$ konvergiert. Die Divergenz des Verfahrens kann so interpretiert werden, dass die Korrektur

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

zwar in die richtige Richtung zeigt, aber „über das Ziel hinausschießt".

Das Hauptproblem des Newton-Verfahrens besteht darin, dass es im Allgemeinen nur für sehr gute Startwerte überhaupt konvergiert. Einen Grund dafür zeigt gerade das obige Beispiel: Ist der Startwert schlecht, so schießt die Newton-Richtung über.

Satz Sei $f \in C^1(\mathbb{R}^n)^n$ mit $\nabla f(x^0)$ regulär und $f(x^0) \neq 0$. Dann ist die Newton-Richtung

$$d = -\nabla f(x^0)^{-1} f(x^0)$$

eine *Abstiegsrichtung* für $g(x) = |f(x)|^2$, d.h. es gibt ein $t_0 > 0$ mit

$$g(x^0 + td) < g(x^0) \quad \text{für alle } 0 < t \leq t_0.$$

Beweis: Mit $F(t) = g(x + td)$ und $|f|^2 = \sum_{i=1}^n (f^i)^2$ folgt

$$\frac{d}{dt}F(t) = \nabla g(x^0 + td)d = 2\sum_{i,j=1}^{n} f^i(x)\partial_j f^i(x)d_j = f(x)^T \nabla f(x)d, \quad x = x^0 + td,$$

und daher

$$\frac{d}{dt}F(0) = -2f(x^0)^T \nabla f(x^0)\nabla f(x^0)^{-1} f(x^0) = -2g(x^0) < 0.$$

□

Mit diesem Satz lassen sich eine Vielzahl von gedämpften Newton-Verfahren definieren. Im einfachsten Fall akzeptiert man die Newton-Iterierte, wenn ein Abstieg von g vorliegt, andernfalls dämpft man:

1) Wähle $x^0 \in \mathbb{R}^n$.

2) Sei $x^k \in \mathbb{R}^n$ bereits bestimmt. Löse das lineare Gleichungssystem

$$\nabla f(x^k)d = -f(x^k).$$

3) a) Sei $t = 1$.
   b) Wenn $|f(x^k + td)| < |f(x^k)|$, setze $x^{k+1} = x^k + td$, $k = k + 1$ und gehe nach 2).
   Andernfalls setze $t = t/2$ und gehe nach 3)b).

Das gedämpfte Newton-Verfahren konvergiert gegen ein lokales Minimum von $|f|$ oder gegen eine Nullstelle von f, sofern es überhaupt konvergiert. Den ersten Fall erkennt man daran, dass es im Lauf der Iteration zu immer mehr Dämpfungsschritten kommt, d.h die Werte der akzeptierten t werden immer kleiner. Um diese im Allgemeinen unerwünschte Situation schnell zu beenden, muss im obigen Algorithmus noch die Zahl der Dämpfungsschritte in jedem Schritt gespeichert und eine Exit-Strategie entwickelt werden.

Gegenüber dem normalen Newton-Verfahren ist der Konvergenzbereich des gedämpften Verfahren sehr viel größer geworden. Bei dem oben angeführten Beispiel $f(x) = \arctan x$ ist es sogar global konvergent. In der Nähe einer Nullstelle wird im gedämpften Newton-Verfahren $t = 1$ genommen und man bekommt die gewohnte quadratische Konvergenz.

4.6 Weiter modifizierte Newton-Verfahren

Numerische Differentiation Statt exakter Auswertung von $\partial_i f^j$ kann man auch einen Differenzenquotienten verwenden, im einfachsten Fall nimmt man für eine Komponente $f = f^j$

$$D_i^h f(x) = \frac{1}{h}\big(f(x + he_i) - f(x)\big) = \partial_i f(x) + r(x), \quad |r(x)| \le c_1 h$$

Da zwei nahezu gleich große Werte voneinander abgezogen werden, ist diese Auswertung numerisch instabil. Statt f bestimmen wir tatsächlich ein $\tilde{f}$ mit

$$|f - \tilde{f}| \le |f|\texttt{eps}.$$

Auf der rechten Seite können wir für die Auswertungen von $f(x)$ und $f(x + he_i)$ das gleiche $|f| = |f(x)|$ nehmen. Für die Auswertung $D_i^h \tilde{f}$ erhalten wir damit die Fehlerabschätzung

$$\begin{aligned} |D_i^h \tilde{f}(x) - \partial_i f(x)| &\le |D_i^h \tilde{f}(x) - D_i^h f(x)| + |D_i^h f(x) - \partial_i f(x)| \\ &\le \frac{2|f|\texttt{eps}}{h} + c_1 h =: g(h). \end{aligned}$$

Es gilt $c_1 \approx |\partial_{ii}^2 f(x)|$. Es lohnt sich jedoch nicht, die Konstante c_1 durch einen Differenzenquotienten für $\partial_{ii}^2 f(x)$ abzuschätzen. Man kann die Konstante c_1 vorgeben oder einfach $c_1 = 1$ setzen. Die rechte Seite in der letzten Gleichung ist strikt konvex in $h > 0$. Das Minimum von g lässt sich daher durch $g'(h) = 0$ bestimmen zu $h = \sqrt{2|f|\texttt{eps}/c_1}$. Für diese Wahl erzielt man eine Genauigkeit von $O(\sqrt{\texttt{eps}})$. Ist die Funktion f durch ein Unterprogramm gegeben, so kann man dieses mit einer höheren Genauigkeit ausstatten und auch obigen Differenzenquotienten in höherer Genauigkeit bestimmen. Nach den Ausführungen in Abschnitt 1.1 steigt dadurch die Rechenzeit nicht an. Daher kann für große n jede partielle Ableitung mit nur einer Auswertung von f bestimmt werden, weil $f(x)$ mehrfach verwendet werden kann. Das dürfte im Allgemeinen günstiger sein als die direkte Berechnung von $\partial_i f$.

Man kann den obigen Mechanismus, dass man bei einer Maschinengenauigkeit `eps` nur eine Genauigkeit von $O(\sqrt{\texttt{eps}})$ für die erste Ableitungen bekommen kann, durch Verwendung von Formeln höherer Ordnung abmildern. Für den zentralen Differenzenquotient

$$D_i^0 f(x) = \frac{1}{2h}\big(f(x + he_i) - f(x - he_i)\big) = \partial_i f(x) + O(h^2)$$

erhält man mit der gleichen Überlegung wie oben für die optimale Schrittweite $h \sim \texttt{eps}^{1/3}$ und kann die erste Ableitung mit einer Genauigkeit von $O(\texttt{eps}^{2/3})$ bestimmen. Dazu muss die Funktion f allerdings dreimal stetig differenzierbar sein und man braucht zwei Auswertungen von f für jede partielle Ableitung.

Vereinfachtes Newton-Verfahren Das Newton-Verfahren ist quadratisch konvergent, wenn man sich in der Nähe der Nullstelle befindet, d.h. in jeder Iteration verdoppelt sich die Anzahl der richtigen Stellen. In diesem Fall ändern sich die Iterierten kaum noch und man kann die einmal berechnete Funktionalmatrix $\nabla f(x)$ sowie ihre LR-Zerlegung für die nächsten Schritte beibehalten. Das Verfahren ist dann nur noch linear konvergent, die Konstante $\overline{c}$ aber sehr klein. Eine Alternative wird im nächsten Abschnitt besprochen.

Rang 1-Update von Broyden Sei zunächst $n = 1$ und $f : \mathbb{R} \to \mathbb{R}$ eine differenzierbare Funktion mit Nullstelle ξ. Im *Sekantenverfahren* wird $f'(x^k)$ in der Newton-Iteration

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

ersetzt durch den Differenzenquotient, der aus den letzten beiden Iterierten gebildet wird

$$x^{k+1} = x^k - B_k^{-1} f(x^k) \quad \text{mit } B_k = \frac{f(x^k) - f(x^{k-1})}{x^k - x^{k-1}}. \tag{4.3}$$

Wir ersetzen also die Steigung der Tangente durch die Steigung der Sekante. Ist $f'(\xi) \neq 0$, besitzt das Verfahren die merkwürdige Konvergenzordnung von $\Phi = (1 + \sqrt{5})/2 = 1.61\ldots$ und ist damit, weil keine erste Ableitung ausgewertet werden muss, durchaus konkurenzfähig zum Newton-Verfahren.

Das Sekantenverfahren lässt sich zunächst nicht auf höhere Dimensionen ausdehnen, weil wir dort keine Differenzenquotienten bilden können. Schreiben wir (4.3) für $f : \mathbb{R}^n \to \mathbb{R}^n$ etwas um, so erhalten wir die *Quasi-Newton-Verfahren*

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

wobei die $(n \times n)$-Matrizen B_k der Quasi-Newton-Bedingung

$$B_k(x^k - x^{k-1}) = f(x^k) - f(x^{k-1}) \tag{4.4}$$

genügen müssen. Durch diese Bedingung ist die Matrix B_k natürlich nicht eindeutig festgelegt. Im Broyden-Verfahren addiert man auf B_{k-1} eine Matrix vom Rang 1, so dass die neue Matrix die Bedingung (4.4) erfüllt. Grundlage ist das folgende

Lemma Seien $A, B \in \mathbb{R}^{n\times n}$ und $p, q \in \mathbb{R}^n$ mit $p \neq 0$ und $Ap = q$. Dann gilt für die „upgedatete“ Matrix

$$B' = B + \frac{1}{|p|^2}(q - Bp)p^T,$$

dass

$$\|B' - A\| \leq \|B - A\|, \quad B'p = Ap = q,$$

d.h. B' ist A mindestens so ähnlich wie B und stimmt auf $\operatorname{span}\{p\}$ mit A überein.

Beweis: Es gilt

$$B'p = Bp + \frac{1}{|p|^2}(Ap - Bp)p^T p = Ap$$

Für $w \perp p$ ist

$$B'w = Bw + \frac{1}{|p|^2}(Ap - Bp)p^T w = Bw.$$

Für allgemeines $v \in \mathbb{R}^n$ gilt die Darstellung $v = w + \alpha p$ mit $w \perp p$ und $|v| = (|w|^2 + \alpha^2|p|^2)^{1/2}$. Dann

$$\begin{aligned} |(B' - A)v| &\leq |(B' - A)w| + |(B' - A)(\alpha p| \\ &= |(B - A)w| + 0 \leq \|B - A\|\,|w| \leq \|B - A\|\,|v|. \end{aligned}$$

□

Wir erhalten damit den

Broyden-Algorithmus

1) Seien $x^0 \in \mathbb{R}^n$, $B_0 \in \mathbb{R}^{n\times n}$ mit $\|B_0 - \nabla f(x^0)\|$ klein.

2) Seien $x^k \in \mathbb{R}^n$ und $B_k \in \mathbb{R}^{n\times n}$ bereits konstruiert.

   Bestimme die Suchrichtung

$$d_k = -B_k^{-1} f(x^k)$$

sowie $x^{k+1} \in \mathbb{R}^n$ und $0 < \lambda_k \leq 1$ mit

$$|f(x^{k+1}| \approx \min_{0<\lambda_k\leq 1} |f(x^k) + \lambda_k d_k)|.$$

Setze dann

$$p_k = x^{k+1} - x^k, \quad q_k = f(x^{k+1}) - f(x^k),$$

$$B_{k+1} = B_k + \frac{1}{|p_k|^2}(q - B_k p_k)p_k^T. \tag{4.5}$$

Setze $k = k + 1$ und gehe nach 2).

In der Nähe der Nullstelle haben wir superlineare Konvergenz des Broyden-Verfahrens, d.h. es gilt dort die Abschätzung

$$|x^{k+1} - \xi| \leq c_k|x^k - \xi| \quad \text{mit } c_k = o(|x^k - \xi|).$$

Die quadratische Konvergenz geht verloren.

Es gibt verschiedene Möglichkeiten, mit Hilfe der Matrix B_{k-1} das lineare Gleichungssystem zur Bestimmung der Suchrichtung mit Matrix B_k effizient zu lösen. Die einfachste besteht darin, einmal die Inverse B^{-1} von B zu bestimmen und mit folgender Formel upzudaten:

$$(B + uv^T)^{-1} = B^{-1} - \frac{B^{-1}uv^TB^{-1}}{1 + v^tB^{-1}u}.$$

Für reguläres B ist $B + uv^T$ genau dann regulär, wenn $1 + v^tB^{-1}u \neq 0$.

Homotopie-Verfahren Auch wenn man mit dem gedämpften Newton-Verfahren arbeitet, braucht man i.A. einen Startwert mit $|x_0 - \xi| << 1$, damit das Verfahren konvergiert. Verwende daher eine *Homotopie*

$$H(t, x) = tf(x) + (1 - t)g(x), \quad t \in [0, 1],$$

mit g „ähnlich“ zu f, aber $g(x) = 0$ leicht zu lösen.

Homotopie-Verfahren

1) Bestimme x^0 aus $H(0, x^0) = g(x^0) = 0$ und wähle $h > 0$.

2) Sei x^k bestimmt für $t = t_k$. Ist $t_k = 1$, so STOP, x^k ist die Lösung. Ist $t_k + h > 1$, so setze $h = 1 - t_k$. Versuche

   $$H(t_k + h, x^{k+1}) = 0 \tag{4.6}$$

   mit dem Newton-Verfahren zu lösen mit Start

   $$x^{k+1,0} = x^k. \tag{4.7}$$

   Falls dies möglich ist, so setze $h = 1.2h$, $t_{k+1} = t_k + h$, $k = k + 1$ und gehe nach 2). Falls dies nicht möglich ist, so setze $h = h/2$ und gehe nach 2).

Es fehlt noch eine Abbruchbedingung, wenn das Verfahren fehlschlägt, z.B.

$$\text{Wenn } h < \texttt{eps}, \text{ so STOP.}$$

Man nennt (4.7) den *Prädiktor* und das Newton-Verfahren für die Lösung von (4.6) den *Korrektor*.

Das Verfahren funktioniert, wenn es eine Kurve $x(t)$ gibt mit

$$H(t, x(t)) = 0, \quad x(0) = x^0$$

und

$$\nabla_x H(t, x(t)) \text{ regulär für alle } t \in [0, 1].$$

Denn nach dem Satz über implizite Funktionen ist dann die Kurve $x(t)$ lokal eindeutig bestimmt und, falls H glatt, ebenfalls glatt.

Im glatten Fall kann man H nach t differenzieren

$$0 = \frac{d}{dt} H(t, x(t)) = H_t + \nabla_x H x',$$

also

$$x'(t) = -\nabla_x H(t, x(t))^{-1} H_t(t, x(t)).$$

Statt (4.7) kann man daher den *Tangentenprädiktor* verwenden,

$$x^{k+1,0} = x^k - h \nabla_x H(t_k, x^k)^{-1} H_t(t_k, x^k),$$

wozu ein lineares Gleichungssystem gelöst werden muss. Alternativ kann der *Sekantenprädiktor* verwendet werden,

$$x^{k+1,0} = x^k + \frac{t_{k+1} - t_k}{t_k - t_{k-1}} (x^k - x^{k-1}),$$

der natürlich billig zu bekommen ist.

Alle diese Möglichkeiten dürfen allerdings nicht darüber hinwegtäuschen, dass das Auffinden einer Homotopie mit $\nabla_x H$ regulär entlang der Lösungskurve das große Problem beim Homotopie-Verfahren ist. Selbstverständlich können auch allgemeinere Homotopien als die eingangs vorgestellte affin lineare Homotopie verwendet werden.

4.7 Polynome und ihre Nullstellen

Das Horner-Schema Wir betrachten Polynome der Form

$$p(x) = a_n x^n + \ldots + a_0, \quad a_j \in \mathbb{R} \text{ oder } \mathbb{C}.$$

Wir schreiben $\operatorname{grad} p = n$, wenn $a_n \neq 0$. Wenn wir das Newton-Verfahren zur Nullstellensuche verwenden, müssen wir das Polynom und seine erste Ableitung in einem Punkt ξ auswerten. Dazu verwendet man das *Horner-Schema*

$$p(\xi) = (\ldots (a_n \xi + a_{n-1}) \xi + a_{n-2}) \ldots) \xi + a_0$$

oder rekursiv

$$b_n = a_n, \quad b_i = b_{i+1} \xi + a_i \text{ für } i = n-1, \ldots, 0.$$

Dann ist $b_0 = p(\xi)$.

Um eine entsprechende Formel für $p'(\xi)$ zu finden, setzen wir

$$p_1(x) = b_n x^{n-1} + \ldots + b_2 x + b_1,$$

mit den gerade bestimmten b_i. Dann gilt

$$(x - \xi) p_1(x) + b_0 = \underbrace{b_n}_{=a_n} x^n + (\underbrace{b_{n-1} - \xi b_n}_{=a_{n-1}}) x^{n-1} + \ldots + (\underbrace{b_0 - b_1 \xi}_{=a_0}) = p(x).$$

Damit $p(x) = (x - \xi) p_1(x) + b_0$ und

$$p'(x) = p_1(x) + (x - \xi) p_1'(x) \;\Rightarrow\; p'(\xi) = p_1(\xi).$$

Die Ableitung von p kann daher mit dem Horner-Schema für das Polynom p_1 bestimmt werden. Das Horner-Schema ist weniger rundungsfehleranfällig als die natürliche Auswertung mit Hilfe der Potenzen ξ^i.

Das Newton-Verfahren für Polynome Verfahren für die Berechnung von Nullstellen von Polynomen braucht man hauptsächlich bei der Bestimmung von Eigenwerten $Ax = \lambda x$, die durch die Nullstellen des charakteristischen Polynoms

$$p(\lambda) = \det(A - \lambda I) \in \mathbb{P}_n \quad \text{für } A \in \mathbb{R}^{n\times n}$$

gefunden werden können. Interessant ist hier auch der Fall einer symmetrischen Matrix A, bei der alle Eigenwerte reell sind.

Satz Besitzt das reelle Polynom p vom Grad n nur reelle Nullstellen mit

$$\xi_1 \geq \xi_2 \geq \ldots \geq \xi_n$$

so konvergiert das Newton-Verfahren für alle Startwerte $x^0 > \xi_1$ streng monoton fallend gegen ξ_1.

Beweis: Sei ohne Beschränkung der Allgemeinheit $p(x) > 0$ für $x > \xi_1$. In der Darstellung

$$p(x) = a_n x^n + \ldots + a_0, \quad a_n \neq 0,$$

muss dann $a_n > 0$ gelten. Die Nullstellen von p' und p'' liegen links von ξ_1, wegen $a_n > 0$ muss dann auch $p'(x), p''(x) > 0$ gelten für $x > \xi_1$.

Für $x^k > \xi_1$ gilt im Newton-Verfahren

$$x^{k+1} = x^k - \frac{p(x^k)}{p'(x^k)} < x^k.$$

Ferner gilt für $x^k > \xi_1$ mit einem $\eta \in (\xi_1, x^k)$ nach Taylor

$$\begin{aligned} 0 = p(\xi_1) &= p(x^k) + (\xi_1 - x^k)p'(x^k) + \frac{1}{2}(\xi_2 - x^k)^2 p''(\eta) \\ &> p(x^k) + (\xi_1 - x^k)p'(x^k), \end{aligned}$$

daher wegen $p'(x^k) > 0$

$$(\xi_1 - x^k) < -\frac{p(x^k)}{p'(x^k)},$$

also

$$\xi_1 < x^k - \frac{p(x^k)}{p'(x^k)} = x^{k+1}.$$

□

Wenn x^k noch weit von der Nullstelle entfernt ist, so

$$x^{k+1} = x^k - \frac{a_n(x^k)^n + \ldots}{n a_n (x^k)^{n-1} + \ldots} \approx x^k(1 - \frac{1}{n})$$

und die Konvergenz ist sehr langsam. Verwende daher das

Doppelschrittverfahren

1) Starte mit $x^0 > \xi_1$.

2) Verwende

$$x^{k+1} = x^k - 2\frac{p(x^k)}{p'(x^k)}$$

solange $p(x^k)p(x^0) > 0$. Andernfalls, wenn also $p(x^k)p(x^0) < 0$, so starte ein normales Newton-Verfahren mit $y^0 = x^k$.

Satz Unter den gleichen Voraussetzungen wie im letzten Satz gilt für das Doppelschrittverfahren

$$\xi_2 < y^0 < \xi_1 \quad \text{und} \quad y^1 > \xi_1.$$

Bemerkung Nachdem y^1 bestimmt wurde, konvergiert das Newton-Verfahren wie im letzten Satz angegeben wieder monoton gegen ξ_1. Ist ξ_1 eine mehrfache Nullstelle, so kann der Fall $\xi_2 < y^0 < \xi_1$ offenbar gar nicht eintreten und zum Umschalten in das normale Newton-Verfahren kommt es erst gar nicht. Daher konvergieren bereits die x^k monoton fallend gegen die Nullstelle.

Abdividieren von Nullstellen Wenn eine Nullstelle ξ_1 von $p(x)$ gefunden wurde, so bilde

$$p(x) = (x - \xi_1)p_1(x) \tag{4.8}$$

und bestimme dann eine Nullstelle von $p_1(x)$. Durch dieses sogenannte Abdividieren werden große Rundungsfehler erzeugt. Mit dem Ansatz

$$p_1(x) = a'_{n-1}x^{n-1} + \ldots + a'_0$$

kann man die Koeffizienten a'_i durch Koeffizientenvergleich aus (4.8) bestimmen. Man geht dann in der Reihenfolge $a'_{n-1}, \ldots, a'_0$ bei betragsmäßig kleinen Nullstellen und in umgekehrter Reihenfolge bei betragsmäßig großen Nullstellen vor. Zudem kann man für die später aufgefundenen Nullstellen noch einen Newton-Schritt mit dem Originalpolynom p durchführen.

5 Interpolation

5.1 Die Lagrangesche Interpolationsaufgabe Mit $\mathbb{P}_n$ bezeichnen wir den Raum der reellen Polynome vom Grad $\leq n$. Gegeben seien $n+1$ verschiedene Stützstellen $x_j \in \mathbb{R}$, $j = 0, \ldots, n$, und $n+1$ nicht notwendig verschiedene Werte $y_0, \ldots, y_n$. In der *Lagrangeschen Interpolationsaufgabe* ist ein Polynom $p \in \mathbb{P}_n$ gesucht mit

$$p(x_j) = y_j, \quad j = 0, 1, \ldots, n. \tag{5.1}$$

Die y_j können wir uns als Werte $y_j = f(x_j)$ einer vorgegebenen Funktion f vorstellen. Wir sagen dann, dass p die Funktion f *interpoliert*.

Die Dimension von $\mathbb{P}_n$ ist $n+1$. Wir haben daher in der Lagrangeschen Interpolationsaufgabe $n+1$ Bedingugen gestellt, aber auch $n+1$ Freiheiten zur Verfügung. Zur Lösung des Interpolationsproblems definieren wir die *Lagrange-Basis* $\{l_j\}_{j=0,\ldots,n}$, $l_j \in \mathbb{P}_n$, durch

$$l_j(x) = \frac{(x-x_0)\ldots(x-x_{j-1})(x-x_{j+1})\ldots(x-x_n)}{(x_j-x_0)\ldots(x_j-x_{j-1})(x_j-x_{j+1})\ldots(x_j-x_n)}. \tag{5.2}$$

Es gilt dann $l_i(x_j) = \delta_{ij}$ und die Interpolationsaufgabe (5.1) wird gelöst durch

$$p(x) = \sum_{j=0}^{n} y_j l_j(x) \in \mathbb{P}_n. \tag{5.3}$$

Satz Die Interpolationsaufgabe (5.1) wird eindeutig gelöst durch das *Lagrangesche Interpolationspolynom* (5.3).

Beweis: Gäbe es zwei Lösungen $p_1, p_2 \in \mathbb{P}_n$ von $p(x_j) = y_j$, so gilt für $q = p_1 - p_2 \in \mathbb{P}_n$, dass $q(x_j) = 0$. Damit hat q $n+1$ Nullstellen und muss das Nullpolynom sein. Daher ist $p_1 = p_2$. □

5.2 Die Newtonsche Interpolationsformel Das Polynom $p_n(x) \in \mathbb{P}_n$ interpoliere die Daten $(x_j, y_j) \in \mathbb{R}^2$, $j = 0, \ldots, n$. Wir nehmen ein weiteres Zahlenpaar $(x_{n+1}, y_{n+1}) \in \mathbb{R}^2$, $x_{n+1} \neq x_j$, $j = 0, \ldots, n$, hinzu. Kann man dann das Interpolationspolynom $p_{n+1}(x) \in \mathbb{P}_{n+1}$ zu den Daten (x_j, y_j), $j = 0, \ldots, n+1$, schreiben als

$$p_{n+1}(x) = p_n(x) + f(x)$$

mit einer leicht berechenbaren Funktion $f(x)$? Wegen $p_n \in \mathbb{P}_n$, $p_{n+1} \in \mathbb{P}_{n+1}$ gilt $f(x) = p_{n+1}(x) - p_n(x) \in \mathbb{P}_{n+1}$ sowie $f(x_j) = 0$ für $j = 0, \ldots, n$. Daher hat f mit einem $a \in \mathbb{R}$ die Gestalt

$$f(x) = a \prod_{j=0}^{n} (x - x_j).$$

a kann man aus der Interpolationsbedingung

$$y_{n+1} = p_{n+1}(x_{n+1}) = p_n(x_{n+1}) + a \prod_{j=0}^{n} (x_{n+1} - x_j)$$

ermitteln

$$a = \frac{y_{n+1} - p_n(x_{n+1})}{(x_{n+1}-x_0)\ldots(x_{n+1}-x_n)}.$$

Wir wollen nun ein einfaches Verfahren zur Berechnung der Zahl a, des führenden Koeffizienten des Interpolationspolynoms, herleiten. Grundlage dafür ist

Satz [Aitken Lemma] Es sei zu $(x_j, y_j) \in \mathbb{R}^2$, $j = 0, \ldots, n$, $x_i \neq x_j$, das Interpolationspolynom $p \in \mathbb{P}_n$ gesucht.

Seien $p_{[0]}, p_{[n]} \in \mathbb{P}_{n-1}$ die Interpolationspolynome mit

$$p_{[0]}(x_j) = y_j,\ j = 0, \ldots, n-1, \quad p_{[n]}(x_j) = y_j,\ j = 1, \ldots, n.$$

Dann gilt

$$p(x) = \frac{p_{[0]}(x)(x - x_n) - p_{[n]}(x)(x - x_0)}{x_0 - x_n}.$$

Beweis: Das angegebene Polynom genügt offenbar den Bedingungen $p(x_j) = y_j$ für $j = 0, \ldots, n$ sowie $p \in \mathbb{P}_n$. □

Für $0 \leq i \leq j \leq n$ sei nun $p_{ij} \in \mathbb{P}_{j-i}$ das Interpolationspolynom mit $p_{ij}(x_k) = y_k$ für $k = i, \ldots, j$. Dann folgt aus dem Aitken Lemma, dass für die Interpolationspolynome die Rekursion gilt

$$p_{ij}(x) = \frac{p_{i\,j-1}(x)(x - x_j) - p_{i+1\,j}(x)(x - x_i)}{x_i - x_j} \tag{5.4}$$

$$= p_{i+1\,j}(x) + \big(p_{i+1\,j}(x) - p_{i\,j-1}(x)\big)\frac{x - x_j}{x_j - x_i} \tag{5.5}$$

Diese Form des Interpolationspolynoms eignet sich besonders, um einzelne Werte zu berechnen, ohne gleich das ganze Polynom aufzustellen. Mit P_{ij}, $i \leq j$ bezeichnen wir den Wert des Interpolationspolynoms $p_{ij}(x)$ für ein festes x. Wir erhalten dafür das folgende *Neville-Schema*

$$\begin{array}{lllll} x_0 & y_0 = P_{00} & & & \\ & & P_{01} & & \\ x_1 & y_1 = P_{11} & & P_{02} & \\ & & P_{12} & & P_{03} \\ x_2 & y_2 = P_{22} & & P_{13} & \\ & & P_{23} & & P_{14} \quad P_{04} = p_{04}(x), \\ x_3 & y_3 = P_{33} & & P_{24} & \\ & & P_{34} & & \\ x_4 & y_4 = P_{44} & & & \end{array}$$

wobei P_{ij} aus $P_{i\,j-1}$ und $P_{i+1\,j}$ mit (5.4) bestimmt wird.

Man kann in diesem Schema auf einfache Weise weitere Stützstellen hinzufügen. In obigem Beispiel hängt man das Paar $(x_5, y_5 = P_{55})$ unten an und berechnet die Werte $P_{45}, \ldots, P_{05}$

Aus der Formel (5.5) erhält man eine weitere Darstellung des Interpolationspolynoms. Da a in

$$p_n = p_{n-1} + a \prod_{j=0}^{n-1}(x - x_j)$$

der Koeffizient der höchsten Potenz x^n in $p_n(x)$ ist, liest man aus (5.5) durch Koeffizientenvergleich sofort ab:

Satz [Newtonsche Interpolationsformel] Das Interpolationspolynom aus (5.1) hat die Gestalt

$$p(x) = \sum_{j=0}^{n} [x_0, \ldots, x_j] \prod_{k=0}^{j-1}(x - x_k), \quad \prod_{k=0}^{-1} = 1, \tag{5.6}$$

wobei die *dividierten Differenzen* $[x_0, \ldots, x_j]$ rekursiv definiert sind durch

$$[x_j] = y_j$$

$$[x_k, \ldots, x_j] = \frac{[x_{k+1}, \ldots, x_j] - [x_k, \ldots, x_{j-1}]}{x_j - x_k}, \quad j > k \geq 0.$$

Die dividierten Differenzen können daher völlig analog zum Neville Schema bestimmt werden

$$\begin{array}{llll} y_0 & [x_0,x_1] & & \\ y_1 & [x_1,x_2] & [x_0,x_1,x_2] & \\ y_2 & [x_2,x_3] & [x_1,x_2,x_3] & [x_0,x_1,x_2,x_3] \\ y_3 & [x_3,x_4] & [x_2,x_3,x_4] & [x_1,x_2,x_3,x_4] \\ y_4 & & & \end{array} \qquad [x_0,x_1,x_2,x_3,x_4]$$

Einzelne Werte des Interpolationspolynoms bestimmt man aus (5.6) unter Verwendung eines Horner-ähnlichen Schemas. Wir schreiben (5.6) in der Form

$$p(x) = \sum_{j=0}^{n} c_j \prod_{k=0}^{j-1}(x - x_k) = (\ldots(c_n(x - x_{n-1}) + c_{n-1})(x - x_{n-2}) + \ldots + c_1)(x - x_0) + c_0$$

und erhalten die Rekursion

$$\begin{aligned} b_n &= c_n, \\ b_i &= b_{i+1}(x - x_i) + c_i, \quad i = n-1, \ldots, 0. \end{aligned}$$

Dann ist $p(x) = b_0$.

Die Newtonsche Interpolationsformel liefert die effizienteste Methode zur Auswertung des Interpolationspolynoms. Selbst wenn man nur an einem Funktionswert interessiert ist, ist der Aufwand geringer als mit dem Neville-Schema. Wegen seiner einfachen Gestalt wird das Neville-Schema dennoch in der Praxis verwendet.

5.3 Der Interpolationsfehler Satz Sei $f \in C^{n+1}[a,b]$ und seien Stützstellen gegeben mit $a \le x_0 < \ldots < x_n \le b$. Sei $p \in \mathbb{P}_n$ das zugehörige Interpolationspolynom mit $p(x_j) = f(x_j)$. Zu jedem $x \in [a,b]$ gibt es ein ξ aus dem kleinsten Intervall I, das die Punkte $x, x_0, x_1, \ldots, x_n$ enthält, mit

$$f(x) - p(x) = \frac{\omega(x)}{(n+1)!} f^{(n+1)}(\xi), \tag{5.7}$$

mit

$$\omega(x) = (x - x_0) \cdots (x - x_n).$$

Beweis: Für $x \neq x_j$ betrachte die Funktion

$$F(x) = f(x) - p(x) - \alpha\omega(x) \tag{5.8}$$

und bestimme ein $\alpha \in \mathbb{R}$ so, dass $F(x) = 0$ erfüllt ist, was wegen $\omega(x) \neq 0$ möglich ist. Dann besitzt F mindestens die Nullstellen $x, x_0, \ldots, x_n$ in I. Nach dem Satz von Rolle besitzt F' mindestens $n+1$ Nullstellen, F'' mindesten n und $F^{(n+1)}$ mindestens eine Nullstelle ξ in I, also

$$0 = F^{(n+1)}(\xi) = f^{(n+1)}(\xi) - 0 - \alpha \cdot (n+1)!$$

und damit $\alpha = f^{(n+1)}(\xi)/(n+1)!$. Die Behauptung folgt aus $F(x) = 0$ in (5.8). □

Für ein Intervall $I = [a,b]$ betrachten wir nun eine Folge von Zerlegungen

$$\Delta_m = \{a = x_0^{(m)} < x_1^{(m)} < \ldots < x_{n_m}^{(m)} = b\}$$

mit Feinheit

$$|\Delta_m| = \max_i |x_{i+1}^{(m)} - x_i^{(m)}|.$$

Aus obiger Fehlerdarstellung erhält man sofort, dass sehr glatte Funktionen wie etwa die e-Funktion auf beschränkten Intervallen immer genauer interpoliert werden. Für alle übrigen Funktionen wird der Interpolationsfehler in der Regel ansteigen, wenn wir viele Stützstellen verwenden. Einige Resultate:

Satz von Faber: Zu jeder Folge $\{\Delta_m\}$ gibt es eine stetige Funktion f, so dass die zugehörigen Interpolationspolynome nicht gleichmäßig gegen f konvergieren.

Runge-Phänomen: Die Folge der Interpolationspolynome auf äquidistanten Zerlegungen ist nicht beschränkt, wenn die Funktion $1/(1+x^2)$ auf dem Intervall $[-5,5]$ interpoliert wird.

Zur Darstellung einer Funktion ist die Polynominterpolation daher ungeeignet.

5.4 Hermite-Interpolation In diesem Abschnitt gehen wir von einem beschränkten Intervall $[a,b]$ und einer Funktion $f \in C^1[a,b]$ aus. Für paarweise verschiedene Stützstellen $x_0, x_1, \ldots, x_n \in [a,b]$ suchen wir ein Polynom $p \in \mathbb{P}_{2n+1}$, das die *Hermitesche Interpolationsaufgabe*

$$\begin{aligned} p(x_j) &= f(x_j) = f_j \\ p'(x_j) &= f'(x_j) = f'_j. \end{aligned} \tag{5.9}$$

für $j = 0, \ldots, n$ erfüllt. Analog zum Vorgehen beim Lagrangen Interpolationsproblem suchen wir eine Basis $\phi_j, \psi_j \in \mathbb{P}_{2n+1}$ mit

$$\begin{aligned} \phi_j(x_i) = \delta_{ij}, \quad \phi'_j(x_i) = 0, \\ \psi_j(x_i) = 0, \quad \psi'_j(x_i) = \delta_{ij}, \end{aligned}$$

für $i, j = 0, \ldots, n$. Mit den Basisfunktionen $l_j \in \mathbb{P}_n$ des Lagrangeschen Interpolationsproblems in (5.2) gilt dann

$$\begin{aligned} \phi_j(x_i) &= \big(1 - 2l'_j(x_j)(x - x_j)\big) l_j^2(x) \\ \psi_j(x_i) &= (x - x_j) l_j^2(x), \end{aligned}$$

was man für $j = 0, \ldots, n$ leicht nachrechnen kann. Die Lösung von (5.9) ist dann

$$p(x) = \sum_{j=0}^{n} f_j \phi_j(x) + \sum_{j=0}^{n} f'_j \psi_j(x). \tag{5.10}$$

Satz Das Hermite-Interpolationspolynom $p \in \mathbb{P}_{2n+1}$ aus (5.10) ist die eindeutige Lösung des Hermiteschen Interpolationsproblems (5.9).

Beweis: Gäbe es zwei Interpolationspolynome $p_1, p_2 \in \mathbb{P}_{2n+1}$, die (5.9) erfüllen, so besäße $q = p_1 - p_2 \in \mathbb{P}_{2n+1}$ $n+1$ Nullstellen und $n+1$ Nullstellen der ersten Ableitung. Damit ist $q = 0$ und $p_1 = p_2$. □

Für den Interpolationsfehler gilt entsprechend Satz 5.3:

Satz Sei $f \in C^{2n+2}[a,b]$, seien die Stützstellen $x_0, \ldots, x_n \in [a,b]$ paarweise verschieden, und sei $p \in \mathbb{P}_{2n+1}$ das Hermitesche Interpolationspolynom, das den Bedingungen (5.9) genügt. Dann gibt es zu jedem $x \in [a,b]$ ein ξ aus dem kleinsten Intervall, das die Punkte $x, x_0, \ldots, x_n$ enthält mit

$$f(x) - p(x) = \frac{\omega^2(x)}{(2n+2)!} f^{(2n+2)}(\xi). \tag{5.11}$$

mit $\omega(x) = (x - x_0) \cdots (x - x_n)$.

Beweis: Für $x = x_j$ ist die Gleichung richtig, sei also $x \neq x_j$ für alle j. Die Funktion

$$F(z) = \big(f(x) - p(x)\big)\omega^2(z) - \big((f(z) - p(z)\big)\omega^2(x)$$

besitzt für jedes x_j mindestens eine doppelte und für x mindestens eine einfache Nullstelle. $(2n+2)$-fache Anwendung des Satzes von Rolle zeigt, dass $F^{(2n+2)}$ mindestens eine Nullstelle ξ besitzt, also

$$0 = F^{(2n+2)}(\xi) = (2n+2)!\big(f(x) - p(x)\big) - f^{(2n+2)}(\xi)\omega^2(x) = 0.$$

Eine einfache Umformung liefert die Behauptung. □

5.5 Splines Die in den vorhergehenden Abschnitten behandelte Polynominterpolation ist zwar leicht durchführbar, hat aber den Nachteil, dass bei Verfeinerung der Zerlegung keine Konvergenz zu erwarten ist. Bessere Konvergenzeigenschaften haben die nun zu besprechenden Splines.

Sei

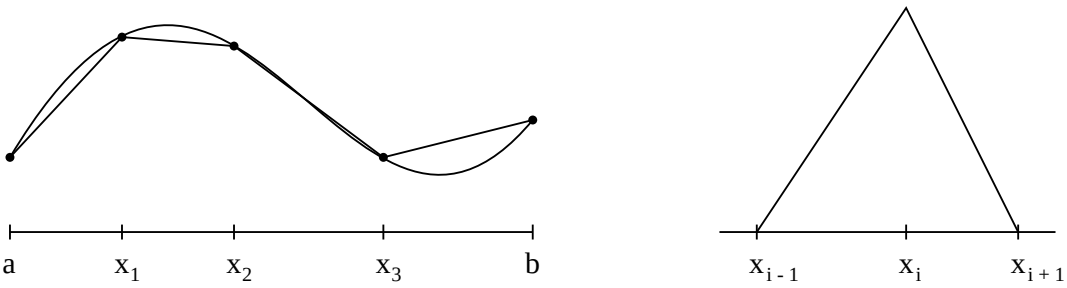
$$\Delta = \{a = x_0 < x_1 < \ldots < x_n = b\} \tag{5.12}$$

eine Zerlegung des Intervalls $[a, b]$. Dann bezeichnen wir mit $S(\Delta, p, q)$, $p, q \in \mathbb{N}_0$, $0 \le q < p$, den Raum aller Funktionen $s \in C^q[a, b]$, die auf jedem Teilintervall $[x_{i-1}, x_i]$, $i = 1, \ldots, n$, mit einem Polynom vom Höchstgrad p übereinstimmen. Jedes $s \in S(\Delta, p, q)$ heißt (Polynom-)Spline vom Grade p der Differenzierbarkeitsklasse q zur Zerlegung Δ.

Wir behandeln nur Splines, bei denen es eine leicht zugängliche Basisdarstellung ähnlich wie die Lagrange-Basis in der Polynominterpolation gibt, die aber zusätzlich noch lokaler Natur ist.

Der Raum der stückweise linearen Funktionen $S(\Delta, 1, 0)$ Sei Δ eine Zerlegung des Intervalls wie in (5.12). Wir definieren

$$S_1 = S(\Delta, 1, 0) = \{s \in C[a, b] : \ s\big|_{[x_{i-1}, x_i]} \in \mathbb{P}_1 \text{ für } i = 1, \ldots, n\}.$$



Offenbar ist eine solche stetige, stückweise lineare Spline-Funktion s durch die Vorgabe ihrer Werte an den Stützstellen $x_0, \ldots, x_n$ eindeutig bestimmt, da sie ja innerhalb der Teilintervalle affin linear ist. Wir definieren die Basisfunktionen $\phi_i \in S_1$ durch

$$\phi_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & \text{falls } x_{i-1} \le x \le x_i \text{ und } i > 0 \\ \dfrac{x_{i+1} - x}{x_{i+1} - x_i} & \text{falls } x_i \le x \le x_{i+1} \text{ und } i < n \\ 0 & \text{sonst} \end{cases}.$$

Es gilt dann

$$\phi_i(x_j) = \delta_{ij}, \quad i, j = 0, \ldots, n..$$

Aufgrund dieser Eigenschaft sind die ϕ_i linear unabhängig und bilden eine Basis von S_1. Die im Inneren des Intervalls liegenden Basisfunktionen haben die charakteristische Hut-Gestalt, wie im Bild oben rechts zu sehen ist.

Für eine auf dem Intervall $[a,b]$ stetige Funktion f ist die *Interpolierende* $I_\Delta f \in S_1$ definiert durch

$$I_\Delta f(x) = \sum_{i=0}^{n} f(x_i)\phi_i(x).$$

$I_\Delta f$ ist damit die eindeutige stetige, stückweise lineare Funktion, die an den Stützstellen mit f übereinstimmt. Ein Beispiel für eine solche Interpolierende ist im Bild oben links zu sehen, aus dem auch klar wird, dass man eine täuschend echte Approximation von f nur erreicht, wenn die Feinheit der Zerlegung sehr klein ist. Zur Darstellung einer Funktion ist also der Raum S_1 weniger geeignet.

Dennoch kann man mit solchen Splines sehr viel anfangen, um schwierige Probleme wie die Approximation gewöhnlicher und partieller Differentialgleichungen zu behandeln. Denn das Stabilitätsverhalten dieser Splines ist unvergleichlich viel besser als das der Polynominterpolation. Aus dem obigen Bild wird beispielsweise sofort klar, dass $\|I_\Delta f\|_\infty \leq \|f\|_\infty$ richtig ist, wobei $\|\cdot\|_\infty$ die Maximumsnorm ist. Weiter ist für die Auswertung von $I_\Delta f$ nur die Bestimmung eines gewichteten Mittelwerts der Werte von f an den benachbarten Stützstellen erforderlich.

Wir leiten nun eine Abschätzung für den Fehler her.

Satz Sei $f \in C^2[a,b]$. Dann gilt mit $|\Delta| = \max_i |x_{i+1} - x_i|$

$$\|f - I_\Delta f\|_\infty \leq \frac{1}{8}|\Delta|^2 \|f''\|_\infty.$$

Beweis: Aus Satz 5.3 erhalten wir für $x \in [x_{i-1}, x_i]$

$$f(x) - I_\Delta f(x) = \frac{1}{2}(x - x_{i-1})(x - x_i) f''(\xi_i), \quad \xi_i \in [x_{i-1}, x_i].$$

Die Funktion in x auf der rechten Seite wird maximal für den Mittelwert von x_{i-1} und x_i, daher

$$|f - I_\Delta f|(x) \leq \frac{1}{8}|x_i - x_{i-1}|^2 \max_{x \in [x_{i-1}, x_i]} |f''(x)|,$$

also

$$\|f - I_\Delta f\|_\infty \leq \frac{1}{8}|\Delta|^2 \|f''\|_\infty.$$

□

Der Raum der kubischen Hermite-Splines $S(\Delta, 3, 1)$ Sei Δ eine Zerlegung des Intervalls wie in (5.12). Wir definieren

$$S_3 = S(\Delta, 3, 1) = \{s \in C^1[a,b] : \ s\big|_{[x_{i-1}, x_i]} \in \mathbb{P}_3 \text{ für } i = 1, \ldots, n\}.$$

In diesem Fall ist die Funktion $s \in S_3$ durch die Vorgabe ihrer Werte und der Werte ihrer Ableitung an den Stützstellen $x_0, \ldots, x_n$ eindeutig bestimmt, da sie innerhalb der Teilintervalle eine kubische Funktion ist. Wir benötigen Basisfunktionen $\phi_i \in S_3$, deren Ableitungen an den Stützstellen verschwinden, mit $\phi_i(x_j) = \delta_{ij}$ und Basisfunktionen $\psi_i \in S_3$, die an den Stützstellen verschwinden,

mit $\psi_i'(x_j) = \delta_{ij}$:

$$\phi_i(x) = \begin{cases} \dfrac{(x - x_{i-1})^2(3x_i - x_{i-1} - 2x)}{(x_i - x_{i-1})^3} & \text{falls } x_{i-1} \le x \le x_i \text{ und } i > 0 \\ \dfrac{(x_{i+1} - x)^2(x_{i+1} - 3x_i + 2x)}{(x_{i+1} - x_i)^3} & \text{falls } x_i \le x \le x_{i+1} \text{ und } i < n \\ 0 & \text{sonst} \end{cases},$$

$$\psi_i(x) = \begin{cases} \dfrac{(x - x_{i-1})^2(x - x_i)}{(x_i - x_{i-1})^2} & \text{falls } x_{i-1} \le x \le x_i \text{ und } i > 0 \\ \dfrac{(x - x_{i+1})^2(x - x_i)}{(x_{i+1} - x_i)^2} & \text{falls } x_i \le x \le x_{i+1} \text{ und } i < n \\ 0 & \text{sonst} \end{cases}.$$

Man rechnet leicht nach, dass dann

$$\left.\begin{array}{ll} \phi_i(x_j) = \delta_{ij}, & \phi_i'(x_j) = 0 \\ \psi_i(x_j) = 0, & \phi_i'(x_j) = \delta_{ij} \end{array}\right\} \; i, j = 0, \ldots, n.$$

Demnach stellen diese Funktionen eine Basis des Raumes S_3 dar, die zudem *lokal* ist: Der Träger der Basisfunktionen ϕ_i, ψ_i besteht lediglich aus dem Intervall $[x_{i-1}, x_{i+1}]$.

Für eine auf dem Intervall $[a, b]$ stetig differenzierbare Funktion f ist die Interpolierende $I_\Delta f \in S_3$ definiert durch

$$I_\Delta f(x) = \sum_{i=0}^{n} \big(f(x_i)\phi_i(x) + f'(x_i)\psi_i(x)\big).$$

$I_\Delta f$ erfüllt dann die Bedingungen

$$f(x_i) = I_\Delta f(x_i), \quad f'(x_i) = I_\Delta f'(x_i) \quad \text{für } i = 0, \ldots, n.$$

Eine Fehlerabschätzung folgt aus (5.11). Für $f \in C^4[a, b]$ gilt für beliebiges $x \in [x_{i-1}, x_i]$

$$f(x) - I_\Delta f(x) = \frac{1}{4!}(x - x_{i-1})^2(x - x_i)^2 f^{(4)}(\xi_i) \;\text{ für ein } \xi_i \in [x_{i-1}, x_i].$$

Auch hier wird die Funktion in x auf der rechten Seite maximal für den Mittelwert von x_{i-1} und x_i,

$$|f - I_\Delta f|(x) \le \frac{1}{4!} \cdot \frac{1}{16}|x_{i-1} - x_i|^4 \max_{x \in [x_{i-1}, x_i]} |f^{(4)}(x)|.$$

Damit haben wir gezeigt

Satz Sei $f \in C^4[a, b]$. Dann gilt mit $|\Delta| = \max_i |x_{i+1} - x_i|$

$$\|f - I_\Delta f\|_\infty \le \frac{1}{384}|\Delta|^4 \|f^{(4)}\|_\infty.$$

5.6 Bézierkurven Wir gehen von der Aufgabe aus, eine Kurve zu zeichnen, von deren Verlauf wir eine gewisse Vorstellung haben. Wie bestimmen wir eine Realisierung $x : [0, 1] \to \mathbb{R}^n$? Es liegt nahe, gewisse Punkte $x^i = (x_1^i, \ldots, x_n^i)$ festzulegen, durch die die Kurve laufen soll, und für jede Komponente eine Polynominterpolation durchzuführen. Wie im Abschnitt über Polynominterpolation ausgeführt, nimmt die Kurve dann in der Regel einen erratischen Verlauf, wenn wir viele Punkte vorgeben. Auch hier ist die Polynominterpolation kein geeignetes Hilfsmittel.

Die Polynome

$$B_i^m(t) = \binom{m}{i} t^i (1-t)^{m-i}, \quad t \in [0,1], \quad i = 0, \ldots, m$$

heißen *Bernstein-Polynome* vom Grad m.

Satz Es gilt:

(a) $0 < B_i^m(t) < 1$ in $(0,1)$.

(b) $\sum_{i=0}^m B_i^m(t) = 1$.

(c) $B_i^m(t)$ hat eine i-fache Nullstelle in $t = 0$ und eine $(m-i)$-fache Nullstelle in $t = 1$.

(d) Es gilt für alle m und $i = 0, \ldots, m+1$ die Rekursion

$$B_i^{m+1}(t) = (1-t)B_i^m(t) + tB_{i-1}^m(t), \tag{5.13}$$

wobei $B_{-1}^m = B_{m+1}^m = 0$ und $B_0^0 = 1$ gesetzt wird.

Beweis: Die Eigenschaften (a) und (c) sind trivialerweise erfüllt.

(b) folgt aus der binomischen Formel wegen

$$\sum_{i=0}^m B_i^m(t) = \sum_{i=0}^m \binom{m}{i} t^i (1-t)^{m-i} = \big(t + (1-t)\big)^m = 1.$$

(d) gilt wegen

$$\begin{aligned} B_i^{m+1}(t) &= \binom{m+1}{i} t^i (1-t)^{m+1-i} = \left(\binom{m}{i} + \binom{m}{i-1}\right) t^i (1-t)^{m+1-i} \\ &= (1-t)\binom{m}{i} t^i (1-y)^{m-i} + t\binom{m}{i-1} t^{i-1} (1-t)^{m-(i-1)} \\ &= (1-t)B_i^m(t) + tB_{i-1}^m(t). \end{aligned}$$

□

Wegen (c) sind die Bernstein-Polynome linear unabhängig, denn aus

$$\phi(t) = \sum_{i=0}^m \alpha_i B_i^m(t) = 0$$

folgt

$$\phi(0) = \sum_{i=0}^m \alpha_i B_i^m(0) = \alpha_0 = 0$$

und daher

$$\phi(t) = \sum_{i=1}^m \alpha_i B_i^m(t) = 0.$$

Genauso folgt aus $\phi'(t) = 0$, dass $\alpha_1 = 0$ und damit $\alpha_i = 0$ für alle i. Die B_i^m bilden daher für $i = 0, \ldots, m$ eine Basis des Polynomraums $\mathbb{P}_m$.

Mit den Bernstein-Polynomen lassen sich auf folgende Art schöne Kurven zeichnen: Es seien $x^i \in \mathbb{R}^n$, $i = 0, \ldots, m$, vorgegeben. Dann heißt

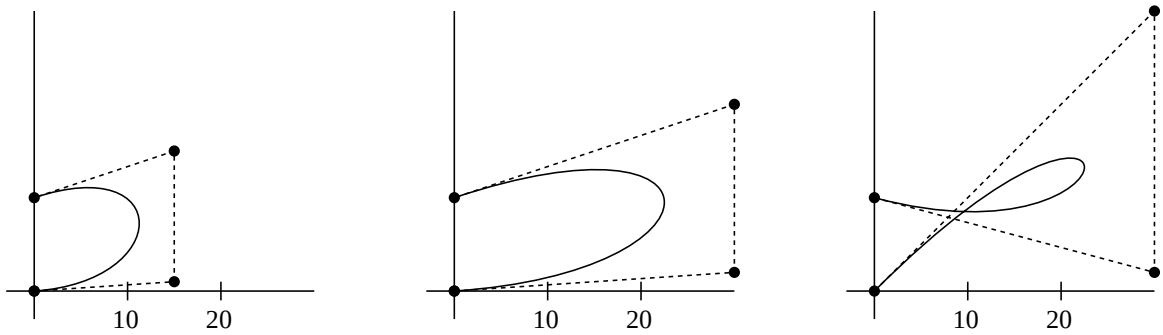
$$F(t) = \sum_{i=0}^m x^i B_i^m(t)$$

die *Bézier-Kurve* vom Grade m zu den *Bézierpunkten* (auch *Kontrollpunkte* genannt) x^i.

Wegen (c) gilt $F(0) = x^0$ und $F(1) = x^m$, die Bézierkurve verbindet demnach die Punkte x^0 und x^m. Die anderen Bézierpunkte dienen dazu, den Kurvenverlauf zu steuern, und liegen in der Regel nicht auf der Kurve.

Wegen (a) und (b) ist $F(t)$ für jedes t eine Konvexkombination der Punkte x^i. Damit liegt die ganze Kurve in der konvexen Hülle der x^i.

In der Programmiersprache postscript gibt es den Befehl „curveto“, mit dem Bézierkurven für $m = 3$ gezeichnet werden können. In der folgenden Abbildung geben wir einige Beispiele für die Ausführung dieses Befehls für $x^0 = (0, 0)$ und $x^3 = (10, 0)$. Mit dem Punkt x^1 lässt sich die erste Ableitung im Punkt x^0 vorgeben und damit gleichzeitig die „Anfangsgeschwindigkeit“ der Kurve. Die gleiche Bedeutung hat der Punkt x^2 für x^3. Die zugehörige Kurve der Kontrollpunkte ist gestrichelt eingezeichnet.



Die Rekursionsformel (5.13) liefert eine einfache Methode zur Auswertung des Bézier-Polynoms an einer festen Stelle, den *Algorithmus von de Casteljau:*

Satz Sei $\hat{t} \in [0, 1]$. Setze $x^{i,0} = x^i$ für $i = 0, \ldots, m$ und bestimme

$$x^{i,k+1} = (1 - \hat{t})x^{i-1,k} + \hat{t}x^{i,k}, \quad k = 0, \ldots, m-1, \quad i = k+1, \ldots, m.$$

Dann gilt

$$x^{m,m} = F(\hat{t}).$$

Beweis: Für $m = 1$ gilt

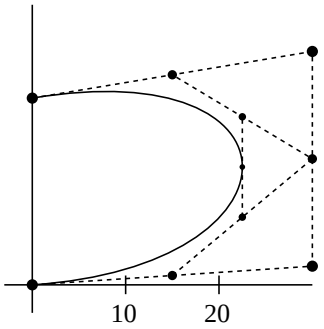
$$x^{1,1} = (1 - \hat{t})x^{0,0} + \hat{t}x^{1,0} = x^0 B_0^1(\hat{t}) + x^1 B_1^1(\hat{t}) = F(\hat{t}).$$

Ist $m \geq 2$ und die Behauptung für $m - 1$ bewiesen, so

$$\begin{aligned}
x^{m,m} &= (1 - \hat{t})x^{m-1,m-1} + \hat{t}x^{m,m-1} \\
&= (1 - \hat{t}) \sum_{i=0}^{m-1} x^i B_i^{m-1}(\hat{t}) + \hat{t} \sum_{i=0}^{m-1} x^{i+1} B_i^{m-1}(\hat{t}) \\
&= (1 - \hat{t})x^0 + \sum_{i=1}^{m-1} x^i \Big((1 - \hat{t}) B_i^{m-1}(\hat{t}) + \hat{t} B_{i-1}^{m-1}(\hat{t}) \Big) + \hat{t}x^m \\
&= \sum_{i=0}^{m} x^i B_i^m(\hat{t}) = F(\hat{t}).
\end{aligned}$$

□

Im Bild rechts wird der Algorithmus von Casteljau für $\hat{t} = 1/2$ gezeigt, dort werden also immer die Seitenmitten benachbarter Seiten miteinander verbunden.



6 Numerische Integration

6.1 Newton-Cotes Formeln In diesem und den folgenden Abschnitten wollen wir das Integral

$$\int_a^b f(x)\,dx$$

durch ein numerisches Verfahren approximieren. Durch die lineare Transformation $x = a + t(b-a)$ können wir es auf die Form

$$\int_a^b f(x)\,dx = (b-a)\int_0^1 f(a+t(b-a))\,dt$$

bringen.

Bei den *interpolatorischen Quadraturformeln* interpolieren wir f an den Stützstellen $x_0, x_1, \ldots, x_n \in [0,1]$ durch ein Polynom $p \in \mathbb{P}_n$ und können $\int_0^1 p(x)\,dx$ als Näherung für $\int_0^1 f(x)\,dx$ nehmen. Mit

$$l_j(x) = \frac{(x-x_0)\ldots(x-x_{j-1})(x-x_{j+1})\ldots(x-x_n)}{(x-x_0)\ldots(x_j-x_{j-1})(x_j-x_{j+1})\ldots(x_j-x_n)}.$$

gilt nach der Lagrangeschen Interpolationsformel

$$p(x) = \sum_{j=0}^{n} f(x_j) l_j(x) \in \mathbb{P}_n$$

und man erhält

$$Q(f) = \int_0^1 \sum_{j=0}^{n} f(x_j) l_j(x)\,dx = \sum_{j=0}^{n} f(x_j) \int_0^1 l_j(x)\,dx = \sum_{j=0}^{n} \alpha_j f(x_j).$$

Die *Gewichte*

$$\alpha_j = \int_0^1 l_j(x)\,dx$$

hängen dabei nur von den gewählten Stützstellen $x_0, \ldots, x_n$, aber nicht von der zu integrierenden Funktion f ab.

Beispiele (i) Für $n = 0$ und $x_0 = 0.5$ gilt $l_0 = 1$, $\int_0^1 1\,dx = 1$, und wir erhalten die Formel

$$\int_0^1 f(x)\,dx \approx f(0.5) = R(f),$$

die *Rechteck-* oder *Mittelpunktsregel* genannt wird.

(ii) Für $n = 1$ und $x_0 = 0$, $x_1 = 1$ ist $l_0(x) = 1 - x$ und $l_1(x) = x$. Für die Gewichte erhält man durch Integration $\alpha_0 = \alpha_1 = 0.5$ und damit die *Trapezregel*

$$\int_0^1 f(x)\,dx \approx \frac{1}{2}\big(f(0) + f(1)\big) = T(f).$$

(iii) Für $n = 2$, $x_0 = 0$, $x_1 = 0.5$, $x_2 = 1$ erhält man entsprechend die *Simpson Regel*

$$\int_0^1 f(x)\,dx \approx \frac{1}{6}\big(f(x_0) + 4f(0.5) + f(1)\big) = S(f),$$

die in der deutschsprachigen Literatur auch *Keplersche Fassregel* genannt wird.

Die in (ii) und (iii) hergeleiteten Formeln, bei denen zwei Stützstellen des Interpolationspolynoms an den Intervallenden gewählt werden, sind Beispiele für (abgeschlossene) *Newton-Cotes Formeln*. Die allgemeine Form dieser Formeln erhält man durch $x_j = j/n$.

$$\int_0^1 f(x)\,dx \approx \sum_{j=0}^{n} \alpha_j^{(n)} f\big(\frac{j}{n}\big).$$

n	$\alpha_j^{(n)}$					Name
1	1/2	1/2				Trapezregel
2	1/6	4/6	1/6			Simpson Regel
3	1/8	3/8	3/8	1/8		3/8 Regel
4	7/90	32/90	12/90	32/90	7/90	Milne Regel

Mit den Gewichten wie in dieser Tabelle erhält man für die Approximation des Integrals $\int_a^b f(x)\,dx$ die Formeln

$$(b-a)\sum_{j=0}^{n} \alpha_j f(x_j), \quad x_j = a + j\frac{b-a}{n} \text{ für } j = 0, 1, \ldots, n.$$

Beispiel (i) zeigt mit der Mittelpunktsformel eine *offene Newton-Cotes Formel*. Bei diesen werden die Randpunkte des Intervalls nicht für die Interpolation verwendet. Außer der Mittelpunktsformel werden diese Formeln nicht mehr benutzt.

Für die abgeschlossenen Formeln treten für $n \geq 8$ wechselnde Vorzeichen auf. Diese Formeln sind daher anfällig für Rundungsfehler. Man benutzt die Newton-Cotes Formeln daher nur für kleine n auf Teilintervallen von $[a, b]$ und summiert auf. Man erhält dann die *summierten Newton-Cotes Formeln* oder *zusammengesetzten Newton-Cotes Formeln*.

Für äquidistante Zerlegungen

$$h = \frac{b-a}{m}, \quad x_j = a + jh,\ j = 0, \ldots, m$$

erhalten wir die *summierte Rechtecksregel*

$$\int_a^b f(x)\,dx \approx h\sum_{j=1}^{m} f(x_j - \frac{h}{2}) = R_h(f)$$

sowie die *summierte Trapezregel*

$$\int_a^b f(x)\,dx \approx h\Big(\frac{1}{2}f(a) + \sum_{j=1}^{m-1} f(x_j) + \frac{1}{2}f(b)\Big) = T_h(f).$$

6.2 Fehler von Quadraturformeln Wir betrachten das Integral

$$I(f) = \int_0^1 f(x)\,dx$$

und eine zugehörige Quadraturformel, die nicht notwendig vom interpolatorischen Typ ist, mit

$$Q(f) = \sum_{i=0}^{n} \omega_i f(x_i), \quad x_i \in [0,1],\ \omega_i \in \mathbb{R} \text{ für } i = 0, \ldots, n.$$

Wir sagen, die Quadraturformel hat die *Fehlerordnung* m, wenn sie auf den Polynomen vom Grad $m-1$ exakt ist, genauer

$$(i)\quad I(p)-Q(p)=0 \text{ für alle Polynome } p\in\mathbb{P}_{m-1},$$
$$(ii)\quad I(p)-Q(p)\neq 0 \text{ für ein Polynome } p\in\mathbb{P}_{m}.$$

Nach Konstruktion ist eine auf n Stützstellen fußende interpolatorische Quadraturformel mindestens von der Ordnung $n+1$. Bei der Trapezregel ist dies auch die exakte Fehlerordnung wegen

$$I(x^2)=\int_0^1 x^2\,dx=\frac{1}{3},\quad T(x^2)=\frac{1}{2}(0^2+1^2)=\frac{1}{2}.$$

Dagegen haben die Rechteckregel und alle Newton-Cotes Formeln mit Stützstellen $0=x_0,\ldots,x_n=1$ mit ungeradem n eine um Eins erhöhte Ordnung. Für die Recheckformel erhält man nämlich

$$I(x)=\frac{1}{2}=R(x),$$

also $m=2$. Für die Simpson Regel ist

$$I(x^3)=\frac{1}{4},\quad S(x^3)=\frac{1}{6}\cdot 0+\frac{4}{6}\cdot 0.5^3+\frac{1}{6}\cdot 1=\frac{1}{4},\quad I(x^4)\neq S(x^4),$$

daher $m=4$. Für die allgemeine Newton-Cotes Formel mit ungeradem n bleibt der Beweis dem Leser überlassen.

Satz Es sei Q eine Quadraturformel der Fehlerordnung $m\geq 1$. Dann gibt es eine Funktion $K:[0,1]\to\mathbb{R}$, so dass der Fehler von Q die Darstellung

$$E(f)=\int_0^1 f(x)\,dx-Q(f)=\int_0^1 K(x)f^{(m)}(x)\,dx \tag{6.1}$$

für alle $f\in C^m[0,1]$ besitzt. Die Funktion K heißt *Peano-Kern* der Quadraturformel Q.

Beweis: Nach dem Satz von Taylor gilt

$$f(x)=\sum_{k=0}^{m-1}\frac{f^{(k)}(0)}{k!}x^k+\frac{1}{(m-1)!}\int_0^x f^{(m)}(t)(x-t)^{m-1}\,dt=p(x)+r(x)$$

und wegen $p\in\mathbb{P}_{m-1}$ folgt mit

$$(t)_+^k=\begin{cases} t^k & \text{falls } t\geq 0,\\ 0 & \text{falls } t<0,\end{cases}\qquad k\in\mathbb{N}_0,$$

$$\begin{aligned}
E(f)&=E(p)+E(r)=E(r)\\
&=\frac{1}{(m-1)!}\left\{\int_0^1\int_0^x f^{(m)}(t)(x-t)^{m-1}\,dt\,dx-\sum_{i=0}^n\omega_i\int_0^{x_i} f^{(m)}(t)(x_i-t)^{m-1}\,dt\right\}\\
&=\frac{1}{(m-1)!}\left\{\int_0^1\int_t^1 f^{(m)}(t)(x-t)^{m-1}\,dx\,dt-\sum_{i=0}^n\omega_i\int_0^1 f^{(m)}(t)(x_i-t)_+^{m-1}\,dt\right\}\\
&=\frac{1}{(m-1)!}\int_0^1\Big(\frac{1}{m}(1-t)^m-\sum_{i=0}^n\omega_i(x_i-t)_+^{m-1}\Big)f^{(m)}(t)\,dt,
\end{aligned}$$

demnach gilt (6.1) mit

$$K(x) = \frac{1}{(m-1)!}\left(\frac{1}{m}(1-x)^m - \sum_{i=0}^{n}\omega_i(x_i-x)_+^{m-1}\right).$$

□

Beispiele (i) Für die Trapezregel gilt $x_0 = 0$, $x_1 = 1$, $\omega_0 = \omega_1 = 0.5$, $m = 2$, also

$$K_T(x) = \frac{1}{2}(1-x)^2 - \frac{1}{2}(1-x) = -\frac{1}{2}x(1-x).$$

(ii) Für die Simpson-Regel haben wir $x_0 = 0$, $x_1 = 0.5$, $x_2 = 1$, $\omega_0 = \omega_2 = 1/6$, $\omega_1 = 2/3$ sowie $m = 4$, daher

$$K_S(x) = \frac{1}{3!}\Big(\frac{1}{4}(1-x)^4 - \frac{1}{6}(1-x)^3 - \frac{2}{3}\big(\frac{1}{2}-x\big)_+^3\Big).$$

Wenn wir wissen, dass der Peano-Kern das Vorzeichen nicht wechselt, so können wir in (6.1) mit dem Mittelwertsatz der Integralrechnung schließen

$$E(f) = f^{(m)}(\xi)\int_0^1 K(x)\,dx = c_m f^{(m)}(\xi), \quad \xi \in (0,1). \tag{6.2}$$

c_m heißt dann *Fehlerkonstante* der Quadraturformel Q.

Für die Trapezregel gilt $K_T(x) = -\frac{1}{2}x(1-x) \le 0$. Da T die Ordnung 2 besitzt, erhalten wir

$$E_T(f) = -\frac{1}{2}f''(\xi)\int_0^1 x(1-x)\,dx = -\frac{1}{12}f''(\xi).$$

Auch für die Simpson Regel haben wir ein einheitliches Vorzeichen,

$$K_S(x) = \frac{1}{3!}\big(\frac{1}{4}(1-x)^4 - \frac{1}{6}(1-x)^3 - \frac{2}{3}(\frac{1}{2}-x)_+^3 \le 0.$$

Durch Integration von K erhält man für den Fehler der Simpson Regel

$$E_S(f) = -\frac{1}{2880}f^{(4)}(\xi).$$

Man kann die Fehlerkonstante auch ohne explizite Kenntnis des Peano-Kerns bestimmen, wenn man nur weiß, dass er das Vorzeichen nicht wechselt. Es gilt nämlich $(x^m)^{(m)} = m!$ und aus (6.2) folgt

$$E(x^m) = m!\cdot c_m.$$

Für die Simpson-Regel gilt

$$E_S(x^4) = \int_0^1 x^4\,dx - \frac{1}{6}\big(0^4 + 4\cdot(\frac{1}{2})^4 + 1^4\big) = \frac{1}{5} - \frac{1}{6}\cdot\frac{5}{4} = -\frac{1}{120},$$

daher

$$c_4 = \frac{1}{4!}\cdot\Big(-\frac{1}{120}\Big) = -\frac{1}{2880}$$

Kommen wir nun zu Fehlerabschätzungen für die summierte Version der Newton-Cotes Formeln. Wir unterteilen dazu das Intervall $[a,b]$ äquidistant in n Intervalle der Länge $h = (b-a)/n$ und wenden die Formel auf jedes dieser Teilintervalle an. Um Fehlerabschätzungen für solch ein summiertes Verfahren zu bekommen, müssen wir erst einmal eine Fehlerabschätzung für ein kleines Intervall der Länge h herleiten. Auf

$$\int_\alpha^{\alpha+h} f(x)\,dx$$

wenden wir die Transformation $x = \alpha + ht$ an

$$\int_\alpha^{\alpha+h} f(x)\,dx = h\int_0^1 g(t)\,dt, \quad g(t) = f(\alpha+ht) = f(x).$$

Für das auf das Intervall $[0,1]$ transformierte Integral verwenden wir die Quadraturformel

$$Q(g) = \sum_{i=0}^{n} \omega_i g(x_i)$$

und erhalten auf dem Intervall $[\alpha, \alpha+h]$ entsprechend

$$Q_{[\alpha,\alpha+h]}(f) = h\sum_{i=0}^{n} \omega_i f(\alpha + hx_i).$$

Für den Fehler gilt

$$\begin{aligned} E_{[\alpha,\alpha+h]}(f) &= \int_\alpha^{\alpha+h} f(x)\,dx - Q_{[\alpha,\alpha+h]}(f) \\ &= h\Big(\int_0^1 g(t)\,dt - Q(g)\Big) = hE(g). \end{aligned} \tag{6.3}$$

In jedem Fall, also auch wenn der Peano-Kern nicht das Vorzeichen wechselt, erhalten wir aus (6.1) die Fehlerabschätzung

$$|E(g)| \le \max_{x\in[0,1]} |g^{(m)}| \int_0^1 |K(x)|\,dx = \tilde{c}_m \max_{x\in[0,1]} |g^{(m)}|.$$

Mit

$$\frac{d^m g}{dt^m} = \frac{d^m f}{dx^m}\cdot\Big(\frac{dx}{dt}\Big)^m = h^m \frac{d^m f}{dx^m}$$

folgt aus (6.3)

$$|E_{[\alpha,\alpha+h]}(f)| \le \tilde{c}_m h^{m+1} \max_{x\in[0,1]} |f^{(m)}|.$$

Für die summierte Quadraturformel

$$Q_h(f) = \sum_{i=1}^{n} Q_{[\alpha+(i-1)h,\alpha+ih]}(f)$$

erhält man aus der letzten Fehlerabschätzung mit der Dreiecksungleichung

$$\begin{aligned} \Big|\int_a^b f(x) - Q_h(f)\Big| &= \left|\sum_{i=1}^{n}\left(\int_{a+(i-1)h}^{a+ih} f(x)\,dx - Q_{[a+(i-1)h,a+ih]}(f)\right)\right| \\ &\le \sum_{i=1}^{n} |E_{[a+(i-1)h,a+ih]}(f)| \\ &\le \sum_{i=1}^{n} h^{m+1}\tilde{c}_m \max_x |f^{(m)}(x)| \\ &= nh^{m+1}\tilde{c}_m \|f^{(m)}\|_\infty = h^m(b-a)\tilde{c}_m\|f^{(m)}\|_\infty. \end{aligned}$$

Für die summierte Trapezregel erhält man daher für den Fehler

$$\Big|\int_a^b f(x)\,dx - T_h(f)\Big| \le \frac{h^2}{12}(b-a)\|f''\|_\infty,$$

und für die summierte Simpson Regel

$$\Big|\int_a^b f(x)\,dx - S_h(f)\Big| \le \frac{h^4}{2880}(b-a)\|f^{(4)}\|_\infty.$$

6.3 Das Romberg-Verfahren In diesem Abschnitt approximieren wir das Integral

$$\int_a^b f(x)\,dx$$

durch die summierte Trapezregel

$$T_h(f) = \frac{h}{2}\left(f(a) + 2\sum_{i=1}^{n-1} f(x_i) + f(b)\right), \quad h = \frac{b-a}{n},\ x_i = a + ih.$$

Wir werden eine asymptotische Entwicklung in h für dieses Verfahren beweisen, die dann den Einsatz eines Extrapolationsverfahrens rechtfertigt.

Satz [Euler-Maclaurinsche Summenformel] Sei $g \in C^{2m+2}[0,n]$, $n \in \mathbb{N}$, und

$$T_1(g) = \frac{1}{2}g(0) + \sum_{i=1}^{n-1} g(j) + \frac{1}{2}g(n)$$

die summierte Trapezformel zur Schrittweite $h = 1$. Dann gibt es Konstanten $C_1, \ldots, C_m \in \mathbb{R}$ und eine beschränkte Funktion $\phi_{2m+2} : \mathbb{R} \to \mathbb{R}$, so dass

$$\int_0^n g(t)\,dt = T_1(g) - \sum_{j=1}^{m} C_j\big(g^{(2j-1)}(n) - g^{(2j-1)}(0)\big) + R_m(g) \tag{6.4}$$

gilt mit

$$R_m(g) = \int_0^n \big(\phi_{2m+2}(t) - \phi_{2m+2}(0)\big)g^{(2m+2)}(t)\,dt. \tag{6.5}$$

Beweis: Mit der periodischen Funktion

$$\phi_1(t) = t - \frac{1}{2}, \quad 0 \le t < 1, \quad \phi_1(t+1) = \phi_1(t), \quad t \in \mathbb{R},$$

erhält man durch partielle Integration für $j = 1, \ldots, n$

$$\begin{aligned}\int_{j-1}^{j} g(t)\,dt &= \phi_1(t)g(t)\big|_{j-1}^{j} - \int_{j-1}^{j} \phi_1(t)g'(t)\,dt \\ &= \frac{1}{2}\big(g(j) + g(j-1)\big) - \int_{j-1}^{j} \phi_1(t)g'(t)\,dt,\end{aligned}$$

und durch Summation

$$\int_0^n g(t) = T_1(g) - \int_0^n \phi_1(t)g'(t)\,dt. \tag{6.6}$$

Um das Integral auf der rechten Seite umzuformen, benötigen wir die Stammfunktionen $\phi_j : \mathbb{R} \to \mathbb{R}$, $j \ge 2$, mit $\phi_j' = \phi_{j-1}$.

ϕ_1 besitzt die Fourier-Reihe

$$\phi_1(t) = -2\sum_{k=1}^{\infty} \frac{\sin(2\pi kt)}{2\pi k}.$$

Durch wiederholte Anwendung gliedweiser Integration erhält man die Darstellungen

$$\phi_{2j}(t) = 2\cdot(-1)^{j+1}\sum_{k=1}^{\infty} \frac{\cos(2\pi kt)}{(2\pi k)^{2j}}, \quad j \ge 1,$$

$$\phi_{2j+1}(t) = 2\cdot(-1)^{j+1}\sum_{k=1}^{\infty} \frac{\sin(2\pi kt)}{(2\pi k)^{2j+1}}, \quad j \ge 0,$$

Nach dem Majorantenkriterium konvergieren die Reihen ϕ_j für $j \geq 2$ gleichmäßig in $\mathbb{R}$. Daher gilt $\phi_j' = \phi_{j-1}$ für $j \geq 3$. Da die Fourierreihe von ϕ_1 für jedes abgeschlossene Intervall, das keine ganzzahligen Punkte enthält, gleichmäßig gegen ϕ_1 konvergiert, gilt auch $\phi_2'(t) = \phi_1(t)$ für alle $t \notin \mathbb{Z}$. Damit folgt aus (6.6)

$$\begin{aligned}\int_0^n g(t)\,dt - T_1(g) &= -\int_0^n \phi_1(t)g'(t)\,dt \\ &= -\phi_2(t)g'(t)\big|_0^n + \int_0^n \phi_2(t)g''(t)\,dt = \dots \\ &= \sum_{j=2}^{2m+1} (-1)^{j+1}\phi_j(t)g^{(j-1)}(t)\big|_0^n - \int_0^n \phi_{2m+1}(t)g^{(2m+1)}(t)\,dt,\end{aligned}$$

und mit nochmaliger partieller Integration

$$\int_0^n \phi_{2m+1}(t)g^{(2m+1)}(t)\,dt = -\int_0^n \big(\phi_{2m+2}(t) - \phi_{2m+2}(0)\big)g^{(2m+2)}(t)\,dt.$$

Offensichtlich gilt

$$\begin{aligned}\phi_{2j+1}(0) = \phi_{2j+1}(n) &= 0, \quad j \in \mathbb{N}, \\ \phi_{2j}(0) = \phi_{2j}(n) &= (-1)^{j+1}\sum_{k=1}^{\infty} \frac{2}{(2\pi k)^{2j}}, \quad j \geq 1.\end{aligned}$$

Daher folgt die Entwicklung (6.4) mit $C_j = \phi_{2j}(0)$. □

Bemerkung Die Zahlen

$$B_j = (-1)^{j+1}(2j)!\,\phi_{2j}(0), \quad j \in \mathbb{N},$$

heißen *Bernoulli-Zahlen*. Mit ihnen lautet die Euler Maclaurinsche Summenformel in der üblichen Darstellung

$$\int_0^n g(t)\,dt - T_1(g) = \sum_{j=1}^{m} \frac{(-1)^j B_j}{(2j)!}\big(g^{(2j-1)}(n) - g^{(2j-1)}(0)\big) + R_m(g).$$

Als Folgerung erhalten wir die asymptotische Entwicklung des Fehlers der summierten Trapezregel:

Korollar Ist $f \in C^{2m+2}[a,b]$ und $h = (b-a)/n$, $n \in \mathbb{N}$, so gilt

$$T_h(f) = \int_a^b f(x)\,dx + \sum_{j=1}^{m} c_j h^{2j} + \psi_{m+1}(h)h^{2m+2}, \tag{6.7}$$

wobei $|\psi_{m+1}(h)| \leq M$ mit einer von h unabhängigen Konstanten M gilt.

Beweis: Mit $x = a + th$ und $g(t) = f(a + th)$ gilt

$$\int_a^b f(x)\,dx = h\int_0^n g(t)\,dt \quad \text{und} \quad T_h(f) = hT_1(g),$$

und daher folgt mit $g^{(j)}(t) = h^j f^{(j)}(a + th)$ aus der Euler-Maclaurinschen Summenformel (6.4)

$$T_h(f) = \int_a^b f(x)\,dx + \sum_{j=1}^{m} C_j\big(f^{(2j-1)}(b) - f^{(2j-1)}(a)\big)h^{2j} - hR_m(g),$$

wobei

$$hR_m(g) = h\int_0^n \big(\phi_{2m+2}(t) - \phi_{2m+2}(0)\big)g^{(2m+2)}(t)\,dt$$
$$= h^{2m+2}\int_a^b \left(\phi_{2m+2}\left(\frac{x-a}{h}\right) - \phi_{2m+2}(0)\right)f^{(2m+2)}(x)\,dx$$

gilt. Da ϕ_{2m+2} als stetige und periodische Funktion beschränkt ist, ist auch

$$\psi_{m+1}(h) = -\int_a^b \left(\phi_{2m+2}\left(\frac{x-a}{h}\right) - \phi_{2m+2}(0)\right)f^{(2m+2)}(x)\,dx$$

beschränkt. □

Wir verwenden das letzte Korollar, um durch Extrapolation zu rasch konvergierenden Integrationsmethoden zu gelangen. Vernachlässigt man bei der Entwicklung von $T_h(f)$ in (6.7) das Restglied, so lässt sich $T_h(f)$ als ein Polynom in h^2 auffassen, das für $h = 0$ den Wert $c_0 := \int_a^b f(x)\,dx$ liefert. Wir bestimmen daher zu verschiedenen Schrittweiten $h_0, h_1, \ldots, h_m > 0$ die Trapezsummen $T_h(f)$, $j = 0, \ldots, m$. Dann gibt es ein eindeutiges Polynom $p \in \mathbb{P}_m$ in h^2 mit

$$p(h_j^2) = T_{h_j}(f), \quad j = 0, \ldots, m.$$

$p(0)$ ist dann eine verbesserte Näherung von $\int_a^b f(x)\,dx$.

Da nur der Wert $p(0)$ bestimmt werden soll, bietet sich das Neville-Schema zu seiner Berechnung an. Sei $h_0 = b - a$, $h_j = h_0/n_j$, $n_j < n_{j+1}$, $j = 1, \ldots, m$, und sei $T_{jj} = T_{h_j}(f)$ die Trapezsumme zur Schrittweite h_j. Weiter sei $p_{ij}(h)$, $0 \le i \le j \le m$, das Polynom vom Grad $j - i$ in h^2, für das gilt

$$p_{ij}(h_k) = T_{kk}, \quad k = i, \ldots, j.$$

Für die extrapolierten Werte $T_{ij} = p_{ij}(0)$ gilt dann nach dem Neville-Schema

$$T_{ij} = T_{i+1\,j} + \frac{T_{i+1\,j} - T_{i\,j-1}}{\left(\frac{h_i}{h_j}\right)^2 - 1}, \quad 0 \le i < j \le m. \tag{6.8}$$

Dieses Verfahren wurde erstmals von Romberg im Jahre 1955 vorgeschlagen mit den Schrittweiten $h_i = (b-a)\cdot 2^{-i}$. In diesem Fall bekommen wir die einfacheren Formeln

$$T_{ij} = \frac{4^{j-i}T_{i+1\,j} - T_{i\,j-1}}{4^{j-i} - 1}.$$

Für die Näherungen von $\int_a^b f(x)\,dx$

$$\begin{array}{ccccc} T_{00} & & & & \\ T_{11} & T_{01} & T_{02} & & \\ T_{22} & T_{12} & T_{13} & T_{03} & T_{04} \\ T_{33} & T_{23} & T_{24} & & \\ T_{44} & T_{34} & & & \end{array}$$

gilt dann: T_{00} ist der Wert der Trapezregel auf dem Intervall $[a, b]$, T_{01} der Wert der Simpson-Regel und T_{02} der Wert der Milne-Regel. Die T_{0j} stellen für $j \ge 3$ keine Newton-Cotes Formeln dar.

Die Romberg-Folge $h_i = (b-a)\cdot 2^{-i}$ hat den Vorteil, dass man für die Berechnung von $T_{i+1\,i+1}$ auf die schon bei T_{ii} berechneten Funktionswerte zurückgreifen kann. Der Nachteil besteht darin, dass die Zahl der Auswertungen trotz des gerade beschriebenen Spareffekts stark ansteigt.

Eine gute Alternative zur Romberg-Folge stellt daher die *Bulirsch-Folge*

$$1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{6} \ldots \times h_0, \quad h_0 = b - a,$$

dar. Auch hier kann man nach den ersten Schritten viele alte Auswertungen wiederverwenden.

Beispiel Für das Integral

$$\int_0^1 e^x \sin 5x \, dx \quad (= 0.05623058659667)$$

erhalten wir für $T_{05} - \int_0^1 f(x)\,dx$ bei Verwendung der

Romberg-Folge	$4.4 \cdot 10^{-11}$	33 Funktionsauswertungen
Bulirsch-Folge	$1.8 \cdot 10^{-10}$	20 Funktionsauswertungen

Die Bulirsch-Folge ist also zumindest in diesem Beispiel klar vorzuziehen.

Für die Theorie der Romberg-Integration muss an die Wahl der Schrittweiten h_j Folgendes vorausgesetzt werden: Es gibt eine Konstante $\eta < 1$, so dass $h_j \leq \eta h_{j-1}$ für alle j erfüllt ist. Für die Romberg-Folge ist diese Bedingung mit $\eta = \frac{1}{2}$ erfüllt und für die Bulirsch-Folge mit $\eta = \frac{3}{4}$. Unter dieser Voraussetzung kann man die verbesserte Konvergenz des Extrapolationsverfahren beweisen. Wir zitieren hier nur ein spezielleres Resultat:

Satz Seien T_{ik} die Werte des Neville-Schemas zur Bestimmung von $\int_a^b f(x)\,dx$ unter Verwendung von Trapezsummen mit der Romberg- oder der Bulirsch-Folge. Für genügend oft auf dem Intervall $[a, b]$ stetig differenzierbarem f gilt: Für jedes $k \in \mathbb{N}$ gibt es ein $\xi \in [a, b]$ mit

$$T_{ik} - \int_a^b f(x)\,dx = (b-a)\frac{B_{k+1}}{(2k+2)!} f^{(2k+2)}(\xi) \prod_{j=0}^{k} h_{i-j}^2.$$

6.4 Quadraturformeln von Gauß Bisher haben wir in den Newton-Cotes Formeln eine äquidistante Verteilung der Stützstellen auf dem Intervall $[a, b]$ zugrunde gelegt. In diesem Abschnitt fragen wir, ob durch eine geschicktere Wahl der Stützstellen noch bessere Fehlerordnungen erzielt werden können.

Wit studieren die numerische Approximation von Integralen der Form

$$I(f) = \int_a^b \omega f(x)\,dx$$

mit der *Gewichtsfunktion* ω. In diesem Abschnitt kann das Integral auch ein uneigentliches sein, d.h. die Grenzen können auch die Werte $-\infty$ bzw. ∞ sein, sofern die Gewichtsfunktion dies zulässt. Das wichtigste Beispiel ist $\omega(x) = \exp(-x^2)$, bei der im obigen Integral über ganz $\mathbb{R}$ integriert werden kann.

Genauer setzen wir an die Gewichtsfunktion voraus:

(i) ω ist messbar mit $\omega \geq 0$ in (a, b).

(ii) Die *Momente*

$$\mu_k = \int_a^b x^k \omega(x)\,dx$$

existieren für alle $k \in \mathbb{N}_0$ und $\mu_0 = \int \omega(x)\,dx > 0$.

(iii) Für jedes Polynom $p(x)$ mit

$$\int_a^b \omega(x)p(x)\,dx = 0 \quad \text{und} \quad p(x) \geq 0 \text{ in } (a,b)$$

gilt

$$p(x) = 0 \quad \text{in } (a,b).$$

Diese Voraussetzungen sind bei einem beschränkten Integrationsintervall für $\omega(x) = 1$ erfüllt.

Aufgabe Gesucht sind Stützstellen $x_1, \ldots, x_n \in [a,b]$ und Gewichte $\omega_1, \ldots, \omega_n \in \mathbb{R}$, so dass die Formel

$$I_n(f) = \sum_{i=1}^{n} \omega_i f(x_i)$$

für glattes f von möglichst hoher Ordnung ist, d.h. $I(p) = I_n(p)$ für alle $p \in \mathbb{P}_k$.

Im Folgenden verwenden wir die Bilinearform

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

Nach den Voraussetzungen (i) und (iii) gilt $(p,p) > 0$ für alle Polynome p mit $p \neq 0$. Auf dem Raum der Polynome ist $(\cdot,\cdot)$ daher ein Skalarprodukt.

Mit $\tilde{\mathbb{P}}_i$ bezeichnen wir die Menge der Polynome vom Grad $i \geq 0$ mit führendem Koeffizienten $a_i = 1$.

Satz [Eigenschaften orthogonaler Polynome] Es gibt eindeutig bestimmte Polynome $p_i \in \tilde{\mathbb{P}}_i$ mit

$$(p_i, p_k) = 0 \quad \text{für } i \neq k.$$

Diese genügen den Rekursionsformeln

$$p_0(x) = 1$$

$$p_{i+1}(x) = (x - \delta_{i+1})p_i(x) - \gamma_i^2 p_{i-1}(x), \quad i \geq 0,$$

mit $p_{-1}(x) = 0$ und

$$\delta_{i+1} = \frac{(xp_i, p_i)}{(p_i, p_i)}, \quad i \geq 0,$$

$$\gamma_{i+1}^2 = \begin{cases} 0 & \text{für } i = 0, \\ \dfrac{(p_i, p_i)}{(p_{i-1}, p_{i-1})} & \text{für } i \geq 1. \end{cases}$$

Beweis: Wir verwenden das Schmidtsche Orthogonalisierungsverfahren. $p_0 = 1$ ist klar. Seien $p_j \in \tilde{\mathbb{P}}_j$ für $0 \leq j \leq i$ bereits konstruiert. Da diese p_j eine Basis von $\mathbb{P}_i$ bilden und $p_{i+1} \in \tilde{\mathbb{P}}_{i+1}$ folgt

$$p_{i+1}(x) = (x + c_i)p_i(x) + c_{i-1}p_{i-1}(x) + \ldots + c_0 p_0(x).$$

Durch geeignete Wahl der c_j müssen wir die Bedingungen

$$p_{i+1} \perp p_j \quad \text{für } j = 0, \ldots, i$$

erfüllen. Wegen der Induktionsvoraussetzung $(p_k, p_l) = 0$ für $0 \leq k < l \leq i$ gilt für die c_i

$$\big((x + c_i)p_i, p_i\big) = 0$$

$$(xp_i, p_j) + c_j(p_j, p_j) = 0 \quad \text{für } j = 0, \ldots, i-1.$$

Nach Voraussetzung (iii) gilt $(p_j, p_j) > 0$, womit die c_j eindeutig bestimmt sind. Für $j \leq i-2$ gilt

$$(xp_i, p_j) = (p_i, xp_j) = 0 \quad \text{wegen } xp_j \in \tilde{\mathbb{P}}_{j+1}$$

und damit $c_j = 0$ für diese j. Für $j = i$ und $j = i-1$ erhalten wir

$$c_i = \frac{(xp_i, p_i)}{(p_i, p_i)},$$

$$c_{i-1} = \frac{(xp_{i-1}, p_i)}{(p_{i-1}, p_{i-1})}$$

Nach Induktionsvoraussetzung gilt

$$p_i(x) = (x - \delta_i)p_{i-1}(x) - \gamma_i^2 p_{i-2}(x)$$

und damit

$$(p_i, p_i) = (xp_{i-1}, p_i).$$

Damit ist wie in der Behauptung angegeben $c_i = \delta_{i+1}$ und $c_{i-1} = \gamma_{i+1}^2 > 0$. □

Korollar Es gilt $(p_n, p) = 0$ für alle $p \in \mathbb{P}_{n-1}$.

Satz [Nullstellen der orthogonalen Polynome] Die Nullstellen $x_1, \ldots, x_n$ von p_n sind einfach, reell und liegen alle im Intervall (a, b).

Beweis: Seien

$$a < x_1 < \ldots < x_l < b$$

die Nullstellen von p_n im Intervall (a, b), in denen p_n das Vorzeichen wechselt. Wenn $l < n$, so setze

$$p(x) = \prod_{i=1}^{l} (x - x_i) \in \mathbb{P}_l.$$

p wechselt in jedem x_i das Vorzeichen, pp_n daher nicht. Dies ist ein Widerspruch zu $(p_n, p) = 0$. □

Lemma Für beliebige $x_1 < x_2 < \ldots < x_n$ ist die Matrix

$$A = \begin{bmatrix} p_0(x_1) & \cdots & p_0(x_n) \\ \vdots & & \vdots \\ p_{n-1}(x_1) & \cdots & p_{n-1}(x_n) \end{bmatrix}$$

regulär.

Beweis: Wäre A singulär, so gäbe es ein $c \in \mathbb{R}^n \setminus \{0\}$ mit

$$q(x_j) = \sum_{i=0}^{n-1} c_i p_i(x_j) = 0 \quad \text{für } j = 1, \ldots, n.$$

Dann wäre wegen $\operatorname{grad} q \leq n-1$ auch $q = 0$. Dies ist ein Widerspruch zur linearen Unabhängigkeit der p_i. □

Satz [Gaußsche Quadraturformel] (a) Seien $x_1, \ldots, x_n$ die Nullstellen von p_n und $\omega_1, \ldots, \omega_n$ die Lösung des linearen Gleichungssystems

$$\sum_{i=1}^{n} p_k(x_i)\omega_i = \begin{cases} (p_0, p_0) & \text{für } k = 0 \\ 0 & \text{für } k = 1, \ldots, n-1 \end{cases}.$$

Dann gilt $\omega_i > 0$ sowie

$$\int_a^b \omega(x)p(x)\,dx = \sum_{i=1}^{n} \omega_i p(x_i) \quad \text{für alle } p \in \mathbb{P}_{2n-1}.$$

(b) Es gibt keine Quadraturformel mit n Stützstellen, die auf $\mathbb{P}_{2n}$ exakt ist.

Beweis: (a) Die ω_i sind als Lösung des linearen Gleichungssystems mit Matrix A aus dem letzten Lemma eindeutig bestimmt. Die Formel

$$I_n(r) = \sum_{i=1}^{n} \omega_i r(x_i)$$

ist nach Konstruktion exakt auf dem Raum $\mathbb{P}_{n-1}$, denn mit

$$r(x) = \sum_{i=0}^{n-1} \alpha_i p_i(x)$$

folgt

$$\int_a^b \omega(x)r(x)\,dx = (r, p_0) = \sum_{i=1}^{n-1} \alpha_i (p_i, p_0) = \alpha_0 (p_0, p_0) = I_n(r).$$

Jedes $p \in \mathbb{P}_{2n-1}$ lässt sich eindeutig in der Form

$$p(x) = p_n(x)q(x) + r(x), \quad q, r \in \mathbb{P}_{n-1}$$

schreiben. Wegen $p_n \perp q$ folgt

$$\int_a^b \omega(x)p(x)\,dx = (p, p_0) = (p_n, q) + (r, p_0) = (r, p_0) = I_n(r).$$

Damit ist die Quadraturformel exakt auf $\mathbb{P}_{2n-1}$.

Für $i \in \{1, \ldots, n\}$ setze

$$q(x) = \prod_{j=1,\, j\neq i}^{n} (x - x_i)^2 \in \mathbb{P}_{2n-1}.$$

Wegen $q \geq 0$, $q \neq 0$, folgt

$$0 < \int_a^b \omega(x)q(x)\,dx = I_n(q) = \omega_i q(x_i)$$

und damit $\omega_i > 0$.

(b) Angenommen, es gibt eine Quadraturformel $\tilde{I}_n$ mit paarweise verschiedenen Stützstellen $x_1, \ldots, x_n \in [a, b]$, die auf $\mathbb{P}_{2n}$ exakt ist. Das Polynom

$$q(x) = \prod_{i=1}^{n} (x - x_i)^2 \in \mathbb{P}_{2n}.$$

erfüllt $q \geq 0$, $q \neq 0$, daher

$$0 < \int_a^b \omega(x)q(x)\,dx = \tilde{I}_n(q) = 0,$$

was einen Widerspruch bedeutet. □

Die Gauß-Formel I_n besitzt daher den optimalen Exaktheitsgrad.

Satz Sei $f \in C^{2n}[a,b]$. Dann gilt für die Gauß-Formel I_n

$$\int_a^b \omega(x) f(x)\,dx - \sum_{i=1}^{n} \omega_i f(x_i) = \frac{f^{(2n)}(\xi)}{(2n)!}(p_n, p_n)$$

für ein $\xi \in (a,b)$.

Beweis: Sei $p \in \mathbb{P}_{2n-1}$ das hermitesche Interpolationspolynom (5.10) an den Stützstellen $x_1, \ldots, x_n$. Nach (5.11) gilt für den Interpolationsfehler

$$r(x) = f(x) - p(x) = \frac{f^{(2n)}(\eta(x))}{(2n)!}(x - x_1)^2 \ldots (x - x_n)^2.$$

mit einem $\eta(x) \in [a,b]$. $p_n \in \tilde{\mathbb{P}}_n$ hat die Nullstellen $x_1, \ldots, x_n$ und der führende Koeffizient von p_n ist 1. Daher ist $p_n(x) = (x - x_1) \ldots (x - x_n)$ und aus der letzten Formel erhalten wir

$$r(x) = \frac{f^{(2n)}(\eta(x))}{(2n)!} p_n^2(x). \tag{6.9}$$

Aus der Regel von de l'Hospital folgt, dass

$$f^{(2n)}(\eta(x)) = \frac{f(x) - p(x)}{p_n^2(x)}(2n)!$$

stetig in $[a,b]$ ist. Wir multiplizieren (6.9) mit der Gewichtsfunktion ω, integrieren und können wegen $\omega(x)p_n^2(x) \geq 0$ den Mittelwertsatz der Integralrechnung anwenden

$$\int_a^b \omega(x) r(x)\,dx = \frac{f^{(2n)}(\xi)}{(2n)!} \int_a^b \omega(x) p_n^2(x)\,dx = \frac{f^{(2n)}(\xi)}{(2n)!}(p_n, p_n), \quad \xi \in [a,b]. \tag{6.10}$$

Wegen $p \in \mathbb{P}_{2n-1}$ folgt

$$\begin{aligned}
\int_a^b \omega(x) r(x)\,dx &= \int_a^b \omega(x) f(x)\,dx - \int_a^b \omega(x) p(x)\,dx \\
&= \int_a^b \omega(x) f(x)\,dx - \sum_{i=1}^{n} \omega_i p(x_i) \\
&= \int_a^b \omega(x) f(x)\,dx - \sum_{i=1}^{n} \omega_i f(x_i) = I(f) - I_n(f).
\end{aligned}$$

Die Behauptung folgt aus der letzten Identität und (6.10). □

Man kann daher auch nichtglatte Integranden behandeln, wenn man die Singularität in ω steckt.

Sei nun $\omega = 1$ und $[a,b] = [-1,1]$. Dann gilt

$$p_k(x) = \frac{k!}{(2k)!} \frac{d^k}{dx^k}(x^2 - 1)^k, \quad k \in \mathbb{N}_0.$$

Die p_k heißen auch *Legendre-Polynome*. Wir zeigen, dass dies die gesuchten Polynome sind.

Der führende Koeffizient von p_k ist

$$\frac{k!}{(2k)!} \frac{d^k}{dx^k} x^{2k} = x^k,$$

also $p_k \in \tilde{\mathbb{P}}_k$. Für $l < k$ folgt mit mehrfacher Anwendung der partiellen Integration

$$\begin{aligned}\int_{-1}^{1} p_l(x)p_k(x)\,dx &= \frac{l!k!}{(2l)!(2k)!}\int_{-1}^{1}\frac{d^l}{dx^l}(x^2-1)^l\frac{d^k}{dx^k}(x^2-1)^k\,dx\\ &= -N(l,k)\int_{-1}^{1}\frac{d^{l+1}}{dx^{l+1}}(x^2-1)^l\frac{d^{k-1}}{dx^{k-1}}(x^2-1)^k\,dx\\ &= \ldots = 0.\end{aligned}$$

Durch Bestimmung der Nullstellen der p_k erhalten wir daher die Gauß-Formeln zur Approximation von $\int_{-1}^{1} f(x)\,dx$:

n	ω_i	x_i
1	$\omega_1 = 2$	$x_1 = 0$
2	$\omega_1 = \omega_2 = 1$	$x_2 = -x_1 = \frac{1}{\sqrt{3}}$
3	$\omega_1 = \omega_3 = \frac{5}{9}$	$x_3 = -x_1 = \sqrt{\frac{3}{5}}$
	$\omega_2 = \frac{8}{9}$	$x_2 = 0$

Beispiel Wir approximieren

$$\int_1^2 \frac{dx}{x} = \ln 2 = 0.693147\ldots$$

Für die Quadraturformel mit $n = 3$ erhalten wir

$$\begin{aligned} x_1 &= 1.112702 \quad \omega_1 = \frac{5}{18}\\ x_2 &= 1.5 \qquad\quad \omega_2 = \frac{4}{9}\\ x_3 &= 1.887298 \quad \omega_3 = \frac{5}{18}\end{aligned}$$

und daher

$$I_3\Big(\frac{1}{x}\Big) = 0.693122.$$

Damit reicht I_3 aus, um eine vierstellige Logarithmentafel aufzustellen.

7 Theorie der Eigenwertprobleme

7.1 Definition und Eigenschaften Sei $A \in \mathbb{C}^{n\times n}$. $\lambda \in \mathbb{C}$ heißt *Eigenwert* (=Rechtseigenwert) von A, wenn

$$Ax = \lambda x \quad \text{für ein } x \in \mathbb{C}^n \setminus \{0\}.$$

x ist dann *Eigenvektor* zu λ.

λ ist daher genau dann Eigenwert, wenn die Matrix $A - \lambda I$ singulär ist und damit das *charakteristische Polynom* von A

$$\phi(\mu) = \det(A - \mu I)$$

in λ eine Nullstelle besitzt.

Die Größe

$$\sigma(\lambda) = \text{Vielfachheit der Nullstelle } \lambda \text{ in } \phi$$

heißt *algebraische Vielfachheit* von λ. Der Vektorraum

$$L(\lambda) = \{x \in \mathbb{C}^n : Ax = \lambda x\}$$

heißt *Eigenraum* zu λ. Ferner heißt

$$\rho(\lambda) = \dim L(\lambda)$$

geometrische Vielfachheit von λ. $\rho(\lambda)$ ist die Zahl der linear unabhängigen Eigenvektoren zu λ.

$$\sigma(A) = \{\lambda \in \mathbb{C} : \lambda \text{ ist Eigenwert von } A\}$$

heißt *Spektrum* von A.

Satz (a) Ist $p(\mu)$ ein Polynom und gilt $Ax = \lambda x$ für ein $x \neq 0$, so besitzt $p(A)$ ebenfalls den Eigenvektor x zum Eigenwert $p(\lambda)$.

(b) λ ist genau dann Eigenwert von A, wenn $\overline{\lambda}$ Eigenwert von A^H ist.

(c) Ähnliche Matrizen besitzen das gleiche charakteristische Polynom, also auch die gleichen Eigenwerte. Wenn

$$B = T^{-1}AT$$

und A besitzt den Eigenwert λ mit Eigenvektor x, so besitzt B den Eigenwert λ mit Eigenvektor $T^{-1}x$.

Beweis: (a) Aus $Ax = \lambda x$ folgt $A^k x = \lambda^k x$ und

$$p(A)x = a_m A^m x + \ldots + a_0 x = p(\lambda)x.$$

(b) $\det(A^H - \overline{\lambda} I) = \det(A - \lambda I)^H = \overline{\det(A - \lambda I)}$.

(c) Mit dem Determinantenmultiplikationssatz folgt

$$\begin{aligned}\det(B - \lambda I) &= \det(T^{-1}AT - \lambda I) = \det\left(T^{-1}(A - \lambda I)T\right)\\ &= \det T^{-1} \det(A - \lambda I) \det T = \det(A - \lambda I).\end{aligned}$$

Ferner gilt

$$BT^{-1}x = T^{-1}Ax = T^{-1}(\lambda x) = \lambda T^{-1}x.$$

□

Beispiel Das Jordan-Kästchen der Länge ν zum Eigenwert λ ist definiert durch

$$C_\nu(\lambda) = \begin{bmatrix} \lambda & 1 & & 0 \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 0 & & & \lambda \end{bmatrix} \in \mathbb{C}^{\nu\times\nu}. \tag{7.1}$$

Wegen

$$\det(C_\nu(\mu) - \lambda I) = (\mu - \lambda)^\nu$$

ist λ Eigenwert mit $\sigma(\lambda) = \nu$, aber $x = e_1$ ist einziger Eigenvektor von C_ν, also $\rho(\lambda) = 1$.

Damit ist gezeigt, dass algebraische und geometrische Vielfachheit nicht übereinstimmen müssen.

Lemma Es gilt $\rho(\lambda) \leq \sigma(\lambda)$.

Beweis: Sei $\rho = \rho(\lambda)$ und $x_1, \ldots, x_\rho$ seien die linear unabhängigen Eigenvektoren von A zu λ. Ergänze $x_1, \ldots, x_\rho$ durch $x_{\rho+1}, \ldots, x_n$ zu einer Basis des $\mathbb{C}^n$. Sei

$$T = [x_1 | \cdots | x_n].$$

Für $1 \leq i \leq \rho$ gilt $Te_i = x_i$ und damit $T^{-1}x_i = e_i$, also

$$T^{-1}ATe_i = T^{-1}Ax_i = T^{-1}(\lambda x_i) = \lambda e_i, \quad 1 \leq i \leq \rho.$$

Also gilt

$$T^{-1}AT = \left[\begin{array}{ccc|c} \lambda & & 0 & \\ & \ddots & & * \\ 0 & & \lambda & \\ \hline & 0 & & * \\ & & & \end{array}\right]$$

und damit

$$\det(T^{-1}AT - \mu I) = (\lambda - \mu)^\rho \det R(\mu).$$

□

7.2 Die Jordansche Normalform Es sei an die Definition des Jordan-Kästchens $C_\nu(\lambda)$ in (7.1) erinnert.

Satz Sei $A \in \mathbb{C}^{n\times n}$, $\lambda_1, \ldots, \lambda_k$ seien die Eigenwerte von A mit geometrischen bzw. algebraischen Vielfachheiten $\rho(\lambda_i)$ und $\sigma(\lambda_i)$. Zu jedem λ_i gibt es Zahlen $\nu_1^{(i)}, \ldots, \nu_{\rho(\lambda_i)}^{(i)}$ mit

$$\sigma(\lambda_i) = \nu_1^{(i)} + \ldots + \nu_{\rho(\lambda_i)}^{(i)}$$

und eine reguläre Matrix $T \in \mathbb{C}^{n\times n}$ mit $J = T^{-1}AT$,

$$J = \begin{bmatrix} C_{\nu_1^{(1)}}(\lambda_1) & & & & & \\ & \ddots & & & & 0 \\ & & C_{\nu_{\rho(\lambda_1)}^{(1)}}(\lambda_1) & & & \\ & & & C_{\nu_1^{(2)}}(\lambda_2) & & \\ & 0 & & & \ddots & \\ & & & & & C_{\nu_{\rho(\lambda_k)}^{(k)}}(\lambda_k) \end{bmatrix}$$

J ist bis auf die Reihenfolge der Jordan-Kästchen eindeutig bestimmt.

Beweis: Den Beweis findet man in jedem Lehrbuch der linearen Algebra. □

Entsprechend den Jordan-Kästchen partitionieren wir die Matrix T

$$T = \left[T_1^{(1)} | \ldots | T_{\rho(\lambda_1)}^{(1)} | T_1^{(2)} | \ldots | T_{\rho(\lambda_k)}^{(k)}\right].$$

Wegen $AT = TJ$ folgt

$$AT_j^{(i)} = T_j^{(i)} C_{\nu_j^{(i)}}(\lambda_i).$$

Mit

$$T_j^{(i)} = \left[t_1 \middle| \dots \middle| t_{\nu_j^{(i)}}\right]$$

erhalten wir

$$(A - \lambda_i I)\left[t_1 \middle| \dots \middle| t_{\nu_j^{(i)}}\right] = \left[t_1 \middle| \dots \middle| t_{\nu_j^i}\right] \begin{bmatrix} 0 & 1 & & 0 \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 0 & & & 0 \end{bmatrix} = \left[0 \middle| t_1 \middle| \dots \middle| t_{\nu_j^i - 1}\right]$$

oder

$$(A - \lambda_i I) t_m = t_{m-1} \quad \text{für } m = \nu_j^{(i)}, \dots, 2, \tag{7.2}$$

$$(A - \lambda_i I) t_1 = 0.$$

Man sagt auch: Die Vektoren $t_1, \dots, t_{\nu_j^{(i)}}$ bilden eine *Jordan-Kette*. t_1 ist ein Eigenvektor, $t_2, \dots, t_{\nu_j^{(i)}}$ heißen *Hauptvektoren*. Die Spalten von T bestehen daher aus Eigen- und Hauptvektoren von A. Da T regulär ist, bilden die Eigen- und Hauptvektoren eine Basis des $\mathbb{C}^n$.

Die Eigenwerte hängen als Nullstellen eines Polynoms stetig von den Koeffizienten und damit auch von den Elementen der Matrix ab. Besitzt A ein Jordan-Kästchen der Länge $\nu > 1$, so zerfällt dieses im Allgemeinen, wenn die Matrix gestört wird. Die Jordansche Normalform lässt sich daher numerisch nur schwer bestimmen.

Eine Matrix heißt *diagonalisierbar*, wenn für alle Eigenwerte λ_i gilt $\rho(\lambda_i) = \sigma(\lambda_i)$. Wenn man dann mehrfache Eigenwerte auch mehrfach zählt, folgt wegen $\nu_j^{(i)} = 1$,

$$J = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}.$$

Anders ausgedrückt: Im diagonalisierbaren Fall gibt es eine Basis aus Eigenvektoren $\{x_1, \dots, x_n\}$ und die Matrix T hat die Gestalt

$$T = [x_1 | \dots | x_n].$$

7.3 Die Schursche Normalform Wir nannten eine Matrix T unitär, wenn $T^H T = T T^H = I$. Sowohl die Zeilen als auch die Spalten einer solchen Matrix bilden eine Orthonormalbasis des $\mathbb{C}^n$.

Satz Zu $A \in \mathbb{C}^{n \times n}$ gibt es eine unitäre Matrix U mit

$$U^H A U = \begin{bmatrix} \lambda_1 & * & \dots & * \\ & \ddots & \ddots & \vdots \\ & & \ddots & * \\ 0 & & & \lambda_n \end{bmatrix}$$

wobei auf der Diagonale die nicht notwendig verschiedenen Eigenwerte von A stehen.

Beweis: durch vollständige Induktion über n. Für $n = 1$ ist $u_{11} = 1$.

Sei der Satz für alle Matrizen der Dimension $n - 1$ richtig. Sei λ_1 ein Eigenwert von A mit $Ax_1 = \lambda x_1$ und $|x_1| = 1$. Ergänze x_1 durch $x_2, \dots, x_n$ zu einer Orthonormalbasis des $\mathbb{C}^n$. Die Matrix $X = [x_1 | \dots | x_n]$ ist dann unitär mit

$$X^H A X e_1 = X^H A x_1 = X^H (\lambda x_1) = \lambda_1 e_1$$

oder

$$X^H A X = \begin{bmatrix} \lambda_1 & a \\ 0 & A_1 \end{bmatrix}.$$

Nach Induktionsvoraussetzung gibt es eine unitäre Matrix $U_1 \in \mathbb{C}^{(n-1)\times(n-1)}$ mit

$$U_1^H A_1 U_1 = \begin{bmatrix} \lambda_2 & & * \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}.$$

Die Matrix

$$U = X \cdot \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & U_1 \end{array}\right]$$

ist als Produkt zweier unitärer Matrizen unitär. Weiter gilt

$$U^H A U = \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & U_1^H \end{array}\right] X^H A X \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & U_1 \end{array}\right] = \begin{bmatrix} \lambda_1 & & * \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}.$$

□

7.4 Hermitesche Matrizen Satz Ist $A \in \mathbb{C}^{n\times n}$ hermitesch, so gibt es eine unitäre Matrix $U = [x_1, \ldots, x_n]$, deren Spalten aus Eigenvektoren besteht, mit

$$U^{-1} A U = U^H A U = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}.$$

mit reellen Eigenwerten $\lambda_1, \ldots, \lambda_n$. Damit ist A diagonalisierbar und die Eigenvektoren bilden eine Orthogonalbasis des $\mathbb{C}^n$. Ordnet man die Eigenwerte

$$\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n,$$

so

$$\lambda_n = \max_{x\in\mathbb{C}^n\setminus\{0\}} R(x), \quad \lambda_1 = \min_{x\in\mathbb{C}^n\setminus\{0\}} R(x),$$

mit dem *Rayleigh-Quotienten*

$$R(x) = \frac{(Ax, x)}{(x, x)}.$$

Ist A reell und damit symmetrisch, so können die Eigenvektoren reell gewählt werden.

Beweis: Nach Satz 7.3 gibt es eine unitäre Matrix U, so dass $U^H A U$ eine rechte obere Dreiecksmatrix ist. Wegen $A = A^H$ gilt

$$(U^H A U)^H = U^H A^H U = U^H A U$$

und $U^H A U = D$ ist eine reelle Diagonalmatrix. Aus $AU = UD$ folgt, dass U aus den Eigenvektoren von A besteht und in D die zugehörigen Eigenwerte stehen.

Wir normieren die Eigenvektoren x_i zu $|x_i| = 1$ und entwickeln $x \in \mathbb{C}^n$

$$x = \sum_{i=1}^{n} \alpha_i x_i.$$

Wegen $(x_i, x_j) = \delta_{ij}$ gilt $(Ax_i, x_j) = \lambda_i \delta_{ij}$ und

$$R(x) = \frac{\sum_{i=1}^n |\alpha_i|^2 \lambda_i}{\sum_{i=1}^n |\alpha_i|^2} \leq \frac{\lambda_n \sum_{i=1}^n |\alpha_i|^2}{\sum_{i=1}^n |\alpha_i|^2} = \lambda_n. \tag{7.3}$$

Andererseits gilt $R(x_n) = \lambda_n$. Das Maximum des Rayleigh-Quotienten wir daher auf dem Eigenvektor x_n angenommen. □

Korollar Eine hermitesche Matrix ist genau dann positiv (semi)definit, wenn alle Eigenwerte von A positiv (nichtnegativ) sind. Dann gilt

$$\lambda_1 |x|^2 \leq (Ax, x) \leq \lambda_n |x|^2 \quad \text{für alle } x \in \mathbb{C}^n.$$

Auch für die Eigenwerte $\lambda_2, \ldots, \lambda_{n-1}$ gibt es Charakterisierungen durch den Rayleigh-Quotienten:

Satz Sei $A \in \mathbb{C}^{n \times n}$ hermitesch mit Eigenwerten $\lambda_1 \leq \ldots \leq \lambda_n$ und zugehörigen Eigenvektoren $x_1, \ldots, x_n$.

(a) (Rayleighsches Minimumprinzip) Mit

$$E_i = \operatorname{span}\{x_1, \ldots, x_i\}, \quad E_0 = \{0\},$$

ist

$$\lambda_i = \min\{R(x) : x \perp E_{i-1},\ x \neq 0\}$$

und das Minimum wird auf einem Vektor des Eigenraums von λ_i angenommen.

(b) (Minmax-Prinzip von Poincaré) Es gilt

$$\lambda_i = \min_{\dim M = i} \max_{x \in M \setminus \{0\}} \{R(x)\}$$

und das Minimum wird auf M für i erste Eigenvektoren von A angenommen.

(c) (Maxmin-Prinzip von Courant-Hilbert-Fischer) Es gilt

$$\lambda_i = \max_{\dim M \leq i-1} \min\{R(x) : x \perp M,\ x \neq 0\}$$

und das Maximum wird auf M für $i-1$ erste Eigenvektoren von A angenommen.

Bemerkung Die etwas schwammig anmutende Formulierung, dass eine Extremum auf den ersten Eigenvektoren angenommen wird, ist der Tatsache geschuldet, dass im Falle mehrfacher Eigenvektoren die Extrema nicht eindeutig angenommen werden (siehe Beweis).

Beweis: (a) Ist $x \perp E_{i-1}$, so $x = \sum_{j=i}^n \alpha_j x_j$ und man schließt wie in (7.3), dass λ_i das Minimum des Rayleigh-Quotienten über solche x ist. Ist $\lambda_i > \lambda_{i-1}$, so gilt für jeden Eigenvektor x_i zu λ_i, dass $\lambda_i = R(x_i)$. Ist $\lambda_i = \lambda_{i-1}$, so orthogonalisieren wir einen Eigenvektor von λ_i an die bereits gefundenen. Für diesen Eigenvektor x_i gilt dann wieder $\lambda_i = R(x_i)$.

(b) Sei M ein i-dimensionaler Teilraum des $\mathbb{C}^n$. Aus Dimensionsgründen gibt es ein $y \in M$, $y \neq 0$, mit $y \perp E_{i-1} = \operatorname{span}\{x_1, \ldots, x_{i-1}\}$. Nach dem Rayleighschen Minimumprizip gilt dann $R(y) \geq \lambda_i$. Für $M = \operatorname{span}\{x_1, \ldots, x_i\}$ gilt $\max_{x \in M} R(x) = R(x_i) = \lambda_i$.

(c) Für $i = 1$ ist die Behauptung richtig, sei also $i > 1$. Wir setzen für Unterräume M mit $\dim M \leq i - 1$

$$h(M) = \min\{R(x) : x \neq 0,\ x \perp M\}.$$

Nach dem Rayleighschen Minimumprinzip gilt $h(E_{i-1}) = \lambda_i$ und damit $\max_M h(M) \geq \lambda_i$. Ist $\dim M \leq i - 1$ und $M \neq E_{i-1}$, so gibt es einen Vektor y in $E_{i-1} \setminus \{0\}$ mit $y \perp M$. Für diesen ist dann $R(y) \leq \lambda_{i-1}$. □

Im Gegensatz zum Rayleighschen Minimumprinzip lassen sich mit (b) und (c) Aussagen über höherer Eigenwerte treffen, ohne die niedrigen zu kennen.

Beispiele (i) Sind $A, B \in \mathbb{R}^{n\times n}$ symmetrisch mit $(Ax, x) \le (Bx, x)$ für alle $x \in \mathbb{R}^n$, so gilt nach (b) oder (c) $\lambda_i(A) \le \lambda_i(B)$ für alle $i = 1, \dots, n$.

(ii) Ist $A \in \mathbb{R}^{n\times n}$ symmetrisch mit $a_{ij} \ge 0$ und $A \neq 0$, so ist der größte Eigenwert positiv und man kann den zugehörigen Eigenvektor nichtnegativ wählen. Wenn nämlich x ein Eigenvektor zum größten Eigenwert ist, so gilt für $\tilde{x} = (|x_1|, \dots, |x_n|)$, dass $R(x) \le R(\tilde{x})$. Ferner ist $R(1, \dots, 1) > 0$, daher $\lambda_n > 0$.

7.5 Eigenwertnäherung bei hermiteschen Matrizen **Satz** Sei $A \in \mathbb{C}^{n\times n}$ hermitesch. Wenn für $\lambda \in \mathbb{R}$ und $x \in \mathbb{C}^n$ mit $|x| = 1$ gilt

$$|Ax - \lambda x| \le \varepsilon,$$

so gibt es einen Eigenwert λ_i von A mit

$$|\lambda - \lambda_i| \le \varepsilon.$$

Beweis: Seien $\lambda_i \in \mathbb{R}$ die Eigenwerte von A mit $(x_i, x_j) = \delta_{ij}$. Da $\{x_i\}$ eine Basis bilden, gilt für $x \in \mathbb{C}^n$ mit $|x| = 1$

$$x = \sum_{i=1}^{n} \alpha_i x_i, \quad 1 = |x|^2 = \Big(\sum_i \alpha_i x_i, \sum_j \alpha_j x_j \Big) = \sum_{i=1}^{n} |\alpha_i|^2$$

sowie

$$|Ax - \lambda x|^2 = \Big(\sum_i \lambda_i \alpha_i x_i - \lambda \alpha_i x_i, \sum_j \lambda_j \alpha_j x_j - \lambda \alpha_j x_j \Big) = \sum_{i=1}^{n} |\alpha_i(\lambda_i - \lambda)|^2 \le \varepsilon^2.$$

Daher

$$\min_j |\lambda_j - \lambda|^2 = \min_j |\lambda_j - \lambda|^2 \sum_{i=1}^{n} |\alpha_i|^2 \le \sum_{i=1}^{n} |\alpha_i(\lambda_i - \lambda)|^2 \le \varepsilon^2.$$

□

Die numerische Berechnung der Eigenwerte einer hermiteschen Matrix ist daher ein gutartiger Algorithmus.

7.6 Normale Matrizen Eine Matrix $A \in \mathbb{C}^{n\times n}$ heißt *normal*, wenn $AA^H = A^H A$.

Satz Eine Matrix ist genau dann normal, wenn es eine unitäre Matrix U gibt mit

$$U^{-1}AU = U^H AU = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}.$$

Bemerkung Normale Matrizen sind daher diagonalisierbar und besitzen orthogonale Eigenvektoren.

Beweis: Sei A normal. Nach Satz 7.3 ist

$$U^H AU = R$$

mit einer rechten oberen Dreiecksmatrix R. Dann

$$\begin{aligned} RR^H &= U^H AUU^H A^H U = U^H AA^H U \\ &= U^H A^H AU = (U^H A^H U)(U^H AU) = R^H R. \end{aligned}$$

Damit gilt

$$(R^H R)_{11} = |r_{11}|^2 = (RR^H)_{11} = \sum_{k=1}^{n} |r_{1k}|^2,$$

daher $r_{1k} = 0$ für $k \geq 2$. Durch wiederholte Anwendung dieses Arguments erschließt man, dass R eine Diagonalmatrix ist.

Sei nun umgekehrt $U^H AU = \operatorname{diag}(\lambda_1, \ldots, \lambda_n) = D$. Dann gilt $A = UDU^H$ und mit $|D|^2 = \operatorname{diag}(|\lambda_1|^2, \ldots, |\lambda_n|^2)$

$$A^H A = UD^H U^H UDU^H = U|D|^2 U^H = AA^H.$$

□

7.7 Singuläre Werte In diesem Abschnitt betrachten wir nicht notwendig quadratische Matrizen $A \in \mathbb{C}^{m\times n}$. Die Matrizen $A^H A$ und AA^H sind dann positiv semidefinit wegen

$$(A^H Ax, x) = (Ax, Ax) = |Ax|^2 \geq 0.$$

Die Eigenwerte von $A^H A$ sind

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0.$$

Wie in der Theorie der singulären Werte üblich, bezeichnen wir hier den kleinsten Eigenwert mit λ_n.

Lemma Seien $A, B \in \mathbb{C}^{m\times n}$. Dann besitzen $AB^H \in \mathbb{C}^{m\times m}$ und $B^H A \in \mathbb{C}^{n\times n}$ – abgesehen von $\lambda = 0$ – die gleichen Eigenwerte: $\sigma(AB^H) \setminus \{0\} = \sigma(B^H A) \setminus \{0\}$.

Beweis: Für $v \neq 0$ sei $AB^H v = \lambda v$ mit $\lambda \neq 0$. Dann ist $B^H v \neq 0$ und wegen

$$(B^H A)B^H v = \lambda B^H v.$$

auch Eigenvektor von $B^H A$ zum Eigenwert λ. Die andere Richtung zeigt man genauso mit dem Eigenwertproblem für $B^H A$. □

Wir schreiben $\lambda_k = \sigma_k^2$ mit $\sigma_k \geq 0$. Die Zahlen

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$$

heißen *singuläre Werte* von A. Aufgrund des obigen Lemmas besitzen $A^H A$ und AA^H die gleichen nichtverschwindenden singulären Werte. Für $|x| = 1$ ist $|Ax| = \sqrt{R(x)}$ mit dem Rayleighquotienten R zu $A^H A$. Nach Satz 7.4 gilt dann

$$\sigma_1 = \|A\| = \max_{x\in\mathbb{C}^n\setminus\{0\}} \frac{|Ax|}{|x|},$$

$$\sigma_n = \min_{x\in\mathbb{C}^n\setminus\{0\}} \frac{|Ax|}{|x|}.$$

Für $m = n$ und A regulär folgt hieraus

$$\kappa(A) = \|A\| \, \|A^{-1}\| = \frac{\sigma_1}{\sigma_n}.$$

Im Fall $m = n$ gibt σ_n den Abstand zur nächsten singulären Matrix an:

Satz Seien $A, E \in \mathbb{C}^{n\times n}$. A besitze die singulären Werte $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$. Dann gilt:

(a) $\|E\| \geq \sigma_n$, falls $A + E$ singulär.

(b) Es gibt eine Matrix E mit $\|E\| = \sigma_n$, so dass $A + E$ singulär ist.

Beweis: (a) Sei $A + E$ singulär, also

$$(A + E)x = 0 \quad \text{für } x \neq 0.$$

Dann

$$\sigma_n|x| \leq |Ax| = |-Ex| \leq \|E\|\,|x|.$$

(b) Sei $\sigma_n > 0$. Dann gibt es $u, v \in \mathbb{C}^n$ $|u| = |v| = 1$ mit

$$|Au| = \sigma_n, \quad v = \frac{1}{\sigma_n} Au.$$

Für $E = -\sigma_n v u^H$ gilt dann

$$(A + E)u = Au - \sigma_n v(u^H u) = Au - \sigma_n v = 0,$$

$$\|E\| = \sigma_n \|v u^H\| = \sigma_n \max_{|x|=1} |v(u^H x)| \leq \sigma_n |v|\,|u| = \sigma_n.$$

Wegen (a) folgt $\|E\| = \sigma_n$. □

Satz [Singulärwertzerlegung] Sei $A \in \mathbb{C}^{m\times n}$. Dann gibt es eine unitäre $m \times m$-Matrix U, eine unitäre $n \times n$-Matrix V mit

$$U^H A V = \Sigma = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix}, \quad D = \operatorname{diag}(\sigma_1, \ldots, \sigma_r),$$

wobei $\sigma_1, \ldots, \sigma_r$ die nichtverschwindenden singulären Werte von A sind.

$A = U\Sigma V^H$ heißt *Singulärwertzerlegung* von A.

Beweis: Wir zeigen dies durch Induktion über m und n, d.h. wir nehmen für die Singulärwertzerlegung eines $A \in \mathbb{C}^{m\times n}$ an, dass die Singulärwertzerlegung für Matrizen $B \in \mathbb{C}^{(m-1)\times(n-1)}$ existiert. Zur Induktionsverankerung müssen wir daher den Fall $A \in \mathbb{C}^{m\times 1}$ betrachten. Ist $A = 0$, so ist $\Sigma = 0$ und als U und V können wir beliebige unitäre Matrizen nehmen. Für $A \neq 0$ wählen wir eine unitäre Matrix $U \in \mathbb{C}^{m\times m}$ mit erster Spalte $A/\|A\|$, $\Sigma = \|A\|e_1 \in \mathbb{C}^{m\times 1}$ sowie $V = 1 \in \mathbb{C}^{1\times 1}$. Dann gilt offenbar $A = U\Sigma V$. Den Fall $A \in \mathbb{C}^{1\times n}$ behandelt man ganz analog.

Sei $A \in \mathbb{C}^{m\times n}$. Wenn $A = 0$, so ist $\Sigma = 0$ und für U und V können wir wieder beliebige unitäre Matrizen wählen. Sei also $A \neq 0$. Wegen $\|A\| = \max_{|v|=1} |Av|$ gibt es einen Vektor v mit $|v| = 1$ und $|Av| = \|A\| = \sigma > 0$. Ferner setze $u = Av/|Av|$. Wir ergänzen v und u zu unitären Matrizen $V = [v|V_1] \in \mathbb{C}^{n\times n}$ und $U = [u|U_1] \in \mathbb{C}^{m\times m}$. Dann gilt

$$\tilde{A} = U^H A V = \begin{bmatrix} u^H \\ U_1^H \end{bmatrix} [Av|AV_1] = \begin{bmatrix} u^H \\ U_1^H \end{bmatrix} [\sigma u|AV_1] = \begin{bmatrix} \sigma & w^H \\ 0 & A_1 \end{bmatrix}$$

mit $w = V_1^H A^H u$ und $A_1 = U_1^H A V_1 \in \mathbb{C}^{(m-1)\times(n-1)}$. Aus

$$\left|\tilde{A}\begin{pmatrix} \sigma \\ w \end{pmatrix}\right|^2 = \left|\begin{pmatrix} \sigma^2 + w^H w \\ A_1 w \end{pmatrix}\right|^2 \geq (\sigma^2 + |w|^2)^2$$

folgt $\|\tilde{A}\|^2 \geq \sigma^2 + |w|^2$. Andererseits ist

$$\sigma^2 = \|A\|^2 = \rho(A^H A) = \rho(V\tilde{A}^H U^H U \tilde{A} V^H) = \rho(\tilde{A}^H \tilde{A}) = \|\tilde{A}\|^2,$$

und daher $w = 0$. Damit besitzt $\tilde{A}$ die Gestalt

$$\begin{bmatrix} \sigma & 0 \\ 0 & A_1 \end{bmatrix}.$$

Nach unserer Induktionsvoraussetzung existiert die Singulärwertzerlegung von A_1. □

Der Beweis zeigt auch, dass man zu einer reellen Matrix eine reelle Singulärwertzerlegung $A = U\Sigma V^T$ bekommen kann mit orthogonalen Matrizen U und V.

Ist $A = U\Sigma V^H$ so folgt

$$A^H = V\Sigma^T U.$$

Die nichtverschwindenden singulären Werte von A und A^H stimmen überein (vergleiche obiges Lemma). Aus dieser Darstellung der Singulärwertzerlegung von A^H folgt auch

$$A^H A = V\Sigma^T \Sigma V^H, \quad AA^H = U\Sigma\Sigma^T U^H.$$

Damit bilden $\{v_1 \ldots, v_n\}$ bzw. $\{u_1, \ldots, u_m\}$ Orthonormalsysteme von Eigenvektoren von $A^H A$ bzw. AA^H.

Satz Sei $A = U^H \Sigma V$ die Singulärwertzerlegung von A mit singulären Werten $\sigma_1 \geq \ldots \geq \sigma_r > \sigma_{r+1} = \ldots = \sigma_{\min\{m,n\}} = 0$ und Matrizen $U = [u_1|\ldots|u_m]$, $V = [v_1|\ldots|v_n]$, so ist

(a) r der Rang von A,

(b) $\text{Kern}(A) = \{x \in \mathbb{C}^n : Ax = 0\} = \text{span}\,\{v_{r+1}, \ldots, v_n\}$,

(c) $\text{Bild}(A) = \{Ax : x \in \mathbb{C}^n\} = \text{span}\,\{u_1, \ldots, u_r\}$,

(d)

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^H = U_r \Sigma_r V_r^H$$

mit $U_r = [u_1|\ldots|u_r]$, $V_r = [v_1|\ldots|v_r]$, $\Sigma_r = \text{diag}\,(\sigma_1, \ldots, \sigma_r)$,

(e)

$$|A|^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2 = \sum_{i=1}^{r} \sigma_i^2.$$

Beweis: (a) Die Multiplikation mit den regulären Matrizen U^H und V ändern den Rang von A nicht, daher $\text{Rang}\, A = \text{Rang}\, \Sigma = r$.

(b) Wegen $V^H v_i = e_i$ ist

$$Av_i = U\Sigma V^H v_i = U\Sigma e_i = 0 \quad \text{für } i = r+1, \ldots, n.$$

Also gilt

$$v_{r+1}, \ldots, v_n \in \text{Kern}(A).$$

Da $\dim \text{Kern}(A) = n - r$ bilden diese Vektoren eine Basis von $\text{Kern}(A)$.

(c) Wegen $A = U\Sigma V^H$ ist

$$\text{Bild}(A) = U \cdot \text{Bild}(\Sigma) = U \cdot \text{span}\,\{e_1, \ldots, e_r\} = \text{span}\,\{u_1, \ldots, u_r\}.$$

(d) Durch Ausmultiplizieren erhalten wir

$$A = U\Sigma V^H = [u_1|\ldots|u_m]\Sigma \begin{bmatrix} v_1^H \\ \vdots \\ v_n^H \end{bmatrix} = \sum_{i=1}^{r} \sigma_i u_i v_i^H.$$

(e) Mit $A = [a_1|\ldots|a_n]$ erhalten wir wegen $|Ux| = |x|$

$$|A|^2 = \sum_{i=1}^{n} |a_i|^2 = \sum_{i=1}^{n} |U^H a_i|^2 = |U^H A|^2.$$

Mit den Zeilen von $U^H A$ kann man genauso argumentieren, daher

$$|A|^2 = |U^H A V|^2 = |\Sigma|^2 = \sum_{i=1}^{r} \sigma_i^2.$$

□

7.8 Spektralradius und induzierte Matrizennormen **Satz** Sei $\|\cdot\|_V$ eine Vektornorm auf $\mathbb{C}^n$ und $\|\cdot\|_{V\to V}$ die induzierte Matrix-Norm. Dann gilt für $A \in \mathbb{C}^{n\times n}$

$$|\lambda| \le \|A\|_{V\to V}$$

für alle Eigenwerte λ von A.

Beweis: Aus $Ax = \lambda x$ folgt $\|Ax\|_V = |\lambda|\,\|x\|_V$ und

$$|\lambda| \le \frac{\|Ax\|_V}{\|x\|_V} \le \|A\|_{V\to V}.$$

□

Wir erinnern daran, dass wir mit $\rho(A) = \max_i |\lambda_i|$ den Spektralradius der Matrix A bezeichnen. Nach dem letzten Satz gilt daher $\rho(A) \le \|A\|_{V\to V}$. Es muss nicht unbedingt eine induzierte Matrixnorm mit $\rho(A) = \|A\|_{V\to V}$ geben. Als Beispiel wählen wir für A das Jordan-Kästchen $C_2(0) \in \mathbb{C}^{2\times 2}$. Es gilt $\rho(J_2(0)) = 0$, aber es ist $J_2(0) \neq 0$. Das folgende Resultat ist daher nicht zu verbessern.

Satz Zu jeder Matrix $A \in \mathbb{C}^{n\times n}$ und jedem $\varepsilon > 0$ gibt es eine Vektornorm $\|\cdot\|_V$ mit

$$\|A\|_{V\to V} = \rho(A) + \varepsilon.$$

Beweis: Nach Jordan gibt es eine reguläre Matrix T mit $TAT^{-1} = J = \operatorname{diag}(C_{\nu_j^i}(\lambda_i))$ mit

$$C_\nu(\lambda) = \begin{bmatrix} \lambda & 1 & & 0 \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 0 & & & \lambda \end{bmatrix} =: \lambda I + E.$$

Sei $D_\varepsilon = \operatorname{diag}(1, \varepsilon, \varepsilon^2, \ldots, \varepsilon^{\nu-1})$. Dann

$$D_\varepsilon^{-1} C_\nu(\lambda) D_\varepsilon = D_\varepsilon^{-1}(\lambda I + E) D_\varepsilon = \lambda I + D_\varepsilon^{-1} E D_\varepsilon.$$

Mit

$$ED_\varepsilon = \begin{bmatrix} 0 & 1 & & 0 \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 0 & & & 0 \end{bmatrix} \begin{bmatrix} 1 & & & 0 \\ & \varepsilon & & \\ & & \ddots & \\ 0 & & & \varepsilon^{\nu-1} \end{bmatrix} = \begin{bmatrix} 0 & \varepsilon & & 0 \\ & \ddots & \ddots & \\ & & \ddots & \varepsilon^{\nu-1} \\ 0 & & & 0 \end{bmatrix}$$

folgt

$$D_\varepsilon^{-1} E D_\varepsilon = \begin{bmatrix} 0 & \varepsilon & & 0 \\ & \ddots & \ddots & \\ & & \ddots & \varepsilon \\ 0 & & & 0 \end{bmatrix}$$

Also können wir äquivalent schreiben

$$\tilde{T} A \tilde{T}^{-1} = \tilde{J}, \tag{7.4}$$

wobei die Jordan-Kästchen von $\tilde{J}$ in der Nebendiagonalen ε statt 1 stehen haben.

Ist $S \in \mathbb{C}^{n\times n}$ regulär, so ist

$$\|x\|_S = |Sx|$$

eine Vektornorm mit induzierter Matrixnorm

$$\begin{aligned}\|A\|_{S\to S} &= \sup_{x\neq 0} \frac{\|Ax\|_S}{\|x\|_S} = \sup_{x\neq 0} \frac{|SAx|}{|Sx|} \\ &= \sup_{x\neq 0} \frac{|SAS^{-1}x|}{|x|} = \|SAS^{-1}\|.\end{aligned}$$

Mit (7.4) folgt daher

$$\|A\|_{\tilde{T}\to\tilde{T}} = \|\tilde{J}\| \leq \rho(A) + \varepsilon.$$

□

7.9 Der Rayleigh-Quotient bei allgemeinen Matrizen Satz [Hausdorff, Bendixson] Sei $A \in \mathbb{C}^{n\times n}$ und

$$R(x) = \frac{(Ax, x)}{(x, x)}$$

der Rayleigh-Quotient zu A. Ferner bezeichne $G(A)$ den Wertebereich von $R(x)$ für $x \in \mathbb{C}^n$. Dann gilt:

(a) $G(A)$ ist konvex und enthält die Eigenwerte von A.

(b) Ist A normal, so ist $G(A)$ die konvexe Hülle der Eigenwerte von A.

(c) Mit

$$H_1 = \frac{1}{2}(A + A^H), \quad H_2 = \frac{1}{2i}(A - A^H)$$

sind H_1, H_2 hermitesch mit $A = H_1 + iH_2$. Für jeden Eigenwert λ von A gilt dann

$$\lambda_{min}(H_1) \leq \operatorname{Re}\lambda \leq \lambda_{max}(H_1),$$

$$\lambda_{min}(H_2) \leq \operatorname{Im}\lambda \leq \lambda_{max}(H_2).$$

Beweis: (a) Für $Ax = \lambda x$, $x \neq 0$, ist

$$R(x) = \frac{(Ax, x)}{(x, x)} = \lambda. \tag{7.5}$$

Die Konvexität von $G(A)$ benötigen wir im Folgenden nicht. Auf den schwierigen Beweis soll daher verzichtet werden.

(b) Für eine normale Matrix haben wir die Darstellung $A = U^H DU$ mit einer unitären Matrix U und einer Diagonalmatrix $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. Dann

$$G(A) = \left\{ \frac{(DUx, Ux)}{(Ux, Ux)} : x \neq 0 \right\} = \left\{ \frac{(Dy, y)}{(y, y)} : y \neq 0 \right\}.$$

Der Wertebereich von $G(A)$ besteht daher genau aus der konvexen Hülle der Eigenwerte.

(c) Wegen (7.5) folgt

$$\begin{aligned}\operatorname{Re}\lambda &\leq \max_{x\neq 0} \operatorname{Re}\frac{(Ax, x)}{(x, x)} = \max_{x\neq 0} \operatorname{Re}\frac{\sum_{ij} a_{ij}x_j\overline{x}_i}{(x, x)} = \max_{x\neq 0} \frac{(Ax, x) + (x, Ax)}{2(x, x)} \\ &= \max_{x\neq 0} \frac{((A + A^H)x, x)}{2(x, x)} = \max_{x\neq 0} \frac{(Hx, x)}{(x, x)}.\end{aligned}$$

Für den Imaginärteil beweist man das ganz genauso. □

7.10 Gerschgorin-Kreise **Lemma** Seien $A, B \in \mathbb{C}^{n\times n}$ und λ sei Eigenwert von A, aber nicht von B. Dann gilt für jede induzierte Matrixnorm $\|\cdot\|$

$$1 \le \|(\lambda I - B)^{-1}(A-B)\| \le \|(\lambda I - B)^{-1}\|\,\|A-B\|.$$

Beweis: Aus

$$(A-B)x = (\lambda I - B)x$$

folgt

$$x = (\lambda I - B)^{-1}(A-B)x$$

und damit

$$\|x\|_V \le \|(\lambda I - B)^{-1}(A-B)\|\,\|x\|_V.$$

□

Wir wenden das Lemma an auf die Vektornorm

$$\|x\|_\infty = \max_{1\le i\le n} |x_i|$$

mit der Zeilensummennorm als zugehöriger Matrixnorm,

$$\|A\|_\infty = \|A\|_{\infty\to\infty} = \sup_{x\neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty} = \max_{1\le i\le n} \sum_{k=1}^{n} |a_{ik}|.$$

Mit $B = \mathrm{diag}\,(a_{11}, \ldots, a_{nn})$ erhalten wir aus dem letzten Lemma

$$1 \le \|(\lambda I - B)^{-1}(A-B)\|_\infty = \max_{1\le i\le n} \frac{1}{|\lambda - a_{ii}|} \sum_{k=1,\,k\neq i}^{n} |a_{ik}|.$$

Damit haben wir gezeigt:

Satz [Gerschgorin] Die Vereinigung der Kreissscheiben

$$K_i = \Big\{\mu \in \mathbb{C} : |\mu - a_{ii}| \le \sum_{k=1,\,k\neq i}^{n} |a_{ik}|\Big\}$$

enthält alle Eigenwerte der Matrix $A \in \mathbb{C}^{n\times n}$.

Zusatz Wenn $\cup_{j=1}^{k} K_{i_j}$ eine Zusammenhangskomponente von $\cup_i K_i$ bildet, so liegen in dieser genau k Eigenwerte.

Beweis: Schreibe $A = D + R$ mit D diagonal und $R_{ii} = 0$. In

$$A(t) = D + tR, \quad t \in [0,1],$$

hängen die Eigenwerte von $A(t)$ stetig von t ab. Sie können daher $\cup_{j=1}^{k} K_{i_j}(t)$ nicht verlassen. □

Man kann die Gerschgorin-Kreise auch auf A^T anwenden. Demnach liegen die Eigenwerte auch in den Kreisen um a_{ii} mit einem Radius, der aus den Elementen der i-ten Spalte gebildet wird.

Eine andere Möglichkeit besteht in der Anwendung der Gerschgorin-Kreise auf $D^{-1}AD$ mit einer Diagonalmatrix $D = \mathrm{diag}\,(d_1, \ldots, d_n)$,

$$K_i = \{\mu : |\mu - a_{ii}| \le \sum_{k=1,\,k\neq i}^{n} \left|\frac{a_{ik} d_k}{d_i}\right| |\}.$$

In jedem Fall erhält man vernünftige Abschätzungen nur für Matrizen mit großen Diagonalelementen.

Beispiel $A \in \mathbb{C}^{n\times n}$ heißt *(strikt) diagonaldominant*, wenn

$$\sum_{k=1,\,k\neq i}^{n} |a_{ik}| < |a_{ii}| \quad \text{für } 1 \le i \le n.$$

Nach dem Satz von Gerschgorin ist eine diagonaldominante Matrix regulär.

7.11 Abschätzungen von Nullstellen von Polynomen Zum Polynom

$$p(\lambda) = a_n\lambda^n + a_{n-1}\lambda^{n-1} + \ldots + a_0, \quad a_n \neq 0,$$

gehört die *Frobeniusmatrix*

$$F = \begin{bmatrix} 0 & & & & -\gamma_0 \\ 1 & \ddots & & 0 & \vdots \\ & \ddots & \ddots & & \vdots \\ & & \ddots & 0 & \vdots \\ 0 & & & 1 & -\gamma_{n-1} \end{bmatrix}, \quad \gamma_i = \frac{a_i}{a_n},$$

denn wenn wir $\det(F - \lambda I)$ nach der letzten Spalte entwickeln, erhalten wir

$$\det(F - \lambda I) = \frac{1}{a_n}(-1)^n p(\lambda).$$

Damit können alle Eigenwertabschätzungen für die Abschätzung von Nullstellen von Polynomen herangezogen werden. Der Satz von Gerschgorin angewendet auf F oder F^H liefert beispielsweise

(a) $|\lambda_k| \leq \max\left\{\left|\frac{a_0}{a_n}\right|, \max\limits_{1\leq i\leq n-1}\left(1 + \left|\frac{a_i}{a_n}\right|\right)\right\}$,

(b) $|\lambda_k| \leq \max\left\{1, \sum\limits_{i=0}^{n-1}\left|\frac{a_i}{a_n}\right|\right\}$.

Beispiel Für das Polynom $p(\lambda) = \lambda^3 - 6\lambda^2 + 11\lambda - 6$ erhalten wir

(a) $\max\{6, 1 + |a_i|\} = 12$,

(b) $\max\{1, 23\} = 23$.

Die Nullstellen sind $\lambda_{1,2,3} = 1, 2, 3$.

8 Numerik von Eigenwertproblemen

8.1 Das Lanczos-Verfahren Mit dem Lanczos-Verfahren bestimmt man für eine hermitesche Matrix $A \in \mathbb{C}^{n\times n}$ eine unitäre Matrix U mit

$$U^H A U = T,$$

wobei T eine reelle symmetrische Tridiagonalmatrix ist,

$$T = \begin{bmatrix} \delta_1 & \gamma_2 & & 0 \\ \gamma_2 & \ddots & \ddots & \\ & \ddots & \ddots & \gamma_n \\ 0 & & \gamma_n & \delta_n \end{bmatrix}$$

Die Eigenwerte von T können dann mit einem anderen Verfahren berechnet werden.

Zur hermiteschen A und $q \in \mathbb{C}^n \setminus \{0\}$ definieren wir die *Krylov-Räume*

$$K_i(q, A) = \operatorname{span}(q, Aq, \ldots, A^{i-1}q), \quad K_0(q, A) = \{0\}.$$

Sei m der größte Index, so dass $q, Aq, \ldots, A^{m-1}q$ linear unabhängig sind. Dann gilt

$$A : K_m(q, A) \to K_m(q, A),$$

denn $A^m q$ lässt sich nach Voraussetzung als Linearkombination der $A^i q$, $0 \le i < m$, darstellen.

Die Idee des Lanczos-Verfahrens besteht darin, eine Orthonormalbasis $q_1, \ldots, q_i$ von $K_i(q, A)$ zu konstruieren, mit der sich A auf eine einfache Matrix transformieren lässt.

Sei $|q| = 1$. Setze als Start

$$q_1 = q$$

Konstruiere Vektoren q_i, die der Rekursion

$$Aq_i = \gamma_i q_{i-1} + \delta_i q_i + \gamma_{i+1} q_{i+1}, \quad i \ge 1 \tag{8.1}$$

genügen mit

$$\begin{aligned} \delta_i &= (Aq_i, q_i) \\ \gamma_{i+1} &= |r_i|, \quad r_i = Aq_i - \gamma_i q_{i-1} - \delta_i q_i \\ q_{i+1} &= \frac{r_i}{\gamma_{i+1}} \quad \text{falls } \gamma_{i+1} \neq 0. \end{aligned}$$

Es gilt dann $\gamma_{i+1} \ge 0$ und $\delta_i = (Aq_i, q_i)$ ist reell, weil A als hermitesch vorausgesetzt wurde. Weiter ist $|q_i| = 1$.

Satz Durch (8.1) werden eindeutige Vektoren $q_1, \ldots, q_m$ bestimmt, so dass $q_1, \ldots, q_i$ eine Orthonormalbasis von $K_i(q, A)$ für $i = 1, \ldots, m$ bilden. Es gilt $\gamma_{m+1} = 0$, so dass (8.1) nach $i = m$ abbricht.

Beweis: Wir zeigen die Behauptung durch Induktion über j. Offenbar ist q_1 eine Orthonormalbasis von $K_1(q, A)$. Im Induktionsschritt nehmen wir an, dass $q_1, \ldots, q_i$ eine Orthonormalbasis von $K_i(q, A)$ bilden für $1 \le i \le j$ und dass $\gamma_i > 0$, also $r_i \neq 0$, erfüllt ist für $i < j$.

Wäre $r_j = 0$, so hätte Aq_j eine Basisdarstellung in $K_j(q, A)$, also $\dim K_{j+1} = \dim K_j$, was nur für $j = m$ sein kann. Für $j < m$ gilt daher $r_j \neq 0$. Damit ist q_{j+1} in (8.1) eindeutig bestimmt mit $|q_{j+1}| = 1$ sowie

$$\begin{aligned} (\gamma_{j+1} q_{j+1}, q_j) &= (Aq_j, q_j) - (\gamma_j q_{j-1}, q_j) - (\delta_j q_j, q_j) \\ &= (Aq_j, q_j) - 0 - (Aq_j, q_j) \cdot 1 = 0, \end{aligned}$$

$$
\begin{aligned}
(\gamma_{j+1}q_{j+1}, q_{j-1}) &= (Aq_j, q_{j-1}) - (\gamma_j q_{j-1}, q_{j-1}) - (\delta_j q_j, q_{j-1}) \\
&= (q_j, Aq_{j-1}) - \gamma_j - 0 \\
&= (q_j, \gamma_{j-1}q_{j-2} + \delta_{j-1}q_{j-1} + \gamma_j q_j) - \gamma_j \\
&= \overline{\gamma}_j - \gamma_j = 0 \quad \text{weil } \gamma_j \text{ reell.}
\end{aligned}
$$

Für $i < j-1$ ist

$$
(\gamma_{j+1}q_{j+1}, q_i) = (Aq_j, q_i) - 0 - 0 = (q_j, Aq_i) = (q_j, \gamma_i q_{i-1} + \delta_i q_i + \gamma_{i+1}q_{i+1}) = 0.
$$

$q_{j+1} \in K_{j+1}(q, A)$ folgt aus der Definition von q_{j+1}. Damit bilden auch die Vektoren $q_1, \ldots, q_{j+1}$ eine Orthonormalbasis von $K_{j+1}(q, A)$. □

Setze

$$
Q_= [q_1 | \ldots | q_i], \quad J_i = \begin{bmatrix} \delta_1 & \gamma_2 & & 0 \\ \gamma_2 & \ddots & \ddots & \\ & \ddots & \ddots & \gamma_i \\ 0 & & \gamma_i & \delta_i \end{bmatrix}
$$

Dann kann man die Rekursion (8.1) als Matrixgleichung schreiben

$$
\begin{aligned}
AQ_i &= Q_i J_i + [0| \ldots |0|\gamma_{i+1}q_{i+1}] \\
&= Q_i J_i + \gamma_{i+1}q_{i+1}e_i^T,
\end{aligned} \tag{8.2}
$$

wobei $e_i \in \mathbb{R}^i$ den i-ten Einheitsvektor bezeichnet. Die j-te Spalte in (8.2) ergibt genau den Schritt j in der Rekursion (8.1). Die J_i sind reell, symmetrisch und unzerlegbar, also $\gamma_j \neq 0$ für $2 \leq j \leq m$. Die Matrizen $Q_i \in \mathbb{C}^{n \times i}$ besitzen orthonormale Spalten, $Q_i^H Q_i = I_i$. Wenn das Verfahren bei m abbricht, ist $\gamma_{m+1} = 0$ und daher

$$
AQ_m = Q_m J_m,
$$

also

$$
Q_m^T A Q_m = J_m.
$$

Die Eigenwerte von J_m sind auch Eigenwerte von A wegen

$$
J_m x = \lambda x \;\Rightarrow\; A(Q_m x) = \lambda Q_m x.
$$

Im Allgemeinen kann man $m = n$ erwarten, womit $Q^T AQ = J$ erfüllt ist. In diesem Fall ist J eine zu A ähnliche reelle, symmetrische Tridiagonalmatrix.

Der Vorteil dieses Verfahrens ist der geringe Rechenaufwand vor allem bei schwach besetzten Matrizen. Ist man nur an den Eigenwerten interessiert, braucht nur die aktuelle Rekursion, nicht aber die gesamte Orthonormalbasis gespeichert zu werden. Der Speicherplatzbedarf ist daher sehr gering. Theoretisch bricht das Verfahren mit einem Index $i = m \leq n$ ab, wenn erstmals $\gamma_{i+1} = 0$ ist, doch wird man wegen des Einflusses der Rundungsfehler in der Rechenpraxis kaum jemals ein $\gamma_{i+1} = 0$ finden. Es ist aber i.a. nicht nötig, das Verfahren solange fortzusetzen bis γ_{i+1} verschwindet, oder auch nur genügend klein wird. Man kann nämlich unter wenig einschränkenden Bedingungen zeigen, dass für $i \to \infty$ die größten bzw. kleinsten Eigenwerte sehr rasch gegen den größten bzw. kleinsten Eigenwert von A konvergieren. Wenn man nur an den extremen Eigenwerten von A interessiert ist, genügen deshalb relativ wenige Iterationen des Verfahrens, um diese mit ausreichender Genauigkeit durch die extremen Eigenwerte einer Matrix J_i mit $i << n$ zu approximieren.

Die Eigenwerte von J können mit dem Verfahren aus dem nächsten Abschnitt oder mit dem QR-Verfahren aus Abschnitt 8.6 bestimmt werden.

8.2 Bestimmung der Eigenwerte einer hermiteschen Tridiagonalmatrix

Sei

$$J = \begin{bmatrix} \delta_1 & \overline{\gamma}_2 & & 0 \\ \gamma_2 & \ddots & \ddots & \\ & \ddots & \ddots & \overline{\gamma}_n \\ 0 & & \gamma_n & \delta_n \end{bmatrix}$$

eine hermitesche Tridiagonalmatrix, insbesondere seien die δ_i reell. Neben dem QR-Verfahren aus Abschnitt 8.6 können die Eigenwerte von T auch direkt aus dem charakteristischen Polynom bestimmt werden.

Wir können J als *unzerlegbar* voraussetzen, d.h $\gamma_i \neq 0$ für alle i. Denn andernfalls zerfällt J in zwei Tridiagonalmatrizen, die man separat untersuchen kann. Wir entwickeln

$$p_i(\mu) = \det(J_i - \mu I_i), \quad J_i = \begin{bmatrix} \delta_1 & \overline{\gamma}_2 & & 0 \\ \gamma_2 & \ddots & \ddots & \\ & \ddots & \ddots & \overline{\gamma}_i \\ 0 & & \gamma_i & \delta_i \end{bmatrix}$$

nach der letzten Spalte:

$$\begin{aligned} p_0(\mu) :&= 1, \quad p_1(\mu) = \delta_1 - \mu \\ p_i(\mu) &= (\delta_i - \mu)p_{i-1}(\mu) - |\gamma_i|^2 p_{i-2}(\mu). \end{aligned}$$

Für das Newton-Verfahren bestimmen wir hieraus auch $p'(\mu)$ durch

$$\begin{aligned} p_0'(\mu) &= 0, \quad p_1'(\mu) = -1 \\ p_i'(\mu) &= -p_{i-1}(\mu) + (\delta_i - \mu)p_{i-1}'(\mu) - |\gamma_i|^2 p_{i-2}'(\mu). \end{aligned}$$

Für die Wahl des Startwerts verwende

Satz Für die Eigenwerte λ_j der Matrix J gilt

$$|\lambda_j| \leq \max_{1 \leq i \leq n} \{|\gamma_i| + |\delta_i| + \gamma_{i+1}|\}, \quad \gamma_1 = \gamma_{n+1} = 0.$$

Beweis: Dies ist gerade die Spaltensummennorm (siehe Satz 7.8). □

In der Praxis sind diese Matrizen häufig positiv-definit und man möchte nur den kleinsten Eigenwert bestimmen. Man hat dann mit $x^0 = 0$ einen guten Startwert für das Newton-Verfahren und kann es mit dem Doppelschrittverfahren aus Abschnitt 4.7 beschleunigen. In diesem Fall ist die hier vorgestellte Methode dem noch zu besprechenden QR-Verfahren, bei dem alle Eigenwerte bestimmt werden, deutlich überlegen.

8.3 Reduktion auf Hessenberggestalt

$$H = \begin{bmatrix} * & & & * \\ * & \ddots & & \\ & \ddots & \ddots & \\ 0 & & * & * \end{bmatrix}$$

heißt *Hessenbergmatrix*, es gilt also $h_{ij} = 0$ für $i < j - 1$.

Für eine allgemeine Matrix möchte man das gleiche Verfahren wie bei der Gauß-Elimination anwenden, um A auf Dreiecksgestalt zu bringen. Hier benötigen wir jedoch eine Ähnlichkeitstransformation, um die Eigenwerte von A zu erhalten. Statt (abgesehen von Permutationen)

$$A_i = L_{i-1} A_{i-1}$$

muss

$$A_i = L_{i-1} A_{i-1} (L_{i-1})^{-1}$$

verwendet werden. Die Matrix-Multiplikation mit $(L_{i-1})^{-1}$ zerstört jedoch die gerade eliminierte Spalte. Eliminiere daher nur unterhalb der Subdiagonalen. Wie bei der Gauß-Elimination verwende die Permutationsmatrizen

$$P_{jk} = \begin{bmatrix} 1 & & & & & & & & & & \\ & \ddots & & & & & & & & & \\ & & 1 & & & & & & & & \\ & & & 0 & \cdots & \cdots & \cdots & 1 & & & \\ & & & \vdots & 1 & & & \vdots & & & \\ & & 0 & \vdots & & \ddots & & \vdots & 0 & & \\ & & & \vdots & & & 1 & \vdots & & & \\ & & & 1 & \cdots & \cdots & \cdots & 0 & & & \\ & & & & & & & & 1 & & \\ & & & & & & & & & \ddots & \\ & & & & & & & & & & 1 \end{bmatrix}, \quad P_{jk}^{-1} = P_{jk}$$

mit Wirkung

$P_{jk}A$ wie A, aber Zeile j wird mit Zeile k vertauscht,

AP_{jk} wie A, aber Spalte j wird mit Spalte k vertauscht.

Die *Frobenius-Matrizen* sind definiert durch

$$G_j = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & 0 & \\ & & \ddots & & & & \\ & & & 1 & & & \\ & & & -l_{j+1\,j} & \ddots & & \\ & 0 & & \vdots & & \ddots & \\ & & & -l_{n\,j} & & & 1 \end{bmatrix}, \quad G_j^{-1} = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & 0 & \\ & & \ddots & & & & \\ & & & 1 & & & \\ & & & l_{j+1\,j} & \ddots & & \\ & 0 & & \vdots & & \ddots & \\ & & & l_{n\,j} & & & 1 \end{bmatrix}.$$

mit Wirkung

AG_j addiert das $-l_{r\,j}$-fache der Spalte r auf die Spalte j von A für $k = j+1, \ldots, n$

$G_j^{-1}A$ addiert das $-l_{r\,j}$-fache der Zeile j auf die Zeile r von A für $k = j+1, \ldots, n$

Algorithmus: Die ersten $i-1$ Spalten von A_{i-1} habe Hessenberggestalt

$$A_{i-1} = \left[\begin{array}{cccc|ccc} * & \cdots & \cdots & * & * & \cdots & * \\ * & \ddots & & \vdots & \vdots & & \vdots \\ & \ddots & \ddots & \vdots & \vdots & & \vdots \\ 0 & & * & * & * & \cdots & * \\ \hline & & & * & * & \cdots & * \\ & 0 & & & \vdots & & \vdots \\ & & & & * & \cdots & * \end{array}\right]$$

Bestimme r mit

$$|a_{r\,i}| = \max_{i+1 \le j \le n} |a_{j\,i}|$$

und bilde

$$A' = P^{-1}_{r\,i+1} A_{i-1} P_{r\,i+1}.$$

$P^{-1}_{r\,i+1} A_{i-1}$ vertauscht wie gewünscht die Zeile $i+1$ mit der Zeile $r \ge i+1$. $A_{i-1} P_{r\,i+1}$ vertauscht daher die Spalte $i+1$ mit der Spalte $r \ge i+1$, lässt mithin die Spalten 1 bis i unverändert, zerstört also nicht die in der i-ten Spalte durchgeführte Pivotisierung.

Eliminiere nun die i-te Spalte mit Hilfe von $a'_{i+1\,i}$:

$$A_i = G^{-1}_{i+1} A' G_{i+1}.$$

Auch hier werden durch $A'G_{i+1}$ die ersten i Spalten von A' nicht verändert. Wir erhalten daher

$$H = T^{-1} A T$$

mit einer Hessenbergmatrix H.

Analog zur QR-Zerlegung zur Lösung eines linearen Gleichungssystems können wir A auch durch Householder-Matrizen auf Hessenberggestalt bringen. Sei

$$A = \begin{bmatrix} a_{11} & c^T \\ b & B \end{bmatrix} \quad \text{mit } b \neq 0.$$

Wir bestimmen dann $w \in \mathbb{C}^{n-1}$ mit $|w| = 1$ und

$$Q_1 b = (I_{n-1} - 2ww^H) b = k e_1.$$

Mit $P_1 = \begin{bmatrix} 1 & 0^T \\ 0 & Q_1 \end{bmatrix}$ gilt dann

$$A_1 = P_1 A P_1 = \left[\begin{array}{c|c} a_{11} & c^T Q_1 \\ \hline k & \\ 0 & \\ \vdots & Q_1 B Q_1 \\ 0 & \end{array}\right].$$

Die übrigen Spalten behandelt man ganz analog.

8.4 Bestimmung der Eigenwerte einer Hessenbergmatrix Auch hier können die Eigenwerte alternativ mit dem QR-Verfahren aus Abschnitt 8.6 bestimmt werden.

Sei B eine unzerlegbare Hessenbergmatrix, also $b_{i+1\,i} \neq 0$ für $i = 1, \ldots, n-1$. Für $\mu \in \mathbb{C}$ bestimmen wir $\alpha, x_1, \ldots, x_{n-1}$ mit

$$(B - \mu I)x = \alpha e_1, \quad x_n = 1$$

oder

$$\begin{array}{rcrcrcrcl} (b_{11}-\mu)x_1 & + & b_{12}x_2 & + & \ldots & + & b_{1\,n-1}x_{n-1} & + & b_{1n}x_n = \alpha \\ b_{21}x_1 & + & (b_{22}-\mu)x_2 & + & \ldots & + & b_{2\,n-1}x_{n-1} & + & b_{2n}x_n = 0 \\ & & & & & & & & \ldots \\ & & & & & & b_{n\,n-1}x_{n-1} & + & (b_{nn}-\mu)x_n = 0. \end{array} \tag{8.3}$$

Mit $x_n = 1$ bestimmt man aus der letzten Gleichung x_{n-1}, dann x_{n-2} und aus der zweiten Gleichung x_1. Aus der ersten Gleichung erhält man das α. Die Größen $x_1, \ldots, x_{n-1}, \alpha$ sind daher eindeutig bestimmt. Fasst man das System als Gleichung in x auf für gegebenes α, so folgt aus der Cramerschen Regel

$$1 = x_n = \frac{\alpha(-1)^{n-1} b_{21} b_{32} \ldots b_{n\,n-1}}{\det(B - \mu I)},$$

also

$$\alpha(\mu) = \frac{(-1)^{n-1}}{b_{21} b_{32} \ldots b_{n\,n-1}} \det(B - \mu I).$$

Damit ist α ein Vielfaches des gesuchten $\det(B - \mu I)$. Wir fassen das System (8.3) als ein System in $x(\mu)$ auf und differenzieren nach μ

$$\begin{array}{rcrcrcrcl} (b_{11}-\mu)x_1' - x_1 & + & b_{12}x_2' & + & \ldots & + & b_{1\,n-1}x_{n-1}' & = \alpha' \\ b_{21}x_1' & + & (b_{22}-\mu)x_2' - x_2 & + & \ldots & + & b_{2\,n-1}x_{n-1}' & = 0 \\ & & & & & & \ldots & \\ & & & & & & b_{n\,n-1}x_{n-1}' - x_n & = 0. \end{array} \tag{8.4}$$

Mit $x_n = 1$ bestimmt man hieraus $x_{n-1}', \ldots, x_1'$ und schließlich α'. Daher

$$\alpha'(\mu) = \frac{(-1)^{n-1}}{b_{21} b_{32} \ldots b_{n\,n-1}} \det(B - \mu I)'.$$

Mit den Größen α und α' kann das Newton-Verfahren durchgeführt werden.

8.5 Potenzmethoden

Die einfache Vektoriteration Sei $A \in \mathbb{C}^{n \times n}$ eine diagonalisierbare Matrix. Für die Eigenwerte λ_i in

$$Ax_i = \lambda_i x_i, \quad |x_i| = 1,$$

gelte

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

Sei v_0 ein Vektor mit

$$v_0 = \sum_{i=1}^{n} \alpha_i x_i, \quad \alpha_1 \neq 0. \tag{8.5}$$

Satz Für das Verfahren

$$v_{j+1} = Av_j = A^{j+1}v_0$$

gilt unter obigen Voraussetzungen

$$\lim_{j\to\infty} \frac{1}{\lambda_1^j} v_j = \alpha_1 x_1, \tag{8.6}$$

oder genauer

$$\left| \frac{v_j}{\lambda_1^j} - \alpha_1 x_1 \right| \leq c \left| \frac{\lambda_2}{\lambda_1} \right|^j .$$

Beweis: Mit der Darstellung von v_0 in (8.5) gilt

$$v_j = A^j v_0 = \sum_{i=1}^{n} \lambda_i^j \alpha_i x_i,$$

daher

$$\left| \frac{v_j}{\lambda_1^j} - \alpha_1 x_1 \right| = \left| \sum_{i=2}^{n} \frac{\lambda_i^j}{\lambda_1^j} \alpha_i x_i \right| \leq c \left| \frac{\lambda_2}{\lambda_1} \right|^j$$

□

In der Praxis normiert man die v_j, beispielsweise durch

$$z_j = \frac{1}{\tau_j} v_j,$$

wobei τ_j die betragsmäßig größte Komponente von v_j ist. Dann folgt aus (8.6)

$$\lim \frac{\tau_{j+1}}{\tau_j} = \lambda_1, \quad \lim z_j = \tilde{\alpha}\alpha_1 x_1,$$

wobei $\tilde{\alpha}$ ein Normierungsfaktor ist.

Natürlich konvergiert das Verfahren auch, wenn λ_1 ein k-facher Eigenwert mit $|\lambda_1| > |\lambda_{k+1}|$ ist, sofern A nach wie vor diagonalisierbar ist. Wie die obigen Formeln zeigen, konvergiert der normierte Vektor v_j gegen $\sum_{i=1}^{k} \alpha_j x_j$, der ebenfalls Eigenvektor zu λ_1 ist.

Das Verfahren versagt offenbar schon, wenn es mehrere verschiedene Eigenwerte mit maximalem Betrag gibt. Insbesondere lässt sich das Verfahren bei reellen Matrizen nicht verwenden, wenn λ_1, λ_2 komplex konjugiert sind.

Bestimmung mehrerer Eigenwerte mit der einfachen Vektoriteration Wenn λ_1 und x_1 mit der einfachen Vektoriteration bestimmt wurden, so kann man bei hermiteschen Matrizen ausnutzen, dass die Eigenvektoren aufeinander senkrecht stehen. Man wählt daher ein $v_0 \perp x_1$ und führt mit v_0 die einfache Vektoriteration durch. Wegen

$$v_j = \sum_{i=2}^{n} \alpha_i \lambda_i^j x_i \perp x_1$$

konvergiert v_j/λ_2^j gegen $\alpha_2 x_2$, sofern $\alpha_2 \neq 0$ und $|\lambda_2| > |\lambda_3|$. Ist $\lambda_1 = \lambda_2$, so konvergieren die normierten v_j gegen einen weiteren Eigenvektor von λ_1.

Aufgrund von Rundungsfehlern ist $v_j \not\perp x_1$. Man bestimmt daher β_j aus $\tilde{v}_j = Av_{j-1}$ mit

$$\tilde{v}_j + \beta_j x_1 \perp x_1, \quad v_j = \tilde{v}_j + \beta_j x_1,$$

was man als *Nachorthogonalisieren* bezeichnet.

Bei unsymmetrischen Matrizen verwende *Matrixdeflation*:

Satz Sei $A \in \mathbb{R}^{n\times n}$ und $Ax = \lambda x$ für $x \neq 0$. Mit $w = x + |x|e_1$ und

$$H = I - 2\frac{ww^T}{|w|^2}$$

gilt dann $H = H^{-1}$ und

$$B = HAH = \begin{bmatrix} \lambda & b^T \\ 0 & C \end{bmatrix}$$

Beweis: Für die Householder-Matrix H gilt bekanntlich $H = H^{-1} = H^T$. Da w in Richtung $x + |x|e_1$ zeigt, steht $x - |x|e_1$ auf w senkrecht. Daher gilt $H(|x|e_1) = x$ und $H^{-1}x = |x|e_1$. Also

$$Be_1 = H^{-1}AHe_1 = HA\left(\frac{x}{|x|}\right) = H\left(\frac{\lambda x}{|x|}\right) = \lambda e_1.$$

□

Nach Bestimmung von B kann die Vektoriteration mit der Matrix C fortgesetzt werden.

Verwendung des Rayleigh-Quotienten Sei A hermitesch und $v_j = A^j v_0$ eine Näherung des ersten Eigenvektors aus der Potenzmethode. Mit Hilfe des Rayleigh-Quotienten lässt sich die zugehörige Näherung des ersten Eigenwerts erheblich verbessern. Verwende

$$\lambda = \frac{(Av_j, v_j)}{(v_j, v_j)} = \frac{(v_{j+1}, v_j)}{(v_j, v_j)}.$$

Dann gilt

$$|\lambda - \lambda_1| \leq c\left|\frac{\lambda_2}{\lambda_1}\right|^{2j} \quad \text{falls } \alpha_1 \neq 0 \text{ in } v_0 = \sum_{i=1}^{n} \alpha_i x_i$$

im Gegensatz zu $O((\lambda_2/\lambda_1)^j)$ aus Satz 8.5.

Beweis: Für $v_0 = \sum_i \alpha_i x_i$ gilt wie im Beweis von Satz 8.5

$$v_j = A^j v_0 = \sum_{i=1}^{n} \lambda_i^j \alpha_i x_i,$$

$$\lambda = \frac{(Av_j, v_j)}{(v_j, v_j)} = \frac{\sum_i |\alpha_i|^2 \lambda_i^{2j+1}}{\sum_i |\alpha_i|^2 \lambda_i^{2j}}$$

$$= \frac{|\alpha_1|^2\lambda_1 + \sum_{i=2}^n |\alpha_i|^2 \left(\frac{\lambda_i}{\lambda_1}\right)^{2j} \lambda_i}{|\alpha_1|^2 + \sum_{i=2}^n |\alpha_i|^2 \left(\frac{\lambda_i}{\lambda_1}\right)^{2j}} = \frac{|\alpha_1|^2\lambda_1 + y}{|\alpha_1|^2 + z} = f(y,z).$$

Wegen $|\lambda_i/\lambda_1| \leq |\lambda_2/\lambda_1|$ ist

$$|y|, |z| \leq c\left|\frac{\lambda_2}{\lambda_1}\right|^{2j} \to 0 \quad \text{für } j \to \infty.$$

Nach Taylor gilt

$$f(y,z) = f(0,0) + f_y(y_1, z_1)y + f_z(y_2, z_2)z$$

mit $|y_1|, |y_2| \leq |y|$ und $|z_1|, |z_2| \leq |z|$. Wegen

$$|f_y| = \left| \frac{1}{|\alpha_1|^2 + z} \right| \leq c \quad \text{für } |z| \leq \frac{1}{2}|\alpha_1|^2$$

$$|f_z| = \left| \frac{|\alpha_1|^2 \lambda_1 + y}{||\alpha_1|^2 + z|^2} \right| \leq c \quad \text{für } |z| \leq \frac{1}{2}|\alpha_1|^2$$

folgt

$$\lambda = \lambda_1 + O\Big(\Big|\frac{\lambda_2}{\lambda_1}\Big|^{2j}\Big).$$

□

Also: Bei hermiteschen Matrizen die vorletzte Iterierte der Potenzmethode immer in den Rayleigh-Quotienten einsetzen!

Allerdings wir diese Regel relativiert durch folgende Anmerkung. Statt der Potenzmethode kann man mit vergleichbarem Aufwand (eine Matrix-Vektor-Multiplikation pro Schritt) auch das Lanczos-Verfahren durchführen. In $J_{i+1} = U_{i+1}^H A U_{i+1}$ bestehen die Spaltenvektoren aus einer Orthonormalbasis des Krylov-Raum $K_{i+1}(A, v_0)$. Dann gilt

$$\lambda_{max}(J_{i+1}) = \max_{x \neq 0} \frac{(J_{i+1}x, x)}{(x, x)} = \max_{x \neq 0} \frac{(U_{i+1}^H A U_{i+1} x, x)}{(x, x)} = \max_{x \neq 0} \frac{(AU_{i+1}x, U_{i+1}x)}{(U_{i+1}x, U_{i+1}x)},$$

Im Lanczos-Verfahren wird der Rayleigh-Quotient von A über den ganzen Krylov-Raum maximiert und nicht nur über den eindimensionalen Teilraum $\operatorname{span} v_i$.

Die inverse Iteration nach Wieland Ist man nur am betragsmäßig kleinsten Eigenwert der regulären Matrix A interessiert, so kann man die inverse Iteration verwenden. Für einen Start $v_0 \in \mathbb{C}^n$ ist sie definiert durch

$$v_j = A^{-1} v_{j-1} = A^{-j} v_0.$$

Da dies mit der einfachen Vektoriteration mit Matrix A^{-1} übereinstimmt, gilt Satz 8.5 sinngemäß.

Ist λ eine Schätzung des Eigenwerts λ_k, so verwende

$$v_j = (A - \lambda I)^{-1} v_{j-1} = (A - \lambda I)^{-j} v_0.$$

Für diagonalisierbares A folgt aus

$$v_0 = \sum_{i=1}^{n} \alpha_i x_i, \quad \alpha_1 \neq 0,$$

dass

$$v_j = \sum_{i=1}^{n} \alpha_i (A - \lambda I)^{-j} x_i = \sum_{i=1}^{n} \frac{\alpha_i}{(\lambda_i - \lambda)^j} x_i$$

daher

$$(\lambda_k - \lambda)^j v_j = \alpha_k x_k + \sum_{i=1,\, i \neq k}^{n} \alpha_i \left(\frac{\lambda_k - \lambda}{\lambda_i - \lambda} \right)^j x_i.$$

Ist die Schätzung von λ_k gut, so konvergiert dieses Verfahren sehr schnell.

8.6 Das QR-Verfahren Das QR-Verfahren ist für allgemeine Matrizen $A \in \mathbb{C}^{n\times n}$ definiert und liefert ziemlich sicher alle Eigenwerte von A. In jedem Schritt des Verfahrens muss eine QR-Zerlegung einer Matrix durchgeführt werden, dessen Aufwand sich vermindert, wenn die Matrix zuvor auf eine einfachere Gestalt gebracht wurde.

Wir definieren $A_1 = A$ und konstruieren schrittweise Matrizen A_i durch

$$A_i = Q_i R_i, \quad A_{i+1} = R_i Q_i,$$

wobei mit $Q_i R_i$ die Householder-Zerlegung in eine unitäre Matrix Q_i und eine rechte obere Dreiecksmatrix R_i gemeint ist.

Satz Die Matrizen A_i, Q_i, R_i sowie

$$P_i = Q_1 \dots Q_i, \quad U_i = R_i \dots R_1$$

haben die folgenden Eigenschaften:

(a) $A_{i+1} = Q_i^H A_i Q_i$,

(b) $A_{i+1} = P_i^H A P_i$, d.h. A_{i+1} ist unitär ähnlich zu A,

(c) $A^i = P_i U_i$.

Beweis: (a) Aus $A_i = Q_i R_i$ folgt $Q_i^{-1} A_i Q_i = R_i Q_i = A_{i+1}$.

(b) folgt aus der sukzessiven Anwendung von (a).

(c) Aus (b) folgt

$$Q_1 \dots Q_i A_{i+1} = A Q_1 \dots Q_i,$$

also

$$\begin{aligned} P_i U_i &= Q_1 \dots Q_{i-1}(Q_i R_i) R_{i-1} \dots R_1 = Q_1 \dots Q_{i-1} A_i R_{i-1} \dots R_1 \\ &= A Q_1 \dots Q_{i-1} R_{i-1} \dots R_1 = A P_{i-1} U_{i-1} = \dots = A^i. \end{aligned}$$

□

Aus $A_{i+1} = R_i Q_i$ und (b) folgt

$$P_i R_i Q_i = P_i A_{i+1} = A P_i.$$

Wir setzen $P_0 = I$ und erhalten aus der letzten Gleichung

$$P_i R_i = A P_{i-1} \quad \text{für alle } i \geq 0. \tag{8.7}$$

Mit dieser Matrizengleichung untersuchen wir nun, was mit dem „vorderen“ Teil von P_i geschieht. Für $1 \leq r < n$ setzen wir

$$P_i = \left[P_i^{(r)} | \hat{P}_i^{(r)}\right], \quad R_i = \begin{bmatrix} R_i^{(r)} & * \\ 0 & \hat{R}_i^{(r)} \end{bmatrix}$$

mit $R_i^{(r)} \in \mathbb{C}^{r\times r}$, $\hat{R}_i^{(r)} \in \mathbb{C}^{(n-r)\times(n-r)}$. Aus $AP_{i-1} = P_i R_i$ in (8.7) folgt

$$A P_{i-1}^{(r)} = P_i^{(r)} R_i^{(r)}.$$

Wir fassen $P_i^{(r)} R_i^{(r)} z$ für $z \in \mathbb{C}^r$ als Linearkombination der orthonormalen Spaltenvektoren von $P_i^{(r)}$ auf. Setze

$$\mathcal{P}_i^{(r)} = \text{Bild}\,(P_i^{(r)}) = \{P_i^{(r)} z : z \in \mathbb{C}^r\}.$$

Damit ist

$$A\mathcal{P}_{i-1}^{(r)} \subset \mathcal{P}_i^{(r)}$$

mit Gleichheit, wenn A und damit $R_i^{(r)}$ regulär ist. Also:

Der QR-Algorithmus ist eine Potenzmethode für die Unterräume $\mathcal{P}_i^{(r)}$.

Nun betrachten wir den Fall $r = 1$ genauer. Mit $P_0 = I$ folgt für die erste Spalte in (8.7)

$$Ae_1 = \kappa p_1$$

mit $\kappa = (R_1)_{11}$. Bezeichnen wir die erste Spalte von P_i mit p_i, so folgt

$$\kappa_i p_i = Ap_{i-1} = A^i e_1,$$

d.h. abgesehen von der Normierung mit κ_i, die wegen $|p_i| = 1$ notwendig ist, erhalten wir die Potenzmethode für den Startvektor e_1. Wenn es also nur einen betragsgrößten Eigenwert gibt und der zugehörige Eigenvektor x_1 in der ersten Spalte von A vorkommt, so gilt $\kappa_i p_i \to x_1$.

Es gibt einen völlig analogen Zusammenhang des QR-Verfahrens mit der inversen Iteration. Aus (8.7) folgt wegen $P_{i-1}^H P_{i-1} = I$ für reguläres A

$$P_i R_i = AP_{i-1} \;\Rightarrow\; R_i P_{i-1}^H = P_i^H A \;\Rightarrow\; P_{i-1}^H A^{-1} = R_i^{-1} P_i^H$$

$$\Rightarrow \quad A^{-H} P_{i-1} = P_i R_i^{-H}, \tag{8.8}$$

wobei hier $B^{-H} = (B^{-1})^H = (B^H)^{-1}$ verwendet wurde. R_i^{-1} ist auch eine rechte obere Dreiecksmatrix, somit ist R_i^{-H} eine linke untere Dreiecksmatrix. Bezeichnen wir hier die letzte Spalte von P_i mit p_i, so folgt aus der letzten Spalte in (8.8)

$$A^{-H} p_{i-1} = \kappa_i p_i,$$

d.h. die p_i sind die Iterierten einer inversen Iteration mit Matrix A^H.

Satz Sei A diagonalisierbar mit

$$|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| > 0. \tag{8.9}$$

Es sei $A = XDX^{-1}$, $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, und die Matrix $Y = X^{-1}$ besitze eine LR-Zerlegung. Dann gilt für die Elemente $a_{jk}^{(i)}$ von A_i

$$\lim_{i\to\infty} a_{jk}^{(i)} \to 0 \quad \text{für } j > k,$$

$$\lim_{i\to\infty} a_{kk}^{(i)} \to \lambda_k \quad \text{für } k = 1, \ldots, n$$

Bemerkung Die Voraussetzung, dass die Matrix Y eine Dreieckszerlegung besitzt, garantiert lediglich, dass die Eigenwerte im Grenzwert der Größe nach auf der Hauptdiagonalen geordnet sind. Aufgrund von Rundungsfehlern dürfte sich dies in jedem Fall einstellen.

Beweis: Sei $X = QR$ die QR-Zerlegung von X mit $r_{ii} > 0$ und sei $Y = LU$ die LR-Zerlegung von Y. Wegen $A = XDX^{-1} = QRDR^{-1}Q^H$ gilt

$$Q^H AQ = RDR^{-1}. \tag{8.10}$$

$Q^H AQ$ ist damit obere Dreiecksmatrix mit Diagonalelementen in der in (8.9) angegebenen Reihenfolge. Weiter ist

$$A^m = XD^m X^{-1} = QRD^m LU = QRD^m LD^{-m} D^m U.$$

Wegen

$$(D^m LD^{-m})_{ij} = l_{ij} \left(\frac{\lambda_i}{\lambda_j}\right)^m = \begin{cases} 0 & \text{für } i < j \\ 1 & \text{für } i = j \\ \to 0 & \text{für } i > j \end{cases}$$

ist

$$D^m L D^{-m} = I + E_m \quad \text{mit} \lim_{m\to\infty} E_m = 0.$$

Daher gilt

$$\begin{aligned} A^m &= QR(I+E_m)D^m U = Q(I + RE_mR^{-1})RD^mU \\ &=: Q(I+F_m)RD^mU \quad \text{mit} \lim_{m\to\infty} F_m = 0. \end{aligned}$$

Nach obigem Lemma hängt die QR-Zerlegung (mit positiver Hauptdiagonale in R) stetig von den Matrixelementen ab. Da $I = I \cdot I$ die QR-Zerlegung von I ist, folgt für die QR-Zerlegung

$$I + F_m = \hat{Q}_m \hat{R}_m, \quad \hat{R}_m \text{ mit positiver Diagonale,}$$

dass $\tilde{Q}_m \to I$ und $R_m \to I$. Wegen $A^i = P_i U_i$ ist

$$A^m = (Q\hat{Q}_m)(\hat{R}_m R D^m U) = P_m U_m,$$

und da die QR-Zerlegung bis auf die Multipikation mit einer Diagonalmatrix eindeutig ist, folgt hieraus: Es existieren unitäre Diagonalmatrizen D_m mit

$$P_m D_m = Q\tilde{Q}_m \to Q.$$

Daher folgt aus $A_{m+1} = P_m^H A P_m$

$$D_m^H A_{m+1} D_m = D_m^H P_m^H A P_m D_m \;\to\; Q^H A Q \overset{(8.10)}{=} RDR^{-1}, \tag{8.11}$$

also

$$\lim_{m\to\infty} a_{ij}^{(m+1)} \frac{d_j^{(m)}}{d_i^{(m)}} = \begin{cases} \lambda_i & \text{falls } i = j \\ \to 0 & \text{falls } i > j \end{cases}.$$

Wegen $|d_i^{(m)}| = 1$ für alle i also

$$\lim_{m\to\infty} a_{ij}^{(m)} = \begin{cases} \lambda_i & \text{falls } i = j \\ \to 0 & \text{falls } i > j \end{cases}.$$

□

Beispiel Für die Matrix

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 3 & -1 \\ -2 & -2 & 2 \end{bmatrix}$$

mit den Eigenwerten $\lambda_1 = 3$, $\lambda_2 = 2$, $\lambda_3 = 1$ und den Eigenvektoren

$$X = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 1 & 1 \\ -2 & -1 & 0 \end{bmatrix}, \quad X^{-1} = \begin{bmatrix} 1 & 1 & 0 \\ -2 & -2 & -1 \\ 0 & 1 & 1 \end{bmatrix}$$

ist die Voraussetzung des letzten Satzes, dass X^{-1} eine LR-Zerlegung besitzen soll, nicht erfüllt. Denn die Hauptuntermatrix $X^{-1}[2]$ ist singulär, so dass spätestens im zweiten Schritt des Gauß-Algorithmus eine Zeilenvertauschung stattfinden muss. Nach 30 Schritten des QR-Verfahrens erhalten wir

$$A_{30} = \begin{bmatrix} 3.0000 & -2.0000 & 2.9999 \\ 1.16e-6 & 0.9999 & -0.9999 \\ 1.16e-6 & 6.69e-5 & 1.9999 \end{bmatrix}$$

Die Diagonalelemente scheinen also gegen die Eigenwerte zu konvergieren, wobei diese nicht dem Betrage nach geordnet sind. Nach weiteren 40 Schritten erhält man

$$A_{70} = \begin{bmatrix} 3.0000 & -3.5355 & 0.7071 \\ 1.48e-13 & 1.9999 & -1.0000 \\ -7.52e-19 & -5.07e-6 & 1.0000 \end{bmatrix}$$

Aufgrund von Rundungsfehlern bekommt man schließlich doch noch Konvergenz gegen eine Dreiecksmatrix mit der Betragsgröße nach geordneten Eigenwerten.

Aus (8.11) folgt, dass die Elemente von A_m oberhalb der Diagonale oszillieren und nur dem Betrage nach konvergieren. Daher sagt man, dass der QR-Algorithmus „im Wesentlichen“ konvergiert.

Ist A eine reelle Matrix, so verläuft der ganze Algorithmus in $\mathbb{R}$. Besitzt A komplexe Eigenwerte, so kann er in der angegebenen Form nicht konvergieren. Die Voraussetzung des obigen Satzes, dass alle Eigenwerte betragsmäßig verschieden sind, ist dann ja auch verletzt.

Ist A hermitesch, so sind wegen $A_{i+1} = P_i^H A P_i$ auch alle Iterierten hermitesch.

8.7 Rotations- und Reflektionsmatrizen Mit $0 \le \phi < 2\pi$ und $c = \cos\phi$, $s = \sin\phi$ ist die *Rotationsmatrix* zum Winkel ϕ auch *Givens-Rotation* genannt, definiert durch

$$U_{12}(\phi) = \begin{bmatrix} c & -s \\ s & c \end{bmatrix}.$$

$U_{12}x$ bewirkt eine Drehung des Vektors x um den Winkel ϕ. Die Inverse ist natürlich die Drehung um den Winkel $-\phi$, also

$$U_{12}(\phi)^{-1} = U_{12}(-\phi) = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}$$

Wegen $U^T = U^{-1}$ sind Drehmatrizen orthogonal.

Im $\mathbb{R}^n$ setzen wir solche Matrizen ein, um Drehungen in der i,j-Ebene zu beschreiben

$$U_{ij} = U_{ij}(\phi) = \begin{bmatrix} 1 & & & & & & & & & & \\ & \ddots & & & & & & & & & \\ & & 1 & & & & & & & & \\ & & & c & \cdots & \cdots & \cdots & -s & & & \\ & & & \vdots & 1 & & & \vdots & & & \\ & 0 & & \vdots & & \ddots & & \vdots & & 0 & \\ & & & \vdots & & & 1 & \vdots & & & \\ & & & s & \cdots & \cdots & \cdots & c & & & \\ & & & & & & & & 1 & & \\ & & & & & & & & & \ddots & \\ & & & & & & & & & & 1 \end{bmatrix}.$$

Auch diese Matrizen sind orthogonal mit $U_{ij}(\phi)^{-1} = U_{ij}^T(\phi)$. Die Multiplikation mit einer Matrix A hat die folgende Wirkung

$U_{ij}A$	wie A, aber die Zeilenvektoren i und j werden verändert,
AU_{ij}	wie A, aber die Spaltenvektoren i und j werden verändert

Solche Drehmatrizen werden zur Elimination von Elementen einer Matrix genutzt. Wir können uns auf den Fall $n = 2$ beschränken und bestimmen für $a, b \in \mathbb{C}$ die Größen $c, s \in \mathbb{C}$ so, dass $U_{12}(a,b)^T = (r,0)^T$,

$$a \neq 0: \quad U_{12} = \frac{1}{\sqrt{|a|^2+|b|^2}} \begin{bmatrix} |a| & \frac{a}{|a|}\overline{b} \\ -\frac{\overline{a}}{|a|}b & |a| \end{bmatrix}, \quad r = \frac{a}{|a|}\sqrt{|a|^2+|b|^2}, \tag{8.12}$$

$$a = 0,\ b \neq 0: \quad U_{12} = \frac{1}{|b|} \begin{bmatrix} 0 & \overline{b} \\ -b & 0 \end{bmatrix}, \quad r = |b|.$$

Offenbar sind beide Matrizen unitär bzw. orthogonal, wenn $a, b \in \mathbb{R}$.

Mit $0 \leq \phi < 2\pi$ und $c = \cos\phi$, $s = \sin\phi$ ist die *Reflektionsmatrix* zum Winkel ϕ definiert durch

$$V_{12}(\phi) = \begin{bmatrix} c & s \\ s & -c \end{bmatrix}$$

Wegen

$$V_{12}(\phi) = \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

setzt sich $V_{12}x$ aus einer Spiegelung und einer Drehung zusammen. V_{12} ist symmetrisch und orthogonal, insbesondere $V_{12}^{-1} = V_{12}$.

Im $\mathbb{R}^n$ oder $\mathbb{C}^n$ verwenden wir entsprechend

$$V_{ij} = V_{ij}(\phi) = \begin{bmatrix} 1 & & & & & & & & & \\ & \ddots & & & & & & & & \\ & & 1 & & & & & & & \\ & & & c & \cdots & \cdots & \cdots & s & & \\ & & & \vdots & 1 & & & \vdots & & \\ & 0 & & \vdots & & \ddots & & \vdots & 0 & \\ & & & \vdots & & & 1 & \vdots & & \\ & & & s & \cdots & \cdots & \cdots & -c & & \\ & & & & & & & & 1 & \\ & & & & & & & & & \ddots \\ & & & & & & & & & 1 \end{bmatrix}.$$

Auch diese Matrizen sind symmetrisch und orthogonal mit $V_{ij}(\phi)^{-1} = V_{ij}(\phi)$. Die Wirkung bei der Multiplikation mit einer Matrix A ist analog zu U_{ij}.

8.8 Beschleunigung des QR Algorithmus Sei $A \in \mathbb{C}^{n\times n}$ eine beliebige Matrix. Ähnlich wie bei der inversen Iteration kann man durch Einführung eines Shift-Parameters k_i den QR-Algorithmus für die Eigenwerte in der Nähe von k_i beschleunigen. Der *QR-Algorithmus mit Shift* ist dann: Setze $A_1 = A$ sowie

$$A_i - k_i I = Q_i R_i,$$

$$A_{i+1} = R_i Q_i + k_i I.$$

Es gilt dann

$$R_i = Q_i^H (A_i - k_i I)$$

und

$$A_{i+1} = Q_i^H (A_i - k_i I) Q_i + k_i I = Q_i^H A_i Q_i, \tag{8.13}$$

also ist A_{i+1} wieder unitär ähnlich zu A_i. Wie zuvor schließt man hieraus, dass A_{i+1} auch zu A unitär ähnlich ist.

Satz Seien

$$P_i = Q_1 \dots Q_i, \quad U_i = R_i \dots R_1$$

Dann gilt

$$P_i U_i = (A - k_i I)(A - k_{i-1} I) \dots (A - k_1 I). \tag{8.14}$$

Beweis: Aus (8.13) erhalten wir

$$A_{i+1} = P_i^H A P_i. \tag{8.15}$$

Wir zeigen (8.14) durch Induktion über i. Für $i = 1$ besagt (8.14)

$$P_1 U_1 = Q_1 R_1 = A - k_1 I,$$

was gerade der erste Schritt des QR-Verfahrens mit Shift ist.

Sei (8.14) für $i - 1$ bewiesen. Dann folgt aus der Definition von A_{i+1} und (8.15)

$$R_i = (A_{i+1} - k_i I) Q_i^H = P_i^H (A - k_i I) P_i Q_i^H = P_i^H (A - k_i I) P_{i-1}.$$

Hieraus erhält man durch Rechtsmultiplikation mit U_{i-1} und Linksmultiplikation mit P_i

$$P_i U_i = (A - k_i I) P_{i-1} U_{i-1} = (A - k_i I)(A - k_{i-1} I) \dots (A - k_1 I)$$

nach Induktionsannahme. □

Aus (8.14) folgt

$$(A^H - \overline{k}_i I)^{-1} \dots (A^H - \overline{k}_1 I)^{-1} e_n = P_i U_i^{-H} e_n.$$

Da mit U_i^H auch U_i^{-H} eine untere Dreicksmatrix ist, gilt

$$P_i U_i^{-H} e_n = \sigma_i P_i e_n$$

für ein $\sigma_i \in \mathbb{C}$. Damit ist die letzte Spalte von P_i ein Vielfaches der i-ten Iterierten des inversen Verfahrens für die Matrix A^H mit Start e_n und Shifts $\overline{k}_1, \dots, \overline{k}_i$.

Man kann also schnelle Konvergenz erwarten, wenn man $\overline{k}_i$ als Rayleigh-Quotienten von A^H an der Stelle $p_n^{(i-1)}$, der letzten Spalte von P_{i-1}, wählt. Es gilt

$$\overline{k}_i = (A^H p_n^{(i-1)}, p_n^{(i-1)}) = (p_n^{(i-1)}, A p_n^{(i-1)}) \overset{(8.15)}{=} (e_n, A_i e_n) = \overline{a}_{nn}^{(i)}.$$

Dieser Shift heißt *Rayleigh-Quotienten Shift.*

Da eine QR-Zerlegung bei einer vollbesetzten Matrix A $O(n^3)$ Operationen kostet, bringt man sie besser zuvor mit den Methoden aus den Abschnitten 8.1 oder 8.3 auf Tridiagonal- oder Hessenberggestalt.

Liegt die Matrix $A \in C^{n \times n}$ in Hessenberggestalt vor, so berechnet man die QR-Zerlegung bessser mit den Drehmatrizen aus dem letzten Abschnitt. Sei $A_1 = A$ und $k_1 \in \mathbb{C}$ ein Shift-Parameter, beispielsweise $k_1 = a_{nn}^{(1)}$. Mit einer Drehmatrix $U_{12} \in \mathbb{C}^{n \times n}$ aus dem vorigen Abschnitt können wir das Element $a_{21}^{(1)}$ eliminieren. Für die folgende Elimination an der Stelle $(3, 2)$ vewenden wir

entsprechend eine Drehmatrix U_{23}. Da Multiplikation von links nur die zweiten und dritten Zeile ändert, bleibt die zuvor an der Stelle $(2,1)$ erzielte Null erhalten. Wir erhalten daher

$$R_1 = U_{n-1\,n} \dots U_{12}(A_1 - k_1 I)$$

mit einer rechten oberen Dreiecksmatrix R_1. Die neue Iterierte ist dann

$$A_2 = R_1 U_{12}^H \dots U_{n-1\,n}^H + k_1 I.$$

Da die Multiplikation von rechts mit $U_{i\,i+1}^H$ nur die i-te und $(i+1)$-te Spalte von R_1 verändert, besitzt A_2 wieder obere Hessenberggestalt. Die obere Hessenberggestalt bleibt also im ganzen Verlauf des Algorithmus erhalten. Da der QR-Schritt hier nur $O(n^2)$ Operationen kostet, lohnt sich die vorbereitende Transformation von A.

Ist A hermitesch, so haben wir bereits gesehen, dass alle A_i hermitesch sind. Ist A eine hermitesche Tridiagonalmatrix, so müssen die A_i hermitesche Hessenbergmatrizen, also wiederum hermitesche Tridiagonalmatrizen sein. In diesem Fall benötigen wir sogar nur $O(n)$ Operationen für die QR-Zerlegung.

Nimmt man bei jedem Schritt den Shift $k_i = a_{nn}^{(i)}$, so kann man zeigen, dass das Element $a_{n\,n-1}^{(i)}$ quadratisch gegen Null konvergiert, sofern der in $a_{nn}^{(i)}$ erscheinende Eigenwert einfach ist. In diesem Fall konvergiert $a_{nn}^{(i)}$ allerdings nicht notwendig gegen den betragsmäßig kleinsten Eigenwert, weil diese Eigenschaft durch den Shift verloren geht. Ist $a_{n\,n-1}^{(i)}$ genügend klein, so kann man den Algorithmus für die $(n-1, n-1)$-Hauptuntermatrix fortsetzen. Analog verfährt man, wenn ein anderes Element der Subdiagonalen klein wird. Will man die Eigenwerte bis zu einer Genauigkeit ε bestimmen, so hat sich die Abfrage

$$|a_{k+1\,k}^{(i)}| \le \varepsilon(|a_{kk}^{(i)}| + |a_{k+1\,k+1}^{(i)}|)$$

bewährt. In diesem Fall fährt man mit den beiden Untermatrizen $A[k]$ und $A[k+1, n]$ fort.

Beispiel Wir testen den QR-Algorithmus mit der Shift-Strategue $k_i = a_{nn}^{(i)}$ an Hand der Hessenbergmatrix

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 1 & 0 \\ 0 & 3 & 2 & 1 \\ 0 & 0 & 2 & 4 \end{bmatrix}.$$

Die folgende Tabelle enthält die Subdiagonalelemente mit dem gerade beschriebenen Abbruchkriterium mit $\varepsilon = 10^{-16}$.

i	a_{21}	a_{32}	a_{43}
1	2	3	2
2	$1.83e0$	$-2.23e0$	$1.61e0$
3	$1.65e0$	$1.43e0$	$3.55e-1$
4	$6.78e-1$	$-3.65e0$	$3.82e-2$
5	$7.63e-1$	$1.17e0$	$-1.63e-3$
6	$8.32e-1$	$-5.32e-1$	$3.93e-6$
7	$1.18e0$	$1.52e-1$	$-3.03e-11$
8	$1.64e0$	$-4.97e-2$	$1.62e-21$
9	$-3.22e0$	$2.89e-5$	
10	$1.81e0$	$5.33e-10$	
11	$-5.32e-1$	$-2.48e-19$	
12	$-4.46e-3$		
13	$5.89e-8$		
14	$-1.06e-17$		

Die quadratische Konvergenz am Ende einer jeden Spalte lässt sich gut erkennen. Für den QR-Algorithmus ohne Shift benötigt man für diese Genauigkeit etwa 300 Iterationen.

Implizite Shifts Der Algorithmus des letzten Abschnitts zusammen mit der beschriebenen Deflationstechnik liefert ein gutes Verfahren zur Bestimmung aller Eigenwerte von A. Es gibt jedoch eine Schwierigkeit. Ist A reell, so ist $k_1 = a_{nn}$ reell und alle Matrizen A_i sowie alle Shiftparameter k_i bleiben reell. Da die inverse Iteration nur dann rasch konvergiert, wenn die Shifts gegen einen Eigenwert von A konvergieren, kann der Algorithmus nicht effizient sein, wenn A komplexe Eigenwerte besitzt. Anstatt mit komplexen Shifts zu arbeiten, wollen wir mit reellen Shifts eine Quasidreiecksgestalt

$$\begin{bmatrix} R_{11} & * & \dots & * \\ & R_{22} & & \vdots \\ & & \ddots & \vdots \\ 0 & & & R_{kk} \end{bmatrix} \quad \text{mit } R_{ii} \in \mathbb{R} \text{ oder } R_{ii} \in \mathbb{R}^{2\times 2}$$

herstellen, um dann die komplexen Eigenwerte aus den $(2,2)$-Diagonalblöcken zu berechnen. Zu diesem Zweck wird eine Variante des QR-Algorithmus hergeleitet, bei der die Shifts nicht explizit durchgeführt werden. Wir hatten eine Hessenbergmatrix B als unzerlegbar bezeichnet, wenn $b_{i+1\,i} \neq 0$ für alle $i = 1, \dots, n-1$.

Lemma Es seien $A, B, Q \in \mathbb{C}^{n\times n}$, Q unitär und B eine unzerlegbare Hessenbergmatrix mit positiven Subdiagonalelementen $b_{i+1\,i}$. Ist $B = Q^H AQ$, so sind B und Q eindeutig durch die erste Spalte von Q festgelegt.

Beweis: Wir geben einen Algorithmus zur Berechnung von B und Q an.

Wir bezeichnen die Spalten von Q mit q_i und die Spalten von B mit b_i. Wir zeigen induktiv, dass die ersten k Spalten von Q und die ersten $k-1$ Spalten von B eindeutig festgelegt sind. Für $k = 1$ ist das richtig, denn die erste Spalte von Q ist gesetzt.

Wegen $QB = AQ$ und weil B Hessenbergmatrix ist, gilt

$$b_{k+1\,k} q_{k+1} + b_{kk} q_k + \dots + b_{1k} q_1 = A q_k. \tag{8.16}$$

Multiplikation dieser Gleichung mit q_i liefert

$$b_{ik} = (Aq_k, q_i), \quad i = 1, \dots, k.$$

Hierdurch ist die k-te Spalte von B außer $b_{k+1\,k}$ festgelegt. Wegen $b_{k+1\,k} \neq 0$ folgt ebenfalls aus (8.16)

$$q_{k+1} = \frac{1}{b_{k+1\,k}} \Big(Aq_k - \sum_{i=1}^{k} b_{ik} q_i \Big)$$

und aus $|q_{k+1}| = 1$ erhält man wegen der Positivität von $b_{k+1\,k}$ auf eindeutige Weise $b_{k+1\,k}$ und q_{k+1}. □

Unterbleibt in diesem Lemma die Forderung, dass $b_{k+1,k} > 0$, so sind q_{k+1} und $b_{k+1\,k}$ nur bis auf einen gemeinsamen Faktor vom Betrag 1 eindeutig bestimmt.

Satz Sei A eine unzerlegbare Hessenbergmatrix, k ein Shift-Parameter,

$$A - kI = QR, \quad B = RQ + kI$$

mit unzerlegbarem B. Dann gilt (vergleiche (8.13))

$$B = Q^H AQ,$$

und man kann B auch mit dem folgendem Algorithmus berechnen:

1. Bestimme eine unitäre Matrix $P \in \mathbb{C}^{n\times n}$, so dass die erste Spalte von P^H mit der ersten Spalte

von Q übereinstimmt.

2. Bestimme eine unitäre Matrix U mit dem Householder Verfahren, so dass $\tilde{B} = UPAP^H U^H$ eine obere Hessenbergmatrix ist.

Dann gilt $Q = P^H U^H$ und $\tilde{B} = B$.

Beweis: Sind $U_1, U_2, \ldots, U_{n-2}$ die Householder Matrizen zur Transformation von PAP^H auf Hessenberggestalt und ist $\tilde{Q}^H = U_{n-2}U_{n-3}\ldots U_1 P$, so gilt $\tilde{B} = \tilde{Q}^H A\tilde{Q}$.

Wegen der speziellen Gestalt der Matrizen

$$U_i = \begin{bmatrix} I_i & 0 \\ 0 & \tilde{U}_i \end{bmatrix}$$

verändert die Linksmultiplikation mit U_i nicht die erste Zeile, also haben $\tilde{Q}^H$ und P dieselbe erste Zeile. Damit haben auch P und Q^H dieselbe erste Zeile. Nach Lemma 8.8 gilt also $Q = \tilde{Q}$ und $B = \tilde{B}$. □

Der obige Algorithmus konzentriert die Wirkung des Shifts k in der Matrix P. Nachdem PAP^H berechnet worden ist, muss nur noch mit einem Standardverfahren PAP^H auf Hessenberggestalt gebracht werden.

Um P zu bestimmen, haben wir die erste Spalte von Q zu berechnen. Es ist $A - kI = QR$, wobei R eine rechte obere Dreiecksmatrix ist. Ist $Q = [q_1|\ldots|q_n]$, so gilt

$$r_{11}q_1 = QRe_1 = (A - kI)e_1,$$

also ist die erste Spalte von Q eine Vielfaches der ersten Spalte

$$a = (a_{11} - k, a_{21}, 0, \ldots, 0)^T$$

von $A - kI$. Wählt man P mit $Pa = \pm|a|e_1$, so gilt $P^H e_1 = \pm a/|a|$ und die ersten Spalten von P^H und Q stimmen überein. Man kann daher P als ebene Drehung wählen, die die zweite Komponenten a_{21} annulliert.

Bildet man hiermit PAP^H, so erhält man eine gestörte obere Hessenbergmatrix, in der zusätzlich das Element an der Stelle $(3, 1)$ besetzt ist. Durch Multiplikation von links mit einer Drehung U_{23} kann man dieses annullieren. Die Multiplikation mit U_{23} von rechts erzeugt ein von Null verschiedenes Element an der Stelle $(4, 2)$. Allgemein hat man im $k-$ten Schritt ein von Null verschiedenes Element an der Stelle $(k+1, k-1)$, das durch eine Rotation $U_{k\,k+1}$ an die Stelle $(k+2, k)$ „gejagt“ wird (=chasing the bulge).

Als Shift wählt man wie vorher den Rayleigh-Quotienten Shift oder auch auch den *Wilkinson-Shift*, d.h. k als den Eigenwert von

$$\begin{bmatrix} a^{(i)}_{n-1\,n-1} & a^{(i)}_{n-1\,n} \\ a^{(i)}_{n\,n-1} & a^{(i)}_{nn} \end{bmatrix},$$

der $a^{(i)}_{nn}$ am nächsten liegt.

Wir beschreiben nun, wie man die impliziten Shifts verwenden kann, um komplexe Eiegenwerte zu bestimmen. Dazu werden zwei QR-Schritte mit den Shifts k_0 und k_1 zu einem Schritt zusammengefasst. Ist A reelle obere Hessenbergmatrix und sind k_0 und k_1 konjugiert komplex, so kann man diesen Doppelschritt in reeller Arithmetik ausführen.

Der erste Schritt mit k_0 werde ausgeführt und führe auf die Matrix

$$A_1 = Q_0^H A Q_0,$$

anschließend folgt der zweite Schritt mit Shift k_1,

$$A_2 = Q_1^H A_1 Q_1 = Q_1^H Q_0^H A Q_0 Q_1.$$

Dies kann man wieder mit folgendem Algorithmus durchführen:

1. Bestimme eine unitäre Matrix U, die dieselbe erste Zeile wie $Q_1^H Q_0^H$ besitzt.

2. Transformiere UAU^H auf obere Hessenberggestalt A_2.

Wir bestimmen zuerst U. Es seien R_0, R_1 die oberen Dreiecksanteile der QR-Zerlegungen von $A - k_0 I$ bzw. $A - k_1 I$. Dann gilt nach (8.14)

$$Q_0 Q_1 R_1 R_0 = (A - k_1 I)(A - k_0 I).$$

Da $R_1 R_0$ eine Dreiecksmatrix ist, ist die erste Spalte von $Q_0 Q_1$ Vielfaches der ersten Spalte a von $(A - k_1 I)(A - k_0 I)$. Wir können also U als Householder-Matrix wählen, die a auf ein Vielfaches von e_1 transformiert.

Da A obere Hessenberggestalt besitzt, sind nur die drei Komponenten a_1, a_2, a_3 von a von Null verschieden. Man rechnet leicht nach, dass

$$\begin{aligned} a_1 &= a_{11}^2 - (k_0 + k_1) a_{11} + k_0 k_1 + a_{12} a_{21}, \\ a_2 &= a_{21}\big(a_{11} + a_{22} - (k_0 + k_1)\big), \\ a_3 &= a_{21} a_{32}. \end{aligned}$$

Wir wählen nun k_0 und k_1 als die Eigenwerte von

$$\begin{bmatrix} a_{n-1\,n-1} & a_{n-1\,n} \\ a_{n\,n-1} & a_{nn} \end{bmatrix}.$$

Dann gilt

$$k_0 + k_1 = a_{n-1\,n-1} + a_{nn}, \quad k_0 k_1 = a_{nn} a_{n-1\,n-1} - a_{n\,n-1} a_{n-1\,n}.$$

Hiermit erhält man für die a_i

$$\begin{aligned} a_1 &= a_{21}\Big(\frac{(a_{nn} - a_{11})(a_{n-1\,n-1} - a_{11}) - a_{n-1\,n} a_{n\,n-1}}{a_{21}} + a_{12}\Big), \\ a_2 &= a_{21}(a_{22} + a_{11} - a_{nn} - a_{n-1\,n-1}), \\ a_3 &= a_{21} a_{32}. \end{aligned}$$

Da man nur an der Richtung von a interessiert, kann man den gemeinsamen Faktor a_{21} fortlassen und hieraus U bestimmen. Auch bei komplex konjugierten Eigenwerten k_0, k_1 sind a_1, a_2, a_3 reell.

Die Matrix UAU^H ist ab der dritten Spalte eine Hessenbergmatrix, besitzt aber in den ersten zwei Spalten die „Beule“

$$B = \begin{bmatrix} a_{31} & a_{32} \\ a_{41} & a_{42} \end{bmatrix}.$$

Ähnlich wie zuvor kann man die Beule nach rechts unten jagen durch Matrizen der Form

$$U_i = \begin{bmatrix} I_i & 0 & 0 \\ 0 & H_i & 0 \\ 0 & 0 & I_{n-i-3} \end{bmatrix}$$

mit einer Householder Matrix $H_i \in \mathbb{R}^{3\times 3}$. Im letzten Schritt wird eine Drehung $U_{n-1\,n}$ benutzt.

Dieser Algorithmus konvergiert i.A. quadratisch. Es gibt allerdings Matrizen wie etwa

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix},$$

die unverändert bleiben. In der Praxis führt man daher zunächst einen Schritt mit zufällig gewählten Shifts k_0, k_1 aus.

8.9 Berechnung der singulären Werte Sei $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$. Falls $m < n$ betrachte man A^T statt A. Mit orthogonalen Matrizen $U \in \mathbb{R}^{m\times m}$ und $V \in \mathbb{R}^{n\times n}$ gilt dann

$$A = U \begin{bmatrix} D \\ 0 \end{bmatrix} V^T, \quad D = \operatorname{diag}(\sigma_1, \ldots, \sigma_n),$$

mit $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$ (siehe Abschnitt 7.7). Dann gilt

$$AA^T = U\Sigma\Sigma^T U^T,\ A^T A = V\Sigma\Sigma^T U^T \quad \text{mit } \Sigma = \begin{bmatrix} D \\ 0 \end{bmatrix},$$

es liegen also Jordansche Normalformen für AA^T und A^TA vor. U besteht aus m orthonormalen Eigenvektoren von AA^T und V aus n orthonormalen Eigenvektoren von A^TA. Man erhält also die Singulärwertzerlegung aus den Eigenvektoren von AA^T und A^TA. Das folgende Beispiel zeigt jedoch, dass diese Berechnungsmethode nicht sonderlich stabil ist.

Beispiel Sei

$$A = \begin{bmatrix} 1 & 1 \\ \varepsilon & 0 \\ 0 & \varepsilon \end{bmatrix}, \quad |\varepsilon| \leq \texttt{eps}$$

mit der Maschinengenauigkeit `eps`. Dann gilt

$$A^TA = \begin{bmatrix} 1 & 0 & \varepsilon \\ 1 & \varepsilon & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \varepsilon & 0 \\ 0 & \varepsilon \end{bmatrix} = \begin{bmatrix} 1 & 1+\varepsilon^2 \\ 1+\varepsilon^2 & 1 \end{bmatrix}.$$

Mit

$$\det(A^TA - \lambda I) = \det \begin{bmatrix} 1-\lambda & 1+\varepsilon^2 \\ 1+\varepsilon^2 & 1-\lambda \end{bmatrix} = (1-\lambda)^2 - (1+\varepsilon^2)^2$$

erhalten wir für die singulären Werte $\sigma_1 = \sqrt{2+\varepsilon^2}$ und $\sigma_2 = |\varepsilon|$. Bei Rechnung mit Maschinengenauigkeit `eps` erhalten wir statt A^TA die Matrix

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

mit singulären Werten $\tilde{\sigma}_1 = \sqrt{2}$ und $\tilde{\sigma}_2 = 0$. Damit stimmen die singulären Werte nicht bis auf Rechengenauigkeit überein.

Das *Verfahren von Golub und Reinsch* vermeidet den gerade demonstrierten Effekt, indem es erst gar nicht das Produkt A^TA bildet, sondern direkt die Matrix A bearbeitet. Im ersten Schritt

dieses Verfahrens wird mit einer Householder-Matrix $Q_1 \in \mathbb{R}^{m\times m}$ die erste Spalte von A auf ein Vielfaches des ersten Einheitsvektors gebracht,

$$A = \begin{bmatrix} * & \cdots & * \\ \vdots & & \vdots \\ \vdots & & \vdots \\ * & \cdots & * \end{bmatrix}, \quad A' = Q_1 A = \begin{bmatrix} * & \cdots & \cdots & * \\ 0 & * & \cdots & * \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 0 & * & \cdots & * \end{bmatrix}.$$

Anschließend werden mit einer $n \times n$-Householder-Matrix der Form

$$P_1 = \begin{bmatrix} 1 & 0 \\ 0 & \tilde{P}_1 \end{bmatrix}$$

die Elemente $a_{13}, \ldots, a_{1n}$ eliminiert,

$$A'' = A'P_1 = \begin{bmatrix} * & * & 0 & \cdots & 0 \\ 0 & * & \cdots & \cdots & * \\ \vdots & \vdots & & & \vdots \\ \vdots & \vdots & & & \vdots \\ 0 & * & \cdots & \cdots & * \end{bmatrix}.$$

Die erste Spalte von A' wird dabei nicht verändert. Durch Fortsetzung dieses Prozesses wird A auf Bidiagonalgestalt gebracht

$$\tilde{Q}A\tilde{P} := Q_n \ldots Q_1 A P_1 \ldots P_{n-2} = \begin{bmatrix} B \\ 0 \end{bmatrix} \tag{8.17}$$

mit

$$B = \begin{bmatrix} \alpha_1 & \beta_1 & & 0 \\ & \ddots & \ddots & \\ & & \ddots & \beta_{n-1} \\ 0 & & & \alpha_n \end{bmatrix}.$$

Wenn $B = \hat{U}\Sigma\hat{V}^T$ die Singulärwertzerlegung von B ist, so gilt

$$\tilde{Q}^T \begin{bmatrix} \hat{U} & 0 \\ 0 & I_{m-n} \end{bmatrix} \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} \hat{V}^T P^T = \tilde{Q}^T \begin{bmatrix} \hat{U}\Sigma\hat{V}^T \\ 0 \end{bmatrix} \tilde{P}^T = A.$$

Wir brauchen daher nur die Singulärwertzerlegung der Bidiagonalmatrix B zu bestimmen.

Die Matrix

$$B^T B = \begin{bmatrix} \alpha_1^2 & \alpha_1\beta_1 & & & 0 \\ \alpha_1\beta_1 & \alpha_2^2 + \beta_1^2 & \alpha_2\beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \alpha_{n-1}^2\beta_{n-2}^2 & \alpha_{n-1}\beta_{n-1} \\ 0 & & & \alpha_{n-1}\beta_{n-1} & \alpha_n^2 + \beta_{n-1}^2 \end{bmatrix}.$$

ist eine symmetrische Tridiagonalmatrix. Wir setzen sie als unzerlegbar voraus, also $\alpha_i, \beta_i \neq 0$ für $i = 1, \ldots, n-1$.

Um $B^T B$ auf Diagonalgestalt zu bringen, wird eine Variante des QR-Verfahrens verwendet, die die explizite Berechnung von $B^T B$ vermeidet. Sei k ein Shiftparameter und Q der orthogonale

Anteil der QR-Zerlegung von $B^TB - kI$. Der Algorithmus besteht aus den folgenden Schritten:

1. Bestimme eine orthogonale Matrix P_0, deren erste Spalte mit Q übereinstimmt.

2. Transformiere BP_0 auf Bidiagonalgestalt mit dem Verfahren aus (8.17).

Sei $\tilde{P} = P_0P_1 \dots P_{n-2}$, wobei $P_1, \dots, P_{n-2}$ die Matrizen aus (8.17) bezeichnen, mit denen BP_0 auf Bidiagonalgestalt gebracht wird. Dann besitzt $\tilde{P}$ dieselbe erste Spalte wie Q und

$$\tilde{B}^T\tilde{B} = \tilde{P}^TB^T(Q_{n-1} \dots Q_1)^T(Q_{n-1} \dots Q_1)B\tilde{P} = \tilde{P}^TB^TB\tilde{P}.$$

Da $\tilde{B}^T\tilde{B}$ und Q^TB^TBQ tridiagonal sind, folgt aus Lemma 8.8, dass $Q = \tilde{P}$ und $\tilde{B}^T\tilde{B} = Q^TB^TBQ$ gilt. $\tilde{B}^T\tilde{B}$ ist also die Tridiagonalmatrix, die man mit einem QR-Schritt mit Shift k ausgehend von B^TB erhält.

Die erste Spalte von $B^TB - kI$ ist gegeben durch

$$(\alpha_1^2 - k, \alpha_1\beta_1, 0, \dots, 0)^T,$$

also kann P_0 als Drehung U_{12} gewählt werden, so dass das Element $\alpha_1\beta_1$ annulliert wird. BP_0 hat dann die Gestalt

$$\begin{bmatrix} * & * & 0 & 0 & 0 & \dots \\ * & * & * & 0 & 0 & \dots \\ 0 & 0 & * & * & 0 & \dots \\ 0 & 0 & 0 & * & * & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix}$$

Wir multiplizieren diese Matrix von links mit einer Drehung U_{12}, die das Element $(2,1)$ eliminiert. Das entstehende nichttriviale Elemente an der Stelle $(1,3)$ wird anschließend mit einer Drehung U_{23} durch Multiplikation von rechts eliminiert.

Allgemein wird das Element, das die Bidiagonalgestalt stört, durch Multiplikation von links mit $U_{i\,i+1}$ von der Position $(i+1, i)$ in die Position $(i, i+2)$ gejagt, danach wird es durch Multiplikation von rechts mit $U_{i+1\,i+2}$ an die Position $(i+2, i+1)$ transportiert.

Als Shift wählt man wieder den Eigenwert von

$$\begin{bmatrix} \alpha_{n-1}^2 + \beta_{n-2}^2 & \alpha_{n-1}\beta_{n-1} \\ \alpha_{n-1}\beta_{n-1} & \alpha_n^2 + \beta_{n-1}^2 \end{bmatrix},$$

der $\alpha_n^2 + \beta_{n-1}^2$ am nächsten ist.

9 Ausgleichsrechnung

9.1 Problemstelllung Eine Reihe von Experimenten soll durchgeführt werden unter bekannten Versuchsbedingungen $z \in \mathbb{R}^m$. Es sollen Größen $x \in \mathbb{R}^n$ bestimmt werden, für die ein Gesetz

$$y = f(z, x), \quad f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R},$$

gelten soll. Führe $m \geq n$ Versuche durch,

$$y_k = f(z_k, x) =: f_k(x), \quad k = 1, \ldots, m.$$

Aufgrund von Messfehlern gilt aber nur $y_k \approx f_k(x)$,. so dass i.A. $m > n$ Versuche durchgeführt werden. Bestimme nun x aus den y_k.

Beispiel Ein chemischer Prozess wird modelliert durch

$$y'' + a_1 y' + a_0 y = 0, \quad y^{(k)}(0) = y_0^{(k)} \text{ mit } k = 0, 1 \text{ bekannt.}$$

Mit dem Ansatz $y(z) = e^{\omega z}$ folgt dann

$$\omega^2 + a_1 \omega + a_0 = 0.$$

Aus dem nichtoszillatorischen Verhalten der Lösung weiß man, dass diese Gleichung zwei reelle Nullstellen α, β besitzt. Die allgemeine Lösung der Differentialgleichung ist dann

$$y(z) = c_1 e^{\alpha z} + c_2 e^{\beta z}.$$

Gesucht sind nun die *Reaktionszeiten* α und β.

Aus $y(0) = y_0$ erhalten wir

$$c_1 + c_2 = y_0$$

und aus $y'(0) = y_0'$

$$\alpha c_1 + \beta c_2 = y_0',$$

was nicht ausgenutzt werden kann, weil die Unbekannten α, β darin vorkommen. Daher ist

$$y(z) = c_1 e^{\alpha z} + (y_0 - c_1) e^{\beta z} = f(z, c_1, \alpha, \beta),$$

wobei c_1, α, β gesucht sind. Führe also für Zeiten $z_1, \ldots, z_m$, $m \geq 3$, Messungen durch,

$$y_k = f(z_k, c_1, \alpha, \beta).$$

Dies ist ein schwieriges und häufiges Problem, nämlich die *Exponentialapproximation.*

Kehren wir zum allgemeinen Problem zurück, ein $x \in \mathbb{R}^n$ zu finden mit $y_k \approx f_k(x)$, $k = 1, \ldots, m$. Die mathematisch einfachste Möglichkeit besteht nun darin, ein Minimum der Funktion

$$F(x) = \sum_{k=1}^{m} |y_k - f_k(x)|^2$$

über $x \in \mathbb{R}^n$ zu finden (=Methode der kleinsten Quadrate). Schwieriger und daher von uns nicht behandelt, ist es, die Funktion

$$F(x) = \max_{k=1,\ldots,m} |y_k - f_k(x)|$$

zu minimieren (=diskretes Tschebyscheff-Problem).

Ein wichtiger Spezialfall ist ein lineares $f(x) = Ax$ mit einer $(m \times n)$-Matrix A. In der Methode der kleinsten Quadrate ist dann das Funktional

$$F(x) = |Ax - y|^2, \quad y \in \mathbb{R}^m, \tag{9.1}$$

zu minimieren, was man als *lineares Ausgleichsproblem* bezeichnet. Die notwenige Bedingung für eine Lösung x dieses Problems ist

$$\operatorname{grad}_x |y - Ax|^2 = 2A^T Ax - 2A^T y = 0.$$

Das System $A^T Ax = A^T y$ heißt *Normalgleichungen* zu (9.1).

9.2 Das lineare Ausgleichsproblem Satz Sei $A \in \mathbb{R}^{m\times n}$. Die Probleme

$$|Ax - y|^2 \;\to\; \text{Min in } x \in \mathbb{R}^n, \quad A^TAx = A^Ty,$$

sind äquivalent: Jede Lösung des einen Problems ist auch eine Lösung des anderen Problems. Für je zwei Lösungen x_1, x_2 gilt $Ax_1 = Ax_2$ und für das Residuuum $r = y - Ax$ einer jeden Lösung x haben wir $A^Tr = 0$.

Beweis: Sei $L = \text{Bild}\,(A) \subset \mathbb{R}^m$. Dann ist $L \oplus L^\perp = \mathbb{R}^m$ und für $y \in \mathbb{R}^m$ gilt dann die eindeutige Zerlegung

$$y = s + r, \quad s \in L,\ r \in L^\perp.$$

Da $s \in \text{Bild}\,(A)$ gibt es ein $x_0 \in \mathbb{R}^n$ mit $Ax_0 = s$. Aus

$$r \perp Ax \quad \forall x \in \mathbb{R}^n$$

folgt

$$0 = (r, Ax_0) = (A^Tr, x_0) \;\Rightarrow\; A^Tr = 0.$$

Daher

$$A^Ty = A^Ts = A^TAx_0,$$

womit x_0 die Normalgleichungen erfüllt.

Für eine Lösung x_0 der Normalgleichungen gilt mit der gleichen Zerlegung $y = r + s$ wie im vorigen Beweisteil

$$0 = (A^T(Ax_0 - y), z) = (Ax_0 - y, Az) \;\Rightarrow\; Ax_0 - y \perp L \;\Rightarrow\; Ax_0 = s \text{ wegen } Ax_0 \in L.$$

Für $x \in \mathbb{R}^n$ setze

$$z = Ax - Ax_0, \quad r = y - Ax_0.$$

Mit $z \in L$ und $r \perp L$ folgt $(r, z) = 0$ und daher

$$|y - Ax|^2 = |r - z|^2 = |r|^2 + |z|^2 \geq |r|^2 = |y - Ax_0|^2.$$

Damit ist x_0 ein Minimum von F. □

Wenn $m \geq n$ und $\text{rang}\,A = n$, so ist A^TA symmetrisch positiv definit und die Gleichung

$$A^TAx = A^Ty$$

kann mit dem Cholesky-Verfahren gelöst werden. Da A^TA aber oft schlecht konditioniert ist, verwendet man besser das im nächsten Abschnitt dargestellte Verfahren.

9.3 Orthogonalisierungsverfahren Wir wollen

$$|y - Ax|^2 \to \text{Min} \;\text{ mit } A \in \mathbb{R}^{m\times n},\ m \geq n,$$

bestimmen. Dazu eliminieren wir die erste Spalte von A mit einer $(m \times m)$-Householder-Matrix P_1 und erhalten mit $A^{(0)} = A$

$$A^{(1)} = P_1A^{(0)} = \begin{bmatrix} * & * & \cdots & * \\ 0 & * & \cdots & * \\ \vdots & \vdots & & \vdots \\ 0 & * & \cdots & * \end{bmatrix}.$$

Der Householder-Algorithmus verläuft daher genauso wie bei quadratischen Matrizen. Nach n Schritten gilt mit einer unitären Matrix $P \in \mathbb{R}^{m\times m}$

$$A^{(n)} = PA = \begin{bmatrix} R \\ 0 \end{bmatrix},$$

wobei $R \in \mathbb{R}^{n\times n}$ eine rechte obere Dreiecksmatrix ist. Da $\operatorname{rang} A = n$, besitzt R nichtverschwindende Elemente auf der Hauptdiagonalen. Für die rechte Seite gilt analog

$$h = Py = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix}$$

mit $h_1 \in \mathbb{R}^n$ und $h_2 \in \mathbb{R}^{m-n}$. Da P unitär folgt $|Pz| = |z|$ und daher

$$|y - Ax|^2 = |P(y - Ax)|^2 = \left| \begin{pmatrix} h_1 - Rx \\ h_2 \end{pmatrix} \right|^2 = |h_1 - Rx|^2 + |h_2|^2.$$

Die Lösung von $Rx = h_1$ ist damit auch die eindeutige Lösung des Minimierungsproblems $F(x) \to$ Min.

9.4 Die Pseudoinverse einer Matrix Zu $A \in \mathbb{R}^{m\times n}$ wollen wir eine *Pseudoinverse* $A^+ \in \mathbb{R}^{n\times m}$ definieren, deren Eigenschaften wir an Hand des linearen Gleichungssystems $Ax = b$ für $b \in \mathbb{R}^m$ festlegen wollen. Da wir keinerlei Voraussetzungen an A stellen, kann das Gleichungssystem unlösbar oder mehrdeutig lösbar sein. Die „Lösung“ $x = A^+b$ wird nach folgenden Kriterien festgelegt: Zuerst wird y aus dem Bild von A bestimmt, das den Fehler $|y - b|$ minimiert. Anschließend wird unter allen x mit $Ax = y$ dasjenige ausgewählt, dessen Norm $|x|$ minimal wird.

Wie immer seien $\operatorname{Bild}(A) \subset \mathbb{C}^m$ das Bild von A und $\operatorname{Kern}(A) \subset \mathbb{C}^n$ der Nullraum von A. Wir definieren die orthogonalen Projektionen

$$P \in \mathbb{C}^{n\times n} : \ \mathbb{C}^n \to \operatorname{Kern}(A)^\perp,$$
$$\overline{P} \in \mathbb{C}^{m\times m} : \mathbb{C}^m \to \operatorname{Bild}(A).$$

Zu $x \in \mathbb{C}^n$ gibt es eindeutige $x_0 \in \operatorname{Kern}(A)$ und $x_\perp \in \operatorname{Kern}(A)^\perp$ mit $x = x_0 + x_\perp$. Dann ist $Px = x_\perp$. Offenbar gilt $P^2 = P$ und wegen

$$(Px, y) = (x_\perp, y) = (x_\perp, y_\perp) = (x_\perp, Py) = (x, Py) = (P^H x, y)$$

ist $P = P^H$. Die Projektion $\overline{P}$ besitzt die gleichen Eigenschaften.

Zu jedem $y \in \operatorname{Bild}(A)$ gibt es genau ein $x \in \operatorname{Kern}(A)^\perp$ mit $Ax = y$. Demnach gibt es eine lineare Abbildung $f : \operatorname{Bild}(A) \to \mathbb{C}^n$ mit

$$Af(y) = y, \quad f(y) \in \operatorname{Kern}(A)^\perp.$$

Ist x eine Lösung von $Ax = y$, so ist mit $x = x_0 + x_\perp$, $x_0 \in \operatorname{Kern}(A)$, $x_\perp \in \operatorname{Kern}(A)^\perp$,

$$A(I - P)x = A(x_0 + x_\perp - x_\perp) = 0,$$

also

$$y = A(Px + (I - P)x) = APx = Ax_1.$$

Es gilt also $f(y) = Px$, wobei x eine beliebige Lösung von $Ax = y$ ist.

Es ist

$$f \circ \overline{P} : \mathbb{C}^m \to \mathbb{C}^n.$$

Die Darstellung von $f \circ \overline{P}$ durch eine $(n \times m)$-Matrix ist dann die gesuchte Pseudo-Inverse A^+. Denn es wird das nächstgelegene $\tilde{y} \in \operatorname{Bild}(A)$ zu y bestimmt und anschließend die betragsmäßig kleinste Lösung von $Ax = \tilde{y}$.

Satz Die Pseudoinverse A^+ hat die folgenden Eigenschaften:

(a) $A^+A = P, \ AA^+ = \overline{P}$,

(b) (i) $A^+A = (A^+A)^H$, (ii) $AA^+ = (AA^+)^H$, (iii) $AA^+A = A$, (iv) $A^+AA^+ = A^+$.

(c) Wenn $Z \in \mathbb{C}^{n\times m}$ die Eigenschaften (b) (i)-(iv) besitzt (an Stelle von A^+), so ist $Z = A^+$.

(d) $A^{++} = A$, $(A^+)^H = (A^H)^+$.

Beweis: (a) Es gilt

$$A^+Ax = f(\overline{P}(Ax)) = f(Ax) = Px$$

sowie

$$AA^+y = A(f(\overline{P}y)) = \overline{P}y,$$

weil $A($$P$

(b) Wegen $A^+A = P$, $AA^+ = \overline{P}$ und $P = P^H$, $P^H = P$ folgen (i) und (ii).

(iii) erhält man aus

$$(AA^+)Ax = \overline{P}Ax = Ax$$

und (iv) aus

$$A^+(AA^+)y = A^+\overline{P}y = A^+y.$$

(c) Es gilt

$$\begin{aligned}
Z &\overset{(iii)\text{ für }Z}{=} ZAZ \overset{(iii)\text{ für }A}{=} ZAA^+Az \\
&\overset{(iv)\text{ für }A}{=} Z(AA^+A)A^+(AA^+A)Z = (ZA)(A^+A)A^+(AA^+)(AZ) \\
&\overset{(i),(ii)\text{ für }A,Z}{=} (A^HZ^H)(A^H(A^+)^H)A^+((A^+)^HA^H)(Z^HA^H) \\
&= (A^HZ^HA^H)(A^+)^HA^+(A^+)^H(A^HZ^HA^H) \\
&\overset{(iii)\text{ für }Z}{=} A^H(A^+)^HA^+(A^+)^HA^H \overset{(i)}{=} A^+AA^+AA^+ \overset{(iv)}{=} A^+AA^+ = A.
\end{aligned}$$

(d) Setze in (c) A^+ statt A. Die Matrix $Z = A$ erfüllt dann (i)-(iv). Genauso argumentiert man bei $(A^+)^H = (A^H)^+$. □

Korollar Das Ausgleichsproblem

$$|Ax - y| \;\to\; \text{Min}$$

wird durch $x = A^+y$ gelöst. Unter allen Lösungen besitzt A^+y die kleinste Norm.

Beweis: Es gilt

$$\min|Ax - y|^2 = \min_{\tilde{y}\in \text{Bild}\,(A)} |\tilde{y} - y|^2,$$

also $\tilde{y} = \overline{P}y$. Wenn $Ax = \tilde{y}$, $x \perp \text{Kern}\,(A)$, so folgt für beliebiges $\tilde{x} \in \text{Kern}\,(A)$

$$A(x + \tilde{x}) = \tilde{y}, \quad |x + \tilde{x}|^2 = |x|^2 + |\tilde{x}|^2.$$

Also besitzt x unter allen Lösungen von $Ax = \tilde{y}$ die kleinste Norm. □

A^+ kann mit der Singulärwertzerlegung bestimmt werden. Wenn

$$A = U\Sigma V^H$$

mit $U \in \mathbb{C}^{m\times m}$, $V \in \mathbb{C}^{n\times n}$ unitär und $\Sigma \in \mathbb{C}^{m\times n}$ mit

$$\Sigma = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix}, \quad D = \text{diag}\,(\sigma_1, \ldots, \sigma_r),\ \sigma_r > 0,$$

so setze

$$\Sigma^+ = \begin{bmatrix} D^{-1} & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{C}^{n\times m}, \quad A^+ = V\Sigma^+ U^H \in \mathbb{C}^{n\times m}.$$

Dann gilt

$$A^+A = V\Sigma^+U^HU\Sigma V^H = V \underbrace{m \times m} \begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix} V^H,$$

also $(A^+A)^H = A^+A$. Analog zeigt man $AA^+ = (AA^+)^H$ sowie

$$AA^+A = U\Sigma V^H V \begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix} V^H = A.$$

Damit sind die Bedingungen (b) (i)-(iv) des vorletzten Satzes alle erfüllt.

9.5 Das nichtlineare Ausgleichsproblem Sei $m \geq n$ und $f : \mathbb{R}^n \to \mathbb{R}^m$ stetig differenzierbar. Zu $y \in \mathbb{R}^n$ soll ein $x \in \mathbb{R}^n$ bestimmt werden mit

$$|y - f(x)|^2 \to \text{Min}.$$

Wir leiten ein Verfahren für dieses Problem durch Linearisierung her. Sei x eine Näherung. Aus der Differenzierbarkeit von f folgt

$$f(\overline{x}) = f(x) + f_x(x)(\overline{x} - x) + o(|\overline{x} - x|).$$

Wir bestimmen daher die Lösung von

$$\min_{z\in\mathbb{R}^n} |y - f(x) - f_x(x)(z - x)|^2. \tag{9.2}$$

Dies ist ein *lineares* Ausgleichsproblem, das als Analogon zum Newton-Verfahren zur Lösung nichtlinearer Gleichungen angesehen werden kann. Ähnlich wie beim Newton-Verfahren können wir zeigen, dass die Suchrichtung eine Abstiegsrichtung für das zu minimierende Funktional ist:

Lemma Sei $\operatorname{rang} f_x(x) = n$, $z = \overline{x}$ sei die eindeutig bestimmte Lösung von (9.2) und $s = \overline{x} - x$ die zugehörige Richtung. Wenn $s \neq 0$, so gibt es ein $\lambda_0 > 0$, so dass

$$\phi(\tau) = |y - f(x + \tau s)|^2$$

in $[0, \lambda_0]$ streng monoton fallend ist.

Beweis: Fast genauso wie im analogen Satz für das gewöhnliche Newton-Verfahren erhalten wir

$$\begin{aligned} \phi'(0) &= \frac{d}{d\tau}(y - f(x + \tau s)^T(y - f(x + \tau s)\big|_{\tau=0} \\ &= -(f_x(x)s)^T(y - f(x)) - (y - f(x)^T(f_x(x)s) \\ &= -2(f_x(x)s)^T(y - f(x)) = -2(f_x(x)s)^T r(x), \quad r(x) = y - f(x). \end{aligned}$$

Nach (9.2) ist $s = \overline{x} - x$ die Lösung von

$$f_x(x)^T f_x(x)s = f_x(x)^T r(x),$$

also

$$\begin{aligned} \phi'(0) &= -2s^T f_x(x)^T r(x) = -2s^T f_x(x) f_x(x)s \\ &= -2|f_x(x)s|^2 < 0 \quad \text{falls } \operatorname{rang} f_x(x) = n \text{ und } s \neq 0. \end{aligned}$$

□

Wir erhalten damit den *Gauß-Newton Algorithmus*:

0) Sei $x^{(0)} \in \mathbb{R}^n$ ein Startvektor. Sei $x^{(i)}$ bereits definiert.

1) Bestimme $s^{(i)}$ durch

$$\min_{s\in\mathbb{R}^n} |r(x^{(i)}) - f_x(x^{(i)})s|^2, \quad r(x^{(i)}) = y - f(x^{(i)}).$$

2) Sei

$$\phi(\tau) = |y - f(x^{(i)} + \tau s^{(i)})|^2.$$

Bestimme die kleinste Zahl $k \in \mathbb{N}_0$ mit $\phi(2^{-k}) < \phi(0)$ oder alternativ $\phi(\tau) \to \text{Min}$ für $\tau > 0$ durch ein Line-Search-Verfahren. Setze $x^{(i+1)} = x^{(i)} + 2^{-k}s^{(i)}$ oder alternativ $= x^{(i)} + \tau s^{(i)}$.

Beispiel Für Probleme der Art

$$y_i = \sum_{j=1}^{n} e^{\alpha_j x_i}, \quad i = 1, \ldots, m$$

sind die Funktionalmatrizen meist sehr schlecht konditioniert. Für $m = n = 2$ erhalten wir beispielsweise

$$f_\alpha(\alpha) = \begin{bmatrix} x_1 e^{\alpha_1 x_1} & x_1 e^{\alpha_2 x_1} \\ x_2 e^{\alpha_1 x_2} & x_2 e^{\alpha_2 x_2} \end{bmatrix}$$

mit

$$\det f_\alpha(\alpha) = x_1 x_2 \big(e^{\alpha_1 x_1 + \alpha_2 x_2} - e^{\alpha_1 x_2 + \alpha_2 x_1}\big) \approx 0 \quad \text{falls } \alpha_1 \approx \alpha_2 \text{ oder } x_1 \approx x_2.$$

Die Normalgleichungen dürfen daher nicht mit einem Verfahren für $f_\alpha^T f_\alpha$ gelöst werden.