GOETHE UNIVERSITÄT FRANKFURT AM MAIN

NUMERISCHE MATHEMATIK

Prof. Dr. Bastian von Harrach

Goethe-Universität Frankfurt am Main
Institut für Mathematik

Wintersemester 2023/2024

https://numerical.solutions

Finale Version des WiSe2324 (kleinere Fehler werden aber weiterhin korrigiert).
Letzte Aktualisierung: 2. Februar 2024

Inhaltsverzeichnis

Kapitel 1

Numerische Quadratur

1.1 Einleitung

In der *numerischen Mathematik* beschäftigen wir uns mit der Frage, wie sich konkrete mathematische Probleme praktisch lösen lassen. Dabei ist meist:

- das Problem durch eine praktische Anwendung motiviert,
- der Einsatz von Computern notwendig/sinnvoll, und
- die Lösung nur näherungsweise möglich.

Unser Ziel ist üblicherweise ein numerischer Algorithmus, d.h. ein (rechnergestütztes) Verfahren, das eine gegen die Lösung konvergierende Folge von Approximationen erzeugt.

Beispiel: Berechnung des Integrals (*numerische Quadratur*)

$$I[f] = \int_a^b f(x)\,\mathrm{d}x$$

einer gegebenen integrierbaren (z.B. stetigen) Funktion $f : [a,b] \to \mathbb{R}$ und Grenzen $a,b \in \mathbb{R}$, $a < b$.

Im Allgemeinen lassen sich Integrale nicht in geschlossener Form lösen, z.B. spielen in der Stochastik Integrale der Form

$$\int_a^b e^{-x^2}\,\mathrm{d}x$$

eine wesentliche Rolle. Für die *Gauß-Funktion* $f(x) = e^{-x^2}$ ist jedoch keine (elementare) Stammfunktion bekannt. Darüber hinaus ist in vielen Anwendungen die zu integrierende Funktion f gar nicht explizit bekannt, sondern es existiert lediglich ein Algorithmus mit dem $f(x)$ für ein gegebenes x ausgerechnet werden kann.

Naive Lösungsansätze:

- Zeichnen der Funktion auf ein Blatt Papier, ausschneiden und abwiegen:
  - einfach, keine Programmierkenntisse erforderlich,
  - hoher manueller Arbeitsaufwand, schwer automatisierbar,
  - Genauigkeit nur begrenzt steigerbar.
- Approximation von f durch einfach integrierbare (z.B. stückweise konstante oder stückweise lineare) Funktionen:
  - Computer-implementierbar,
  - prinzipiell beliebig hohe Genauigkeit möglich.

1.2 Erste Quadraturverfahren

In diesem Kapitel seien stets $a, b \in \mathbb{R}$, $a < b$ und $f : [a,b] \to \mathbb{R}$ integrierbar.

1.2.1 Mittelpunkts- und Trapezformel

Erste einfache Quadraturverfahren erhalten wir durch Approximation der Funktion durch eine konstante oder lineare Funktion.

- **Mittelpunktsformel:**

$$\int_a^b f(x)\,\mathrm{d}x \approx (b-a) f\left(\frac{a+b}{2}\right) =: M[f]$$

- **Trapezformel:**

$$\int_a^b f(x)\,\mathrm{d}x \approx \frac{b-a}{2} f(a) + \frac{b-a}{2} f(b) =: T[f]$$

Indem wir $[a,b]$ in n gleich große Teilintervalle zerlegen,

$$x_i := a+(i-1)h, \quad i=1,\dots,n+1, \quad h := \frac{b-a}{n},$$

und auf jedes Teilintervall die Trapezformel anwenden, erhalten wir die folgende zusammengesetzte Formel.

- **Zusammengesetzte Trapezformel:**

$$\begin{aligned}\int_a^b f(x)\,\mathrm{d}x &= \sum_{i=1}^{n}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \\ &\approx \sum_{i=1}^{n}\frac{x_{i+1}-x_i}{2}\left(f(x_i)+f(x_{i+1})\right) \\ &= \frac{h}{2}f(a)+h\sum_{i=2}^{n}f(x_i)+\frac{h}{2}f(b) =: T_n[f].\end{aligned}$$

Wir bezeichnen mit $C^k([a,b])$, $k\in\mathbb{N}_0$, die Menge aller auch über die Randpunkte hinweg k-mal stetig differenzierbaren Funktionen, also

$$C^k([a,b]) := \{\tilde{f}|_{[a,b]} \,:\, \tilde{f}\in C^k(\mathbb{R})\}.$$

Außerdem sei für $f\in C([a,b]) := C^0([a,b])$

$$\|f\|_{[a,b]} := \max_{a\le x\le b}|f(x)|$$

die *Maximumsnorm* über dem Interval $[a,b]$.[1]

Satz 1.1 (Fehler der Trapezformel)
Sei $f\in C^2([a,b])$. Dann gilt

$$|I[f]-T_n[f]| \le \frac{b-a}{12}\left\|f''\right\|_{[a,b]}h^2.$$

Beweis: Wir betrachten zuerst $n=1$. Es ist

$$\begin{aligned}I[f]-T_1[f] &= \int_a^b f(x)\,\mathrm{d}x-\frac{b-a}{2}(f(a)+f(b)) \\ &= \int_a^b\left(f(a)+\int_a^x f'(t)\,\mathrm{d}t\right)\mathrm{d}x-\frac{b-a}{2}(f(a)+f(b)) \\ &= \frac{b-a}{2}(f(a)-f(b))+\int_a^b\int_a^x f'(t)\,\mathrm{d}t\,\mathrm{d}x.\end{aligned}$$

[1] In der Analysis zeigt man, dass dies tatsächlich eine Norm auf dem Vektorraum $C([a,b])$ ist. $C([a,b])$ ist bzgl. dieser Norm vollständig.

Für $x,t \in [a,b]$ ist $a \le t \le x$ äquivalent zu $t \le x \le b$ und damit

$$\chi_{[a,x]}(t) = \chi_{[t,b]}(x) \qquad \text{für alle } x,t \in [a,b].$$

Mit

$$\begin{aligned}\int_a^b \int_a^x f'(t)\,\mathrm{d}t\,\mathrm{d}x &= \int_a^b \int_a^b \chi_{[a,x]}(t) f'(t)\,\mathrm{d}t\,\mathrm{d}x = \int_a^b \int_a^b \chi_{[t,b]}(x) f'(t)\,\mathrm{d}t\,\mathrm{d}x \\ &= \int_a^b \int_a^b \chi_{[t,b]}(x)\,\mathrm{d}x f'(t)\,\mathrm{d}t = \int_a^b (b-t) f'(t)\,\mathrm{d}t\end{aligned}$$

und $f(a) - f(b) = -\int_a^b f'(t)\,\mathrm{d}t$ erhalten wir

$$I[f] - T_1[f] = \int_a^b f'(t)\left(b - t - \frac{b-a}{2}\right)\mathrm{d}t = \int_a^b f'(t)\left(\frac{b+a}{2} - t\right)\mathrm{d}t.$$

Nun verwenden wir, dass $-\frac{1}{2}(t-a)(t-b)$ eine Stammfunktion von $\frac{b+a}{2} - t$ ist, und erhalten durch partielle Integration

$$I[f] - T_1[f] = \frac{1}{2}\int_a^b f''(t)(t-a)(t-b)\,\mathrm{d}t$$

und damit

$$\begin{aligned}|I[f] - T_1[f]| &\le \frac{1}{2}\left\|f''\right\|_{[a,b]} \int_a^b |(t-a)(t-b)|\,\mathrm{d}t \\ &= \frac{1}{2}\left\|f''\right\|_{[a,b]} \int_a^b (t-a)(b-t)\,\mathrm{d}t \\ &= \frac{1}{12}\left\|f''\right\|_{[a,b]} \left(b^3 - 3ab^2 + 3a^2 b - a^3\right) = \frac{1}{12}\left\|f''\right\|_{[a,b]} (b-a)^3.\end{aligned}$$

Wir verwenden dieses Ergebnis jetzt für jedes Teilintervall $[x_i, x_{i+1}]$, $i = 1,\ldots,n$ und erhalten

$$\begin{aligned}|I[f] - T_n[f]| &\le \sum_{i=1}^n \frac{1}{12}\left\|f''\right\|_{[x_i,x_{i+1}]} h^3 \le \frac{1}{12}\left\|f''\right\|_{[a,b]} h^3 n \\ &= \frac{b-a}{12}\left\|f''\right\|_{[a,b]} h^2. \qquad \square\end{aligned}$$

1.2.2 Allgemeine Quadraturverfahren

Satz 1.1 zeigt, dass der mit dem Trapezverfahren ermittelte Wert (im Grenzwert der Verwendung unendlich vieler Teilintervalle) gegen den wahren Wert des Integrals konvergiert. Darüberhinaus ermöglicht es Satz 1.1, die Konvergenzgeschwindigkeit abzuschätzen (zumindest asymptotisch, d.h. bis auf die oft unbekannte multiplikative Konstante $\|f''\|_{[a,b]}$): Eine Verdopplung der Anzahl der Teilintervalle bewirkt eine Viertelung des Fehlers.

Um noch schnellere Verfahren zu entwickeln, betrachten wir einen allgemeinen Ansatz für eine *Quadraturformel* Q. Es liegt nahe, dass Q nur endlich viele Auswertungen der Funktion $f(x_i)$ benutzen sollte (die Auswertungspunkte $x_i \in [a,b]$, $i = 1,\dots,m$, nennen wir *Knoten*). Außerdem sollte Q linear von den Funktionsauswertungen $f(x_i)$, $i = 1,\dots,m$, abhängen. Wir machen daher den Ansatz

$$I[f] = \int_a^b f(x)\,\mathrm{d}x \approx \sum_{i=1}^{m} w_i f(x_i) =: Q[f]$$

mit *Gewichten* $w_i \in \mathbb{R}$.

Wie bei der Trapezformel erhalten wir aus einer Quadraturformel ein *(zusammengesetztes) Quadraturverfahren* Q_n, indem wir $[a,b]$ in n-Teilintervalle der Länge $h = \frac{b-a}{n}$ unterteilen, in denen wir jeweils die Quadraturformel anwenden.

Meist können wir ohne zusätzlichen Aufwand das noch etwas allgemeinere Problem behandeln, ein Quadraturverfahren für Integrale der Form

$$I[f;w] = \int_a^b f(x)w(x)\,\mathrm{d}x$$

mit einer festen *Gewichtsfunktion* $w(x)$ zu entwickeln. Im gesamten Kapitel sei dabei stets $w \in C([a,b])$ und $w(x) > 0$ für $x \in (a,b)$.[2]

Für die Gewichtsfunktion $w = 1$ nähert die Trapezformel die zu integrierende Funktion durch eine lineare Funktion (also ein Polynom vom Höchstgrad 1) an und verwendet das Integral dieses Polynoms als Näherung an $I[f]$. Ist f selbst ein Polynom vom Grad 1, so liefert dieses Vorgehen bereits den exakten Integralwert. Dies motiviert:

Definition 1.2
Eine Quadraturformel Q für das Integral $I[\cdot;w]$ hat Exaktheitsgrad *q, falls sie Polynome vom Höchstgrad q exakt integriert, also*

$$Q[p] = I[p;w] \quad \forall p \in \Pi_q,$$

[2] Achtung: Zwischen den Gewichten w_i der Quadraturformel und der Gewichtsfunktion $w(x)$ besteht kein unmittelbarer Zusammenhang. (Aber natürlich wird in die Bestimmung optimaler Gewichte die Aufgabenstellung und damit a, b und $w(x)$ einfließen.)

wobei Π_q der Raum der Polynome (mit reellen Koeffizienten) vom Höchstgrad q ist.

Für die Gewichtsfunktion $w = 1$ besitzt die Trapezformel also Exaktheitsgrad $q = 1$.

1.3 Polynominterpolation

Um Verfahren höheren Exaktheitsgrades zu konstruieren werden wir die zu integrierende Funktion durch ein Polynom höheren Grades approximieren. Dazu betrachten wir die folgende *Interpolationsaufgabe*.

Gegeben seien m Knoten x_i und Werte y_i, $i = 1, \ldots, m$. Gibt es ein Polynom p, das diese Werte interpoliert, d.h.

$$p(x_i) = y_i \quad \forall i \in \{1, \ldots, m\}?$$

Bemerkung 1.3 (Naheliegender Ansatz)
Wir schreiben das zu bestimmende Polynom als

$$p(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \ldots + \alpha_k x^k \in \Pi_k$$

und versuchen die Koeffizienten $\alpha_0, \ldots, \alpha_k \in \mathbb{R}$ so zu bestimmen, dass die Interpolationsbedingungen erfüllt sind.

Wir erhalten m lineare Gleichungen

$$\alpha_0 + \alpha_1 x_i + \alpha_2 x_i^2 + \ldots + \alpha_k x_i^k = p(x_i) = y_i, \quad i = 1, \ldots, m,$$

für die $k+1$ Unbekannten $\alpha_0, \ldots, \alpha_k \in \mathbb{R}$. Im Allgemeinen können wir Lösbarkeit ab $k+1 \geq m$ erwarten. Wir versuchen also $k := m-1$ und schreiben das lineare Gleichungssystem in Matrix-Vektor-Form

$$\begin{pmatrix} x_1^0 & x_1^1 & \ldots & x_1^{m-1} \\ x_2^0 & x_2^1 & \ldots & x_2^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_m^0 & x_m^1 & \ldots & x_m^{m-1} \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{m-1} \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}.$$

Diese Matrix ist aus der Linearen Algebra bekannt als Vandermonde-Matrix. *Mit vollständiger Induktion lässt sich zeigen, dass ihre Determinante gegeben ist durch*

$$\prod_{1 \leq j < l \leq m} (x_l - x_j).$$

Sind die Knoten paarweise verschieden, dann ist die Matrix invertierbar. Das Gleichungssystem für die Koeffizienten von p besitzt dann also genau eine Lösung, d.h. es existiert genau ein Interpolationspolynom $p \in \Pi_{m-1}$ für m vorgegebene (paarweise verschiedene) Interpolationswerte.

Der naheliegende Ansatz führt also zum Erfolg. Es gibt aber einen noch einfacheren und expliziteren Weg, das Interpolationspolynom zu konstruieren, wie wir im folgenden zeigen.

Definition 1.4
Zu einem Gitter *aus m paarweise verschiedenen, aufsteigend angeordneten Knoten*

$$\{x_1, x_2, \ldots, x_m\} \subset \mathbb{R}, \qquad x_1 < x_2 < \ldots < x_m,$$

definieren wir das Knotenpolynom

$$\omega(x) = \prod_{i=1}^{m} (x - x_i) \in \Pi_m$$

und die Lagrange-Grundpolynome *($i = 1, \ldots, m$)*

$$l_i(x) = \prod_{\substack{j=1 \\ j \neq i}}^{m} \frac{x - x_j}{x_i - x_j} \in \Pi_{m-1}.$$

Offenbar gilt für alle $i, j = 1, \ldots, m$

$$l_i(x_j) = \delta_{ij} := \begin{cases} 1 & \text{für } i = j, \\ 0 & \text{für } i \neq j, \end{cases} \tag{1.1}$$

und man rechnet leicht nach, dass

$$l_i(x) = \frac{\omega(x)}{(x - x_i)\omega'(x_i)}.$$

Satz 1.5 (Interpolationspolynom)
Zu einem Gitter

$$\{x_1, x_2, \ldots, x_m\} \subset \mathbb{R}, \qquad x_1 < x_2 < \ldots < x_m,$$

und Werten $y_i \in \mathbb{R}$, $i = 1, \ldots, m$ existiert genau ein Interpolationspolynom vom Höchstgrad $m - 1$, d.h. genau ein

$$p \in \Pi_{m-1} \quad \text{mit} \quad p(x_i) = y_i \quad \text{für alle} \quad i \in \{1, \ldots, m\},$$

nämlich

$$p(x) = \sum_{i=1}^{m} y_i l_i(x).$$

Beweis: Aus (1.1) folgt sofort, dass

$$p(x) = \sum_{i=1}^{m} y_i l_i(x) \in \Pi_{m-1}$$

die Interpolationsaufgabe löst.

Um die Eindeutigkeit zu zeigen, seien $p, q \in \Pi_{m-1}$ zwei Interpolationspolynome. Dann besitzt die Differenz $p - q \in \Pi_{m-1}$ m verschiedene Nullstellen, $p - q$ muss also das Nullpolynom sein. □

Bemerkung 1.6

(a) Jedes Polynom $p \in \Pi_{m-1}$ interpoliert seine eigenen Funktionsauswertungen $y_i = p(x_i)$. Aus Satz 1.5 folgt also

$$p(x) = \sum_{i=1}^{m} p(x_i) l_i(x) \quad \forall p \in \Pi_{m-1}. \tag{1.2}$$

(b) Der Polynomraum Π_{m-1} bildet offensichtlich einen Vektorraum der Dimension m. Eine Basis bilden z.B. die Monome *$(x^0, x^1, x^2, \ldots, x^{m-1})$.*

Satz 1.5 zeigt, dass auch die Lagrange-Grundpolynome $(l_1(x), \ldots, l_m(x))$ eine Basis des Π_{m-1} bilden. (Die Erzeugendeneigenschaft folgt aus (1.2) und die lineare Unabhängigkeit aus (1.1).)

Satz 1.7 (Interpolationsfehler)
Sei $f \in C^m([a,b])$ und $p \in \Pi_{m-1}$ das Interpolationspolynom zu m paarweise verschiedenen Knoten

$$x_1, \ldots, x_m \in [a,b], \qquad x_1 < x_2 < \ldots < x_m,$$

und Werten $y_i = f(x_i)$, $i = 1, \ldots, m$.

Dann gibt es zu jedem $x \in [a,b]$ ein $\xi \in [\min\{x, x_1\}, \max\{x, x_m\}] \subseteq [a,b]$ mit

$$f(x) - p(x) = \frac{f^{(m)}(\xi)}{m!} \omega(x).$$

Beweis: Für $x = x_i$ ist die Behauptung offenbar richtig. Sei also $x \in [a,b]$ mit $x \notin \{x_1, \ldots, x_m\}$.

Betrachte (zu diesem festen x) die Funktion

$$h(t) := f(t) - p(t) - \frac{\omega(t)}{\omega(x)} (f(x) - p(x)), \quad t \in \mathbb{R}.$$

Dann ist $h \in C^m([a,b])$ und h besitzt (mindestens) $m+1$ paarweise verschiedene Nullstellen, nämlich $t = x_i$, $i = 1,\dots,m$, und $t = x$. Das kleinste Intervall, das alle diese Nullstellen enthält, ist gegeben durch

$$I := [\min\{x, x_1\}, \max\{x, x_m\}].$$

Zwischen je zwei benachbarten Nullstellen, also an (mindestens) m verschiedenen Werten im Intervall I, liegt nach dem Satz von Rolle eine Nullstelle der Ableitung $h'(t)$. Zwischen je zwei benachbarten Nullstellen der Ableitung, also an (mindestens) $m-1$ verschiedenen Werten in I, liegt wiederum nach dem Satz von Rolle eine Nullstelle der zweiten Ableitung h''. Wir fahren so fort und erhalten $m-2$ verschiedene Nullstellen von h''', $m-3$ verschiedene Nullstellen von $h^{(4)}$, usw., und schließlich eine Nullstelle $\xi \in I$ von $h^{(m)}$.

Wegen $p \in \Pi_{m-1}$ ist $p^{(m)} = 0$. Außerdem ist

$$\omega(t) = \prod_{i=1}^{m} (t - x_i) = t^m + q(t), \qquad \text{mit } q \in \Pi_{m-1}$$

und damit $\omega^{(m)}(t) = m!$. Es folgt, dass

$$0 = h^{(m)}(\xi) = f^{(m)}(\xi) - \frac{m!}{\omega(x)}(f(x) - p(x)). \qquad \square$$

Bemerkung 1.8

(a) Aus Satz 1.7 folgt nicht*, dass der Interpolationsfehler für $m \to \infty$ gegen Null konvergiert. Tatsächlich lassen sich selbst auf äquidistanten Gittern Beispiele konstruieren, bei denen das Interpolationspolynom nicht punktweise gegen die interpolierte Funktion konvergiert. In der Praxis beobachtet man bei hohem Interpolationsgrad unerwünschte starke Oszillationen zwischen den Interpolationspunkten (vgl. Übungsaufgabe 4 auf Blatt 2).*

Aus Satz 1.7 folgt aber

$$|f(x) - p(x)| \le \frac{\left\| f^{(m)} \right\|_{[a,b]}}{m!} |b-a|^m.$$

Für festes m konvergiert der Fehler also für kleiner werdende Intervallbreite $(b-a) \to 0$ gegen Null, und m steuert, wie schnell dies geschieht.

In vielen Teilgebieten der Numerik werden deshalb Funktionen nicht global *durch ein Polynom möglichst hohen Grades angenähert, sondern auf möglichst kleinen Stücken durch Polynome von festem (oder an die Größe der Stücke angepasstem) Grad approximiert.*

(b) Hermite-Interpolation: Die Interpolationsaufgabe lässt sich verallgemeinern zu der Aufgabe zu $f \in C^m([a,b])$ und Knoten

$$x_1 < x_2 < \ldots < x_m$$

ein Polynom p zu finden, das f und seine Ableitungen interpoliert, d.h.

$$p^{(j)}(x_i) = f^{(j)}(x_i), \quad \text{für alle } j = 0,\ldots,J_i - 1 \in \mathbb{N}_0,\ i = 1,\ldots,m.$$

Es lässt sich zeigen[3], dass auch für diese Aufgabe genau ein Interpolationspolynom $p \in \Pi_{M-1}$ existiert, wobei der Höchstgrad $M-1 \in \mathbb{N}_0$ der entsprechend der Anzahl der Ableitungen mehrfach gezählten Knotenanzahl entspricht,

$$M = \sum_{i=1}^{m} J_i.$$

Mit dem entsprechend definierten Knotenpolynom

$$\omega(x) = \prod_{i=1}^{m} (x - x_i)^{J_i} \in \Pi_M$$

folgt analog Satz 1.7 die Fehlerabschätzung

$$f(x) - p(x) = \frac{f^{(M)}(\xi)}{M!}\omega(x)$$

mit einem $\xi = \xi(x) \in [\min\{x, x_0\}, \max\{x, x_m\}] \subseteq [a,b]$.

1.4 Newton-Cotes Formeln

1.4.1 Konstruktion und Fehlerabschätzung

Weiterhin seien stets $a,b \in \mathbb{R}$, $a < b$. $f:\ [a,b] \to \mathbb{R}$ sei integrierbar und für $w \in C([a,b])$ gelte $w(x) > 0$ für $x \in (a,b)$.

Wir kehren zurück zur Aufgabe, eine Quadraturformel mit möglichst hohem Exaktheitsgrad zu entwickeln für

$$I[f;w] = \int_a^b f(x)w(x)\,\mathrm{d}x.$$

Idee:

[3] Für einen wichtigen Spezialfall zeigen wir dies in Übungsaufgabe 2.2.

- Wir wählen ein Gitter von Knoten

$$\{x_1, x_2, \ldots, x_m\} \subset [a,b], \qquad x_1 < x_2 < \ldots < x_m, \quad m \in \mathbb{N}.$$

- Wir interpolieren auf diesem Gitter die Funktionswerte durch ein Interpolationspolynom, d.h. wir bestimmen

$$p \in \Pi_{m-1} \quad \text{mit} \quad p(x_i) = f(x_i) \quad \forall i = 1, \ldots, m.$$

- Wir verwenden das Integal über das Interpolationspolynom p als Näherung für das Integral über die Funktion f:

$$I[f;w] = \int_a^b f(x)w(x)\,\mathrm{d}x \approx \int_a^b p(x)w(x)\,\mathrm{d}x.$$

Mit der expliziten Darstellung von p aus Satz 1.5 erhalten wir mit dieser Idee die Quadraturformel

$$\begin{aligned} Q[f] &= \int_a^b p(x)w(x)\,\mathrm{d}x = \int_a^b \sum_{i=1}^m f(x_i) l_i(x) w(x)\,\mathrm{d}x \\ &= \sum_{i=1}^m \underbrace{\int_a^b l_i(x)w(x)\,\mathrm{d}x}_{=:w_i} f(x_i) = \sum_{i=1}^m w_i f(x_i). \end{aligned}$$

Aufgrund der Konstruktion erwarten wir, dass der Exaktheitsgrad von Q (mindestens) $m-1$ beträgt. Tatsächlich ist dies die einzige Möglichkeit (zu gegebenen Knoten) einen so hohen Exaktheitsgrad zu erhalten:

Satz 1.9
$Q[\cdot]$ sei eine Quadraturformel mit Knoten

$$\{x_1, x_2, \ldots, x_m\} \subset [a,b], \qquad x_1 < x_2 < \ldots < x_m, \quad m \in \mathbb{N},$$

und Gewichten $w_i \in \mathbb{R}$, $i = 1, \ldots, m$, also

$$Q[f] = \sum_{i=1}^m w_i f(x_i).$$

Q hat genau dann Exaktheitsgrad $q \geq m-1$, wenn für die Gewichte gilt, dass

$$w_i = \int_a^b l_i(x)w(x)\,\mathrm{d}x, \quad i = 1, \ldots, m. \tag{1.3}$$

Beweis: Die Quadraturformel habe Exaktheitsgrad $q \geq m-1$. Dann integriert sie insbesondere die Lagrange-Grundpolynome $l_j \in \Pi_{m-1}$ exakt, also

$$\int_a^b l_j(x)w(x)\,\mathrm{d}x = Q[l_j] = \sum_{i=1}^m w_i l_j(x_i) = w_j, \quad j = 1,\ldots,m.$$

Um die Rückrichtung zu zeigen, sei $f \in \Pi_{m-1}$. Dann ist f ein Interpolationspolynom zu seinen eigenen Funktionswerten $y_i := f(x_i)$, $i = 1,\ldots,m$. Aus Satz 1.5 folgt, dass

$$f(x) = \sum_{i=1}^m f(x_i)l_i(x).$$

Erfüllen die Gewichte die Bedingung (1.3), so folgt wie oben

$$\begin{aligned} I[f;w] &= \int_a^b f(x)w(x)\,\mathrm{d}x = \int_a^b \sum_{i=1}^m f(x_i)l_i(x)w(x)\,\mathrm{d}x \\ &= \sum_{i=1}^m \underbrace{\int_a^b l_i(x)w(x)\,\mathrm{d}x}_{=w_i} f(x_i) = \sum_{i=1}^m w_i f(x_i) = Q[f]. \end{aligned}$$

$f \in \Pi_{m-1}$ wird also exakt integriert. □

Bemerkung 1.10
Satz 1.9 lässt sich auch wie folgt interpretieren (und beweisen). Die Abbildungen

$$\begin{aligned} I[\cdot;w] :\ f &\mapsto \int_a^b f(x)w(x)\,\mathrm{d}x \\ Q[\cdot] :\ f &\mapsto \sum_{i=1}^m w_i f(x_i) \end{aligned}$$

sind lineare Abbildungen von dem Vektorraum Π_{m-1} nach $\mathbb{R}$. Sie stimmen also genau dann auf Π_{m-1} überein, wenn sie auf einer Basis des Π_{m-1} übereinstimmen.

Die Bedingung (1.3) ist äquivalent dazu, dass

$$Q[l_i] = I[l_i;w] \quad \forall i = 1,\ldots,m$$

und die Lagrange-Grundpolynome l_i bilden nach Bemerkung 1.6(b) eine Basis des Π_{m-1}.

Aus Satz 1.7 erhalten wir die folgende Fehlerabschätzung:

Satz 1.11
Sei $f \in C^m([a,b])$ und $Q[\cdot]$ sei eine Quadraturformel mit Knoten

$$\{x_1, x_2, \ldots, x_m\} \subset [a,b], \qquad x_1 < x_2 < \ldots < x_m, \quad m \in \mathbb{N},$$

und Gewichten gemäß (1.3).

Dann gilt

$$|I[f;w] - Q[f]| \leq \frac{\left\|f^{(m)}\right\|_{[a,b]}}{m!} \int_a^b |\omega(x)| w(x)\,\mathrm{d}x,$$

wobei $\omega(x) = \prod_{i=1}^m (x - x_i)$ das Knotenpolynom ist.

Beweis: Sei $f \in C^m([a,b])$ und $p \in \Pi_{m-1}$ das dazugehörige Interpolationspolynom mit $p(x_i) = f(x_i)$ für alle $i = 1, \ldots, m$. Aus Satz 1.7 folgt, dass

$$|f(x) - p(x)| \leq \frac{\left\|f^{(m)}\right\|_{[a,b]}}{m!} |\omega(x)| \quad \forall x \in [a,b].$$

Wegen Satz 1.9 ist $Q[f] = Q[p] = I[p;w]$ und damit

$$\begin{aligned} |I[f;w] - Q[f]| = |I[f-p;w]| &\leq \int_a^b |f(x) - p(x)| w(x)\,\mathrm{d}x \\ &\leq \frac{\left\|f^{(m)}\right\|_{[a,b]}}{m!} \int_a^b |\omega(x)| w(x)\,\mathrm{d}x. \qquad \square \end{aligned}$$

Bemerkung 1.12

(a) Aus einer Quadraturformel Q erhalten wir ein (zusammengesetztes) Quadraturverfahren Q_n, indem wir $[a,b]$ in n-Teilintervalle der Länge $h = \frac{b-a}{n}$ unterteilen, in denen wir jeweils Q anwenden. Mit Satz 1.11 können wir den Fehler auf jedem der n Teilintervalle abschätzen durch

$$\frac{\left\|f^{(m)}\right\|_{[a,b]}}{m!} h h^m \|w\|_{[a,b]}$$

und wir erhalten die Fehlerabschätzung

$$|I[f;w] - Q_n[f]| \leq \frac{\|w\|_{[a,b]} (b-a)}{m!} \left\|f^{(m)}\right\|_{[a,b]} h^m.$$

Für festes (hinreichend glattes) f fällt der Fehler (mindestens) so schnell wie h^m. Den Exponenten m bezeichnet man auch als Konsistenzordnung *des Verfahrens.*

(b) *Im Fall $w = 1$ und äquidistanter Knoten $a = x_1 < \ldots < x_m = b$ heißen die gemäß Satz 1.9 aufgestellten Quadraturformeln* (abgeschlossene) Newton-Cotes-Formeln.

(c) *Es ist nicht gesichert, dass die gemäß (1.3) aufgestellten Gewichte positiv sind. Bei den (abgeschlossenen) Newton-Cotes-Formeln treten ab $m = 8$ Knoten erstmalig negative Gewichte auf.*

1.4.2 Beispiele: Trapez- und Simpsonformel

Die Trapezformel besitzt die Knoten $x_1 = a$ und $x_2 = b$ und integriert (bzgl. der Gewichtsfunktion $w = 1$) Polynome 1. Grades exakt. Die Trapezformel ist also die abgeschlossene Newton-Cotes-Formel mit zwei Knoten.

Wir bestimmen nun die abgeschlossene Newton-Cotes-Formel $S[\cdot]$ mit drei Knoten. Sie besitzt die äquidistanten Knoten

$$x_1 = a, \quad x_2 = \frac{a+b}{2}, \quad x_3 = b$$

und Gewichte gemäß (1.3). In diesem einfachen Fall lassen sich die Gewichte schneller per Hand bestimmen, indem wir ansetzen

$$S[f] = w_1 f(a) + w_2 f\left(\frac{a+b}{2}\right) + w_3 f(b)$$

und $w_1, w_2, w_3 \in \mathbb{R}$ so bestimmen, dass

$$S[p] = I[p]$$

für $p = 1$, $p = (x-a)$ und $p = (x-a)(x-b)$ gilt. Offenbar bilden auch diese drei Polynome eine Basis des Π_2, so dass aus der Linearität von $S[\cdot]$ und $I[\cdot]$ dann der gewünschte Exaktheitsgrad 2 folgt (vgl. Bemerkung 1.10).

Wir erhalten

$$\begin{aligned} w_1 + w_2 + w_3 = S[1] &= I[1] \\ &= \int_a^b 1 \,\mathrm{d}x = b - a, \\ w_2(b-a)/2 + w_3(b-a) = S[x-a] &= I[x-a] \\ &= \int_a^b (x-a) \,\mathrm{d}x = (b-a)^2/2, \\ -w_2(b-a)^2/4 = S[(x-a)(x-b)] &= I[(x-a)(x-b)] \\ &= \int_a^b (x-a)(x-b) \,\mathrm{d}x = (a-b)^3/6, \end{aligned}$$

und damit $w_2 = 2/3(b-a)$, $w_3 = 1/6(b-a) = w_1$. Insgesamt ergibt sich

$$\int_a^b f(x)\,\mathrm{d}x \approx S[f] = \frac{b-a}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right),$$

die sogenannte *Simpson-Formel*.

Für das *zusammengesetzte Simspon-Verfahren* zerlegen wir $[a,b]$ in n Teilintervalle und wenden auf jedem Teilintervall die Simpsonformel an. Abweichend von der bisherigen Notation nummerieren wir die Knoten über alle Teilintervalle hinweg und verwenden

$$h := (b-a)/2n, \quad \text{und } x_i := a + (i-1)h, \quad i = 1,\dots,2n+1.$$

Damit ist

$$\begin{aligned}\int_a^b f(x)\,\mathrm{d}x \approx \frac{h}{3}\,&(f(a) + 4f(x_2) + 2f(x_3) + 4f(x_4) + \dots \\ &+ 2f(x_{2n-1}) + 4f(x_{2n}) + f(b)) =: S_n[f].\end{aligned}$$

Die Simpson-Formel ist sogar noch besser als sich aus der Konstruktion erwarten lässt. Offensichtlich ergänzt $p(x) = \left(x - \frac{a+b}{2}\right)^3$ jede Basis von Π_2 zu einer Basis des Π_3 und aus Symmetriegründe gilt

$$S\left[\left(x - \frac{a+b}{2}\right)^3\right] = 0 = I\left[\left(x - \frac{a+b}{2}\right)^3\right],$$

so dass die Simpsonformel sogar Exaktheitsgrad 3 besitzt. Einen höheren Exaktheitsgrad besitzt sie nicht, denn für $a = -1$ und $b = 1$ ist

$$I[x^4] = \int_{-1}^{1} x^4\,\mathrm{d}x = \frac{2}{5} \neq \frac{2}{3} = S[x^4].$$

Entsprechend erhalten wir auch eine höhere Konsistenzordnung:

Satz 1.13
Sei $f \in C^4([a,b])$, $n \in \mathbb{N}$ und $h := \frac{b-a}{2n}$. Dann gilt

$$|I[f] - S_n[f]| \leq \frac{b-a}{180}\left\|f^{(4)}\right\|_{[a,b]} h^4.$$

Beweis: Sei $[c,d]$ eines der n Teilintervalle der Breite $2h$, auf die die Simpson-Formel angewandt wird. Nach Bemerkung 1.8 (und analog zu Übungsaufgabe 2 auf Blatt 2) existiert ein Polynom $p \in \Pi_3$ mit

$$p(c) = f(c), \quad p\left(\frac{c+d}{2}\right) = f\left(\frac{c+d}{2}\right),$$
$$p(d) = f(d), \quad p'\left(\frac{c+d}{2}\right) = f'\left(\frac{c+d}{2}\right).$$

Mit dem Knotenpolynom $\omega = (x-c)(x-d)\left(x-\frac{c+d}{2}\right)^2$ folgt aus Satz 1.7 und Bemerkung 1.8

$$|f(x)-p(x)| \leq \frac{\left\|f^{(4)}\right\|_{[a,b]}}{4!}|\omega(x)| \quad \forall x \in [c,d]$$

und damit

$$\left|\int_c^d (f(x)-p(x))\,\mathrm{d}x\right| \leq \frac{\left\|f^{(4)}\right\|_{[a,b]}}{4!}\int_c^d (x-c)(d-x)\left(x-\frac{c+d}{2}\right)^2 \mathrm{d}x$$
$$= \frac{d-c}{180}\left(\frac{d-c}{2}\right)^4 \left\|f^{(4)}\right\|_{[a,b]}.$$

Mit $\frac{d-c}{2} = h$ folgt die Behauptung durch Summation über alle Teilintervalle. □

1.5 Gauß-Quadratur

1.5.1 Exaktheitsgrad und Orthogonalpolynome

Es gelte weiterhin, dass $a,b \in \mathbb{R}$, $a < b$, $f : [a,b] \to \mathbb{R}$ integrierbar seien. Für die Gewichtsfunktion gelte $w \in C([a,b])$, $w(x) > 0$ für $x \in (a,b)$ (beachte aber Bemerkung 1.19 am Ende dieses Abschnittes).

Wir betrachten wieder unseren allgemeinen Ansatz für eine Quadraturformel $Q[\cdot]$

$$\int_a^b f(x)w(x)\,\mathrm{d}x = I[f;w] \approx Q[f] = \sum_{i=1}^m w_i f(x_i)$$

mit Knoten

$$\{x_1,x_2,\ldots,x_m\} \subset [a,b], \qquad x_1 < x_2 < \ldots < x_m, \quad m \in \mathbb{N},$$

und Gewichten $w_i \in \mathbb{R}$, $i = 1, \ldots, m$.

Aus Satz 1.9 wissen wir, dass wir für jede Wahl der Knoten stets mindestens Exaktheitsgrad $m-1$ erreichen können und zwar genau dadurch, dass wir die Gewichte gemäß (1.3) wählen.

In diesem Abschnitt untersuchen wir, welche Wahl der Knoten (zusammen mit gemäß (1.3) gewählten Gewichten) optimalen Exaktheitsgrad liefert.

Wir benötigen noch ein Hilfsresultat aus der Analysis.

Lemma 1.14
Sei $g \in C([a,b])$ nichtnegativ. Dann ist

$$\int_a^b g(x) = 0 \quad \textit{genau dann wenn} \quad g = 0.$$

Beweis: Um die nicht-triviale Implikation zu zeigen, sei $g \neq 0$. Dann existiert $\xi \in (a,b)$ mit $g(\xi) > 0$. Zu $\delta := g(\xi)/2 > 0$ existiert dann wegen der Stetigkeit ein $\varepsilon > 0$, so dass $g(x) > \delta$ für $x \in (\xi - \varepsilon, \xi + \varepsilon) \subset [a,b]$. Also ist

$$\int_a^b g(x)\,\mathrm{d}x \geq \int_{\xi-\varepsilon}^{\xi+\varepsilon} g(x)\,\mathrm{d}x \geq 2\varepsilon\delta > 0. \qquad \square$$

Jetzt können wir den höchstmöglichen Exaktheitsgrad angeben:

Satz 1.15
(a) Der Exaktheitsgrad von $Q[\cdot]$ ist höchstens $2m-1$.

(b) $Q[\cdot]$ hat genau dann Exaktheitsgrad $q = 2m-1$, wenn für das Knotenpolynom $\omega = \prod_{i=1}^m (x - x_i)$ gilt

$$\int_a^b \omega(x) p(x) w(x)\,\mathrm{d}x = 0 \quad \forall p \in \Pi_{m-1} \tag{1.4}$$

und die Gewichte w_i durch Integration der Lagrange-Grundpolynome, also durch (1.3), bestimmt wurden.

Beweis: (a) Wir betrachten das quadrierte Knotenpolynom $\omega^2(x) \in \Pi_{2m}$. Hierfür gilt

$$Q[\omega^2] = \sum_{i=1}^m w_i \omega^2(x_i) = 0.$$

Auf der anderen Seite ist $\omega^2(x)w(x)$ eine stetige, nichtnegative Funktion, die nicht die Nullfunktion ist. Aus 1.14 folgt also

$$I[\omega^2; w] = \int_a^b \omega^2(x) w(x)\,\mathrm{d}x \neq 0 = Q[\omega^2].$$

Eine Quadraturformel kann es also niemals schaffen, ihr quadriertes Knotenpolynom exakt zu integrieren.

(b) Der Exaktheitsgrad sei $q = 2m-1$. Dann müssen die Gewichte nach Satz 1.9 der Bedingung (1.3) genügen. Außerdem wird dann für alle $p \in \Pi_{m-1}$ das Polynom $p\omega \in \Pi_{2m-1}$ exakt integriert also

$$\int_a^b \omega(x)p(x)w(x)\,\mathrm{d}x = I[p\omega;w] = Q[p\omega] = 0.$$

Um die Rückrichtung zu zeigen, seien die Gewichte und das Knotenpolynom so, dass (1.3) und (1.4) erfüllt seien. Offenbar bildet

$$(x^0, x^1, \ldots, x^{m-1}, \omega(x), x\omega(x), x^2\omega(x), \ldots, x^{m-1}\omega(x))$$

eine Basis des Π_{2m-1}. Da (1.3) erfüllt ist, werden nach Satz 1.9 die ersten m Basiselemente $x^0, \ldots, x^{m-1}$ exakt integriert und (1.4) garantiert, dass

$$I[x^j\omega;w] = \int_a^b x^j\omega(x)w(x)\,\mathrm{d}x = 0 = Q[x^j\omega] \quad \forall j = 1,\ldots,m-1.$$

Aus der Linearität von $Q[\cdot]$ und $I[\cdot;w]$ folgt damit, dass Q Exaktheitsgrad $2m-1$ besitzt (vgl. Bemerkung 1.10). □

Bemerkung 1.16 (Naheliegender Ansatz)
Ein naheliegender Ansatz ist es zu versuchen, die m Knoten $x_i \in [a,b]$ so zu bestimmen, dass (1.4) erfüllt ist. Mit der Monom-Basis von Π_{m-1} ist (1.4) äquivalent zu den m-Gleichungen

$$\int_a^b \prod_{i=1}^m (x-x_i)w(x)x^j\,\mathrm{d}x = 0 \quad \forall j = 0,\ldots,m-1$$

für die m Unbekannten x_i. Die Gleichungen sind nicht-linear und es ist nicht klar, ob überhaupt eine Lösung existiert. Dieser naheliegende Ansatz führt also nicht zum Erfolg.

Setzen wir hingegen ω wie in Bemerkung 1.3 als allgemeines Polynom in Π_m an, dann ist nicht klar, ob ω von der Form $\prod_{i=1}^m (x-x_i)$ mit $x_i \in [a,b]$, also ein geeignetes Knotenpolynom ist. Der nächste Satz zeigt, dass dies aber tatsächlich immer der Fall ist.

Satz 1.17
Erfüllt ein Polynom $\omega = x^m + \ldots \in \Pi_m \setminus \Pi_{m-1}$ die Bedingung (1.4), dann besitzt es m reelle, paarweise verschiedene Nullstellen, die alle im offenen Intervall (a,b) liegen.

Beweis: Nach dem Fundamentalsatz der Algebra existieren genau m (möglicherweise komplexe und mit Vielfachheit gezählte) Nullstellen $x_1, \dots, x_m \in \mathbb{C}$.

Angenommen eine der Nullstellen wäre nicht reell, o.B.d.A. sei dies $x_1 \in \mathbb{C} \setminus \mathbb{R}$. Da ω reelle Koeffizienten besitzt, ist $\omega(\overline{x_1}) = \overline{\omega(x_1)} = 0$, also ist $\overline{x}_1$ eine weitere (von x_1 verschiedene) Nullstelle. O.B.d.A. sei dies $x_2 = \overline{x_1}$. Es gilt also

$$\omega(x) = (x - x_1)(x - \overline{x_1}) \prod_{i=3}^{m} (x - x_i).$$

Da $(x - x_1)(x - \overline{x_1}) = x^2 - 2\mathrm{Re}(x_1)x + |x_1|^2$ ein Polynom mit reellen Koeffizienten ist, folgt durch Polynomdivision (siehe z.B. [Fischer, Abschnitt 2.3.12]), dass auch das Polynom $p(x) = \prod_{i=3}^{m}(x - x_i)$ reelle Koeffizienten besitzt. $\omega(x)p(x)$ ist daher ein reelles Polynom und

$$\omega(x)p(x) = |x - x_1|^2 p(x)^2 \geq 0 \qquad \text{für alle } x \in \mathbb{R}.$$

Da $\omega(x)p(x)w(x)$ nur endlich viele Nullstellen besitzt, folgt aus Lemma 1.14, dass

$$\int_a^b \omega(x)p(x)w(x)\,\mathrm{d}x > 0.$$

Für dieses $p \in \Pi_{m-2}$ wäre dann also die Bedingung (1.4) verletzt, womit gezeigt ist, dass ω nur reelle Nullstellen besitzen kann.

Dass die Nullstellen paarweise verschieden sind und in (a,b) liegen wird in Aufgabe 1 auf Blatt 4 gezeigt. □

Dank Satz 1.17 können wir einen Ansatz wie in 1.3 verwenden, um Polynome zu finden, die (1.4) erfüllen. Es geht jedoch noch einfacher:

Bemerkung 1.18
Die Bedingung (1.4) lässt sich als Orthogonalitätsbedingung interpretieren. Wir betrachten dazu

$$\langle \cdot, \cdot \rangle : \ (f,g) \mapsto \langle f, g \rangle := \int_a^b f(x)g(x)w(x)\,\mathrm{d}x.$$

Dies definiert ein Skalarprodukt *auf dem Raum $C([a,b])$ (siehe Blatt 4, Aufgabe 2).*

Für $\omega \in \Pi_m$ ist Bedingung (1.4) äquivalent dazu, dass (bzgl. dieses Skalarproduktes)

$$\omega \perp \Pi_{m-1}, \quad \textit{d.h.} \quad \langle \omega, p \rangle = 0 \quad \forall p \in \Pi_{m-1}.$$

Aus der Linearen Algebra wissen wir, dass jedes linear unabhängige System in ein Orthogonal-System umgewandelt werden kann (z.B. mit dem Gram-Schmidtschen-Orthogonalisierungsverfahren, vgl. z.B. [Fischer, 6.5.5]). So können wir beginnend mit $\omega_0 := 1$ *eine Folge von* Orthogonalpolynomen $\omega_m \in \Pi_m$ *konstruieren mit* $\omega_m \in \Pi_m$ *und* $\omega_m \perp \omega_j$ *für alle* $j < m$, *also* $\omega_m \perp \Pi_{m-1}$.

Für jedes m erhalten wir so (mit den m Nullstellen von ω_m *als Knoten und aus (1.3) bestimmten Gewichten) ein Quadraturverfahren, dass mit m Knoten den maximal möglichen Exaktheitsgrad* $2m-1$ *erreicht. Die so konstruierten Verfahren heißen* Gauß-Verfahren.

Bemerkung 1.19
Man kann zeigen, dass alle bisher gezeigten Resultate auch unter der allgemeineren Voraussetzung gelten, dass die Gewichtsfunktion w nicht-negativ und integrierbar ist, und nur endlich viele Nullstellen besitzt.

1.5.2 Gauß-Legendre-Formeln

Wir betrachten nun die Konstruktion der Gauß-Verfahren für die Gewichtsfunktion $w = 1$ auf dem Intervall $[-1,1]$, die sogenannten *Gauß-Legendre-Formeln* $G_m[\cdot]$. In diesem Abschnitt bezieht sich Orthogonalität immer auf das Skalarprodukt

$$\langle f,g\rangle := \int_{-1}^{1} f(x)g(x)\,\mathrm{d}x.$$

Beispiel 1.20

(a) Für die erste Gauß-Legendre-Formel benötigen wir ein Polynom $\omega_1 \in \Pi_1$, *das senkrecht auf allen Polynomen nullten Grades steht, also*

$$\int_{-1}^{1} \omega_1(x)\,\mathrm{d}x = \langle \omega_1, 1\rangle = 0.$$

Wir erhalten $\omega_1(x) = x$ *mit der Nullstelle* $x_1 = 0$. *Durch Integration des zugehörigen Lagrange-Grundpolynoms* $l_1(x) = 1$ *erhalten wir* $w_1 = 2$ *und damit die erste Gauß-Legendre-Formel, die Mittelpunktsformel*

$$G_1[f] = 2f(0)$$

mit Exaktheitsgrad 1.

(b) Für die zweite Gauß-Legendre-Formel suchen wir ein Polynom

$$\omega_2(x) = x^2 + ax + b \in \Pi_2$$

mit $\omega_2 \perp \Pi_1$. *Mit*

$$2/3 + 2b = \int_{-1}^{1} \omega_2(x)\,\mathrm{d}x = 0$$
$$2/3a = \int_{-1}^{1} x\,\omega_2(x)\,\mathrm{d}x = 0$$

erhalten wir $\omega_2(x) = x^2 - 1/3$. *Die Nullstellen sind*

$$x_1 = -\frac{1}{\sqrt{3}}, \quad x_2 = \frac{1}{\sqrt{3}}.$$

Durch Integration der Lagrange-Grundpolynome, den Ansatz in Abschnitt 1.4.2 oder eine schlichte Symmetrieüberlegung erhalten wir die Gewichte $w_1 = 1$ *und* $w_2 = 1$. *Die zweite Gauß-Legendre-Formel lautet also*

$$G_2[f] = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

und besitzt Exaktheitsgrad 3.

Auch für allgemeine m lassen sich die Orthogonalpolynome explizit angeben:

Definition 1.21
Die Funktionen

$$P_m(x) := \frac{1}{2^m m!}\frac{d^m}{dx^m}(x^2-1)^m$$

heißen Legendre-Polynome. *Offenbar gilt* $P_m \in \Pi_m$.

Satz 1.22
(a) Es existiert ein $p \in \Pi_{m-2}$ *mit*

$$P_m(x) = \frac{(2m)!}{(m!)^2 2^m}x^m + p(x).$$

(b) Es gilt

$$\int_{-1}^{1} P_n(x)P_m(x)\,\mathrm{d}x = \frac{2}{2m+1}\delta_{nm}.$$

Insbesondere gilt also $P_m \perp \Pi_{m-1}$.

Beweis: (a) Mit

$$q(x) := (x^2-1)^m - x^{2m} \in \Pi_{2m-2} \quad \text{und} \quad p(x) := \frac{1}{2^m m!}\frac{d^m}{dx^m}q(x) \in \Pi_{m-2}$$

folgt die Aussage aus

$$\begin{aligned} P_m(x) &= \frac{1}{2^m m!}\frac{d^m}{dx^m}(x^2-1)^m = \frac{1}{2^m m!}\frac{d^m}{dx^m}(x^{2m}+q(x)) \\ &= \frac{1}{2^m m!}(2m)\cdot\ldots\cdot(m+1)x^m + \frac{1}{2^m m!}\frac{d^m}{dx^m}q(x) \\ &= \frac{1}{2^m m!}\frac{(2m)!}{m!}x^m + p(x). \end{aligned}$$

(b) Das Polynom $(x^2-1)^n$ besitzt n-fache Nullstellen in -1 und 1, es ist also

$$\frac{d^{n-j}}{dx^{n-j}}(x^2-1)^n|_{x=-1} = 0 = \frac{d^{n-j}}{dx^{n-j}}(x^2-1)^n|_{x=1}, \quad j = 1,\ldots,n.$$

Durch partielle Integration erhalten wir deshalb

$$\begin{aligned} &2^n n! 2^m m! \int_{-1}^{1} P_n(x)P_m(x)\,\mathrm{d}x \\ &= \int_{-1}^{1} \frac{d^n}{dx^n}(x^2-1)^n \frac{d^m}{dx^m}(x^2-1)^m dx \\ &= -\int_{-1}^{1} \frac{d^{n-1}}{dx^{n-1}}(x^2-1)^n \frac{d^{m+1}}{dx^{m+1}}(x^2-1)^m dx \\ &= \ldots = (-1)^n \int_{-1}^{1} (x^2-1)^n \frac{d^{m+n}}{dx^{m+n}}(x^2-1)^m dx. \end{aligned}$$

Aus $(x^2-1)^m \in \Pi_{2m}$ folgt, dass

$$\frac{d^{m+n}}{dx^{m+n}}(x^2-1)^m = 0 \quad \text{falls } n > m.$$

Es gilt also

$$\int_{-1}^{1} P_n(x)P_m(x)\,\mathrm{d}x = 0$$

für $m < n$ und (durch Vertauschung von n und m) auch für $m > n$.

Für $m = n$ ist

$$\frac{d^{m+n}}{dx^{m+n}}(x^2-1)^m = \frac{d^{2m}}{dx^{2m}}(x^{2m}+\ldots) = (2m)!.$$

und damit

$$\int_{-1}^{1} P_m^2(x)\,\mathrm{d}x = \frac{(-1)^m(2m)!}{(2^m m!)^2}\int_{-1}^{1}(x^2-1)^m\,\mathrm{d}x$$

Wiederum mit m-maliger partieller Integration erhalten wir

$$\begin{aligned}\int_{-1}^{1}(x^2-1)^m\,\mathrm{d}x &= \int_{-1}^{1}(x-1)^m(x+1)^m\,\mathrm{d}x\\ &= -\int_{-1}^{1}\frac{1}{m+1}(x-1)^{m+1}m(x+1)^{m-1}\,\mathrm{d}x\\ &= \ldots = \frac{(-1)^m m!}{(m+1)\cdot\ldots\cdot 2m}\int_{-1}^{1}(x-1)^{2m}\,\mathrm{d}x\\ &= \frac{(-1)^m(m!)^2}{(2m)!}\frac{-1}{2m+1}(-2)^{2m+1}\end{aligned}$$

also insgesamt

$$\begin{aligned}\int_{-1}^{1}P_m(x)P_m(x)\,\mathrm{d}x &= \frac{(-1)^m(2m)!}{(2^m m!)^2}\frac{(-1)^m(m!)^2}{(2m)!}\frac{-1}{2m+1}(-2)^{2m+1}\\ &= \frac{2}{2m+1},\end{aligned}$$

so dass (b) bewiesen ist. □

Satz 1.22(b) zeigt, dass die Legendre-Polynome die gesuchten Orthogonalpolynome zur Gewichtsfunktion $w=1$ sind. Verwenden wir ihre Nullstellen als Knoten (und wie üblich durch Integration der Lagrange-Grundpolynome gewonnene Gewichte), so erhalten wir ein Quadraturverfahren mit maximalem Exaktheitsgrad!

Wir zeigen noch eine Fehlerabschätzung für die m-te Gauß-Legendre-Formel:

Satz 1.23
Für $f\in C^{2m}([-1,1])$ gilt

$$|I[f]-G_m[f]| \le \varepsilon_m \left\| f^{(2m)} \right\|_{[-1,1]}$$

mit

$$\varepsilon_m = \frac{2}{2m+1}\frac{4^m(m!)^4}{((2m)!)^3} \approx \frac{\sqrt{\pi}}{2\sqrt{m}}(4m/\mathrm{e})^{-2m}$$

wobei hier „$\approx$" im Sinne asymptotischer Gleichheit zu verstehen ist, d.h.

$$\varepsilon_m \Big/ \frac{\sqrt{\pi}}{2\sqrt{m}}(4m/\mathrm{e})^{-2m} \to 1 \quad \textit{für } m\to\infty.$$

Beweis: Sei $f\in C^{2m}([-1,1])$. Nach Übungsaufgabe 2 auf Blatt 2 existiert ein Polynom $p\in\Pi_{2m-1}$, das f zusammen mit seiner 1. Ableitung in den Knoten x_i der m-ten Gauß-Legendre-Formel interpoliert, also

$$p(x_i)=f(x_i),\quad p'(x_i)=f'(x_i)\quad \forall i=1,\ldots,m.$$

Aus Satz 1.7 und Bemerkung 1.8 folgt

$$|f(x)-p(x)| \leq \frac{\left\|f^{(2m)}\right\|_{[-1,1]}}{(2m)!}\prod_{i=1}^{m}(x-x_i)^2.$$

Aus Satz 1.22(a) wissen wir, dass

$$\prod_{i=1}^{m}(x-x_i) = \frac{(m!)^2 2^m}{(2m)!}P_m(x)$$

und erhalten zusammen mit Satz 1.22(b)

$$\begin{aligned}|I[f]-G_m[f]| = |I[f]-G_m[p]| &\leq \int_{-1}^{1}|f(x)-p(x)|\,\mathrm{d}x\\ &\leq \frac{\left\|f^{(2m)}\right\|_{[-1,1]}}{(2m)!}\left(\frac{(m!)^2 2^m}{(2m)!}\right)^2\int_{-1}^{1}P_m^2(x)\,\mathrm{d}x\\ &= \frac{\left\|f^{(2m)}\right\|_{[-1,1]}}{(2m)!}\left(\frac{(m!)^2 2^m}{(2m)!}\right)^2\frac{2}{2m+1}.\end{aligned}$$

Für den zweiten Teil der Behauptung verwenden wir die *Stirling-Formel* (siehe z.B. [Heuser, Abschnitt 96])

$$m! \approx m^m e^{-m}\sqrt{2\pi m}$$

und erhalten mit

$$\begin{aligned}\varepsilon_m &= \frac{2}{2m+1}\frac{4^m(m!)^4}{((2m)!)^3} \approx \frac{2}{2m+1}\frac{4^m(m^m e^{-m}\sqrt{2\pi m})^4}{((2m)^{2m}e^{-2m}\sqrt{4\pi m})^3}\\ &= \frac{2}{2m+1}\frac{1}{4^{2m}}\frac{1}{m^{2m}}e^{2m}\frac{\sqrt{\pi m}}{2} \approx \frac{\sqrt{\pi}}{2\sqrt{m}}(4m/e)^{-2m}\end{aligned}$$

die behauptete asymptotische Abschätzung. □

1.5.3 Der Golub-Welsh Algorithmus

Die Nullstellen der Gauß-Legendre Polynome und auch die Knoten der zugehörigen Gauß-Legendre-Quadraturformeln können über die Lösung eines Eigenwertproblems bestimmt werden. Hierfür benötigen wir noch folgendes Resultat:

Satz 1.24
Die Legendre-Polynome genügen der dreistufigen Rekursionformel

$$(m+1)P_{m+1}(x) = (2m+1)xP_m(x) - mP_{m-1}(x), \quad \forall m \in \mathbb{N}.$$

Beweis: Aus Satz 1.22(a) wissen wir, dass

$$\begin{aligned}(m+1)P_{m+1}(x) &\in (m+1)\frac{(2m+2)!}{((m+1)!)^2 2^{m+1}}x^{m+1} + \Pi_{m-1} \\ &= (2m+1)\frac{(2m)!}{(m!)^2 2^m}x^{m+1} + \Pi_{m-1} \\ (2m+1)xP_m(x) &\in (2m+1)x\frac{(2m)!}{(m!)^2 2^m}x^m + \Pi_{m-1}.\end{aligned}$$

Es ist also

$$p(x) := (m+1)P_{m+1}(x) - (2m+1)xP_m(x) \in \Pi_{m-1}. \tag{1.5}$$

$(P_0, \ldots, P_{m-1})$ bildet eine Orthogonalbasis des Π_{m-1}. Wir entwickeln p in dieser Basis

$$p = \sum_{n=0}^{m-1} a_n P_n, \quad a_n \in \mathbb{R},$$

und erhalten für die Koeffizienten $a_n = \frac{\langle p, P_n\rangle}{\langle P_n, P_n\rangle}$.

Die Rekursionsformel ist bewiesen, wenn wir zeigen können, dass

$$a_n = \begin{cases} 0 & \text{für } n \le m-2, \\ -m & \text{für } n = m-1. \end{cases}$$

Dazu verwenden wir

$$\begin{aligned}\langle p, P_n\rangle &= \langle (m+1)P_{m+1} - (2m+1)xP_m, P_n\rangle \\ &= (m+1)\langle P_{m+1}, P_n\rangle - (2m+1)\langle xP_m, P_n\rangle \\ &= (m+1)\langle P_{m+1}, P_n\rangle - (2m+1)\langle P_m, xP_n\rangle.\end{aligned}$$

Wegen $P_{m+1} \perp \Pi_m$, $P_m \perp \Pi_{m-2}$ folgt daraus direkt

$$a_n = 0 \quad \text{für alle } n \le m-2.$$

Um a_{m-1} auszurechnen, verwenden (1.5) mit $m-1$ statt m und erhalten

$$mP_m(x) - (2m-1)xP_{m-1}(x) \in \Pi_{m-1}.$$

Da $P_m \perp \Pi_{m-1}$, gilt also

$$\langle P_m, mP_m - (2m-1)xP_{m-1}\rangle = 0.$$

und damit $(2m-1)\langle P_m, xP_{m-1}\rangle = \langle P_m, mP_m\rangle$.

Insgesamt ist also

$$\begin{aligned} a_{m-1} &= \frac{\langle p, P_{m-1}\rangle}{\langle P_{m-1}, P_{m-1}\rangle} = \frac{\langle (m+1)P_{m+1}(x) - (2m+1)xP_m(x), P_{m-1}\rangle}{\langle P_{m-1}, P_{m-1}\rangle} \\ &= \frac{-(2m+1)\langle P_m, xP_{m-1}\rangle}{\langle P_{m-1}, P_{m-1}\rangle} = -\frac{2m+1}{2m-1}\,\frac{(2m-1)\langle P_m, xP_{m-1}\rangle}{\langle P_{m-1}, P_{m-1}\rangle} \\ &= -\frac{(2m+1)m}{2m-1}\frac{\langle P_m, P_m\rangle}{\langle P_{m-1}, P_{m-1}\rangle} = -m. \end{aligned}$$

wobei wir im letzten Schritt 1.22(b) verwendet haben. □

Bemerkung 1.25

(a) Die Legendre-Polynome bilden ein Orthogonalsystem aber noch kein Orthonormalsystem bzgl. $\langle\cdot,\cdot\rangle$. Ein solches erhalten wir aber leicht durch Normierung. Für

$$Q_n := P_n / \langle P_n, P_n\rangle^{1/2} = \sqrt{\frac{2n+1}{2}}\,P_n \in \Pi_n$$

gilt offenbar $\langle Q_n, Q_m\rangle = \delta_{nm}$.

Aus der Rekursionsformel für die P_n erhalten wir

$$\frac{(m+1)}{\sqrt{2m+3}}Q_{m+1}(x) = \sqrt{2m+1}\,xQ_m(x) - \frac{m}{\sqrt{2m-1}}Q_{m-1}(x)$$

und damit

$$\beta_{m+1}Q_{m+1}(x) = xQ_m(x) - \beta_m Q_{m-1}(x) \quad \forall m \in \mathbb{N},$$

wobei $\beta_m = \frac{m}{\sqrt{4m^2-1}}$.

(b) Auch für die Orthogonalpolynome zu allgemeinen Gewichtsfunktion $w(x)$ lassen sich ähnliche dreistufige Rekursionsformeln herleiten, siehe z.B. [Hanke, Satz 33.1].

Mit Hilfe der Rekursionsformel können wir die Knoten und Gewichte der Gauß-Legendre-Verfahren durch ein Eigenwert-Problem beschreiben.

Satz 1.26
Wie betrachten die Matrix

$$A := \begin{pmatrix} 0 & \beta_1 & & & \\ \beta_1 & 0 & \ddots & & \\ & \ddots & \ddots & \beta_{m-2} & \\ & & \beta_{m-2} & 0 & \beta_{m-1} \\ & & & \beta_{m-1} & 0 \end{pmatrix} \in \mathbb{R}^{m\times m}, \quad \beta_n = \frac{n}{\sqrt{4n^2-1}}.$$

$\lambda_1,\dots,\lambda_m \in \mathbb{R}$ *seien die Eigenwerte von A und* $v^{(1)},\dots,v^{(m)} \in \mathbb{R}^m$ *die zugehörigen Eigenvektoren, wobei* $v^{(i)} = (v_j^{(i)})_{j=1}^m \in \mathbb{R}^m$.
Dann lautet die m-te Gauß-Legendre-Formel

$$G_m[f] = \sum_{i=1}^{m} \frac{2\left(v_1^{(i)}\right)^2}{\left\|v^{(i)}\right\|^2} f(\lambda_i),$$

wobei $\left\|v^{(i)}\right\|^2 = \sum_{j=1}^m \left(v_j^{(i)}\right)^2$ *die Euklidnorm ist. Insbesondere besitzen die Gauß-Legendre-Formeln nichtnegative Gewichte.*

Beweis: (a) **Knoten der** m**-ten Gauß-Legendre-Formel:** Wir schreiben die Rekursionsformeln für Q_n für $n = 1,\dots,m-1$ in Matrix-Vektor-Form: Für jedes $x \in \mathbb{R}$ gilt

$$x\begin{pmatrix} Q_1(x) \\ \vdots \\ Q_{m-1}(x) \end{pmatrix} = \begin{pmatrix} \beta_1 & 0 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{m-2} & 0 & \beta_{m-1} \\ & & & \beta_{m-1} & 0 \end{pmatrix} \begin{pmatrix} Q_0(x) \\ Q_1(x) \\ \vdots \\ Q_{m-1}(x) \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \beta_m Q_m(x) \end{pmatrix}.$$

Aus $Q_0(x) = \sqrt{\frac{1}{2}}$ und $Q_1(x) = \sqrt{\frac{3}{2}}P_1(x) = \sqrt{\frac{3}{2}}x$ folgt, dass

$$xQ_0(x) = \frac{1}{\sqrt{3}}Q_1(x) = \beta_1 Q_1(x).$$

Wir nehmen diese Gleichung in unser Matrix-Vektor-System auf und erhalten (für alle $x \in \mathbb{R}$)

$$x\begin{pmatrix} Q_0(x) \\ Q_1(x) \\ \vdots \\ Q_{m-1}(x) \end{pmatrix} = \begin{pmatrix} 0 & \beta_1 & & & \\ \beta_1 & 0 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{m-2} & 0 & \beta_{m-1} \\ & & & \beta_{m-1} & 0 \end{pmatrix} \begin{pmatrix} Q_0(x) \\ Q_1(x) \\ \vdots \\ Q_{m-1}(x) \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \beta_m Q_m(x) \end{pmatrix}$$

Für die Nullstellen $x_1, \ldots, x_m \in (-1,1)$ von Q_m ergibt sich

$$x_i \begin{pmatrix} Q_0(x_i) \\ Q_1(x_i) \\ \vdots \\ Q_{m-1}(x_i) \end{pmatrix} = \begin{pmatrix} 0 & \beta_1 & & & \\ \beta_1 & 0 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{m-2} & 0 & \beta_{m-1} \\ & & & \beta_{m-1} & 0 \end{pmatrix} \begin{pmatrix} Q_0(x_i) \\ Q_1(x_i) \\ \vdots \\ Q_{m-1}(x_i) \end{pmatrix}.$$

Jede Nullstelle x_i, $i = 1, \ldots, m$ von Q_m (bzw. P_m) ist also ein Eigenwert von A und

$$\begin{pmatrix} Q_0(x_i) & \ldots & Q_{m-1}(x_i) \end{pmatrix}^T \in \mathbb{R}^m$$

ist ein dazugehöriger Eigenvektor.

Da die Nullstellen paarweise verschieden sind und A höchstens m verschiedene Eigenwerte haben kann, folgt daraus, dass die Eigenwerte von A genau die Nullstellen der Legendre-Polynome (und damit die Knoten von G_m) sind.

(b) **Gewichte von G_m:** Da m paarweise verschiedene Eigenwerte existieren, müssen die zugehörigen Eigenräume eindimensional sein. Jeder Eigenvektor $v^{(i)}$ zum Eigenwert x_i ist also ein skalares Vielfaches von

$$\begin{pmatrix} Q_0(x_i) & \ldots & Q_{m-1}(x_i) \end{pmatrix}^T \in \mathbb{R}^m,$$

und damit gilt

$$\frac{2\left(v_1^{(i)}\right)^2}{\left\|v^{(i)}\right\|^2} = \frac{2Q_0(x_i)^2}{\sum_{j=0}^{m-1} Q_j(x_i)^2} = \frac{1}{\sum_{j=0}^{m-1} Q_j(x_i)^2}.$$

Sei $w_i = \int_{-1}^{1} l_i(x)\,\mathrm{d}x$ das i-te Gewicht von G_m. Wir müssen zeigen, dass

$$w_i^{-1} = \sum_{j=0}^{m-1} Q_j(x_i)^2. \tag{1.6}$$

Hierzu entwickeln wir $l_i \in \Pi_{m-1}$ in die Basis $\{Q_0, \ldots, Q_{m-1}\}$ des Π_{m-1}

$$l_i = \sum_{n=0}^{m-1} a_n Q_n \quad \text{mit } a_0, \ldots, a_{m-1} \in \mathbb{R}.$$

Für die Koeffizienten gilt

$$a_n = \langle Q_n, l_i \rangle = I[Q_n l_i] = G_m[Q_n l_i] = \sum_{j=1}^{m} w_j Q_n(x_j) l_i(x_j) = w_i Q_n(x_i),$$

wobei wir ausgenutzt haben, dass $Q_n l_j \in \Pi_{2m-2}$ exakt integriert wird.
Da auch $l_i^2 \in \Pi_{2m-2}$ exakt integriert wird, folgt

$$w_i = \sum_{j=1}^{n} w_j l_i(x_j)^2 = G_m[l_i^2] = I[l_i^2] = \left\langle \sum_{n=0}^{m-1} a_n Q_n, \sum_{n=0}^{m-1} a_n Q_n \right\rangle$$
$$= \sum_{n=0}^{m-1} a_n^2 = w_i^2 \sum_{n=0}^{m-1} Q_n(x_i)^2.$$

Dies beweist (1.6) und damit die Behauptung. □

Kapitel 2

Lineare Gleichungssysteme: Direkte Verfahren

In diesem Kapitel betrachten wir das Problem, *lineare Gleichungssysteme* (LGS) zu lösen. Gegeben Koeffizienten $a_{ij} \in \mathbb{C}$, $i = 1,\dots,m$, $j = 1,\dots,n$ und rechten Seiten $b_i \in \mathbb{C}$ versuchen wir also n Unbekannte $x_j \in \mathbb{C}$ so zu bestimmen, dass die m Gleichungen

$$\sum_{j=1}^{n} a_{ij}x_j = a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n = b_i, \quad i = 1,\dots,m$$

erfüllt sind. In Matrix-Vektor-Schreibweise entspricht das dem Problem, einen Vektor $x \in \mathbb{C}^n$ so zu bestimmen, dass

$$Ax = b$$

wobei

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} \in \mathbb{C}^{m\times n},\ b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} \in \mathbb{C}^m,\ x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{C}^n$$

Bemerkung 2.1 (Naheliegender Ansatz)
Aus der Linearen Algebra ist eine explizite Lösungsformel für lineare Gleichungssysteme bekannt, die Cramersche Regel. *Sei $A \in \mathbb{C}^{n\times n}$ invertierbar. Dann ist die Lösung $x = (x_j)_{j=1}^n \in \mathbb{C}^n$ von $Ax = b$ eintragsweise gegeben durch*

$$x_j = \frac{\det(A_j)}{\det(A)},$$

wobei $A_j \in \mathbb{C}^{n\times n}$ diejenige Matrix ist, die sich durch Ersetzen der j-ten Spalte von A durch b ergibt.

Die Berechnung von x erfordert also

- *$n+1$ Determinanten von $n\times n$-Matrizen, und*
- *n Divisionen.*

Mit dem Laplaceschen Entwicklungssatz („Entwicklung nach einer Zeile oder Spalte") erfordert die Deteminante einer einzelnen $n\times n$-Matrix ($n \geq 2$)

- *n Determinanten von $(n-1)\times(n-1)$-Matrizen,*
- *n Multiplikationen und*
- *$n-1$ Additionen,*

also über $n!$ Operationen. Schon für moderat große n ist das ein in der Praxis nicht mehr realisierbarer Rechenaufwand.

Wir werden in diesem Kapitel Verfahren studieren, die ein LGS mit $O(n^3)$ Rechenoperationen lösen.

2.1 Die LR-Zerlegung

2.1.1 Gaußsches Eliminationsverfahren: Ein Beispiel

In diesem Abschnitt ist stets $m = n$. Wir betrachten das einfache Beispiel

$$\begin{aligned} x_1 + \ x_2 + \ x_3 &= 3,\\ x_1 + 2x_2 + 3x_3 &= 4,\\ x_1 + 4x_2 + 9x_3 &= 6, \end{aligned}$$

d.h.

$$\begin{pmatrix} 1 & 1 & 1\\ 1 & 2 & 3\\ 1 & 4 & 9 \end{pmatrix} \begin{pmatrix} x_1\\ x_2\\ x_3 \end{pmatrix} = \begin{pmatrix} 3\\ 4\\ 6 \end{pmatrix}.$$

Zur systematischen Lösung verwenden wir das *Gaußsches Eliminationsverfahren*: Wir eliminieren x_1 aus der zweiten und dritten Gleichung, indem wir jeweils

ein Vielfaches der 1. Gleichung dazu addieren. In diesem Beispiel addieren wir (-1)mal die 1. Gleichung zur 2. und zur 3. Gleichung und erhalten

$$\left.\begin{aligned} x_1 + x_2 + x_3 &= 3, \\ x_2 + 2x_3 &= 1, \\ 3x_2 + 8x_3 &= 3, \end{aligned}\right\} \quad \text{bzw.} \quad \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 3 & 8 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \\ 3 \end{pmatrix}.$$

Nun verwenden wir analog die 2. Gleichung um x_2 aus der 3. Gleichung zu eliminieren, d.h. wir addieren das (-3)fache der 2. Gleichung zur dritten:

$$\left.\begin{aligned} x_1 + x_2 + x_3 &= 3, \\ x_2 + 2x_3 &= 1, \\ 2x_3 &= 0. \end{aligned}\right\} \quad \text{bzw.} \quad \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \\ 0 \end{pmatrix}.$$

Das entstandene Gleichungssytem hat *Stufenform*. Wir können nun x_3 aus der dritten Gleichungen bestimmen, damit dann x_2 aus der zweiten und schließlich x_1 aus der 1. Gleichung bestimmen (sogenannte *Rückwärtssubstitution*):

$$x_3 = 0, \quad x_2 = 1, \quad x_1 = 2.$$

2.1.2 Die LR-Zerlegung ohne Pivotsuche

Das Verfahren lässt sich unabhängig von der konkreten rechten Seite formulieren, indem die Umformungsschritte als Matrixmultiplikationen formuliert werden. Betrachte die allgemeine Matrix

$$A = (a_{ij})_{i,j=1,\dots,n} \in \mathbb{C}^{n\times n}.$$

Zur Elimination der Einträge in der 1. Spalte (ab der 2. Zeile), addieren wir geeignete Vielfache der 1. Zeile zu allen nachfolgenden Zeilen

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -l_{21} & 1 & 0 & \dots & 0 \\ -l_{31} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ -l_{n1} & 0 & 0 & \dots & 1 \end{pmatrix}}_{=:L_1} \underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix}}_{=A} = \underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & a_{32}^{(2)} & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(2)} & a_{n3}^{(2)} & \dots & a_{nn}^{(2)} \end{pmatrix}}_{=:A_2},$$

wobei $l_{i1} := a_{i1}/a_{11}$, $i = 2,\dots,n$ und die Elimination nur funktioniert, wenn das sogenannte *Pivot-Element* a_{11} ungleich Null ist.

Den nächsten Eliminationsschritt formulieren wir analog:

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & -l_{32} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & -l_{n2} & 0 & \dots & 1 \end{pmatrix}}_{=:L_2} \underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & a_{32}^{(2)} & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(2)} & a_{n3}^{(2)} & \dots & a_{nn}^{(2)} \end{pmatrix}}_{=:A_2} = \underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \dots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{n3}^{(3)} & \dots & a_{nn}^{(3)} \end{pmatrix}}_{=:A_3},$$

wobei $l_{i2} := a_{i2}^{(2)} / a_{22}^{(2)}$, $i = 3, \dots, n$ und der Schritt benötigt, dass das Pivot-Element $a_{22}^{(2)}$ nicht Null ist.

Wir fahren so fort und erhalten nach $n-1$ solcher Schritte eine rechte obere Dreiecksmatrix $R := A_n$ mit

$$R = A_n = L_{n-1}A_{n-1} = \ldots = L_{n-1}L_{n-2}\cdots L_1 A,$$

und damit die *LR-Zerlegung* (engl.: *LU-decomposition*) von A

$$A = LR, \qquad \text{wobei } L = L_1^{-1}L_2^{-1}\ldots L_{n-1}^{-1}.$$

Die Inverse der Eliminationsmatrizen und L können explizit angegeben werden:

Lemma 2.2
Hier und im folgenden bezeichnen wir mit

$$I = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}, \quad e_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad e_k^* = \begin{pmatrix} 0 & \dots & 0 & 1 & 0 \dots 0 \end{pmatrix}$$

die $n \times n$-Einheitsmatrix, den k-ten Einheitsvektor und seinen hermitesch Adjungierten.

Gegeben seien Vektoren

$$l_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ l_{k+1,k} \\ \vdots \\ l_{n,k} \end{pmatrix} \in \mathbb{C}^n, \qquad k = 1, \dots, n-1$$

und dazugehörige Eliminationsmatrizen

$$L_k = I - l_k e_k^* = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & \ddots & & & & \vdots \\ & \ddots & 1 & 0 & & \\ & & -l_{k+1,k} & 1 & & \\ \vdots & & \vdots & & \ddots & \vdots \\ 0 & \dots & -l_{n,k} & \dots & & 1 \end{pmatrix}.$$

Dann gilt $L_k^{-1} = I + l_k e_k^*$ *und*

$$L = L_1^{-1} \dots L_{n-1}^{-1} = I + l_1 e_1^* + \dots + l_{n-1} e_{n-1}^* = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ l_{31} & l_{32} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ l_{n1} & l_{n2} & \dots & l_{n,n-1} & 1 \end{pmatrix}.$$

Beweis: Für $j,k = 1,\dots,n$ ist

$$e_j^* l_k = \begin{cases} 0 & \text{für } j \le k, \\ l_{jk} & \text{für } j > k. \end{cases} \tag{2.1}$$

Damit folgt

$$L_k(I + l_k e_k^*) = (I - l_k e_k^*)(I + l_k e_k^*) = I + l_k e_k^* - l_k e_k^* - l_k e_k^* l_k e_k^* = I.$$

$I + l_k e_k^* \in \mathbb{C}^{n\times n}$ ist also eine Rechtsinverse von L_k. Da L_k quadratisch ist, folgt, dass L_k invertierbar ist und dass $L_k^{-1} = I + l_k e_k^*$.

Zusammen mit (2.1) folgt damit auch

$$\begin{aligned} L_1^{-1} L_2^{-1} L_3^{-1} \dots L_{n-1}^{-1} &= (I + l_1 e_1^*)(I + l_2 e_2^*)(I + l_3 e_3^*) \dots (I + l_{n-1} e_{n-1}^*) \\ &= (I + l_1 e_1^* + l_2 e_2^*)(I + l_3 e_3^*) \dots (I + l_{n-1} e_{n-1}^*) \\ &= (I + l_1 e_1^* + l_2 e_2^* + l_3 e_3^*) \dots (I + l_{n-1} e_{n-1}^*) \\ &= \dots = I + l_1 e_1^* + \dots + l_{n-1} e_{n-1}^* \end{aligned}$$

und damit die behauptete Gestalt von L. □

Insgesamt erhalten wir also (falls kein Pivotelement $a_{kk}^{(k)}$ Null ist) die Zerlegung von A in eine linke untere und eine rechte obere Dreiecksmatrix

$$A = LR = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ l_{31} & l_{32} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ l_{n1} & l_{n2} & \dots & l_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \dots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & a_{nn}^{(n)} \end{pmatrix}.$$

Die LR-Zerlegung lässt sich speicherplatzschonend implementieren, indem die Einträge von A direkt durch die entsprechenden Einträge von L und R überschrieben werden. Der folgende Algorithmus transformiert eine gegebene Matrix $A = (a_{ij}) \in \mathbb{C}^{n\times n}$ in die Matrix

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ l_{21} & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} \\ l_{31} & l_{32} & a_{33}^{(3)} & \dots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \dots & a_{nn}^{(n)} \end{pmatrix},$$

aus der sich sowohl L als auch R ablesen lassen.

Algorithm 1 LR-Zerlegung ohne Pivotsuche

```
for j = 1,...,n-1 do
  if a_jj = 0 then
    FEHLER: Pivotelement gleich Null
  end if
  for k = j+1,...,n do
    a_kj := a_kj / a_jj                 % = l_kj
    for i = j+1,...,n do
      a_ki := a_ki - a_kj * a_ji        % = a_ki^(j+1)
    end for
  end for
end for
return L = tril(A,-1) + I, R = triu(A)
```

Den benötigten Rechenaufwand können wir direkt ablesen. Zur Berechnung der LR-Zerlegung einer $n \times n$-Matrix gemäß Algorithmus 1 benötigen wir:

(a) Anzahl Additionen/Subtraktionen:

$$\begin{aligned}\sum_{j=1}^{n-1}\sum_{k=j+1}^{n}\sum_{i=j+1}^{n} 1 &= \sum_{j=1}^{n-1}\sum_{k=j+1}^{n}(n-j) = \sum_{j=1}^{n-1}(n-j)^2 \\ &= \sum_{j=1}^{n-1} j^2 = \frac{n(n-1)(2n-1)}{6} \\ &= \frac{1}{3}n^3 + O(n^2).\end{aligned}$$

(b) Anzahl Multiplikationen/Divisionen:

$$\begin{aligned}\sum_{j=1}^{n-1}\sum_{k=j+1}^{n}\left(1+\sum_{i=j+1}^{n} 1\right) &= \sum_{j=1}^{n-1}\left(n-j+(n-j)^2\right) = \sum_{j=1}^{n-1}(j+j^2) \\ &= \frac{n(n-1)}{2} + \frac{n(n-1)(2n-1)}{6} \\ &= \frac{1}{3}n^3 + O(n^2).\end{aligned}$$

2.1.3 LGS-Lösung mit der LR-Zerlegung

Ist für eine Matrix $A \in \mathbb{C}^{n\times n}$ eine Zerlegung $A = LR$ in eine linke untere und eine rechte obere Dreiecksmatrix mit nicht-verschwindenden Diagonalelementen bekannt, so können wir damit leicht ein lineares Gleichungssystem

$$Ax = b$$

mit rechter Seite $b \in \mathbb{C}^n$ lösen. Zunächst lösen wir

$$\begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ l_{31} & l_{32} & l_{33} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ l_{n1} & l_{n2} & \dots & l_{n,n-1} & l_{nn} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

(hier $l_{11} = \ldots = l_{nn} = 1$) durch *Vorwärtssubstitution*, d.h. wir berechnen beginnend mit y_1 nacheinander $y_1, \ldots, y_n$ durch

$$y_k = \frac{1}{l_{kk}}\left(b_k - \sum_{j=1}^{k-1} l_{k,j} y_j\right), \quad k = 1, \ldots, n.$$

Dann lösen wir

$$\begin{pmatrix} r_{11} & r_{12} & r_{13} & \dots & r_{1n} \\ 0 & r_{22} & r_{23} & \dots & r_{2n} \\ 0 & 0 & r_{33} & \dots & r_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & r_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix}$$

durch *Rückwärtssubstitution*, d.h. wir berechnen beginnend mit x_n die Unbekannten $x_n, x_{n-1}, \dots, x_1$ durch

$$x_k = \frac{1}{r_{kk}} \left(y_k - \sum_{j=k+1}^{n} r_{kj} x_j \right).$$

Verwenden wir die Einheitsvektoren $e_1, \dots, e_n$ als rechte Seiten und lösen die n Gleichungssysteme

$$Ax = e_1, \quad \dots, \quad Ax = e_n,$$

so erhalten wir durch dieses Verfahren auch spaltenweise die inverse Matrix A^{-1}. **Dies ist jedoch im Allgemeinen nicht notwendig!** Vorwärts- und Rückwärtssubstitution benötigen jeweils $\frac{n^2}{2} + O(n)$ Multiplikationen/Divisionen und $\frac{n^2}{2} + O(n)$ Additionen/Subtraktionen. Zum Vergleich: Mittels der Inversen A^{-1} benötigt die Berechnung von $x = A^{-1}b$ durch Matrix-Vektor Multiplikation ebenfalls jeweils $n^2 + O(n)$ Multiplikationen/Divisionen und $n^2 + O(n)$ Additionen/Subtraktionen. Mittels L und R lassen sich lineare Gleichungssysteme also genauso so schnell lösen wie mit der Inversen A^{-1}, es ist nicht nötig (und unnötig aufwändig) A^{-1} zu berechnen!

Bemerkung 2.3 (Implementierung in Matlab)
In Matlab kann die unnötige Berechnung der Inversen A^{-1} vermieden werden, indem zur Berechnung $x = A^{-1}b$ (für eine *rechte Seite b) der Befehl* `x=A\b` *statt* `x=inv(A)*b` *verwendet wird. In der Praxis ist* `x=A\b` *etwa doppelt so schnell wir* `x=inv(A)*b`. *(Matlab verwendet für beides intern die LR-Faktorisierung.)*

Soll $A^{-1}b_j$ für mehrere *rechte Seiten b_1, b_2, b_3 berechnet werden, so ist es ratsam die Inverse zwischenzuspeichern, d.h.*

```
Ainv=inv(A); x1=Ainv*b1; x2=Ainv*b2; x3=Ainv*b3;...
```

ist schneller als

```
x1=A\b1; x2=A\b2; x3=A\b3;...
```

Es ist jedoch noch effizienter (wiederum etwa doppelt so schnell), statt dessen die LR-Faktorisierung zwischenzuspeichern, d.h.

```
[L,U]=lu(A); x1=U\(L\b1); x2=U\(L\b2); x3=U\(L\b3);...
```

2.1.4 Pivotsuche

Algorithmus 1 funktioniert nicht für jede invertierbare Matrix, z.B. ist

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$$

offensichtlich invertierbar ($\det A = -1$) aber das erste Pivotelement ist bereits Null.

Das Problem kann behoben werden, indem in A entweder die Zeilen oder die Spalten vertauscht werden. Ist A die Koeffizientenmatrix eines LGS so entspricht ein Zeilentausch dem Umnummerieren der Gleichungen, und ein Spaltentausch der Umnummerierung der Unbekannten. Durch beides ändert sich die Lösung nicht (zumindest solange wir den Überblick darüber behalten welche Vertauschungen wir vorgenommen haben).

Vertauschen der i-ten und j-ten Zeile von A lässt sich als Matrixmultiplikation von links mit einer Permutationsmatrix schreiben, z. B. ist

$$\begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix}.$$

Allgemein gilt: Ist $P \in \mathbb{R}^{n\times n}$ die Matrix die sich durch Vertauschung der i-ten und j-ten Zeile der Einheitsmatrix ergibt, so ergibt sich PA durch Vertauschung der i-ten und j-ten Zeile in A und AP durch Vertauschung der i-ten und j-ten Spalte in A.

Wir führen nun vor, wie sich Zeilenvertauschungen in den Algorithmus zur LR-Zerlegung integrieren lassen. Im k-ten Schritt der LR-Zerlegung suchen wir im unteren Teil der k-ten Spalte von A_k

$$\begin{pmatrix} a_{kk}^{(k)} \\ a_{k+1,k}^{(k)} \\ \vdots \\ a_{n,k}^{(k)} \end{pmatrix}$$

nach einem von Null verschiedenen Element und vertauschen entsprechend die Zeilen von A_k, d.h. wir schreiben

$$A_{k+1} = L_k P_k A_k$$

wobei P_k die k-te Zeile mit der j-ten Zeile $j \in \{k, \ldots, n\}$ vertauscht. Falls sich durch diese *Spaltenpivotsuche* immer ein von Null verschiedenes Pivotelement finden lässt, dann ergibt sich

$$R = L_{n-1}P_{n-1}L_{n-2}P_{n-2}\ldots L_1P_1A_1.$$

Mit dem folgenden Lemma können wir alle Permutationsmatrizen nach rechts „durchschieben":

Lemma 2.4
Mit der Darstellung $L_k = I - l_k e_k^*$ aus Lemma 2.2 gilt für $i > k$

$$P_i L_k = \tilde{L}_k P_i,$$

wobei $\tilde{L}_k = I - \tilde{l}_k e_k^*$, $\tilde{l}_k = P_i l_k$, d.h. $\tilde{L}_k$ ergibt sich aus L_k durch Vertauschen der Einträge unterhalb der Diagonale entsprechend der durch P_i beschriebenen Permutation.

Beweis: Wegen $i > k$ gilt $e_k^* P_i = e_k^*$ und es folgt, dass

$$P_i L_k = P_i(I - l_k e_k^*) = P_i - P_i l_k e_k^* = (I - (P_i l_k)e_k^*)P_i,$$

also $P_i L_k = \tilde{L}_k P_i$. □

Durch entsprechendes Vertauschen der Elemente in den Eliminationsmatrizen L_k ergibt sich also

$$R = L_{n-1}\tilde{L}_{n-2}\ldots\tilde{L}_1P_{n-1}\ldots P_1A_1$$

und damit die Zerlegung

$$PA = LR,$$

wobei $P = P_{n-1}\ldots P_1$ und $L = \tilde{L}_1^{-1}\ldots\tilde{L}_{n-2}^{-1}L_{n-1}^{-1}$.

Das „Durchschieben" der P_i durch entsprechendes Vertauschen der schon berechneten l_{ij} lässt sich besonders einfach implementieren, indem wir die LR-Zerlegung wie in Algorithmus 1 direkt an den Matrixeinträgen ausführen. Dann werden nämlich durch Vertauschen zweier Zeilen nicht nur die Einträge von A_k sondern auch in korrekter Weise die schon berechneten Einträge der L_k mitvertauscht. Analog lässt sich P konstruieren, indem beginnend mit einer Einheitsmatrix die Zeilenvertauschungen nacheinander durchgeführt werden.

Satz 2.5
Ist A invertierbar, dann findet Algorithmus 2 immer ein von Null verschiedenes Pivotelement und konstruiert die LR-Zerlegung

$$PA = LR.$$

Algorithm 2 LR-Zerlegung mit Spaltenpivotsuche

Setze $P = I$.
for $j = 1, \ldots, n-1$ **do**
  Suche Zeile $m \geq j$ mit $a_{mj} \neq 0$.
  Vertausche m-te und j-te Zeile in A.
  Vertausche m-te und j-te Zeile in P.
  for $k = j+1, \ldots, n$ **do**
    $a_{kj} := a_{kj}/a_{jj}$ $\quad\%\ = l_{kj}$
    for $i = j+1, \ldots, n$ **do**
      $a_{ki} := a_{ki} - a_{kj} * a_{ji}$ $\quad\%\ = a_{ki}^{(j+1)}$
    end for
  end for
end for
return $P, L = \texttt{tril}(A, -1) + I, R = \texttt{triu(A)}$

Beweis: Nach Konstruktion ist nur zu zeigen, dass für eine invertierbare Matrix durch Spaltenpivotsuche immer ein von Null verschiedenes Pivotelement gefunden werden kann. Angenommen das ist nicht der Fall, d.h. für eine der Zwischenmatrizen gilt

$$A_k = \begin{pmatrix} a_{11} & \ldots & a_{1,k-1} & a_{1,k} & a_{1,k+1} & \ldots & a_{1n} \\ 0 & \ddots & \vdots & \vdots & \vdots & & \vdots \\ 0 & & a_{k-1,k-1}^{(k-1)} & a_{k-1,k}^{(k-1)} & a_{k-1,k+1}^{(k-1)} & \ldots & a_{k-1,n}^{(k-1)} \\ 0 & & 0 & 0 & a_{k,k+1}^{(k)} & \ldots & a_{k,n}^{(k)} \\ 0 & & 0 & 0 & a_{k+1,k+1}^{(k)} & \ldots & a_{k+1,n}^{(k)} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & & 0 & 0 & a_{n,k+1}^{(k)} & \ldots & a_{n,n}^{(k)} \end{pmatrix}.$$

Offenbar spannen die ersten k-Vektoren von A_k nur einen $k-1$-dimensionalen Unterraum auf, sind also linear abhängig, d.h. A_k ist nicht invertierbar.

Wir wählen k kleinstmöglich. Dann war in allen vorherigen Schritten die Pivotsuche erfolgreich, d.h. $A_k = L_{k-1}P_{k-1}\ldots L_1P_1A_1$. Die Matrizen L_j und P_j sind offenbar invertierbar, so dass $A_1 = A$ nicht invertierbar sein kann, was aber der Voraussetzung widerspricht. □

Bemerkung 2.6
Mit der LR-Zerlegung lassen sich auch effizient Determinanten berechnen. Findet der Algorithmus kein von Null verschiedenes Pivotelement, so ist $\det(A) = 0$,

ansonsten ist

$$\det(P)\det(A) = \det(PA) = \det(LR) = \det(L)\det(R),$$

also $\det(A) = \frac{\det(L)\det(R)}{\det(P)}$. *Da die Diagonaleinträge der Dreiecksmatrix L eins sind, ist* $\det(L) = 1$. *Für die Dreiecksmatrix R ist* $\det(R)$ *das Produkt der Diagonaleinträge von R. Da jedes Vertauschen zweier Zeilen das Vorzeichen der Determinante umdreht, ist schließlich* $\det(P) = (-1)^k$, *wobei k die Anzahl der durchgeführten Zeilenvertauschungen (also das Signum der dazugehörigen Permutation) ist.*

2.1.5 Einschub: Vektor- und Matrixnormen, Kondition

Wir erinnern an einige wichtige Normen auf dem Vektorraum $\mathbb{C}^n$. Zu einem Vektor $x = (x_j)_{j=1}^n \in \mathbb{C}^n$ definieren wir

- Betragssummennorm: $\|x\|_1 := \sum_{j=1}^{n} |x_j|$,
- Euklidnorm: $\|x\| := \|x\|_2 := \left(\sum_{j=1}^{n} |x_j|^2 \right)^{1/2}$,
- Maximumsnorm: $\|x\|_\infty := \max_{j=1,\ldots,n} |x_j|$.

Aus der Analysis und der Linearen Algebra wissen wir, dass auf $\mathbb{C}^n$ alle Normen äquivalent sind, d.h. für jede Norm $\|\cdot\|_*$ existieren Konstanten $c, C > 0$, so dass

$$c\,\|x\|_* \leq \|x\|_\infty \leq C\,\|x\|_* \quad \forall x \in \mathbb{C}^n,$$

und eine Folge in $\mathbb{C}^n$ konvergiert genau dann, wenn sie komponentweise konvergiert.

Gebräuchliche Normen auf dem Matrizenraum $\mathbb{C}^{m\times n}$ sind
(für $A = (a_{ij})_{i=1,\ldots,m,\, j=1,\ldots,n} \in \mathbb{C}^{m\times n}$)

- Spaltensummennorm: $\|A\|_1 := \max_{j=1,\ldots,n} \sum_{i=1}^{m} |a_{ij}|$,
- Zeilensummennorm: $\|A\|_\infty := \max_{i=1,\ldots,m} \sum_{j=1}^{n} |a_{ij}|$,

- Frobeniusnorm: $\|A\|_F := \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2}$.

Für den Rest dieses Unterabschnitts betrachten wir der Einfachheit halber quadratische Matrizen $A \in \mathbb{C}^{n\times n}$ und eine fest gewählte Vektornorm auf dem $\mathbb{C}^n$. Analoge Definitionen und Aussagen gelten auch für nicht-quadratische Matrizen $A \in \mathbb{C}^{m\times n}$ und für den Fall unterschiedlicher Normen auf $\mathbb{C}^n$ und $\mathbb{C}^m$.

Definition 2.7
Eine Norm $\|\cdot\|_M$ auf $\mathbb{C}^{n\times n}$ heißt

- submultiplikativ, *falls*

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

- verträglich *mit der Vektornorm $\|\cdot\|_*$ auf $\mathbb{C}^n$, falls*

$$\|Ax\|_* \leq \|A\|_M \|x\|_* \quad \forall A \in \mathbb{C}^{n\times n}, x \in \mathbb{C}^n.$$

Definition und Satz 2.8
Sei $\|\cdot\|_*$ eine Norm in $\mathbb{C}^n$. Durch

$$\|A\|_{ind} := \sup_{x\neq 0} \frac{\|Ax\|_*}{\|x\|_*} = \max_{\|x\|_*=1} \|Ax\|_* \quad A \in \mathbb{C}^{n\times n}$$

wird eine Matrixnorm in $\mathbb{C}^{n\times n}$, die sogenannte induzierte Norm *definiert.*

Die induzierte Norm ist submultiplikativ und mit der Ausgangsnorm verträglich. Außerdem gilt für jede andere mit der Ausgangsnorm verträgliche Norm $\|\cdot\|_{vertr}$ in $\mathbb{C}^{n\times n}$.

$$\|A\|_{ind} \leq \|A\|_{vertr} \quad \forall A \in \mathbb{C}^{n\times n}.$$

Beweis: Übungsaufgabe 1 auf Blatt 7. □

Beispiel 2.9
(a) Die Spaltensummennorm ist die durch die Betragssummennorm induzierte Norm.

(b) Die Zeilensummennorm ist die durch die Maximumsnorm induzierte Norm.

(c) Die Frobeniusnorm ist mit der Euklid-Norm verträglich, jedoch nicht die dadurch induzierte Norm.

Beweis: Übungsaufgabe 2 auf Blatt 7. □

Die von der Euklid-Norm induzierte Norm werden wir in Abschnitt 2.3.3 charakterisieren.

Wir betrachten nun die Frage, wie stark sich bei der Lösung eines linearen Gleichungssystems $Ax = b$ Änderungen/Fehler der rechten Seite auf die Lösung auswirken: Es seien $\|\cdot\|_*$ eine Norm auf $\mathbb{C}^n$ und $\|\cdot\|_M$ eine dazu verträgliche Matrixnorm. Bezeichne $0 \neq b \in \mathbb{C}^n$ die exakte rechte Seite und $\Delta b \in \mathbb{C}^n$ eine additive Störung. Für die zugehörigen Lösungen

$$x := A^{-1}b \quad \text{und} \quad x + \Delta x := A^{-1}(b + \Delta b), \text{ d.h. } \Delta x = A^{-1}\Delta b,$$

gilt dann:

$$\frac{\|\Delta x\|_*}{\|x\|_*} = \frac{\left\|A^{-1}\Delta b\right\|_*}{\|x\|_*} \leq \left\|A^{-1}\right\|_M \frac{\|\Delta b\|_*}{\|b\|_*} \frac{\|Ax\|_*}{\|x\|_*} \leq \left\|A^{-1}\right\|_M \|A\|_M \frac{\|\Delta b\|_*}{\|b\|_*}.$$

Mit dem Faktor $\left\|A^{-1}\right\|_M \|A\|_M$ lässt sich also abschätzen, welchen relativen Fehler $\frac{\|\Delta x\|_*}{\|x\|_*}$ ein in der rechten Seite vorhandener relativer Fehler $\frac{\|\Delta b\|_*}{\|b\|_*}$ verursacht.

Definition 2.10
Für eine invertierbare Matrix $A \in \mathbb{C}^{n\times n}$ heißt

$$\operatorname{cond}_M(A) = \left\|A^{-1}\right\|_M \|A\|_M$$

die Kondition *der Matrix A bezüglich der Norm $\|\cdot\|_M$.*

Bemerkung 2.11
Bekanntlich ist eine Matrix $A \in \mathbb{C}^{n\times n}$ genau dann invertierbar, wenn $\det(A) \neq 0$ *gilt. In numerischen Rechnungen ist die Frage nach Singularität oft wenig sinnvoll, da bereits kleinste Rechenfehler eine nicht-invertierbare Matrix invertierbar machen können, z.B. ist für*

$$A_1 := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad A_2 := \begin{pmatrix} 1 & 0 \\ 0 & 10^{-16} \end{pmatrix},$$

A_1 singulär und die nur ganz gering abgeänderte Matrix A_2 regulär. Die Fehlerverstärkung durch Inversion der „fast singulären" Matrix A_2 ist jedoch enorm. Zum Beispiel verursacht für

$$b := \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \Delta b := \begin{pmatrix} 0 \\ 10^{-16} \end{pmatrix}, \quad x := A_2^{-1}b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \Delta x := A_2^{-1}\Delta b = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

ein relativer Fehler von $\frac{\|\Delta b\|_\infty}{\|b\|_\infty} = 10^{-16}$ in der rechten Seite einen Fehler von $\frac{\|\Delta x\|_\infty}{\|x\|_\infty} = 1$ in der Lösung.

Die Determinante ist kein gutes Kriterium dafür, „wie regulär bzw. singulär eine Matrix ist". So besitzt z.B. A_2 die gleiche Determinante $\det A_2 = 10^{-16}$ *wie die offensichtlich ohne relative Fehlerverstärkung einfach zu invertierende Matrix*

$$A_3 = \begin{pmatrix} 10^{-8} & 0 \\ 0 & 10^{-8} \end{pmatrix}, \quad \det(A_3) = 10^{-16}.$$

Ein bessere Kriterium für die Regularität einer Matrix ist deshalb die Kondition:

$$\mathrm{cond}_\infty(A_2) = 10^{16}, \qquad \mathrm{cond}_\infty(A_3) = 1.$$

2.1.6 Pivotsuche und Stabilität

Kleine Pivotelemente können dazu führen, dass sich die Kondition der auftretenden Matrizen verschlechtert. Wir demonstrieren das an dem Beispiel

$$A := \begin{pmatrix} 10^{-3} & -1 \\ 1 & 0 \end{pmatrix}.$$

Es gilt $\mathrm{cond}_F(A) \approx 2$, die Matrix ist also vergleichsweise gut konditioniert, beim Lösen des Gleichungssystems $Ax = b$ erwarten wir, dass sich der relative Fehler höchstens verdoppelt.

Führen wir die LR-Zerlegung ohne Vertauschung durch, so erhalten wir

$$A = LR, \qquad L = \begin{pmatrix} 1 & 0 \\ 1000 & 1 \end{pmatrix}, \qquad R = \begin{pmatrix} 10^{-3} & -1 \\ 0 & 1000 \end{pmatrix}$$

mit $\mathrm{cond}_F(L) \approx 1000000$, $\mathrm{cond}_F(R) \approx 1000000$. L und R sind schlecht konditioniert, nach Vorwärts- und Rückwärtssubstitution ist der relative Fehler (genauer: seine obere Schranke) auf sein 10^{12}-faches angewachsen.

Führen wir hingegen die LR-Zerlegung mit Vertauschung der zwei Zeilen durch, so erhalten wir

$$PA = LR, \quad P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad L = \begin{pmatrix} 1 & 0 \\ 10^{-3} & 1 \end{pmatrix}, \quad R = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

mit $\mathrm{cond}_F(L) \approx \mathrm{cond}_F(R) \approx 2$. L, R und sind also ähnlich gut konditioniert wie die Original-Matrix.

Das Beispiel suggeriert, dass es empfehlenswert ist, möglichst große Pivotelemente zu wählen, um die zur Addition der Gleichungen verwendeten Faktoren

(d.h. die Einträge in L) möglichst klein zu halten. Man wird daher bei der Spaltenpivotsumme nicht irgendein von Null verschiedenes Element wählen, sondern möglichst das Betragsgrößte.

Mehr noch: Die Umnormierung einer der Gleichungen des linearen Gleichungssystem, also ihre Multiplikation mit einer beliebigen (z.B. sehr großen) Zahl, sollte die Pivotsuche nicht beeinflussen. Deshalb hat sich in der Praxis die *relative Spaltenpivotsuche* bewährt, bei der das relativ zur Betragssummennorm der jeweiligen Zeile betragsgrößte Pivotelement gewählt wird.

2.2 Die Cholesky-Zerlegung

Wir erinnnern an ein Ergebnis aus der Linearen Algebra:

Definition und Satz 2.12

(a) Für $A = (a_{ij}) \in \mathbb{C}^{m\times n}$ ist die adjungierte Matrix A^* *diejenige Matrix, die durch Transposition und Komplexkonjugation der Einträge von A entsteht, d.h. der (i,j)-te Eintrag von A^* ist $\overline{a_{ji}}$.*

Wir verwenden die Notation auch für Vektoren $x \in \mathbb{C}^n = \mathbb{C}^{n\times 1}$ und schreiben das komplexe Skalarprodukt zweier Vektoren als

$$x^*y = \sum_{j=1}^{m} \overline{x_j} y_j \quad x,y \in \mathbb{C}^n.$$

Insbesondere ist also $x^*x = \|x\|_2^2$.

Es gelten die Rechenregeln

$$\begin{aligned}(\alpha A + \beta B)^* &= \overline{\alpha} A^* + \overline{\beta} B^* \quad \forall \alpha,\beta \in \mathbb{C},\ A,B \in \mathbb{C}^{m\times n},\\ (AB)^* &= B^*A^* \quad \forall A \in \mathbb{C}^{m\times n}, B \in \mathbb{C}^{n\times p},\\ (A^*)^* &= A \quad \forall A \in \mathbb{C}^{m\times n}.\end{aligned}$$

Ist $A \in \mathbb{C}^{n\times n}$ invertierbar, so ist auch A^* invertierbar und es ist

$$(A^*)^{-1} = (A^{-1})^* =: A^{-*}$$

(b) Eine Matrix $A \in \mathbb{C}^{n\times n}$ heißt hermitesch *(oder* selbst-adjungiert*), falls $A = A^*$.*

(c) Eine hermitesche Matrix $A = A^* \in \mathbb{C}^{n\times n}$ heißt positiv definit, (bzw. positiv semi-definit) falls

$$x^*Ax > 0 \quad (\text{ bzw. } x^*Ax \geq 0) \quad \forall 0 \neq x \in \mathbb{C}^n$$

Ist A hermitesch und positiv definit, so ist A offensichtlich auch injektiv und damit sogar bijektiv.

In diesem Abschnitt leiten wir eine spezielle hermitesche Version der Zerlegung in eine linke untere und eine rechte obere Dreiecksmatrix her:

Definition 2.13
Eine Faktorisierung $A = LL^*$ mit linker unterer Dreiecksmatrix L mit positiven Diagonaleinträgen heißt Cholesky-Zerlegung *von $A \in \mathbb{C}^{n\times n}$.*

Wir erhalten sofort eine notwendige Bedingung für die Existenz einer Cholesky-Zerlegung:

Lemma 2.14
Besitzt $A \in \mathbb{C}^{n\times n}$ eine Cholesky-Zerlegung $A = LL^*$, so ist A hermitesch und positiv definit.

Beweis: Die Hermitizität von A folgt sofort aus

$$A^* = (LL^*)^* = (L^*)^*L^* = LL^* = A$$

Zum Beweis der positiven Definitheit bemerken wir zunächst, dass

$$x^*Ax = x^*LL^*x = (L^*x)^*L^*x = \|L^*x\|^2 .$$

Die Diagonaleinträge von L (und damit auch die von L^*) sind positiv, also insbesondere nicht Null, woraus $\det L^* \neq 0$ folgt. Für alle $x \neq 0$ ist also $L^*x \neq 0$ und damit $\|L^*x\|^2 > 0$. □

Wir werden zeigen, dass diese beiden Bedingungen auch hinreichend für die Existenz der Cholesky-Zerlegung sind. Dazu betrachten wir zunächst eine Blockversion der LR-Zerlegung und versuchen A so zu zerlegen, dass

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = \begin{pmatrix} I & 0 \\ L_{21} & I \end{pmatrix} \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix}.$$

Lemma 2.15
Es sei

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \in \mathbb{C}^{n\times n}$$

zerlegt in $A_{11} \in \mathbb{C}^{p\times p}$, $A_{12} \in \mathbb{C}^{p\times(n-p)}$, $A_{21} \in \mathbb{C}^{(n-p)\times p}$ und $A_{22} \in \mathbb{C}^{(n-p)\times(n-p)}$, wobei $1 \leq p \leq n-1$. Dann gilt:

(a) Ist A_{11} invertierbar, so ist

$$A = \begin{pmatrix} I & 0 \\ A_{21}A_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} \\ 0 & S \end{pmatrix}, \quad S := A_{22} - A_{21}A_{11}^{-1}A_{12}.$$

S heißt das Schur-Komplement *von A_{11} in A.*

(b) Sei A hermitesch und positiv definit. Dann ist $A_{11} \in \mathbb{C}^{p\times p}$ invertierbar und sowohl A_{11} als auch S sind hermitesch und positiv definit.

Beweis: (a) ist trivial.

Für (b) verwenden wir zuerst, dass A hermitesch ist und erhalten

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = A = A^* = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}^* = \begin{pmatrix} A_{11}^* & A_{21}^* \\ A_{12}^* & A_{22}^* \end{pmatrix},$$

also $A_{11} = A_{11}^*$, $A_{22} = A_{22}^*$ und $A_{12}^* = A_{21}$.

Da A positiv definit ist, folgt insbesondere für alle $0 \neq x \in \mathbb{C}^p$

$$0 < \begin{pmatrix} x \\ 0 \end{pmatrix}^* \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} x \\ 0 \end{pmatrix} = \begin{pmatrix} x \\ 0 \end{pmatrix}^* \begin{pmatrix} A_{11}x \\ A_{21}x \end{pmatrix} = x^*A_{11}x.$$

A_{11} ist also hermitesch und positiv definit und damit auch invertierbar.

Für das Schur-Komplement gilt

$$S^* = (A_{22} - A_{21}A_{11}^{-1}A_{12})^* = A_{22}^* - A_{12}^*A_{11}^{-*}A_{21}^* = A_{22} - A_{21}A_{11}^{-1}A_{12} = S.$$

und es bleibt nur noch die positive Definitheit von S zu zeigen.

Hierfür sei $0 \neq y \in \mathbb{C}^{n-p}$. Wir setzen $x := -A_{11}^{-1}A_{12}y \in \mathbb{C}^p$ und erhalten

$$\begin{aligned} 0 < \begin{pmatrix} x \\ y \end{pmatrix}^* \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} &= \begin{pmatrix} x \\ y \end{pmatrix}^* \begin{pmatrix} A_{11}x + A_{12}y \\ A_{21}x + A_{22}y \end{pmatrix} \\ = \begin{pmatrix} x \\ y \end{pmatrix}^* \begin{pmatrix} -A_{12}y + A_{12}y \\ -A_{21}A_{11}^{-1}A_{12}y + A_{22}y \end{pmatrix} &= \begin{pmatrix} x \\ y \end{pmatrix}^* \begin{pmatrix} 0 \\ Sy \end{pmatrix} = y^*Sy, \end{aligned}$$

womit auch die positive Definitheit von S gezeigt ist. □

Satz 2.16
Eine Matrix $A \in \mathbb{C}^{n\times n}$ besitzt genau dann eine Cholesky-Zerlegung, wenn sie hermitesch und positiv definit ist.

Beweis: Die Hinrichtung wurde in Lemma 2.14 gezeigt. Wir zeigen die Rückrichtung durch Induktion über die Dimension n.

Für $n = 1$ sei $A = a_{11} \in \mathbb{C}^{1\times 1}$ hermitesch und positiv definit. Dann ist $a_{11} = \overline{a}_{11}$ also a_{11} reell und aus der positiven Definitheit folgt sofort $a_{11} > 0$. Es ist also $A = LL^*$ mit der 1×1-Matrix $L = \sqrt{a_{11}}$.

Für den Induktionsschritt nehmen wir an, dass die Behauptung für ein $n \in \mathbb{N}$ gelte. Es sei $A \in \mathbb{C}^{(n+1)\times(n+1)}$ hermitesch und positiv definit. Wir betrachten die Blockzerlegung

$$A = \begin{pmatrix} a_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \in \mathbb{C}^{(n+1)\times(n+1)},$$

wobei $a_{11} \in \mathbb{C}$, $A_{12} \in \mathbb{C}^{1\times n}$, $A_{21} \in \mathbb{C}^{n\times 1}$, $A_{22} \in \mathbb{C}^{n\times n}$.

Aus Lemma 2.15 folgt dass a_{11} und das Schur-Komplement $S \in \mathbb{C}^{n\times n}$ hermitesch und positiv definit sind. Aus der Induktionsannahme erhalten wir also, dass S eine Cholesky-Zerlegung $S = L_S L_S^*$ besitzt. Außerdem folgt wie oben $a_{11} > 0$.

Mit

$$l_{11} := \sqrt{a_{11}} > 0 \quad \text{und} \quad L := \begin{pmatrix} l_{11} & 0 \\ A_{21}/l_{11} & L_S \end{pmatrix}$$

folgt (beachte $A_{21}^* = A_{12}$, da A hermitesch)

$$\begin{aligned} LL^* &= \begin{pmatrix} l_{11} & 0 \\ A_{21}/l_{11} & L_S \end{pmatrix} \begin{pmatrix} l_{11} & A_{12}/l_{11} \\ 0 & L_S^* \end{pmatrix} \\ &= \begin{pmatrix} a_{11} & A_{12} \\ A_{21} & A_{21}a_{11}^{-1}A_{12} + L_S L_S^* \end{pmatrix} = \begin{pmatrix} a_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = A. \end{aligned}$$

A besitzt also eine Cholesky-Zerlegung. □

Der Beweis von Satz 2.16 ist konstruktiv und liefert uns einen rekursiven Algorithmus zur Berechnung der Cholesky-Zerlegung von A, siehe Algorithmus 3. Der Beweis von Satz 2.16 zeigt außerdem, dass die Konstruktion genau dann funktioniert, wenn A hermitesch ist und in jedem Dimensionsreduktionsschritt $a_{11} > 0$ ist.

Mit diesem Algorithmus benötigt die Cholesky-Zerlegung einer $n \times n$-Matrix:

- 1 Wurzel (zur Berechnung von l_{11})
- $(n-1)$ Divisionen (zur Berechnung von L_{21})
- jeweils $(n-1)n/2$ Multiplikationen und Additionen (zur Berechnung von S, beachte die Hermitizität!)

Algorithm 3 Cholesky-Zerlegung einer hermiteschen Matrix $A \in \mathbb{C}^{n\times n}$

if $a_{11} \leq 0$ **then**
  FEHLER: Matrix nicht positiv definit.
end if
$l_{11} := \sqrt{a_{11}}$
if $n = 1$ **then**
  return $L = (l_{11})$
else
  Zerlege A gemäß $A = \begin{pmatrix} a_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$.
  $L_{21} := A_{21}/l_{11}$, $S := A_{22} - L_{21}L_{21}^*$
  Berechne Cholesky-Zerlegung $S = L_S L_S^*$
  return $L := \begin{pmatrix} l_{11} & 0 \\ A_{21}/l_{11} & L_S \end{pmatrix}$
end if

- eine $(n-1)\times(n-1)$ Cholesky-Zerlegung

und die Cholesky-Zerlegung einer 1×1-Matrix benötigt eine Wurzeloperation.

Insgesamt benötigt die Cholesky-Zerlegung einer $n\times n$-Matrix also

$$\sum_{k=1}^{n} k^2/2 + O(n^2) = \frac{1}{6}n^3 + O(n^2)$$

Multiplikationen/Divisionen/Wurzeln sowie $\frac{1}{6}n^3 + O(n^2)$ Additionen/Subtraktionen. Der Aufwand ist also nur halb so hoch wie für die LR-Zerlegung. Zusätzlich zum geringeren Rechenaufwand liegt der Vorteil der Cholesky-Zerlegung vor allem in der Tatsache, dass auch beim Auftreten von Rechenfehlern in L das Produkt LL^* stets hermitesch und positiv definit ist. Bei der LR-Zerlegung einer hermitesch und positiv definiten Matrix können Rechenfehler hingegen diese Eigenschaften zerstören.

2.3 Lineare Ausgleichsrechnung

In diesem Abschnitt betrachten wir lineare Gleichungssysteme

$$Ax = b$$

wobei $A \in \mathbb{C}^{m\times n}$ möglicherweise nicht quadratisch und insbesondere möglicherweise nicht invertierbar ist. Im Allgemeinen existiert also keine Lösung oder die Lösung ist nicht eindeutig.

2.3.1 Die Gaußschen Normalengleichungen

Wenn keine exakte Lösung existiert, dann liegt es nahe zu versuchen, das LGS doch wenigstens bestmöglich zu lösen, d.h. wir suchen $x \in \mathbb{C}^n$ so, dass Ax möglichst nahe an b liegt, d.h. $\|Ax-b\|$ möglichst klein ist.

Definition und Satz 2.17
Sei $A \in \mathbb{C}^{m\times n}$ und $b \in \mathbb{C}^m$. Äquivalent sind

(a) $x \in \mathbb{C}^n$ *minimiert das* Residuum $\|Ax-b\|$.

(b) $x \in \mathbb{C}^n$ *löst die* Gaußschen Normalengleichungen $A^*Ax = A^*b$.

Beweis: Für eine komplexe Zahl $\alpha \in \mathbb{C}$ bezeichnen wir ihren Realteil mit

$$\mathrm{Re}(\alpha) := \frac{1}{2}(\alpha+\overline{\alpha}).$$

Es ist

$$\begin{aligned}
&\|Ax-b\|^2 - \|Ay-b\|^2 \\
&= (Ax-b)^*(Ax-b) - (Ay-b)^*(Ay-b) \\
&= x^*A^*Ax \underbrace{-x^*A^*b - b^*Ax}_{=-2\mathrm{Re}(x^*A^*b)} + b^*b - y^*A^*Ay + \underbrace{y^*A^*b + b^*Ay}_{=2\mathrm{Re}(y^*A^*b)} - b^*b \\
&= x^*A^*Ax - 2\mathrm{Re}(x^*A^*b) - y^*A^*Ay + 2\mathrm{Re}(y^*A^*b) \\
&\quad - \|A(y-x)\|^2 + (Ay)^*(Ay) - 2\mathrm{Re}((Ay)^*Ax) + (Ax)^*(Ax) \\
&= 2\mathrm{Re}(x^*A^*(Ax-b)) + 2\mathrm{Re}(y^*A^*(b-Ax)) - \|A(y-x)\|^2 \\
&= 2\mathrm{Re}((x-y)^*(A^*Ax-A^*b)) - \|A(y-x)\|^2 .
\end{aligned}$$

(b)$\Longrightarrow$(a): Löst x die Normalengleichungen, dann ist $A^*Ax - A^*b = 0$ und damit

$$\|Ax-b\|^2 - \|Ay-b\|^2 = -\|A(y-x)\|^2 \le 0,$$

also $\|Ax-b\| \le \|Ay-b\|$ für alle $y \in \mathbb{C}^n$.

(a)$\Longrightarrow$(b): Sei nun $x \in \mathbb{C}^n$ ein Minimierer von $\|Ax-b\|$. Dann ist

$$2\mathrm{Re}((x-y)^*(A^*Ax-A^*b)) - \|A(y-x)\|^2 = \|Ax-b\|^2 - \|Ay-b\|^2 \le 0$$

für alle $y \in \mathbb{C}^n$. Betrachte speziell die Folge $y_k := x - \frac{1}{k}(A^*Ax-A^*b)$, $k \in \mathbb{N}$. Mit ihr erhalten wir

$$\frac{2}{k}\|A^*Ax-A^*b\|^2 - \frac{1}{k^2}\|A(A^*Ax-A^*b)\|^2 \le 0,$$

also

$$\|A^*Ax - A^*b\|^2 \leq \frac{1}{2k} \|A(A^*Ax - A^*b)\|^2 \quad \forall k \in \mathbb{N}$$

und damit $A^*Ax - A^*b = 0$. □

Bemerkung 2.18 (Das orthogonale Komplement)
Aus der Linearen Algebra ist bekannt, dass sich jede Basis eines Untervektorraums $V \subseteq \mathbb{C}^n$ zu einer Basis des $\mathbb{C}^n$ ergänzen lässt und dass sich jede Basis orthogonalisieren lässt (etwa mit dem schon im Rahmen der Gauß-Quadratur angesprochenen Gram-Schmidtschen-Orthogonalisierungsverfahren). Es müssen daher mindestens $n - \dim V$ linear unabhängige Vektoren in dem Raum

$$V^\perp := \{x \in \mathbb{C}^n \,:\, x^*v = 0 \quad \forall v \in V\}$$

existieren.

Da offensichtlich $V \cap V^\perp = \{0\}$ gilt, folgt

$$\mathbb{C}^n = V \oplus V^\perp \quad \textit{und} \quad \dim V^\perp = n - \dim V.$$

Offensichtlich ist auch $V \subseteq (V^\perp)^\perp$ und aus Dimensionsgründen folgt

$$(V^\perp)^\perp = V.$$

$V^\perp$ heißt das orthogonale Komplement *von V.*

Definition und Satz 2.19
Wir bezeichnen mit

$$\mathscr{R}(A) = \{Ax \,:\, x \in \mathbb{C}^n\} \quad \textit{bzw.} \quad \mathscr{N}(A) = \{x \in \mathbb{C}^n \,:\, Ax = 0\}$$

das Bild bzw. den Kern einer Matrix $A \in \mathbb{C}^{m\times n}$. Offenbar sind $\mathscr{R}(A)$ bzw. $\mathscr{N}(A)$ Untervektorräume von $\mathbb{C}^m$ bzw. $\mathbb{C}^n$. Es gilt

(a) $\mathscr{R}(A)^\perp = \mathscr{N}(A^*)$,

(b) $\mathscr{N}(A)^\perp = \mathscr{R}(A^*)$.

Beweis: (a) Es gilt

$$\begin{aligned}\mathscr{R}(A)^\perp &= \{y \in \mathbb{C}^m \,:\, y^*Ax = 0 \quad \forall x \in \mathbb{C}^n\} \\ &= \{y \in \mathbb{C}^m \,:\, (A^*y)^*x = 0 \quad \forall x \in \mathbb{C}^n\} \\ &= \{y \in \mathbb{C}^m \,:\, A^*y = 0\} = \mathscr{N}(A^*),\end{aligned}$$

wobei die letzte Gleicheit durch Wahl von $x := A^*y$ folgt.

(b) Durch Verwendung von (a) folgt

$$\mathscr{N}(A)^\perp = \mathscr{N}((A^*)^*)^\perp = (\mathscr{R}(A^*)^\perp)^\perp = \mathscr{R}(A^*),$$

womit auch (b) bewiesen ist. □

Bemerkung 2.20
Löst $x \in \mathbb{C}^n$ die Normalengleichungen, dann gilt $Ax - b \in \mathscr{N}(A^*) = \mathscr{R}(A)^\perp$. Das Residuum *$Ax - b$ steht also senkrecht auf $\mathscr{R}(A)$, d.h. Ax ist die Orthogonalprojektion von b auf das Bild von A.*

Satz 2.21
Sei $A \in \mathbb{C}^{m\times n}$. Es gilt

$$\mathscr{N}(A^*A) = \mathscr{N}(A) \quad \textit{und} \quad \mathscr{R}(A^*A) = \mathscr{R}(A^*).$$

Beweis: Offenbar gilt $\mathscr{N}(A) \subseteq \mathscr{N}(A^*A)$. Sei nun $x \in \mathscr{N}(A^*A)$. Dann ist

$$0 = x^*(A^*A)x = (Ax)^*Ax = \|Ax\|^2,$$

also $Ax = 0$ und damit $x \in \mathscr{N}(A)$. Damit ist $\mathscr{N}(A^*A) = \mathscr{N}(A)$ gezeigt.

Bei der zweiten Behauptung ist die Inklusion $\mathscr{R}(A^*A) \subseteq \mathscr{R}(A^*)$ trivial. Die Umkehrung folgt aus Dimensionsgründen, denn es ist

$$\begin{aligned}\dim\mathscr{R}(A^*) &= n - \dim\mathscr{R}(A^*)^\perp = n - \dim\mathscr{N}(A)\\ &= n - \dim(\mathscr{N}(A^*A)) = n - \dim\mathscr{R}(A^*A)^\perp = \dim\mathscr{R}(A^*A).\end{aligned}$$ □

Folgerung 2.22
Sei $A \in \mathbb{C}^{m\times n}$ und $b \in \mathbb{C}^m$. Es existiert (mindestens) eine Lösung $x \in \mathbb{C}^n$ der Normalengleichungen

$$A^*Ax = A^*b.$$

x minimiert das Residuum des betrachteten LGS, d.h.

$$\|Ax - b\| = \min_{\xi\in\mathbb{C}^n} \|A\xi - b\|.$$

Die Normalengleichungen sind eindeutig lösbar (d.h. A^*A ist invertierbar), genau dann wenn A injektiv ist.

Beweis: Nach Satz 2.21 ist $A^*b \in \mathscr{R}(A^*A)$, es existiert also ein Vektor $x \in \mathbb{C}^n$ mit $A^*b = A^*Ax$. Die Minimierungseigenschaft haben wir in Satz 2.17 bewiesen. Die Eindeutigkeit der Lösung folgt aus $\mathscr{N}(A^*A) = \mathscr{N}(A)$. □

2.3.2 Singulärwertzerlegung, Moore-Penrose Inverse

Die Normalengleichung kann mehrere Lösungen besitzen. Wir werden nun zeigen, wie wir diejenige Lösung mit minimaler Norm erhalten können.

Aus der linearen Algebra ist bekannt, dass eine hermitesche Matrix stets ein Spektralsystem, d.h. ein Orthonormalsystem aus Eigenvektoren besitzt. Für eine allgemeine Matrix $A \in \mathbb{C}^{m\times n}$ gilt dies nicht, aber die folgende Zerlegung erweist sich als wertvolle Alternative:

Definition und Satz 2.23 (Singulärwertzerlegung)
Sei $A \in \mathbb{C}^{m\times n}$ eine Matrix mit Rang $p := \dim \mathscr{R}(A)$.

(a) A besitzt eine Singulärwertzerlegung*, d.h. es existieren reelle Zahlen (die* Singulärwerte*)*

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p > 0$$

und Orthonormalbasen $(v_1,\ldots,v_n)$ und $(u_1,\ldots,u_m)$ des $\mathbb{C}^n$ und $\mathbb{C}^m$ (die Singulärvektoren*), sodass*

$$\begin{array}{llll} Av_j = \sigma_j u_j & \text{und} & A^*u_j = \sigma_j v_j & \text{für alle } j \in \{1,\ldots,p\}, \\ Av_j = 0 & \text{und} & A^*u_k = 0 & \text{für alle } j,k > p. \end{array}$$

(b) $\sigma_1^2,\ldots,\sigma_p^2$ sind die von Null verschiedenen Eigenwerte von A^*A und $v_1,\ldots,v_p$ sind zugehörige Eigenvektoren.

(c) Es ist

$$A = \sum_{j=1}^{p} \sigma_j u_j v_j^*.$$

(d) Wir ordnen die Singulärvektoren spaltenweise in Matrizen an:

$$U = \begin{pmatrix} u_1 & u_2 & \ldots & u_m \end{pmatrix} \in \mathbb{C}^{m\times m}, \quad V = \begin{pmatrix} v_1 & v_2 & \ldots & v_n \end{pmatrix} \in \mathbb{C}^{n\times n}$$

und bilden eine (möglicherweise unvollständige und nicht-quadratische) Diagonalmatrix aus den Singulärwerten

$$\Sigma := \begin{pmatrix} D_\Sigma & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{C}^{m\times n}, \quad D_\Sigma = \operatorname{diag}(\sigma_1,\ldots,\sigma_p) = \begin{pmatrix} \sigma_1 & 0 & \ldots & 0 \\ 0 & \sigma_2 & \ldots & 0 \\ \vdots & & \ddots & \\ 0 & \ldots & 0 & \sigma_p. \end{pmatrix}$$

Dann sind U und V unitär (d.h. $U^*U = I = UU^*$, $V^*V = I = VV^*$) und es gilt

$$A = U\Sigma V^*.$$

Beweis: (a) Da $A^*A \in \mathbb{C}^{n\times n}$ hermitesch ist, existiert eine Orthonormalbasis aus Eigenvektoren $v_j \in \mathbb{C}^n$, $j = 1,\dots,n$ und zugehörigen reellen Eigenwerten $\lambda_j \in \mathbb{R}$, $j = 1,\dots,n$, von denen

$$\dim \mathscr{N}(A^*A) = \dim \mathscr{N}(A) = n - \dim \mathscr{R}(A) = n - p$$

Eigenwerte gleich Null sind. O.B.d.A. seien dies $\lambda_{p+1},\dots,\lambda_n$. Die restlichen Eigenwerte sind ungleich Null und wegen

$$0 \le \left\|Av_j\right\|^2 = v_j^* A^* A v_j = \lambda_j \left\|v_j\right\|^2 = \lambda_j$$

auch nicht-negativ, also positiv. O.B.d.A. seien sie so sortiert, dass

$$\lambda_1 \ge \ldots \ge \lambda_p > 0 = \lambda_{p+1} = \ldots = \lambda_n.$$

Dann definieren wir

$$\sigma_j := \sqrt{\lambda_j}, \quad u_j := \frac{1}{\sigma_j} A v_j, \quad j = 1,\dots,p.$$

Wegen

$$u_j^* u_k = \frac{1}{\sigma_j \sigma_k} v_j^* A^* A v_k = \frac{\lambda_k}{\sigma_j \sigma_k} v_j^* v_k = \delta_{jk}$$

bilden die u_j eine Orthonormalbasis von $\mathscr{R}(A)$, die wir durch Hinzunahme von $m-p$ Vektoren $u_{p+1},\dots,u_m \in \mathscr{R}(A)^\perp = \mathscr{N}(A^*)$ zu einer Orthonormalbasis des $\mathbb{C}^m$ ergänzen.

Damit erhalten wir die gewünschten Eigenschaften

$$\begin{array}{llll} Av_j = \sigma_j u_j & \text{und} & A^* u_j = \sigma_j v_j & \text{für alle } j \in \{1,\dots,p\}, \\ Av_j = 0 & \text{und} & A^* u_k = 0 & \text{für alle } j,k > p, \end{array}$$

wobei $Av_j = 0$ aus $v_j \in \mathscr{N}(A^*A) = \mathscr{N}(A)$ für $j > p$ folgt.

(b) Besitzt A eine Singulärwertzerlegung, so gilt offenbar $A^*Av_j = \sigma_j^2 v_j$, σ_j^2 ist also ein Eigenwert von A^*A und v_j ein dazugehöriger Eigenvektor.

(c) Sei $x \in \mathbb{C}^n$. Dann kann x in die ONB der v_j entwickelt werden,

$$x = \sum_{j=1}^{n} \alpha_j v_j,$$

und für die Entwicklungskoeffizienten gilt $\alpha_j = v_j^* x$. Es folgt, dass

$$Ax = \sum_{j=1}^{n} (Av_j)(v_j^* x) = \sum_{j=1}^{p} \sigma_j u_j v_j^* x.$$

(d) Die Unitarität von U und V folgt sofort aus der Orthonormalität ihrer Spalten. Alles weitere ist nur die Matrixformulierung von der Aussage in (c). □

Definition 2.24
Mit den Bezeichnungen aus Satz 2.23 sei

$$\Sigma^+ := \begin{pmatrix} D_\Sigma^{-1} & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{C}^{n\times m},$$

$$D_\Sigma^{-1} = \operatorname{diag}(\sigma_1^{-1},\dots,\sigma_p^{-1}) = \begin{pmatrix} 1/\sigma_1 & 0 & \dots & 0 \\ 0 & 1/\sigma_2 & \dots & 0 \\ \vdots & & \ddots & \\ 0 & \dots & 0 & 1/\sigma_p \end{pmatrix}.$$

Die Abbildung

$$A^+ = \sum_{j=1}^{p} \sigma_j^{-1} v_j u_j^* = V\Sigma^+ U^*$$

heißt verallgemeinerte Inverse, Pseudoinverse *oder* Moore-Penrose Inverse *von A.*

Satz 2.25
Sei $A \in \mathbb{C}^{m\times n}$ und $A^+ \in \mathbb{C}^{n\times m}$ die Moore-Penrose Inverse.

(a) $\mathcal{N}(A^+) = \mathcal{N}(A^*) = \mathcal{R}(A)^\perp$, $\mathcal{R}(A^+) = \mathcal{R}(A^*) = \mathcal{N}(A)^\perp$.

(b) A^+ *erfüllt die* Moore-Penrose Axiome[1]

$$AA^+A = A, \qquad (AA^+)^* = AA^+,$$
$$A^+AA^+ = A^+, \qquad (A^+A)^* = A^+A.$$

(c) A^+ *besitzt die Singulärwertzerlegung*

$$A^+u_j = \sigma_j^{-1}v_j, \quad (A^+)^*v_j = \sigma_j^{-1}u_j.$$

Beweis: Übungsaufgabe 9.2 □

Satz 2.26
Sei $A \in \mathbb{C}^{m\times n}$ und $b \in \mathbb{C}^m$. Dann ist $x^+ := A^+b \in \mathbb{C}^n$ die eindeutige Lösung der Normalengleichungen mit minimaler Norm, d.h. es gilt

$$A^*Ax^+ = A^*b \quad \text{und} \quad \left\|x^+\right\| < \|x\|$$

für alle $x \in (A^*A)^{-1}A^*b := \{\xi : A^*A\xi = A^*b\}$ mit $x \neq x^+$.

[1] Man kann zeigen, dass A^+ die einzige Matrix ist, die diese vier Eigenschaften erfüllt.

Beweis: Nach Satz 2.25 ist

$$AA^+b - b \in \mathcal{N}(A^+) = \mathcal{N}(A^*),$$

also $A^*Ax^+ = A^*b$.

Sei nun $x \in \mathbb{C}^n$ eine weitere Lösung der Normalengleichungen, also $A^*Ax = A^*b$ und $x \neq x^+$. Dann ist $x - x^+ \in \mathcal{N}(A^*A) = \mathcal{N}(A)$. Mit $x^+ \in \mathcal{R}(A^+) = \mathcal{N}(A)^\perp$ folgt $(x-x^+)^*x^+ = 0$.

Damit ist

$$\|x\|^2 = \left\|(x-x^+)+x^+\right\|^2 = \left\|x-x^+\right\|^2 + \left\|x^+\right\|^2 > \left\|x^+\right\|^2,$$

womit die Behauptung gezeigt ist. □

Folgerung 2.27
Ist A injektiv, so ist $A^+ = (A^*A)^{-1}A^*$ und zu jedem $b \in \mathbb{C}^m$ ist A^+b die eindeutige Lösung der Normalengleichungen und damit der eindeutige globale Minimierer des Residuums $\|Ax-b\|$.

Ist A bijektiv, so ist $A^+ = A^{-1}$.

Beweis: Ist A injektiv, so ist nach Satz 2.21 auch A^*A injektiv. Da A^*A quadratisch ist, folgt daraus die Bijektivität von A^*A. Es gibt also genau eine Lösung der Normalengleichungen und dies ist nach Satz 2.26 das Bild der Moore-Penrose-Inversen.

Ist A bijektiv, so ist auch A^* bijektiv und $A^+ = (A^*A)^{-1}A^* = A^{-1}(A^*)^{-1}A^* = A^{-1}$.

2.3.3 Spektralnorm und Kondition

Ist $A \in \mathbb{C}^{m\times n}$ injektiv, so ist $\mathcal{N}(A^*A) = \mathcal{N}(A^*) = 0$, d.h. auch $A^*A \in \mathbb{C}^{n\times n}$ ist injektiv und damit (da A^*A quadratisch ist) auch bijektiv. In diesem Fall existiert genau eine Lösung der Normalengleichungen

$$A^*Ax = A^*b$$

und es ist $A^+ = (A^*A)^{-1}A^*$.

Es liegt dann nahe, die Normalengleichungen mittels Cholesky-Zerlegung zu lösen. Wir werden aber in diesem Abschnitt zeigen, dass dies die Kondition des Problems verschlechtert und im nächsten Abschnitt ein besseres Verfahren herleiten (das auch zusätzlich den Vorteil besitzt, dass es sich auch auf nicht injektive A erweitern lässt, siehe Bemerkung 2.38).

Zunächst beantworten wir dazu eine offene Frage aus Abschnitt 2.1.5 und charakterisieren die durch die euklidische Norm induzierte Matrixnorm.

Definition und Satz 2.28
Die durch die euklidische Norm induzierte Matrixnorm ist

$$\|A\| := \|A\|_2 = \begin{cases} \sigma_1 & \text{für } 0 \neq A \in \mathbb{C}^{m\times n}, \\ 0 & \text{für } A = 0. \end{cases}$$

wobei (für $A \neq 0$) σ_1 der größte Singulärwert von A (also die Wurzel aus dem größten Eigenwert von A^*A) ist.

Es gilt $\|A^*\| = \|A\|$. $\|\cdot\|$ heißt Spektralnorm.

Beweis: Für $A = 0$ ist dies offensichtlich richtig. Sei also $A \neq 0$ und

$$(\sigma_j)_{j=1}^p \subset \mathbb{R}, \quad (u_j)_{j=1}^m \subset \mathbb{C}^m, \quad (v_j)_{j=1}^n \subset \mathbb{C}^n \qquad (p := \operatorname{rang}(A))$$

die zugehörige Singulärwertzerlegung. Nach Definition 2.8 müssen wir zeigen, dass

$$\|A\|_2 = \max_{\|x\|=1} \|Ax\| = \sigma_1.$$

Es ist

$$\|v_1\| = 1 \quad \text{und} \quad \|Av_1\| = \|\sigma_1 u_1\| = \sigma_1,$$

also gilt $\|A\|_2 = \max_{\|x\|=1} \|Ax\| \geq \sigma_1$.

Um $\max_{\|x\|=1} \|Ax\| \leq \sigma_1$ zu zeigen, bemerken wir zunächst, dass für jede Linearkombination mit Koeffizienten $\alpha_1, \ldots, \alpha_m \in \mathbb{C}$ aufgrund der Orthonormalität der u_j gilt, dass

$$\left\| \sum_{j=1}^m \alpha_j u_j \right\|^2 = \left(\sum_{j=1}^m \alpha_j u_j \right)^* \left(\sum_{k=1}^m \alpha_k u_k \right) = \sum_{j=1}^m |\alpha_j|^2.$$

Analoges gilt für jede Linearkombinationen der v_j. Damit erhalten wir für jedes $x \in \mathbb{C}^n$ mit $\|x\| = 1$ aus Definition und Satz 2.23(b)

$$\begin{aligned} \|Ax\|^2 &= \left\| \sum_{j=1}^p \sigma_j u_j v_j^* x \right\|^2 = \sum_{j=1}^p |\sigma_j|^2 |v_j^* x|^2 \leq |\sigma_1|^2 \sum_{j=1}^p |v_j^* x|^2 \\ &\leq |\sigma_1|^2 \sum_{j=1}^n |v_j^* x|^2 = |\sigma_1|^2 \left\| \sum_{j=1}^n (v_j^* x) v_j \right\|^2 = |\sigma_1|^2 \|x\|^2 \end{aligned}$$

und damit $\|Ax\| \leq \sigma_1$ für alle $x \in \mathbb{C}^n$ mit $\|x\| = 1$. Insgesamt folgt $\|A\|_2 = \sigma_1$.

$\|A\|_2 = \|A^*\|_2$ folgt daraus, dass offenbar (σ_j, v_j, u_j) eine SWZ von A^* ist. □

Lemma 2.29
Sei $A \in \mathbb{C}^{n\times n}$ invertierbar. Seien $\sigma_1^2 \geq \ldots \geq \sigma_n^2 > 0$ die Eigenwerte von A^*A. Dann ist

$$\operatorname{cond}(A) := \operatorname{cond}_2(A) = \|A\| \left\|A^{-1}\right\| = \sigma_1/\sigma_n.$$

Beweis: Da A invertierbar ist, ist auch A^*A invertierbar, besitzt also n positive Eigenwerte, sodass A tatsächlich n von Null verschiedene Singulärwerte besitzt.

Aus Satz 2.25(c) und Satz 2.28 folgt $\|A^+\| = 1/\sigma_n$ und wir erhalten zusammen mit Folgerung 2.27

$$\operatorname{cond}(A) := \operatorname{cond}_2(A) = \|A\| \left\|A^{-1}\right\| = \|A\| \left\|A^+\right\| = \sigma_1/\sigma_n,$$

womit die Behauptung gezeigt ist. □

Definition 2.30
Für nicht notwendig bijektive Matrizen $A \in \mathbb{C}^{m\times n}$ definieren wir

$$\operatorname{cond}(A) := \|A\| \left\|A^+\right\| = \sigma_1/\sigma_p,$$

wobei wieder σ_p der kleinste von Null verschiedene Singulärwert sei.

Bemerkung 2.31
Wie in Abschnitt 2.1.5 betrachten wir ein lineares Gleichungssystem mit Matrix $A \in \mathbb{C}^{m\times n}$ und einer Störung $\Delta b \in \mathbb{C}^m$ einer exakten rechten Seite $b \in \mathbb{C}^m$. Seien $x := A^+ b$ und $x + \Delta x = A^+(b + \Delta b)$ die zugehörige exakte und gestörte verallgemeinerte Lösung.

Ist A nicht surjektiv, so ist $\mathscr{R}(A)^\perp \neq 0$ und wir können b durch ein beliebig großes $b + b^\perp$ mit $b^\perp \in \mathscr{R}(A)^\perp$ ersetzen, ohne dass sich x und Δx ändern. Es kann daher (für $\Delta x \neq 0$) keine relative Fehlerabschätzung der Form

$$\frac{\|\Delta x\|}{\|x\|} \leq C \frac{\|\Delta b\|}{\|b\|}$$

gelten.

Mit $\tilde{b} := Ax$ gilt aber weiterhin[2]

$$\frac{\|\Delta x\|}{\|x\|} \leq \left\|A^+\right\| \frac{\|\Delta b\|}{\|x\|} \leq \left\|A^+\right\| \|A\| \frac{\|\Delta b\|}{\left\|\tilde{b}\right\|}.$$

In diesem Sinne ist also auch für die verallgemeinerte Lösung eines linearen Gleichungssystems die Kondition ein Maß für die relative Fehlerverstärkung.

[2]Der Vektor $\tilde{b}$ ist (wegen $b - Ax \in \mathscr{R}(A)^\perp$) die Orthogonalprojektion von b auf $\mathscr{R}(A)$ und kann als der erreichbare Anteil in der rechte Seite b interpretiert werden. Durch Ausnutzung von $\Delta x = A^+(\Delta b) = A^+AA^+(\Delta b)$ kann in der Ungleichung auch Δb durch $\Delta\tilde{b} = A\Delta x$ ersetzt werden.

Bemerkung 2.32
Offenbar sind die Singulärwerte von A^*A gegeben durch $\sigma_1^2 \geq \ldots \sigma_p^2 > 0$. Es ist also

$$\operatorname{cond}(A^*A) = \sigma_1^2/\sigma_p^2 = \operatorname{cond}(A)^2,$$

d.h. falls A schlecht konditioniert ist, so sind die Normalengleichungen nochmal deutlich schlechter konditioniert.

2.3.4 Die QR-Zerlegung

Wir stellen in diesem Abschnitt noch eine weitere Matrixzerlegung vor, mit der sich lineare Ausgleichsprobleme konditionsneutral lösen lassen, und die außerdem auch zur Eigenwertberechnung einer quadratischen Matrix verwendet werden kann.

Sei $A \in \mathbb{C}^{m\times n}$. Wir beschränken uns zunächst auf den Fall, dass A injektiv ist (also $n = \operatorname{Rang}(A) \leq m$). Den allgemeinen Fall betrachten wir in Bemerkung 2.38.

Definition 2.33
Sei $A \in \mathbb{C}^{m\times n}$ injektiv. Wir sagen, dass A eine QR-Zerlegung *besitzt, falls eine unitäre Matrix $Q \in \mathbb{C}^{m\times m}$ und eine (verallgemeinerte) rechte obere Dreiecksmatrix*

$$R = \begin{pmatrix} r_{11} & \ldots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \\ 0 & \ldots & 0 \\ \vdots & & \vdots \\ 0 & \ldots & 0 \end{pmatrix} \in \mathbb{C}^{m\times n}$$

mit von Null verschiedenen Diagonalelementen $r_{11}, \ldots, r_{nn}$ existieren, so dass

$$A = QR.$$

Bevor wir uns der Konstruktion solch einer QR-Zerlegung zuwenden, zeigen wir, welche Vorteile sie besitzt. Ist A bijektiv (also $m = n$), so lässt sich

$$Ax = b$$

durch Lösung von $Rx = Q^*b$ durch Rückwärtssubstitution berechnen. Da Q unitär ist, ist

$$\|Qx\|^2 = x^*Q^*Qx = x^*x = \|x\|^2 \quad \forall x \in \mathbb{C}^m,$$

also $\|Q\| = 1$ und genauso folgt $\left\|Q^{-1}\right\| = \|Q^*\| = 1$ und damit $\operatorname{cond}(Q) = 1$. Aus $R^*R = R^*Q^*QR = A^*A$ folgt außerdem $\operatorname{cond}(R) = \operatorname{cond}(A)$.

Die Lösung des LGS durch Verwendung der QR-Zerlegung hat also die gleiche Kondition wie die ursprüngliche Matrix, es findet keine zusätzliche Fehlerverstärkung statt! Für die LR-Zerlegung trifft dies nicht zu, vgl. z.B. unser Beispiel in Abschnitt 2.1.6). Die Multiplikation mit Q oder Q^* ist allerdings mit jeweils $n^2 + O(n)$ Multiplikationen und Additionen teurer als die Vorwärtssubstitution mit L.

Nun betrachten wir das lineare Ausgleichsproblem

$$\|Ax - b\| \to \min!$$

Es sei $c = Q^*b$. Wir bezeichnen die oberen n Zeilen von c und R mit c_1 und R_1, d.h.

$$c = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \in \mathbb{C}^m, \quad c_1 \in \mathbb{C}^n, \qquad R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix} \in \mathbb{C}^{m\times n}, \quad R_1 \in \mathbb{C}^{n\times n},$$

und lösen $R_1 x = c_1$ durch Rückwärtssubstitution. Dann löst x das lineare Ausgleichsproblem, denn das Residuum

$$\|b - Ax\|^2 = \|Q^*(b - Ax)\|^2 = \left\| \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} - \begin{pmatrix} R_1 \\ 0 \end{pmatrix} x \right\|^2 = \|c_1 - R_1 x\|^2 + \|c_2\|^2$$

wird offensichtlich minimiert durch $x = R_1^{-1} c_1$. Die QR-Zerlegung erlaubt also auch die Lösung des linearen Ausgleichsproblems und zwar (im Gegensatz zur Verwendung der Normalengleichungen) ohne Konditionsverschlechtung!

Zur Konstruktion der QR-Zerlegung gehen wir analog der LR-Zerlegung vor, ersetzen jedoch die Eliminationsmatrizen L_i durch unitäre Matrizen:

Definition und Satz 2.34
Eine Matrix der Gestalt

$$P = I - \frac{2}{v^*v} vv^* \in \mathbb{C}^{r\times r}, \quad 0 \neq v \in \mathbb{C}^r$$

heißt Householder-Transformation.

Für jedes $0 \neq v \in \mathbb{C}^r$ ist die zugehörige Householder-Transformation P hermitesch, unitär und erfüllt

$$Pv = -v \quad \textit{und} \quad Pw = w \quad \forall w \in \operatorname{span}(v)^\perp.$$

Beweis: $P = P^*$, $Pv = -v$ und $Pw = w$ für alle $w \in \operatorname{span}(v)^\perp$ folgen direkt aus der Definition von P. Da $\mathbb{C}^r = \operatorname{span}(v) + \operatorname{span}(v)^\perp$ folgt damit auch $I = P^2 = P^*P$ also die Unitarität von P. □

Durch Multiplikation mit den Eliminationsmatrizen L_i konnten wir Spalten der ursprünglichen Matrix in Vielfache des ersten Einheitsvektors überführen. Analog suchen wir nun eine Householder-Trafo, die einen gegebenen Vektor $x \in \mathbb{C}^r$ auf ein Vielfaches von $e_1 \in \mathbb{C}^r$ überführt. D.h. wir suchen $v \in \mathbb{C}^r$ so, dass

$$x - \frac{2}{v^*v} vv^*x = Px \in \operatorname{span}(e_1).$$

Bemerkung 2.35

(a) Im reellen lassen sich Householder-Transformationen als Spiegelungen an der Ebene $\operatorname{span}(v)^\perp$ *interpretieren. Mit einfachen geometrischen Überlegungen (vgl. das in der Vorlesung gemalten Bild) erhalten wir für die gewünschte Spiegelung* $v = x/\|x\| - e_1$ *und tatsächlich ist (mit* $x_1 = e_1^* x$*)*

$$\begin{aligned} x - \frac{2}{v^*v} vv^*x &= x - \frac{2}{\left(\frac{x}{\|x\|} - e_1\right)^* \left(\frac{x}{\|x\|} - e_1\right)} \left(\frac{x}{\|x\|} - e_1\right) \left(\frac{x}{\|x\|} - e_1\right)^* x \\ &= x - \frac{2\left(\|x\| - x_1\right)}{2 - \frac{2\operatorname{Re}(x_1)}{\|x\|}} \left(\frac{x}{\|x\|} - e_1\right) \\ &= x - \frac{\|x\| - x_1}{\|x\| - \operatorname{Re}(x_1)} \left(x - \|x\| e_1\right) \end{aligned}$$

ein Vielfaches von e_1 *falls* $x_1 = \operatorname{Re}(x_1)$*, also* $x_1 \in \mathbb{R}$ *(und der Nenner nicht Null wird).*

(b) Für $x \in \mathbb{C}^r$ *verwenden wir* x/x_1 *statt* x *und erhalten, dass*

$$v = \frac{x/x_1}{\|x/x_1\|} - e_1 = \frac{x}{\|x\|} \frac{|x_1|}{x_1} - e_1$$

das gewünschte tut. Dabei sei $\frac{|x_1|}{x_1}$ *in* $x_1 = 0$ *durch 1 fortgesetzt. Liegt* x *schon nahe an* $\operatorname{span}(e_1)$ *so wird* v *allerdings sehr klein und die Differenzen in obiger Rechnung nähern sich Null, so dass durch Auslöschungseffekte Rechenfehler enorm verstärkt werden.*

(c) Um solche numerische Instabilitäten zu vermeiden nehmen wir in der obigen Überlegung $-e_1$ *anstelle von* e_1 *und erhalten*

$$v = \frac{x}{\|x\|} \frac{|x_1|}{x_1} + e_1$$

(wiederum mit $\frac{|x_1|}{x_1} := 1$ *für* $x_1 = 0$*). Der erste Eintrag von* $\frac{x}{\|x\|} \frac{|x_1|}{x_1}$ *ist stets nicht-negativ, sodass stets ein gewisser Abstand zu* $-e_1$ *besteht.*

Tatsächlich ergibt sich so

$$v^*v = \left(\frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1\right)^* \left(\frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1\right) = 2 + 2\frac{|x_1|}{\|x\|}$$

und damit ist

$$\begin{aligned} x - \frac{2}{v^*v}vv^*x &= x - \frac{2}{v^*v}v(v^*x) = x - \frac{2}{v^*v}(v^*x)v \\ &= x - \frac{2\left(\frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1\right)^* x}{v^*v}\left(\frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1\right) \\ &= x - \frac{2\left(\|x\|\frac{|x_1|}{\overline{x_1}} + x_1\right)}{2 + 2\frac{|x_1|}{\|x\|}}\left(\frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1\right) = -\frac{\left(\|x\|\frac{|x_1|}{\overline{x_1}} + x_1\right)}{1 + \frac{|x_1|}{\|x\|}}e_1 \end{aligned}$$

(wiederum mit der Konvention $\frac{|x_1|}{x_1} := 1$ *für* $x_1 = 0$*).*

Für jedes vorgegebene $x \neq 0$ *ergibt also diese Wahl von* v *eine Householder-Transformation* P*, die* x *auf ein Vielfaches des ersten Einheitsvektors* e_1 *überführt.*

Analog zur LR-Zerlegung bringen wir nun durch Multiplikation mit Householder-Transformationen eine Matrix auf (verallgemeinerte) obere-rechte Dreiecksform: Sei $A \in \mathbb{C}^{m\times n}$ und $x \in \mathbb{C}^m$ die erste Spalte von A. Ist $x \neq 0$ so können wir definieren

$$v := \frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1 \quad \text{und} \quad P_1 := I - \frac{2}{v^*v}vv^* \in \mathbb{C}^{m\times m}$$

und gemäß Bemerkung 2.35 besitzt P_1A die Form

$$P_1A = \begin{pmatrix} r_{11} & * & \dots & * \\ 0 & * & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & * & \dots & * \end{pmatrix} = \begin{pmatrix} r_{11} & * \\ 0 & A_2 \end{pmatrix}.$$

mit $A_2 \in \mathbb{C}^{(m-1)\times(n-1)}$ und $r_{11} \in \mathbb{C}\setminus\{0\}$.

Für den nächsten Schritt sei $x \in \mathbb{C}^{m-1}$ die erste Spalte von A_2. Ist $x \neq 0$, so erhalten wir mit $v := \frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1$ und $P_2' := I - \frac{2}{v^*v}vv^* \in \mathbb{C}^{(m-1)\times(m-1)}$

$$P_2P_1A = \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & P_2' \end{pmatrix}}_{=:P_2}\begin{pmatrix} r_{11} & * \\ 0 & A_2 \end{pmatrix} = \begin{pmatrix} r_{11} & * & * \\ 0 & r_{22} & * \\ 0 & 0 & A_3 \end{pmatrix} \text{ mit } A_3 \in \mathbb{C}^{(m-2)\times(n-2)}.$$

Wir fahren so fort und konstruieren also im $j+1$-ten Schritt die Householder-Matrix $P'_{j+1} \in \mathbb{C}^{(m-j)\times(n-j)}$ wie oben mittels der ersten Spalte des unteren rechten $(m-j)\times(n-j)$-Blockes der Matrix $P_j \dots P_1 A$. Damit definieren wir

$$P_{j+1} := \begin{pmatrix} I & 0 \\ 0 & P'_{j+1} \end{pmatrix}$$

und erhalten die Matrix $P_{j+1} \dots P_1 A$, in der die ersten $j+1$-Spalten rechte obere Dreiecksgestalt besitzen. Schließlich ist

$$P_n \dots P_1 A = \begin{pmatrix} r_{11} & \dots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \\ 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \end{pmatrix} =: R \in \mathbb{C}^{m\times n}$$

und wir erhalten die QR-Zerlegung

$$A = QR,$$

mit $Q = P_1^{-1} \dots P_n^{-1} = P_1^* \dots P_n^*$.

Satz 2.36
Ist $A \in \mathbb{C}^{m\times n}$ injektiv. Dann gilt für den ersten Spaltenvektor $x \in \mathbb{C}^{m-j+1}$ der j-ten Zwischenmatrix $A_j \in \mathbb{C}^{(m-j+1)\times(n-j+1)}$ stets $x \neq 0$, dem beschriebenen Verfahren gelingt also für jede injektive Matrix die Konstruktion einer QR-Zerlegung.

Beweis: Angenommen nicht. Dann existieren unitäre Matrizen $P_1,\dots,P_{j-1}$ (mit $1 \leq j \leq n-1$), so dass

$$P_{j-1} \dots P_1 A = \begin{pmatrix} r_{11} & \dots & * & * & * & \dots & * \\ 0 & \ddots & \vdots & \vdots & \vdots & & \vdots \\ \vdots & & r_{j-1,j-1} & * & * & \dots & * \\ 0 & \dots & 0 & 0 & * & \dots & * \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & 0 & * & \dots & * \end{pmatrix}.$$

Die ersten j Spalten von $P_{j-1} \dots P_1 A$ müssen also linear abhängig sein, d.h. die Matrix $P_{j-1} \dots P_1 A$ wäre nicht injektiv.

Für jedes $\xi \neq 0$ ist aber wegen der Injektivität von A und der Unitarität der Householder-Transformationen

$$0 \neq \|A\xi\| = \|P_1 A\xi\| = \dots = \left\|P_{j-1} \dots P_1 A\xi\right\|,$$

also $P_{j-1} \dots P_1 A$ doch injektiv und damit die Annahme widerlegt. □

Implementierung. Die naive Matrix-Multiplikation $P_1 A$ benötigt $O(m^2 n)$ Multiplikationen und Additionen. Viel schneller ist es, die spezielle Gestalt der Householder-Transformationen auszunutzen. Die Berechnung mittels

$$P_1 A = A - \left(\left(\frac{2}{v^* v} \right) v \right) (v^* A)$$

benötigt

- mn Multiplikationen und $(m-1)n$ Additionen für $v^* A$
- $2m+1$ Multiplikationen und $m-1$ Additionen für $\left(\frac{2}{v^* v}\right) v$
- mn Multiplikationen und mn Additionen für $A - \left(\left(\frac{2}{v^* v}\right) v \right) (v^* A)$,

also nur jeweils $2mn + O(m)$ Multiplikationen und Additionen (beachte $m \geq n$).

Algorithmus 4 zeigt eine Q-freie Implementierung des QR-Algorithmus, indem zu gegebener injektiver Matrix $A \in \mathbb{C}^{m \times n}$ und gegebener rechter Seite $b \in \mathbb{C}^m$ die Matrix R der QR-Zerlegung und $Q^* b$ berechnet werden. Wie zu Beginn des Abschnitts beschrieben lässt sich daraus durch Rückwärtssubstitution die verallgemeinerte Lösung $A^+ b$ des LGS $Ax = b$ berechnen.

Zur Anwendung auf mehrere rechte Seiten sollte auch die Matrix Q aufgestellt werden, z.B. indem in Algorithmus 4 b durch die Einheitsmatrix I ersetzt wird (der Algorithmus transformiert dann I zu Q^*). Alternativ kann auch die in Q enthaltene Information zwischengespeichert werden, indem die erzeugenden Householder-Vektoren zwischengespeichert werden.[3]

Bemerkung 2.37 (Aufwand der QR-Zerlegung)
Im j-ten Schritt des Algorithmus 4 benötigt die Berechnung der Zwischenmatrizen

$$A_j \in \mathbb{C}^{(m-j+1) \times (n-j+1)}$$

jeweils

$$2(m-j+1)(n-j+1) + O(m)$$

Multiplikationen (inkl. Divisionen) und Additionen (inkl. Subtraktionen). Die Berechnung von v, w und b_j benötigt nur $O(m)$ Multiplikationen und Additionen.

[3] Man sieht leicht, dass der erste Eintrag von v immer positiv (sogar ≥ 1) ist und dass Vielfache von v dieselbe Householder-Transformation erzeugen. Wenn wir v noch so normieren, dass der erste Eintrag stets 1 ist, dann müssen nur die restlichen Einträge von v zwischengespeichert werden und das kann wie bei der LR-Zerlegung in den freiwerdenden Einträgen der Matrix A geschehen.

Algorithm 4 Q-freie QR-Zerlegung einer injektiven Matrix $A \in \mathbb{C}^{m\times n}$ zu gegebener rechten Seite $b \in \mathbb{C}^m$

for j=1,...,n **do**

Es bezeichne A_j den unteren rechten $(m-j+1)\times(n-j+1)$-Block von A, x des Blockes erste Spalte und x_1 der Spalte ersten Eintrag.

Der Vektor b_j enthalte die unteren $m-j+1$-Einträge von b.

$v := \frac{x}{\|x\|}\frac{|x_1|}{x_1} + e_1$ % mit $\frac{|x_1|}{x_1} := 1$ *für* $x_1 = 0$

$w := \left(\frac{2}{v^*v}\right) v$

$A_j := A_j - w\left(v^*A_j\right)$

$b_j := b_j - w\left(v^*b_j\right)$

end for

return A ist zu R und b zu Q^*b transformiert.

Wegen $m \geq n$ benötigt Algorithmus 4 insgesamt also jeweils

$$\begin{aligned}
&O(mn) + 2\sum_{j=1}^{n}(m-j+1)(n-j+1)\\
&= O(mn) + 2\sum_{k=1}^{n}(m+k-n)k\\
&= O(mn) + 2\sum_{k=1}^{n}k^2 + 2(m-n)\sum_{k=1}^{n}k\\
&= O(mn) + 2\frac{n^3}{3} + O(n^2) + 2(m-n)\left(\frac{n^2}{2} + O(n)\right)\\
&= mn^2 - \frac{1}{3}n^3 + O(mn).
\end{aligned}$$

Multiplikationen und Additionen.

Für quadratische Gleichungssysteme ($m = n$) ist die QR-Zerlegung im Vergleich zur LR-Zerlegung demnach etwa doppelt so aufwändig (aber dafür konditionsneutral).

Bemerkung 2.38 (QR-Zerlegung für nicht-injektive Matrizen)
Ist A nicht injektiv, so kann im bisher betrachteten Verfahren eine Zwischenmatrix $A_j \in \mathbb{C}^{(m-j+1)\times(n-j+1)}$ den Nullvektor als erste Spalte enthalten. Durch Spaltenvertauschungen (also Multiplikationen von rechts mit den aus Abschnitt 2.1.4 bekannten Permutationsmatrizen) lässt sich das Verfahren noch so lange durchführen bis eine Zwischenmatrix die Nullmatrix ist. So erhält man eine Faktorisierung

$$Q^*AP = \begin{pmatrix} R_1 & R_2 \\ 0 & 0 \end{pmatrix}$$

wobei $P \in \mathbb{C}^{n\times n}$ eine Permutationsmatrix ist, $R_1 \in \mathbb{C}^{p\times p}$ eine rechte obere Dreiecksmatrix und $R_2 \in \mathbb{C}^{p\times(n-p)}$ ist. (Man überlegt sich leicht, dass $p = \operatorname{rang}(A)$ ist.)

Wir zerlegen wie zu Beginn des Abschnitts $c := Q^* b \in \mathbb{C}^m$ in

$$c = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \quad \text{mit } c_1 \in \mathbb{C}^p,\ c_2 \in \mathbb{C}^{m-p}.$$

Analog zerlegen wir $y := P^{-1}x \in \mathbb{C}^n$

$$y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \quad \text{mit } y_1 \in \mathbb{C}^p,\ y_2 \in \mathbb{C}^{n-p}.$$

Dann erhalten wir (beachte $x = Py$) für das lineare Ausgleichsproblem

$$\begin{aligned} \|b - Ax\|^2 = \|Q^*(b - APy)\|^2 &= \left\| \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} - \begin{pmatrix} R_1 & R_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \right\|^2 \\ &= \|c_1 - R_1 y_1 - R_2 y_2\|^2 + \|c_2\|^2 . \end{aligned}$$

Offensichtlich wird dies durch $y_1 := R_1^{-1} c_1$, $y_2 := 0$ minimiert, d.h.

$$x := P \begin{pmatrix} R_1^{-1} c_1 \\ 0 \end{pmatrix}$$

ist eine *Lösung des linearen Ausgleichsproblems.*

In Matlab liefert der Befehl `A\b` *für nicht-quadratische Matrizen diese über die QR-Zerlegung bestimmte Lösung des linearen Ausgleichsproblems.*

Aufgrund der Konstruktion hat die über die QR-Zerlegung bestimmte Lösung x den Vorteil, nur wenige, nämlich höchstens $p = \operatorname{rang}(A)$, von Null verschiedene Einträge zu besitzen. x ist jedoch im Allgemeinen weder die Lösung mit minimaler Norm (also $x \neq A^+ b$) noch die Lösung mit den wenigstmöglichen Nichtnulleinträgen. Zum Beispiel ergibt sich für

$$A := \begin{pmatrix} 2 & 0 & 1 \\ 0 & 2 & 1 \end{pmatrix}, \quad b := \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

über das QR-Verfahren die Lösung $x = (\frac{1}{2}, \frac{1}{2}, 0)^T$. Jedoch ist $(0,0,1)^T$ eine Lösung mit weniger Nichtnulleinträgen und $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})^T$ ist eine Lösung mit kleinerer Norm.

2.4 Ausblick: Eigenwertprobleme

Die QR-Zerlegung kann auch zur Bestimmung der Eigenwerte und -vektoren einer Matrix verwendet werden. Wir skizzieren nur grob die grundsätzliche Idee.

Die Potenzmethode. Wir betrachten zunächst eine Diagonalmatrix

$$A = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_n \end{pmatrix}.$$

Wiederholtes Anwenden von A auf einen Vektor $x = (x_i)_{i=1}^n \in \mathbb{C}^n$ liefert

$$A^k x = \begin{pmatrix} \lambda_1^k x_1 \\ \vdots \\ \lambda_n^k x_n \end{pmatrix} = \sum_{i=1}^n \lambda_i^k x_i e_i,$$

wobei e_i der i-te Einheitsvektor ist. Gibt es genau einen betragsgrößten Eigenwert λ_1 und ist $x_1 \neq 0$, dann wird der dazugehörige Summand schneller wachsen (bzw. langsamer fallen für $|\lambda_1| < 1$) als alle anderen Summanden. Er wird die Summe also immer mehr dominieren. Wir erwarten daher, dass für

$$A^k x \approx \lambda_1^k x_1 e_1 + \text{ langsamer wachsendere Terme.}$$

Analoges sollte für eine bezüglich einer ONB diagonalisierbaren Matrix mit genau einem betragsgrößten Eigenwert λ_1 und zugehörigem Eigenvektor v_1 gelten. Auch hier erwarten wir, dass für jeden Startwert $x = \sum_{j=1}^n (v_j^* x) v_j$ mit $x_1 \neq 0$

$$A^k x \approx \lambda_1^k x_1 v_1 + \text{ langsamer wachsendere Terme.}$$

Wenn wir das Ergebnis in jedem Schritt normalisieren,

$$\tilde{x}^{(k)} := A x^{(k-1)}, \quad x^{(k)} := \frac{\tilde{x}^{(k)}}{\left\| \tilde{x}^{(k)} \right\|},$$

dann sollte sich der am schnellsten wachsende (bzw. der am langsamste fallende) Term durchsetzen, $\tilde{x}^{(k)}$ sollte also immer mehr zu einem Vielfachen von v_1 werden und $(x^{(k)})^* A x^{(k)}$ gegen λ_1 konvergieren (die sogenannte *Potenzmethode*, siehe Algorithmus 5). Für Konvergenzresultate verweisen wir auf [Hanke, §25].

QR-Verfahren zur Eigenwertbestimmung. Wir betrachten nun das Problem, *alle* Eigenwerte und -vektoren einer Matrix zu bestimmen. Dazu können wir die Potenzmethode auf n Startvektoren[4]

$$x_1, \ldots, x_n \in \mathbb{C}^n$$

[4] Achtung: Im Unterschied zum Abschnitt über die Potenzmethode bezeichnen x_1, x_2, u.s.w. jetzt verschiedene Vektoren im $\mathbb{C}^n$ und nicht verschiedene Einträge desselben Vektors.

Algorithm 5 Potenzmethode (von Mises-Verfahren)

Gegeben Matrix $A \in \mathbb{C}^{n\times n}$ und Startvektor $x \in \mathbb{C}^n$.
repeat
$\tilde{x} := Ax$
$\lambda := x^* \tilde{x}$
$x := \tilde{x} / \|\tilde{x}\|$
until STOP
return $\lambda \in \mathbb{C}$, $x \in \mathbb{C}^n$ approximiert betragsgrößten Eigenwert von A und dazugehörigen Eigenvektor.

gleichzeitig anwenden. In Matrixnotation beginnen wir also mit

$$\begin{pmatrix} x_1 & \dots & x_n \end{pmatrix} \in \mathbb{C}^{n\times n}$$

und berechnen im Wesentlichen (d.h. bis auf Normierung)

$$A^k \begin{pmatrix} x_1 & \dots & x_n \end{pmatrix} = \begin{pmatrix} A^k x_1 & \dots & A^k x_n \end{pmatrix}.$$

Im Allgemeinen wird jedoch an alle Startvektoren der Eigenvektor zum betragsgrößten Eigenwert λ_1 beteiligt sein und alle Spalten $A^k x_i$ werden gegen einen Eigenvektor zu λ_1 konvergieren.

Um das zu verhindern, orthonormalisieren wir die Vektoren in jedem Schritt. Wir nehmen also im ersten Schritt nicht

$$\begin{pmatrix} Ax_1 & \dots & Ax_n \end{pmatrix}$$

sondern Vektoren

$$\begin{pmatrix} x_1^{(1)} & \dots & x_n^{(1)} \end{pmatrix}$$

mit den folgenden Eigenschaften:

- Für jedes $k = 1,\dots,n$ sollen die ersten k Vektoren jeweils den gleichen Raum aufspannen, also

$$\operatorname{span}(x_1^{(1)},\dots,x_k^{(1)}) = \operatorname{span}(Ax_1,\dots,Ax_k).$$

- $x_1^{(1)},\dots,x_n^{(1)}$ sollen orthonormiert sind, also

$$(x_j^{(1)})^* x_k^{(1)} = \delta_{jk}.$$

Im nächsten Schritt bilden wir

$$\begin{pmatrix} Ax_1^{(1)} & \dots & Ax_n^{(1)} \end{pmatrix}$$

und orthonormalisieren diese zu Vektoren

$$\begin{pmatrix} x_1^{(2)} & \dots & x_n^{(2)} \end{pmatrix}$$

u.s.w.

Für den ersten Vektor entspricht dieses Verfahren gerade der Potenzmethode. Wir können also erwarten, dass er gegen einen Eigenvektor v_1 zum betragsgrößten Eigenwert λ_1 konvergiert. Beim zweiten Vektor wird bei diesem Verfahren die Potenzmethode angewandt und zusätzlich durch die Orthogonalisierung der Anteil in Richtung des ersten Vektors eliminiert. Wir können also erwarten, dass der zweite Vektor gegen einen Eigenvektor zum größten Eigenwert der Matrix A eingeschränkt auf den Unterraum

$$\operatorname{span}(v_1)^{\perp} = \{v \,:\, v_1^* v = 0\}$$

konvergiert. Dies wird (zumindest wenn die Eigenvektoren senkrecht aufeinander stehen) gerade der betragsmäßig zweitgrößte Eigenwert sein. Enstprechend erwarten wir Konvergenz aller n-Vektoren gegen Eigenvektoren zu allen n Eigenwerten. Die Eigenwerte erhalten wir dann als Diagonaleinträge von

$$\begin{pmatrix} x_1^{(k)} & \dots & x_n^{(k)} \end{pmatrix}^* A \begin{pmatrix} x_1^{(k)} & \dots & x_n^{(k)} \end{pmatrix}.$$

Für die Orthogonalisierung bietet sich das Gram-Schmidtsche Orthogonalisierungsverfahren an. Tatsächlich wird dies aber bereits durch die QR-Zerlegung realisiert. Sei

$$A = QR$$

die QR Zerlegung von A. Da Q unitär ist, sind die Spalten von Q orthonormiert. Da R obere rechte Dreiecksgestalt hat, ergibt sich die k-te Spalte von A durch Kombination von k Spalten von Q, d.h. die jeweils ersten k Spalten von A und Q spannen den selben Raum auf. Q enthält also eine Orthonormalisierung der Spaltenvektoren von A.

Es erscheint natürlich, als Startvektoren die Einheitsvektoren zu wählen, also

$$\begin{pmatrix} x_1^{(0)} & \dots & x_n^{(0)} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Dann ist im ersten Schritt keine Orthonormalisierung nötig und es ergibt sich (durch Anwendung von A) im ersten Schritt immer A, so dass wir auch gleich mit

$$A_0 := \begin{pmatrix} x_1^{(0)} & \dots & x_n^{(0)} \end{pmatrix} = A$$

starten können. Insgesamt erhalten wir so Algorithmus 6.

Algorithm 6 Simultane Potenzmethode

Gegeben Matrix $A \in \mathbb{C}^{n\times n}$. Setze $\tilde{A}_0 = A$.
for $k = 0, 1, \dots$ **do**
  Berechne QR-Zerlegung $\tilde{A}_k = \tilde{Q}_k \tilde{R}_k$
  $\tilde{A}_{k+1} := A\tilde{Q}_k$
end for
return $\tilde{Q}_k$ approximiert spaltenweise Eigenvektoren von A. Die Diagonaleinträge von $\tilde{Q}_k^* A \tilde{Q}_k$ approximieren die Eigenwerte.

In der Literatur wird dieser QR-Algorithmus für Eigenwertprobleme üblicherweise in Form von Algorithmus 7 formuliert. Dabei kann man zeigen, dass in jedem Schritt A_k aus Algorithmus 7 und $\tilde{Q}_k^* A \tilde{Q}_k$ aus Algorithmus 6 übereinstimmen. Für die Berechnung der Eigenvektoren in Algorithmus 7, die effiziente Berechnung der benötigten QR-Zerlegungen und Konvergenzaussagen zum QR-Algorithmus verweisen wir wieder auf [Hanke, §26+27].

Algorithm 7 QR-Verfahren

Gegeben Matrix $A \in \mathbb{C}^{n\times n}$. Setze $A_0 = A$.
for $k = 0, 1, \dots$ **do**
  Berechne QR-Zerlegung $A_k = Q_k R_k$
  $A_{k+1} := R_k Q_k$
end for
return Die Diagonaleinträge von A_k approximieren die Eigenwerte.

Kapitel 3

Iterative Verfahren

Sind die betrachteten linearen Gleichungssysteme zu groß, um sie mit den im letzten Kapitel betrachteten *direkten* Verfahren exakt zu lösen, so bieten sich als Ausweg *iterative* Verfahren an. Dabei wird eine Folge von Approximationen $x^{(k)}$, $k \in \mathbb{N}$, konstruiert, die gegen die exakte Lösung konvergiert. Die Durchführung einer gewissen Anzahl von Iterationsschritten ist typischerweise schneller als die direkte Lösung des LGS, liefert aber oft schon eine hinreichend genaue Approximation der Lösung. Grundlage vieler Iterationsverfahren für lineare und nichtlineare Probleme ist der Banachsche Fixpunktsatz.

3.1 Der Banachsche Fixpunktsatz

Satz 3.1 (Banachscher Fixpunktsatz)
Sei $K \subseteq \mathbb{C}^n$ eine abgeschlossene, nicht-leere Teilmenge und $\|\cdot\|_*$ eine Norm auf dem $\mathbb{C}^n$. Sei Φ eine bezüglich dieser Norm kontrahierende *(möglicherweise nichtlineare) Selbstabbildung, d.h. $\Phi: K \to K$ und es existiert ein $0 \leq q < 1$, sodass*

$$\|\Phi(x) - \Phi(y)\|_* \leq q \|x - y\|_* \quad \forall x, y \in K.$$

Dann besitzt Φ genau einen Fixpunkt $\hat{x}$, d.h. es existiert genau ein $\hat{x} \in K$ mit $\Phi(\hat{x}) = \hat{x}$.

Sei $x^{(0)} \in K$ ein beliebiger Startwert. *Definiere die Folge $(x^{(k)})_{k \in \mathbb{N}_0}$ durch* Fixpunktiteration

$$x^{(k+1)} := \Phi(x^{(k)}) \quad \forall k \in \mathbb{N}_0.$$

Dann konvergiert $x^{(k)}$ gegen $\hat{x}$. Außerdem gelten die Abschätzungen

(a) $\left\|x^{(k)} - \hat{x}\right\|_* \leq q \left\|x^{(k-1)} - \hat{x}\right\|_*$ *(Monotonie),*

(b) $\left\|x^{(k)}-\hat{x}\right\|_* \leq \frac{q^k}{1-q}\left\|x^{(1)}-x^{(0)}\right\|_*$ *(a-priori Schranke),*

(c) $\left\|x^{(k)}-\hat{x}\right\|_* \leq \frac{q}{1-q}\left\|x^{(k)}-x^{(k-1)}\right\|_*$ *(a-posteriori Schranke).*

Beweis: (i) Konvergenz der Fixpunktiteration:

Wir zeigen zuerst, dass die durch Fixpunktiteration definierte Folge für jeden Startwert konvergiert. Sei also $x^{(0)} \in K$ und $(x^{(k)})_{k\in\mathbb{N}_0}$ definiert durch $x^{(k+1)} := \Phi(x^{(k)})$ für alle $k \in \mathbb{N}_0$.

Für jedes $k \in \mathbb{N}$ erhalten wir

$$\begin{aligned}\left\|x^{(k+1)}-x^{(k)}\right\|_* &= \left\|\Phi(x^{(k)})-\Phi(x^{(k-1)})\right\|_* \leq q\left\|x^{(k)}-x^{(k-1)}\right\|_* \\ &\leq \ldots \leq q^k\left\|x^{(1)}-x^{(0)}\right\|_*.\end{aligned}$$

Damit folgt für alle $m,l \in \mathbb{N}$ mit $l > m$

$$\begin{aligned}&\left\|x^{(l)}-x^{(m)}\right\|_* \\ &\leq \left\|x^{(l)}-x^{(l-1)}\right\|_* + \left\|x^{(l-1)}-x^{(l-2)}\right\|_* + \ldots + \left\|x^{(m+1)}-x^{(m)}\right\|_* \\ &\leq \left(q^{l-1}+q^{l-2}+\ldots+q^m\right)\left\|x^{(1)}-x^{(0)}\right\|_* \\ &= q^m\left(q^0+\ldots+q^{l-m-2}+q^{l-m-1}\right)\left\|x^{(1)}-x^{(0)}\right\|_* \\ &\leq \frac{q^m}{1-q}\left\|x^{(1)}-x^{(0)}\right\|_*\end{aligned}$$

und damit (beachte $q < 1$)

$$\lim_{l,m\to\infty}\left\|x^{(l)}-x^{(m)}\right\|_* = 0.$$

$(x^{(k)})_{k\in\mathbb{N}_0}$ ist also ein Cauchy-Folge. Da der Raum $\mathbb{C}^n$ vollständig ist, existiert der Grenzwert

$$\hat{x} := \lim_{k\to\infty} x^{(k)}$$

und, da K abgeschlossen ist, folgt $\hat{x} \in K$. Für jeden Startwert konvergiert also die durch Fixpunktiteration definierte Folge gegen einen Grenzwert $\hat{x} \in K$.

(ii) Existenz und Eindeutigkeit des Fixpunktes:

Nun zeigen wir, dass jeder solche Grenzwert $\hat{x} \in K$ ein Fixpunkt von Φ ist. Aus der Kontraktionseigenschaft folgt insbesondere dass Φ auf K Lipschitzstetig ist und damit

$$\hat{x} = \lim_{k\to\infty} x^{(k)} = \lim_{k\to\infty} \Phi(x^{(k-1)}) = \Phi\left(\lim_{k\to\infty} x^{(k-1)}\right) = \Phi(\hat{x}).$$

Insgesamt ist damit also gezeigt, dass Φ (mindestens) einen Fixpunkt $\hat{x}$ besitzt und die Fixpunktiteration für jeden Startwert gegen einen Fixpunkt konvergiert.

Als nächstes zeigen wir die Eindeutigkeit des Fixpunktes. Seien $\hat{x}, \tilde{x} \in K$ Fixpunkte, also

$$\hat{x} = \Phi(\hat{x}) \quad \text{und} \quad \tilde{x} = \Phi(\tilde{x}).$$

Dann ist

$$\|\hat{x} - \tilde{x}\|_* = \|\Phi(\hat{x}) - \Phi(\tilde{x})\|_* \leq q \|\hat{x} - \tilde{x}\|_*$$

und aus $q < 1$ folgt damit $\|\hat{x} - \tilde{x}\|_* = 0$, d.h. $\hat{x} = \tilde{x}$.

(iii) Beweis der Abschätzungen (a)–(c):

Es verbleibt noch, die drei Abschätzungen (a)–(c) zu zeigen. (a) folgt sofort aus

$$\left\|x^{(k)} - \hat{x}\right\|_* = \left\|\Phi(x^{(k-1)}) - \Phi(\hat{x})\right\|_* \leq q \left\|x^{(k-1)} - \hat{x}\right\|_*.$$

(b) folgt mit der oben gezeigten Abschätzung

$$\left\|\hat{x} - x^{(m)}\right\|_* = \lim_{l\to\infty} \left\|x^{(l)} - x^{(m)}\right\|_* \leq \frac{q^m}{1-q} \left\|x^{(1)} - x^{(0)}\right\|_*.$$

Schließlich folgt mit

$$\begin{aligned} & q \left\|x^{(k)} - x^{(k-1)}\right\|_* \\ & \geq \left\|x^{(k+1)} - x^{(k)}\right\|_* \geq \left\|x^{(k)} - \hat{x}\right\|_* - \left\|x^{(k+1)} - \hat{x}\right\|_* \\ & \geq \left\|x^{(k)} - \hat{x}\right\|_* - q \left\|x^{(k)} - \hat{x}\right\|_* = (1-q) \left\|x^{(k)} - \hat{x}\right\|_* \end{aligned}$$

auch die Abschätzung (c). □

Bemerkung 3.2

(a) Ist Φ bezüglich irgendeiner *Norm $\|\cdot\|_*$ eine Kontraktion, so konvergiert die Fixpunktiteration (wegen der Äquivalenz aller Normen auf dem $\mathbb{C}^n$) bezüglich* jeder *Norm. Die Abschätzungen gelten jedoch im Allgemeinen nur bezüglich $\|\cdot\|_*$.*

(b) Satz 3.1 gilt offenbar genauso für Kontraktionen auf abgeschlossenen Teilmengen des $\mathbb{R}^n$. Mehr noch: im Beweis von Satz 3.1 wurden nur die Metrik-Eigenschaften der Norm und die Vollständigkeit von K verwendet. Tatsächlich gilt (mit dem gleichen Beweis) der Banachsche Fixpunktsatz also für Kontraktionen $\Phi : K \to K$ auf vollständigen metrischen Räumen K. (In diesem Fall gilt aber die Konvergenz im Allgemeinen auch nur in dieser Metrik.)

Beispiel 3.3

(a) Die Nullabbildung $\Phi : \mathbb{R}^n \to \mathbb{R}^n$, $\Phi(x) := 0$ ist offensichtlich eine kontrahierende Selbstabbildung des $\mathbb{R}^n$ und besitzt genau den Nullvektor als Fixpunkt. Für jede Wahl des Startwerts $x^{(0)} \in \mathbb{R}^n$ konvergiert die Fixpunktiteration $x^{(k+1)} := \Phi(x^{(k)}) = 0$ offensichtlich gegen diesen Fixpunkt.

(b) Die Abbildung $\Phi(x) := \cos(x)$ bildet das Intervall $[0,1]$ auf das Intervall $[\cos(1),1] \subseteq [0.5,1]$ ab, ist also insbesondere eine Selbstabbildung

$$\Phi : \ [0,1] \to [0,1].$$

Nach dem Mittelwertsatz der Differentialgleichung gilt

$$\begin{aligned} |\Phi(x) - \Phi(y)| &\leq \sup_{\xi \in [0,1]} |\Phi'(\xi)||x-y| = \sup_{\xi \in [0,1]} |\sin(\xi)||x-y| \\ &= \sin(1)|x-y| \leq 0.85|x-y|. \end{aligned}$$

Der Cosinus besitzt also genau einen Fixpunkt $\hat{x}$ in $[0,1]$ (das folgt auch direkt aus dem Zwischenwertsatz für stetige Funktionen) und für jeden Startwert $x^{(0)} \in [0,1]$ konvergiert

$$x^{(k)} := \underbrace{\cos\cos\ldots\cos}_{\text{k}-\text{mal}} x^{(0)}$$

gegen diesen Fixpunkt $\hat{x} \approx 0.739$.

3.2 Zwei einfache Iterationsverfahren

3.2.1 Jacobi-Verfahren und Gauß-Seidel-Verfahren

Wir verwenden nun die Fixpunktiteration für die Lösung linearer Gleichungssysteme. Es sei $A \in \mathbb{C}^{n\times n}$ und $b \in \mathbb{C}^n$. Es gibt viele verschiedene Möglichkeiten, das lineare Gleichungssystem

$$Ax = b$$

in Fixpunktgestalt zu bringen, z.B. ist $Ax = b$ äquivalent zu

$$x = \Phi(x) := Ax - b + x = (A + I)x - b.$$

Das folgende Lemma zeigt, wann eine solche affin-lineare Fixpunktgleichung die Voraussetzungen des Banachschen Fixpunktsatzes erfüllt.

Lemma 3.4
Sei Φ eine affin-lineare *Abbildung*

$$\Phi : \mathbb{C}^n \to \mathbb{C}^n, \quad \Phi(x) = Tx + c, \quad T \in \mathbb{C}^{n\times n}, c \in \mathbb{C}^n.$$

Sei $\|\cdot\|_*$ eine Norm auf dem $\mathbb{C}^n$ und $\|\cdot\|_M$ eine dazu verträgliche Matrixnorm auf dem $\mathbb{C}^{n\times n}$. Ist $\|T\|_M < 1$, so ist Φ eine Kontraktion (und erfüllt damit die Voraussetzungen des Banachschen Fixpunktsatzes).

Beweis: Es gilt

$$\|\Phi(x) - \Phi(y)\|_* = \|T(x-y)\|_* \le \|T\|_M \|x-y\|_*$$

für alle $x, y \in \mathbb{C}^n$. □

Bemerkung 3.5
Wenn die Fixpunktgleichung durch Äquivalenzumformung aus einem eindeutig lösbaren linearen Gleichungssystem hervorgegangen ist, so ist die Existenz eines eindeutigen Fixpunktes bereits garantiert. In diesem Fall kann man zeigen, dass die Fixpunktiteration in Lemma 3.4 genau dann für alle Startwerte konvergiert, wenn $\rho(T) < 1$, wobei $\rho(T)$ der Spektralradius von T, d.h. der Betrag des betragsgrößten Eigenwertes ist, siehe Aufgabe 10.1.

Unsere erste Idee einer Fixpunktform $x = \Phi(x) := (A + I)x - b$ wird im Allgemeinen nicht die benötigte Kontraktionseigenschaft besitzten, z.B. gilt für den Trivialfall $A = I$ in jeder Norm $\|\Phi(x) - \Phi(y)\|_* = 2\|x - y\|_*$.

Wir betrachten deshalb den allgemeineren Ansatz, $A = M - N$ in zwei Matrizen $M, N \in \mathbb{C}^{n\times n}$ zu zerlegen, wobei M invertierbar sei. Dann ist $b = Ax = (M - N)x$ äquivalent zur Fixpunktgleichung

$$x = \Phi(x) := M^{-1}(Nx + b).$$

Damit die Fixpunktiteration durchführbar ist, sollte dabei M leicht invertierbar sein. Gleichzeitig sollte der in N verbleibende Rest möglichst wenig sein, damit eine Chance auf $\left\|M^{-1}N\right\|_M < 1$ besteht. In diesem Abschnitt betrachten wir die folgenden zwei Möglichkeiten für die Wahl von M:

- **Gesamtschrittverfahren (Jacobi-Verfahren)**: M ist die aus den Diagonaleinträgen von A bestehende Diagonalmatrix.

  Ist $x^{(k)} = \left(x_j^{(k)}\right)_{j=1,\dots,n} \in \mathbb{C}^n$ die k-te Iterierte, so ergibt sich die $(k+1)$-te Iterierte aus

  $$x^{(k+1)} := \Phi(x^{(k)}) = M^{-1}(Nx^{(k)} + b),$$

  mit

  $$M = \begin{pmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ & & \ddots & \\ 0 & 0 & \dots & a_{nn} \end{pmatrix}, \quad N = -\begin{pmatrix} 0 & a_{12} & \dots & a_{1n} \\ a_{21} & 0 & \dots & a_{2n} \\ & & \ddots & \\ a_{n1} & a_{n2} & \dots & 0 \end{pmatrix},$$

  d.h. komponentenweise durch

  $$x_j^{(k+1)} = \frac{1}{a_{jj}} \left(b_j - \sum_{\substack{l=1 \\ l \neq j}}^{n} a_{jl} x_l^{(k)} \right).$$

  Dies entspricht in jedem Schritt also gerade dem Auflösen der j-ten Gleichung von $Ax = b$ nach der j-ten Unbekannten x_j und der Verwendung der Näherungen des letzten Schrittes für die benötigten anderen Unbekannten.

- **Einzelschrittverfahren (Gauß-Seidel-Verfahren)**: M ist die aus dem linken unteren Dreiecksanteil von A bestehende linke untere Dreiecksmatrix.

  Ist $x^{(k)} = \left(x_j^{(k)}\right)_{j=1,\dots,n} \in \mathbb{C}^n$ die k-te Iterierte, so ergibt sich die $(k+1)$-te Iterierte aus

  $$x^{(k+1)} := \Phi(x^{(k)}) = M^{-1}(Nx^{(k)} + b),$$

  mit

  $$M = \begin{pmatrix} a_{11} & 0 & \dots & 0 \\ a_{21} & a_{22} & \dots & 0 \\ & & \ddots & \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}, \quad N = -\begin{pmatrix} 0 & a_{12} & \dots & a_{1n} \\ 0 & 0 & \dots & a_{2n} \\ & & \ddots & \\ 0 & 0 & \dots & 0 \end{pmatrix},$$

  also komponentenweise durch

  $$x_j^{(k+1)} = \frac{1}{a_{jj}} \left(b_j - \sum_{l=1}^{j-1} a_{jl} x_l^{(k+1)} - \sum_{l=j+1}^{n} a_{jl} x_l^{(k)} \right). \tag{3.1}$$

  Dies entspricht also in jedem Schritt wieder gerade dem Auflösen der j-ten Gleichung von $Ax = b$ nach der j-ten Unbekannten x_j. Im Unterschied zum Gesamtschrittverfahren werden aber für die anderen Unbekannten aber schon die in diesem Schritt bereits erhaltenen neuen Näherungen verwendet.

3.2.2 Konvergenz der Verfahren

Definition 3.6
Eine Matrix $A \in \mathbb{C}^{n\times n}$ heißt (zeilenweise) strikt-diagonaldominant, *falls für jede Zeile $j = 1,\dots,n$ gilt*

$$|a_{jj}| > \sum_{\substack{l=1\\ l\neq j}}^{n} |a_{jl}|.$$

Satz 3.7
Ist $A \in \mathbb{C}^{n\times n}$ (zeilenweise) strikt-diagonaldominant, dann ist A invertierbar und für jeden Startvektor $x^{(0)} \in \mathbb{C}^n$ konvergieren sowohl das Gesamt- als auch das Einzelschrittverfahren gegen die eindeutige Lösung von $Ax = b$.

Beweis: Da A (zeilenweise) strikt diagonaldominant ist, sind insbesondere die Diagonalelemente von Null verschieden und damit ist (sowohl für das Gesamt- als auch das Einzelschrittverfahren) M invertierbar. $Ax = b$ ist also äquivalent zu der Fixpunktgleichung $x = \Phi(x) := M^{-1}(Nx + b)$ und die Invertierbarkeit folgt aus der Existenz eines eindeutigen Fixpunktes. Wegen Lemma 3.4 genügt es also zu zeigen, dass eine mit einer Vektornorm verträgliche Matrixnorm $\|\cdot\|_M$ existiert, so dass $\left\|M^{-1}N\right\|_M < 1$.

Gesamtschrittverfahren: Für die Zeilensummennorm von

$$T := M^{-1}N = -\begin{pmatrix} a_{11}^{-1} & 0 & \dots & 0 \\ 0 & a_{22}^{-1} & \dots & 0 \\ & & \ddots & \\ 0 & 0 & \dots & a_{nn}^{-1} \end{pmatrix} \begin{pmatrix} 0 & a_{12} & \dots & a_{1n} \\ a_{21} & 0 & \dots & a_{2n} \\ & & \ddots & \\ a_{n1} & a_{n2} & \dots & 0 \end{pmatrix}$$
$$= \begin{pmatrix} 0 & \frac{a_{12}}{a_{11}} & \dots & \frac{a_{1n}}{a_{11}} \\ \frac{a_{21}}{a_{22}} & 0 & \dots & \frac{a_{2n}}{a_{22}} \\ & & \ddots & \\ \frac{a_{n1}}{a_{nn}} & \frac{a_{n2}}{a_{nn}} & \dots & 0 \end{pmatrix}$$

gilt wegen der strikten Diagonaldominanz

$$\|T\|_\infty = \max_{j=1,\dots,n} \sum_{\substack{l=1\\ l\neq j}} \frac{|a_{jl}|}{|a_{jj}|} < 1.$$

Die Aussage ist also für das Gesamtschrittverfahren bewiesen.

Einzelschrittverfahren: Wir werden wiederum zeigen, dass $\|T\|_\infty < 1$. Dazu verwenden wir (siehe Beispiel 2.9)

$$\|T\|_\infty = \max_{\|x\|_\infty = 1} \|Tx\|_\infty$$

Sei dafür $x \in \mathbb{C}^n$ mit $\|x\|_\infty = 1$. Wir müssen zeigen, dass für alle Komponenten von $y := Tx$ gilt $|y_j| < 1$. Für diese gilt gemäß (3.1):

$$y_j = \frac{1}{a_{jj}} \left(-\sum_{l=1}^{j-1} a_{jl} y_l - \sum_{l=j+1}^{n} a_{jl} x_l \right)$$

und damit

$$\begin{aligned} |y_j| &\leq \frac{1}{|a_{jj}|} \left(\sum_{l=1}^{j-1} |a_{jl}|\,|y_l| + \sum_{l=j+1}^{n} |a_{jl}|\,|x_l| \right) \leq \sum_{\substack{l=1 \\ l \neq j}} \frac{|a_{jl}|}{|a_{jj}|} \max_{l=1,\ldots,j-1} \{|y_l|, 1\} \\ &< \max_{l=1,\ldots,j-1} \{|y_l|, 1\}. \end{aligned}$$

Mit trivialer Induktion folgt daraus $|y_j| < 1$ für alle $j = 1, \ldots, n$ und damit die Aussage für das Einzelschrittverfahren. □

Bemerkung 3.8
Es lassen sich sowohl Beispiele konstruieren, in denen das Einzelschrittverfahren schneller als das Gesamtschrittverfahren konvergiert als auch umgekehrt. Intuitiv wird man dem Einzelschritt-Verfahren aber eher eine schnellere Konvergenz zutrauen, da es in jedem Schritt schon die bereits berechneten wohl besseren Näherungen des aktuellen Schrittes verwendet statt der wohl schlechteren Näherungen des letzten Schrittes. Tatsächlich wird deshalb in der Praxis meist das Einzelschritt-Verfahren bevorzugt.

Kapitel 4

Nichtlineare Gleichungen

In diesem Kapitel untersuchen wir Lösungsalgorithmen für nicht-lineare Gleichungen. Dabei beschränken wir uns auf reelle Probleme. Komplexe Problemstellungen lassen sich mit der Identifikation $\mathbb{C}^n = \mathbb{R}^{2n}$ darauf zurückführen.

4.1 Fixpunktiterationen

4.1.1 Konvergenz von Fixpunktverfahren

In Beispiel 3.3 haben wir (durch Abschätzung der Ableitung des Cosinus) gesehen, dass der Cosinus eine kontrahierende Selbstabbildung auf $[0,1]$ ist und deshalb die nichlineare Fixpunktiteration

$$x^{(k+1)} := \Phi(x^{(k)}) := \cos x^{(k)}$$

für jeden Startwert $x^{(0)} \in [0,1]$ gegen den Fixpunkt

$$0.739 \approx \hat{x} = \cos(\hat{x}) \in [0,1]$$

konvergiert.

Das Beispiel lässt sich auf allgemeine, mehr-dimensionale, nicht-lineare Funktionen übertragen. Dazu erinnern wir an den wichtigen Mittelwertsatz der Differentialrechnung im $\mathbb{R}^n$:

Lemma 4.1 (Mittelwertsatz der Differentialrechnung)
$\Phi : \mathscr{D}(\Phi) \to \mathbb{R}^m$ *sei eine auf einer offenen Menge* $\mathscr{D}(\Phi) \subseteq \mathbb{R}^n$ *definierte, stetig differenzierbare Funktion.*

Seien $x, y \in \mathbb{R}^n$ Punkte, deren Verbindungsstrecke in $\mathscr{D}(\Phi)$ liegt, also

$$x + t(y-x) \in \mathscr{D}(\Phi) \quad \text{für alle } t \in [0,1].$$

Dann gilt

$$\Phi(y) - \Phi(x) = \int_0^1 \Phi'(x + t(y-x))(y-x)\,\mathrm{d}t,$$

wobei $\Phi' : \mathscr{D}(\Phi) \to \mathbb{R}^{m\times n}$ die totale Ableitung (Jacobi-Matrix) von Φ bezeichnet.

Beweis: Betrachte die Funktion

$$f : [0,1] \to \mathbb{R}^n, \quad f(t) := \Phi(x + t(y-x)).$$

f ist stetig differenzierbar und die Ableitung $f' : [0,1] \to \mathbb{R}^m$ ist

$$f'(t) = \Phi'(x + t(y-x))(y-x).$$

Mit

$$\begin{aligned}\Phi(y) - \Phi(x) = f(1) - f(0) &= \int_0^1 f'(t)\,\mathrm{d}t \\ &= \int_0^1 \Phi'(x + t(y-x))(y-x)\,\mathrm{d}t\end{aligned}$$

folgt daher die Behauptung. □

Definition 4.2
Eine Menge $K \subseteq \mathbb{R}^n$ heißt konvex*, falls für jedes Paar von Punkten $x, y \in K$ die Verbindungsstrecke in K liegt, also*

$$x + t(y-x) \in K \quad \text{für alle } x, y \in K,\ t \in [0,1].$$

Wir zeigen nun unter welchen Voraussetzungen, die Fixpunktiteration global oder lokal konvergiert. Für die folgenden beiden Sätze sei stets

$$\Phi : \mathscr{D}(\Phi) \to \mathbb{R}^n$$

eine auf einer offenen Menge $\mathscr{D}(\Phi) \subseteq \mathbb{R}^n$ definierte, stetig differenzierbare Funktion, $\|\cdot\|_*$ sei eine Norm auf dem $\mathbb{R}^n$, und $\|\cdot\|_M$ sei eine dazu verträgliche Matrixnorm auf dem $\mathbb{R}^{n\times n}$.

Satz 4.3
$K \subseteq \mathscr{D}(\Phi)$ sei abgeschlossen und konvex. Außerdem gelte $\Phi(K) \subseteq K$. Existiert ein $0 \leq q < 1$ mit

$$\left\|\Phi'(x)\right\|_M \leq q \quad \forall x \in K,$$

so ist Φ eine kontrahierende Selbstabbildung auf K. Insbesondere besitzt dann Φ genau einen Fixpunkt $\hat{x}$ in K und die Fixpunktiteration

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

konvergiert für jeden Startwert $x^{(0)} \in K$ gegen $\hat{x}$.

Beweis: Seien $x, y \in K$. Aus dem Mittelwertsatz der Differentialgleichung Lemma 4.1 folgt

$$\begin{aligned}
\|\Phi(y) - \Phi(x)\|_* &= \left\| \int_0^1 \Phi'(x + t(y-x))(y-x)\,\mathrm{d}t \right\|_* \\
&\leq \int_0^1 \left\|\Phi'(x + t(y-x))(y-x)\right\|_* \mathrm{d}t \\
&\leq \int_0^1 \left\|\Phi'(x + t(y-x))\right\|_M \|y-x\|_* \,\mathrm{d}t \\
&\leq q\,\|y-x\|_*
\end{aligned}$$

und damit die Behauptung. □

Beispiel 4.4
Betrachte

$$\Phi\colon\ \mathbb{R}^2 \to \mathbb{R}^2, \quad \Phi(x) = \begin{pmatrix} \Phi_1(x_1,x_2) \\ \Phi_2(x_1,x_2) \end{pmatrix} := \begin{pmatrix} \cos x_2 - \frac{x_1}{10} \\ \cos x_1 \end{pmatrix}.$$

Φ ist eine Selbstabbildung auf der offensichtlich abgeschlossenen und konvexen Menge $[0,1]^2$, denn für alle $x_1, x_2 \in [0,1]^2$ gilt

$$\begin{aligned}
1 \geq \cos x_2 - \frac{x_1}{10} &\geq \cos(1) - \frac{x_1}{10} \geq 0 \\
1 \geq \cos x_1 &\geq 0,
\end{aligned}$$

also $\Phi([0,1]^2) \subseteq [0,1]^2$.

Außerdem ist Φ eine Kontraktion bzgl. der $\|\cdot\|_\infty$-Norm, denn für alle Vektoren $x = (x_1,x_2)^T \in [0,1]^2$ ist

$$\begin{aligned}
\left\|\Phi'(x)\right\|_\infty &= \left\| \begin{pmatrix} \frac{\partial \Phi_1}{\partial x_1} & \frac{\partial \Phi_1}{\partial x_2} \\ \frac{\partial \Phi_2}{\partial x_1} & \frac{\partial \Phi_2}{\partial x_2} \end{pmatrix} \right\|_\infty = \left\| \begin{pmatrix} -\frac{1}{10} & -\sin x_2 \\ -\sin x_1 & 0 \end{pmatrix} \right\|_\infty \\
&= \max\{1/10 + \sin(x_2), \sin(x_1)\} \\
&\leq 1/10 + \sin(1) \leq 0.95 < 1.
\end{aligned}$$

Nach Satz 4.3 existiert also genau ein Fixpunkt $\hat{x} = (\hat{x}_1, \hat{x}_2)^T$ *in* $[0,1]^2$ *und die Fixpunktiteration*

$$\begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \end{pmatrix} = x^{(k+1)} = \Phi(x^{(k)}) = \begin{pmatrix} \cos x_2^{(k)} - x_1^{(k)}/10 \\ \cos x_1^{(k)} \end{pmatrix}$$

konvergiert für jeden Startwert $x^{(0)} = (x_1^{(0)}, x_2^{(0)})^T \in [0,1]^2$ *gegen* $\hat{x}$.

Satz 4.5
Ist $\hat{x} \in \mathscr{D}(\Phi)$ *ein Fixpunkt von* Φ *und gilt*

$$\left\|\Phi'(\hat{x})\right\|_M < 1,$$

dann existiert eine abgeschlossene Kugel $K = \overline{B_\varepsilon(\hat{x})} \subseteq \mathscr{D}(\Phi)$ *um* $\hat{x}$, $\varepsilon > 0$, *sodass* Φ *auf* K *eine kontrahierende Selbstabbildung ist. Insbesondere ist also* $\hat{x}$ *der einzige Fixpunkt in* K *und die Fixpunktiteration*

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

konvergiert für jeden Startwert $x^{(0)} \in K$ *gegen* $\hat{x}$.

Beweis: Sei $q > 0$ so, dass $\|\Phi'(\hat{x})\|_M < q < 1$. Da $\mathscr{D}(\Phi)$ offen ist und Φ' stetig, existiert ein $\varepsilon > 0$, so dass für die abgeschlossene Kugel $K := \overline{B_\varepsilon(\hat{x})}$ gilt

$$K \subseteq \mathscr{D}(\Phi) \quad \text{und} \quad \left\|\Phi'(x)\right\|_M < q \quad \text{für alle } x \in K.$$

Aus dem Mittelwertsatz der Differentialgleichung erhalten wir wieder

$$\|\Phi(x) - \hat{x}\|_* = \|\Phi(x) - \Phi(\hat{x})\|_* \leq q \|x - \hat{x}\|_*$$

und damit $\Phi(x) \in K$ für alle $x \in K$. Damit folgt die Behauptung aus Satz 4.3. □

Bemerkung 4.6
Satz 4.5 garantiert, dass das iterative Verfahren (hier: die Fixpunktiteration) den gesuchten Punkt $\hat{x}$ *(hier: den Fixpunkt) findet, wenn der Startwert hinreichend nahe am gesuchten Punkt liegt. Diese Eigenschaft nennt man auch* lokale Konvergenz.

Im Gegensatz dazu spricht man von globaler Konvergenz, *wenn das Verfahren für alle Startwerte* $x^{(0)} \in \mathbb{R}^n$ *(oder zumindest für alle Startwerte in einer a-priori bekannten Menge) gilt.*

Beispiel 4.7 (Das Heron-Verfahren)
Wir betrachten die Berechnung der Quadratwurzel einer positiven Zahl $a > 0$. Offensichtlich ist für $x > 0$ die Gleichung $x^2 = a$ äquivalent zu der Fixpunktgleichung

$$x = \frac{1}{2}\left(x + \frac{a}{x}\right) =: \Phi(x)$$

(siehe auch Beispiel 4.10 und Abschnitt **??** *zur Motivation gerade dieser Fixpunktform).*

Es ist $\Phi'(x) = \frac{1}{2}\left(1 - \frac{a}{x^2}\right)$. Am Fixpunkt $\hat{x} := \sqrt{a}$ gilt also $\Phi'(\hat{x}) = 0$. Nach Satz 4.5 konvergiert also die Fixpunktiteration (das sogenannte Heron-Verfahren*)*

$$x^{(k+1)} = \frac{1}{2}\left(x^{(k)} + \frac{a}{x^{(k)}}\right) \tag{4.1}$$

lokal gegen $\sqrt{a}$.

Man kann zeigen (siehe Übungsaufgabe 13.3), dass die Fixpunktiteration tatsächlich für jeden Startwert $x^{(0)} \in (0, \infty)$ gegen $\sqrt{a}$ konvergiert.

4.1.2 Konvergenzgeschwindigkeit

Sind die Voraussetzungen des Banachschen Fixpunktsatzes mit einer Kontraktionskonstante $q < 1$ erfüllt, so gilt für den Fehler der k-ten Iterierten der Fixpunktiteration (nach der a-priori Abschätzung aus Satz 3.1)

$$\left\|x^{(k)} - \hat{x}\right\| \leq \frac{q^k}{1-q}\left\|x^{(1)} - x^{(0)}\right\| \approx Cq^k.$$

Wir erwarten also, dass die Anzahl der korrekten Dezimalstellen in $x^{(k)}$ linear mit der Iterationszahl wächst, z.B. eine korrekte Stelle pro Iteration für $q = 1/10$ oder eine korrekte Stelle pro 3.3 Schritte für $q = 1/2$ (da $0.5^{3.3} \approx 0.1$).

Erfüllt eine gegen $\hat{x}$ konvergente Folge sogar

$$\left\|x^{(k+1)} - \hat{x}\right\| \leq C\left\|x^{(k)} - \hat{x}\right\|^p,$$

mit einem $C > 0$ und $p > 1$ so erwarten wir eine ver-p-fachung der Anzahl der korrekten Stellen in jedem Iterationsschritt. Dies motiviert:

Definition und Satz 4.8
(a) Für eine reelle nichtnegative Nullfolge $(\varepsilon_k)_{k\in\mathbb{N}}$ heißt

$$\kappa := \limsup_{k\to\infty} \varepsilon_k^{1/k}$$

asymptotischer Konvergenzfaktor. *Die Folge $(\varepsilon_k)_{k\in\mathbb{N}}$ heißt*

- linear konvergent *für* $0 < \kappa < 1$,
- sublinear konvergent *für* $\kappa = 1$,
- superlinear konvergent *für* $\kappa = 0$.

(b) Existieren für eine reelle nichtnegative Nullfolge $C > 0$ *und* $p > 1$*, so dass*

$$\varepsilon_{k+1} \leq C\varepsilon_k^p \quad \text{für hinreichend große } k \in \mathbb{N}$$

dann konvergiert die Folge superlinear. p *heißt* Konvergenzordnung *der Folge.*

(c) Für eine gegen $\hat{x}$ *konvergente Folge* $(x^{(k)})_{k\in\mathbb{N}} \subseteq \mathbb{R}^n$ *verwenden wir die entsprechenden Bezeichnungen, wenn sie für die Fehlerfolge*

$$\varepsilon_k := \left\| x^{(k)} - \hat{x} \right\|$$

zutreffen. (Offenbar ist wegen der Äquivalenz aller Normen auf dem $\mathbb{R}^n$ *diese Bezeichnung unabhängig von der gewählten Norm.)*

Beweis: Wir zeigen, dass eine Folge mit Konvergenzordnung $p > 1$ superlinear konvergiert. Sei $(\varepsilon_k)_{k\in\mathbb{N}}$ eine Folge mit Konvergenzordnung $p > 1$, d.h. es existiere ein $C > 0$, so dass

$$\varepsilon_{k+1} \leq C\varepsilon_k^p \quad \text{für hinreichend große } k \in \mathbb{N}.$$

O.B.d.A. sei auch $\varepsilon_k \neq 0$ für hinreichend große $k \in \mathbb{N}$, da ansonsten $\varepsilon_k = 0$ für alle hinreichend großen $k \in \mathbb{N}$ gilt und daraus trivialerweise $\limsup_{k\to\infty} \varepsilon_k^{1/k} = 0$ folgt.

Sei $\delta > 0$. Da $\varepsilon_k \to 0$ konvergiert, ist für hinreichend große $k \in \mathbb{N}$, $C\varepsilon_k^{p-1} \leq \delta$. Insgesamt existiert also ein $K \in \mathbb{N}$, so dass für alle $k \geq K$

$$\varepsilon_k \neq 0, \quad \varepsilon_{k+1} \leq C\varepsilon_k^p \quad \text{und} \quad C\varepsilon_k^{p-1} \leq \delta$$

und daher $\frac{\varepsilon_{k+1}}{\varepsilon_k} \leq C\varepsilon_k^{p-1} \leq \delta$ gilt. Hieraus folgt dann aber, dass für alle $k \geq K$

$$\varepsilon_k = \frac{\varepsilon_k}{\varepsilon_{k-1}} \frac{\varepsilon_{k-1}}{\varepsilon_{k-2}} \cdots \frac{\varepsilon_{K+1}}{\varepsilon_K} \varepsilon_K \leq \delta^{k-K} \varepsilon_K = \frac{\varepsilon_K}{\delta^K} \delta^k.$$

Somit ist

$$\limsup_{k\to\infty} \varepsilon_k^{1/k} \leq \limsup_{k\to\infty} \left(\frac{\varepsilon_K}{\delta^K} \delta^k \right)^{1/k} = \delta.$$

Da dies für alle $\delta > 0$ gilt, folgt dass $\limsup_{k\to\infty} \varepsilon_k^{1/k} = 0$. □

Auf dem Banachschen Fixpunktsatz beruhende Verfahren konvergieren wie oben skizziert im Allgemeinen linear mit Konvergenzfaktor q. Wir zeigen aber eine Verschärfung der eindimensionalen Version von Satz 4.5:

Satz 4.9
$\Phi: \mathscr{D}(\Phi) \to \mathbb{R}$ *sei eine auf einer offenen Menge* $\mathscr{D}(\Phi) \subseteq \mathbb{R}$ *definierte, p-mal stetig differenzierbare Funktion,* $p \geq 2$.

(a) Ist $\hat{x} \in \mathscr{D}(\Phi)$ *ein Fixpunkt von* Φ *und gilt*

$$0 = \Phi'(\hat{x}) = \ldots = \frac{d^{p-1}\Phi(\hat{x})}{dx^{p-1}},$$

dann ist die Fixpunktiteration $x^{(k+1)} = \Phi(x^{(k)})$ *lokal superlinear konvergent gegen* $\hat{x}$ *und die Konvergenzordnung ist mindestens p.*

(b) Ist zusätzlich zu (a)

$$\frac{d^p}{dx^p}\Phi(\hat{x}) \neq 0,$$

dann ist die Konvergenzordnung genau p.

Beweis: Nach Satz 4.5 konvergiert die Fixpunktiteration für alle Startwerte aus einer abgeschlossenen Kugel $K = \overline{B_\varepsilon(\hat{x})}$.

Für jedes $x^{(0)} \in K$ und die dazugehörige Fixpunktiterationsfolge $x^{(k)}$ erhalten wir durch Taylorentwicklung

$$\begin{aligned} x^{(k+1)} = \Phi(x^{(k)}) &= \Phi(\hat{x}) + \sum_{i=1}^{p-1} \frac{1}{i!} \frac{d^i\Phi(\hat{x})}{dx^i} \left(x^{(k)} - \hat{x}\right)^i + \frac{1}{p!} \frac{d^p\Phi(\xi^{(k)})}{dx^p} \left(x^{(k)} - \hat{x}\right)^p \\ &= \hat{x} + \frac{1}{p!} \frac{d^p\Phi(\xi^{(k)})}{dx^p} \left(x^{(k)} - \hat{x}\right)^p, \end{aligned}$$

mit einer Zwischenstelle $\xi^{(k)}$ zwischen $x^{(k)}$ und $\hat{x}$.

(a) Mit $C := \frac{1}{p!} \sup_{\xi \in K} \left| \frac{d^p\Phi(\xi)}{dx^p} \right|$ folgt, dass

$$|x^{(k+1)} - \hat{x}| \leq C|x^{(k)} - \hat{x}|^p.$$

Die Konvergenzordnung ist also mindestens p.

(b) Wäre die Konvergenzordnung q mit $q > p$, so würde ein $C' > 0$ existieren mit

$$|x^{(k+1)} - \hat{x}| \leq C'|x^{(k)} - \hat{x}|^q \quad \text{für fast alle } k \in \mathbb{N}$$

und damit

$$\frac{|x^{(k+1)} - \hat{x}|}{|x^{(k)} - \hat{x}|^p} \leq C' |x^{(k)} - \hat{x}|^{q-p} \to 0.$$

Da $x^{(k)} \to \hat{x}$, konvergieren aber auch die Zwischenstellen $\xi^{(k)} \to \hat{x}$ und wegen $\frac{d^p}{dx^p}\Phi(\hat{x}) \neq 0$ und der Stetigkeit von $\frac{d^p\Phi}{dx^p}$ existiert daher ein $c > 0$, sodass für fast alle $k \in \mathbb{N}$

$$\frac{1}{p!}\left|\frac{d^p\Phi(\xi^{(k)})}{dx^p}\right| \geq c.$$

Damit folgt, dass für fast alle $k \in \mathbb{N}$

$$\frac{|x^{(k+1)} - \hat{x}|}{|x^{(k)} - \hat{x}|^p} \geq c,$$

was der obigen Folgerung aus einer Konvergenzordnung $q > p$ widerspricht. Die Konvergenzordnung kann also nicht größer als p sein. □

Beispiel 4.10
Für das Heron-Verfahren aus Beispiel 4.7 gilt

$$\Phi'(\hat{x}) = 0 \neq \Phi''(\hat{x}).$$

Das Heron-Verfahren konvergiert also genau quadratisch.

4.2 Nullstellenbestimmung reeller Funktionen

Wir betrachten jetzt das Problem, die Nullstelle einer reellen Funktion

$$f : [a,b] \to \mathbb{R}$$

($b > a$) zu finden.

4.2.1 Intervallhalbierungsverfahren

Wir beginnen mit einem sehr einfachen, aber global konvergenten Verfahren, dem Intervallhalbierungs- oder Bisektionsverfahren. Ist f stetig und erfüllt $f(a) < 0$ und $f(b) > 0$, so muss nach dem Zwischenwertsatz für stetige Funktionen mindestens eine Nullstelle $\hat{x} \in (a,b)$ von f existieren. Wir halbieren das Intervall und betrachten $f(\frac{a+b}{2})$. Ist $f(\frac{a+b}{2}) > 0$, so muss eine Nullstelle in der vorderen Hälfte

liegen und wir fahren mit dem Intervall $[a, \frac{a+b}{2}]$ fort. Ist $f(\frac{a+b}{2}) < 0$, so muss eine Nullstelle in der hinteren Hälfte liegen und wir fahren mit dem Intervall $[\frac{a+b}{2}, b]$ fort. In jedem Schritt ergibt sich also eine Halbierung des Intervalls und wir können auf diese Art und Weise eine Nullstelle mit beliebiger Genauigkeit approximieren. Analoges gilt natürlich auch für $f(a) > 0$ und $f(b) < 0$.

Algorithm 8 Intervallhalbierungsverfahren

Gegeben $a, b \in \mathbb{R}$, $b > a$ und stetiges $f : [a,b] \to \mathbb{R}$ mit $f(a)f(b) < 0$
$y_a := f(a)$, $y_b := f(b)$
repeat
  $c := \frac{a+b}{2}$, $y_c := f\left(\frac{a+b}{2}\right)$
  if $y_c = 0$ **then**
    break and return c
  end if
  if $y_a y_c < 0$ **then**
    $b := c$; $y_b := y_c$
  else
    $a := c$; $y_a := y_c$
  end if
until $b - a$ hinreichend klein
return $[a,b]$

Satz 4.11
Seien $a, b \in \mathbb{R}$, $b > a$, $f : [a,b] \to \mathbb{R}$ stetig und $f(a)f(b) < 0$. $[a_k, b_k]$ sei das Ergebnis von k Schleifendurchläufen in Algorithmus 8.

Dann konvergiert $(a_k)_{k\in\mathbb{N}}$ monton wachsend und mit linearer Geschwindigkeit gegen eine Nullstelle $\hat{x}$ von f. $(b_k)_{k\in\mathbb{N}}$ konvergiert linear und monoton fallend gegen $\hat{x}$ und $\hat{x} \in [a_k, b_k]$ für alle $k \in \mathbb{N}$.

Beweis: Nach Konstruktion ist $(a_k)_{k\in\mathbb{N}}$ monoton wachsend, $(b_k)_{k\in\mathbb{N}}$ monoton fallend und es gilt $a_k \leq b_l$ für alle $k, l \in \mathbb{N}$. Insbesondere ist $(a_k)_{k\in\mathbb{N}}$ also beschränkt und besitzt damit einen Grenzwert $\hat{x}$. Aus $a_k \leq b_l$ für alle $k, l \in \mathbb{N}$ folgt $\hat{x} \in [a_k, b_k]$ für alle $k \in \mathbb{N}$.

Nach Konstruktion gilt außerdem $b_k - a_k = 2^{-k}(b-a) \to 0$. $(b_k)_{k\in\mathbb{N}}$ konvergiert also ebenfalls gegen $\hat{x}$ und die Konvergenzgeschwindigkeit ist für beide Folgen linear.

Schließlich enthält wegen dem Zwischenwertsatz jedes Intervall $[a_k, b_k]$ eine Nullstelle x_k von f. Da $a_k \to \hat{x}$, $b_k \to \hat{x}$, konvergiert auch die Folge dieser Nullstellen $(x_k)_{k\in\mathbb{N}}$ gegen $\hat{x}$ und aus der Stetigkeit von f folgt $f(\hat{x}) = 0$. □

4.2.2 Das Newton-Verfahren

Um ein superlinear konvergentes Verfahren zu entwickeln, versuchen wir gemäß Satz 4.9 die Nullstellenaufgabe

$$f(x) \stackrel{!}{=} 0$$

auf Fixpunktgestalt zu transformieren

$$\Phi(x) \stackrel{!}{=} x$$

mit $\Phi'(\hat{x}) = 0$ an der Nullstelle $\hat{x}$ von f.

Wir machen den Ansatz

$$\Phi(x) := x + f(x)g(x)$$

mit einer noch zu bestimmenden Funktion $g(x)$. Für $g(x) \neq 0$ gilt dann

$$f(x) = 0 \quad \Leftrightarrow \quad x = x + f(x)g(x) = \Phi(x)$$

Die Forderung $\Phi'(\hat{x}) \stackrel{!}{=} 0$ führt auf

$$0 \stackrel{!}{=} \Phi'(\hat{x}) = 1 + f'(\hat{x})g(\hat{x}) + f(\hat{x})g'(\hat{x}),$$

also $g(\hat{x}) = -\frac{1}{f'(\hat{x})}$. Eine naheliegende Wahl ist also

$$g(x) = -\frac{1}{f'(x)}, \quad \text{d.h.} \quad \Phi(x) := x - \frac{f(x)}{f'(x)}.$$

Die zugehörige Fixpunktiteration

$$x_{k+1} := x_k - \frac{f(x_k)}{f'(x_k)}$$

heißt *Newton-Verfahren*.

Bemerkung 4.12
Wir geben noch eine weitere Motivation für das Newton-Verfahren. Ersetzen wir in der Nullstellenaufgabe $f(x) \stackrel{!}{=} 0$ die nicht-lineare Funktion $f(x)$ durch ihr lineares Taylorpolynom um unsere aktuelle Iterierte x_k, so erhalten wir

$$0 \stackrel{!}{=} f(x) \approx f(x_k) + f'(x_k)(x - x_k).$$

Offensichtlich ist die nächste Newton-Iterierte gerade die Nullstelle dieser linearen Approximation (vgl. die in der Vorlesung gemalte Skizze).

Satz 4.13
$f \in C^3(a,b)$ besitze eine Nullstelle $\hat{x} \in (a,b)$ mit

$$f(\hat{x}) = 0 \quad \text{und} \quad f'(\hat{x}) \neq 0.$$

Dann konvergiert das Newton-Verfahren lokal (mindestens) quadratisch gegen die Nullstelle $\hat{x}$.

Beweis: Da f' stetig ist und $f'(\hat{x}) \neq 0$ existiert eine Umgebung von $\hat{x}$, in der $f'(x) \neq 0$ gilt. Auf dieser Umgebung ist die Nullstellenaufgabe $f(x) = 0$ äquivalent zur Fixpunktaufgabe $\Phi(x) = x$ mit $\Phi(x) := x - f(x)/f'(x)$. Aus $f \in C^3$ und $f'(x) \neq 0$ folgt $\Phi \in C^2$, $\Phi(\hat{x}) = \hat{x}$ und $\Phi'(\hat{x}) = 0$. Die Behauptung folgt damit aus Satz 4.9. □

Beispiel 4.14

(a) Die Berechnung der Quadratwurzel einer Zahl $a > 0$ können wir als Nullstellenaufgabe schreiben

$$f(x) := x^2 - a \stackrel{!}{=} 0.$$

Das Newton-Verfahren lautet

$$x_{k+1} := x_k - \frac{f(x_k)}{f'(x_k)} = x_k - \frac{x_k^2 - a}{2x_k} = \frac{1}{2}\left(x_k + \frac{a}{x_k}\right),$$

das ist also das schon bekannte (sogar global für alle $x_0 > 0$) quadratisch konvergente Heron-Verfahren.

(b) Für das Nullstellenproblem

$$f(x) := \frac{x}{\sqrt{x^2+1}} \stackrel{!}{=} 0.$$

ist

$$f'(x) = \frac{1}{\sqrt{x^2+1}} - \frac{x^2}{(x^2+1)^{3/2}} = \frac{1}{(x^2+1)^{3/2}}$$

und es ergibt sich als Newton-Iteration

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k - x_k(x_k^2+1) = -x_k^3.$$

Für jeden Startwert $x_0 \in (-1,1)$ konvergiert das Newton-Verfahren also (sogar kubisch) gegen das Minimum $\hat{x} = 0$.

Für $|x_0| \geq 1$ divergieren die Iterierten jedoch. Für $|x_0| > 1$ gehen sie betragsmäßig gegen unendlich und für $|x_0| = 1$ alternieren sie zwischen -1 und 1.

Bemerkung 4.15 (Das mehrdimensionale Newton-Verfahren)
Wir betrachten nun die mehrdimensionale Nullstellenaufgabe $F(x) = 0$, wobei

$$F : \mathbb{R}^n \to \mathbb{R}^n$$

eine stetig differenzierbare Funktion mit Nullstelle $\hat{x} \in \mathbb{R}^n$ sei.

Analog Bemerkung 4.12 können wir die nicht-lineare Funktion durch ihr lineares Taylorpolynom um unsere aktuelle Iterierte $x^{(k)}$ ersetzen und erhalten

$$0 \overset{!}{=} F(x) \approx F(x^{(k)}) + F'(x^{(k)})(x - x^{(k)})$$

mit der Jacobi-Matrix $F'(x^{(k)}) \in \mathbb{R}^{n \times n}$.

So erhalten wir die Iterationsvorschrift

$$x^{(k+1)} := x^{(k)} - F'(x^{(k)})^{-1} F(x^{(k)})$$

bzw.

$$x^{(k+1)} := x^{(k)} + h^{(k)}, \quad \text{wobei } h^{(k)} \text{ das LGS } F'(x^{(k)})h^{(k)} = -F(x^{(k)}) \text{ löst.}$$

Im Rahmen der Vorlesung Optimierung und inverse Probleme *im Sommersemester 2024 werden wir zeigen, dass das mehrdimensionale Newton-Verfahren lokal superlinear konvergiert, falls die Jacobi-Matrix $F(\hat{x})$ invertierbar ist. Ist F' zusätzlich Lipschitz-stetig in einer Umgebung von $\hat{x}$, dann konvergiert das Newton-Verfahren sogar quadratisch (vgl. auch Übungsaufgabe 13.4 für ein numerisches Beispiel). Das entsprechende Resultat gilt auch für $n = 1$ und zeigt, dass Satz 4.13 auch unter der schwächen Voraussetzung gilt, dass f' in einer Umgebung von $\hat{x}$ Lipschitz-stetig ist.*

Kapitel 5

Splineinterpolation

5.1 Motivation und Definition

Zum Abschluss der Vorlesung gehen wir nocheinmal auf die Interpolationsaufgabe aus Abschnitt 1.3 ein. Gegeben sei in diesem Kapitel stets ein nichtleeres Intervall $[a,b] \subset \mathbb{R}$, $a,b \in \mathbb{R}$ und ein Gitter aus $m+1$ paarweise verschiedenen, aufsteigend angeordneten Knoten

$$\Delta := \{x_0, x_1, \ldots, x_m\} \subset [a,b], \qquad a = x_0 < x_1 < \ldots < x_m = b.$$

Gesucht ist eine Funktion

$$\varphi : \ [a,b] \to \mathbb{R},$$

die vorgegebene Werte $y_i \in \mathbb{R}$, $i = 0, \ldots, m+1$ auf diesem Gitter auf möglichst „gute" Art und Weise interpoliert, also

$$\varphi(x_i) = y_i \quad \forall i \in \{0, 1, \ldots, m\}.$$

Kriterien für eine „gute" Interpolation sind dabei:

(a) φ sollte eine möglichst glatte Funktion sein.

(b) Sind die Werte y_i aus Auswertung einer glatten Funktion $f : [a,b] \to \mathbb{R}$ entstanden, so sollte die interpolierenden Funktion φ für $m \to \infty$ gegen die ursprüngliche Funktion f konvergieren.

Das sind keine mathematisch präzisen Kriterien. Die Eigenschaft „glatt" in (a) kann z.B. einen hohen Grad an Differenzierbarkeit, aber auch geringe Steigung oder Krümmung bedeuten. Der Konvergenzbegriff von Funktionen in (b) hängt

von der verwendeten Topologie ab, z.B. punktweise, gleichmäßig, bzgl. der Norm $\|\cdot\|_{L^2}$, etc.

In Abschnitt 1.3 haben wir gesehen, dass die Interpolationsaufgabe stets durch ein Polynom $\varphi \in \Pi_m$ gelöst werden kann, dieses aber für große m immer stärke Oszillationen aufweist und damit keinen gute Approximationseigenschaften im Sinne von (b) besitzt.

Wie im Kapitel über numerische Quadratur ersetzen wir deshalb das Polynom durch ein *stückweises* Polynom. Die Polynomstücke sollen dabei jedoch jetzt möglichst glatt zueinander passen:

Definition 5.1
Ein Spline vom Grad $n \in \mathbb{N}_0$ *ist eine Funktion* $s : (a,b] \to \mathbb{R}$ *mit den Eigenschaften*

(a) Auf jedem Intervall $(x_{i-1}, x_i]$, $i = 1, \ldots, m$ *ist* s *ein Polynom* n*-ten Grades.*

(b) $s \in C^{n-1}((a,b])$

Für $n > 0$ *identifizieren wir* s *auch mit seiner stetigen Fortsetzung*

$$s : \ [a,b] \to \mathbb{R}.$$

Die Menge aller solchen Splines bezeichnen wir mit $S_{n,\Delta}$. *Offensichtlich ist* $S_{n,\Delta}$ *ein Vektorraum und es gilt* $\Pi_n \subseteq S_{n,\Delta}$. *sowie*

$$s \in S_{n+1,\Delta} \quad \Longleftrightarrow \quad s' \in S_{n,\Delta}, \quad (n \in \mathbb{N}).$$

Beispiel 5.2
(a) $S_{0,\Delta}$ *ist der Raum aller stückweiser konstanter Funktionen*

$$s(x) = \sum_{i=1}^{m} y_i \chi_i(x) \qquad \textit{mit } y_i \in \mathbb{R}, \ i = 1, \ldots m,$$

wobei

$$\chi_i(x) = \chi_{(x_{i-1},x_i]}(x) = \begin{cases} 1 & \textit{für } x \in (x_{i-1}, x_i] \\ 0 & \textit{sonst} \end{cases}$$

die charakteristische Funktion des i*-ten Intervalls ist.*

Mit dieser Darstellung gilt

$$s(x_i) = y_i, \qquad \textit{für } i = 1, \ldots, m.$$

Ein so konstruierter Spline 0*-ter Ordnung erfüllt also die Interpolationsaufgabe in allen Knoten bis auf* $a = x_0$.

(b) (vgl. die in der Vorlesung gemalten Skizzen)

$S_{1,\Delta}$ ist der Raum aller stückweise linearer Funktionen

$$s(x) = \sum_{i=0}^{m} y_i \Lambda_i(x) \quad \text{mit } y_i \in \mathbb{R},\ i = 0, \dots m,$$

*wobei Λ_i diejenige auf Δ stückweise lineare Funktion ist mit $\Lambda_i(x_j) = \delta_{ij}$, $i, j = 0, \dots, m$ (*Hutfunktion*).*

Mit dieser Darstellung gilt

$$s(x_i) = y_i, \quad \text{für } i = 0, \dots, m.$$

Ein so konstruierter Spline 1-ter Ordnung erfüllt also die Interpolationsaufgabe in allen $m+1$ Knoten.

Lemma 5.3
Die Dimension von $S_{n,\Delta}$ ist $n+m$.

Beweis: Wir führen den Beweis durch Induktion über $n \in \mathbb{N}_0$. Aus Beispiel 5.2(a) folgt $\dim(S_{0,\Delta}) = m$ und $\dim(S_{1,\Delta}) = m+1$. Für den Induktionsschritt $n \mapsto n+1$ sei $(s_1, \dots, s_{m+n})$ eine Basis von $S_{n,\Delta}$ für ein $n \geq 1$.

Seien $(\tilde{s}_1, \dots, \tilde{s}_{m+n})$ Stammfunktionen von $(s_1, \dots s_{m+n})$. Für jedes $s \in S_{n+1,\Delta}$ gilt $s' \in S_{n,\Delta}$. Es existieren also $\lambda_1, \dots, \lambda_{m+n} \in \mathbb{R}$ mit

$$s'(x) = \sum_{i=1}^{m+n} \lambda_i s_i(x), \quad \text{also} \quad \left(s(x) - \sum_{i=1}^{m+n} \lambda_i \tilde{s}_i(x) \right)' = 0.$$

$s(x) - \sum_{i=1}^{m+n} \lambda_i \tilde{s}_i(x)$ ist also konstant, etwa $\lambda \in \mathbb{R}$. Damit ist

$$s = \lambda 1(x) + \sum_{i=1}^{m+n} \lambda_i \tilde{s}_i(x),$$

d.h. $(1, \tilde{s}_1, \dots, \tilde{s}_{m+n})$ ist ein Erzeugendensystem von $S_{n+1,\Delta}$.

Zum Beweise der linearen Unabhängigkeit sei

$$\lambda 1(x) + \sum_{i=1}^{m+n} \lambda_i \tilde{s}_i(x) = 0 \quad \forall x \in [a, b].$$

Dann folgt durch Ableiten $\sum_{i=1}^{m+n} \lambda_i s_i(x) = 0$. Aus der linearen Unabhängigkeit der s_i ergibt sich dann $\lambda_i = 0$, $i = 1, \dots, m+n$ und damit auch $\lambda = 0$. □

5.2 Konstruktion kubischer Splines

In der Praxis weit verbreitet ist die Interpolation durch kubische Splines. Auf einem Gitter mit $m+1$ Punkten besitzen diese $m+3$ Freiheitsgrade, wovon nur $m+1$ durch die Interpolationbedingungen festgelegt sind.

Definition 5.4
Ein kubischer Spline $s \in S_{\Delta,3}$ der die Interpolationsaufgabe $s(x_i) = y_i$, $i = 0,\dots,m$ erfüllt heißt

(a) vollständig interpolierend, *falls $s'(x_0) = y'_0$ und $s'(x_m) = y'_m$ zu zwei zusätzlich vorgegebenen Werten $y'_0, y'_m \in \mathbb{R}$ gilt.*

(b) natürlich, *falls $s''(x_0) = 0 = s''(x_m)$.*

(c) periodisch, *falls $s'(x_0) = s'(x_m)$ und $s''(x_0) = s''(x_m)$.*

Für die Untersuchung von Existenz und Eindeutigkeit, aber auch zur numerischen Berechnung ist folgende Überlegung nützlich. Dabei bezeichne im Folgenden stets $h_i := x_i - x_{i-1}$, $i = 1,\dots,m$

Lemma 5.5
(a) Sei $s \in S_{\Delta,3}$. Mit den Bezeichnungen[1]

$$s_i := s(x_i), \quad s'_i := s'(x_i), \quad \gamma_i := s''(x_i) \qquad (i = 0,\dots,m)$$

gilt für alle $x \in [x_{i-1}, x_i]$, $i = 1,\dots,m$

$$s(x) = s_i + s'_i(x - x_i) + \gamma_i \frac{(x-x_i)^2}{2} + \frac{\gamma_i - \gamma_{i-1}}{h_i} \frac{(x-x_i)^3}{6} \tag{5.1}$$

(b) Seien $s_i, s'_i, \gamma_i \in \mathbb{R}$ $(i = 0,\dots,m)$ gegeben. Die abschnittsweise auf $(x_{i-1}, x_i]$ durch (5.1) definierte Funktion $s : (a,b] \to \mathbb{R}$ ist genau dann ein kubischer Spline mit

$$s(x_i) = s_i, \quad s'(x_i) = s'_i, \quad \text{und} \quad s''(x_i) = \gamma_i \quad \text{für alle } i = 0,\dots,m,$$

falls für alle $i \in \{0,\dots,m-1\}$

$$s_i = s_{i+1} - s'_{i+1} h_{i+1} + \frac{1}{3}\gamma_{i+1} h_{i+1}^2 + \frac{1}{6}\gamma_i h_{i+1}^2, \tag{5.2}$$

$$s'_i = s'_{i+1} - \frac{1}{2}\gamma_{i+1} h_{i+1} - \frac{1}{2}\gamma_i h_{i+1}. \tag{5.3}$$

[1] Die γ_i werden eine besondere Rolle spielen. Sie heißen auch *Momente* des kubischen Splines.

Beweis: (a) s'' ist auf $[x_{i-1},x_i]$ ein lineares Polynom mit Randwerten $s''(x_i)=\gamma_i$ und $s''(x_{i-1})=\gamma_{i-1}$.

Definiere $f:\ [x_{i-1},x_i]\to\mathbb{R}$ durch die rechte Seite von (5.1). Dann ist für alle $x\in[x_{i-1},x_i]$

$$f''(x)=\gamma_i+\frac{\gamma_i-\gamma_{i-1}}{h_i}(x-x_i),$$

also ist f'' ebenfalls ein lineares Polynom und es gilt

$$f''(x_i)=\gamma_i=s''(x_i)\quad\text{ und }f''(x_{i-1})=\gamma_{i-1}=s''(x_{i-1}).$$

Da lineare Polynome durch ihre Randwerte eindeutig festgelegt sind, gilt $s''(x)=f''(x)$ für alle $x\in[x_{i-1},x_i]$.

Da außerdem $f'(x_i)=s'_i=s'(x_i)$ und $f(x_i)=s_i=s(x_i)$, folgt, dass

$$f'(x)=\int_{x_i}^{x}f''(t)\,\mathrm{d}t+f'(x_i)=\int_{x_i}^{x}s''(t)\,\mathrm{d}t+s'(x_i)=s'(x)\ \text{ für alle } x\in[x_{i-1},x_i],$$

und damit

$$f(x)=\int_{x_i}^{x}f'(t)\,\mathrm{d}t+f(x_i)=\int_{x_i}^{x}s'(t)\,\mathrm{d}t+s(x_i)=s(x)\ \text{ für alle } x\in[x_{i-1},x_i].$$

(b) s ist auf jedem Teilintervall $(x_{i-1},x_i]$ ein Polynom dritten Grades, so dass s genau dann ein kubischer Spline ist, wenn $s\in C^2([a,b])$ und dieses ist genau dann erfüllt, wenn für alle $i=1,\dots,m-1$

$$\begin{aligned}\lim_{x\to x_i^-}s(x)=s(x_i)&=\lim_{x\to x_i^+}s(x)\\ \lim_{x\to x_i^-}s'(x)=s'(x_i)&=\lim_{x\to x_i^+}s'(x)\\ \lim_{x\to x_i^-}s''(x)=s''(x_i)&=\lim_{x\to x_i^+}s''(x)\end{aligned}$$

Aus (5.1) auf $(x_{i-1},x_i]$ erhalten wir für alle $i\in\{1,\dots,m\}$, dass

$$s(x_i)=\lim_{x\to x_i^-}s(x)=s_i,\quad s'(x_i)=\lim_{x\to x_i^-}s'(x)=s'_i,\quad s''(x_i)=\lim_{x\to x_i^-}s''(x)=\gamma_i$$

und aus (5.1) auf $(x_i, x_{i+1}]$ folgt dass für alle $i \in \{0, \dots, m-1\}$

$$\begin{aligned}
\lim_{x \to x_i^+} s(x) &= s_{i+1} - s'_{i+1} h_{i+1} + \gamma_{i+1} \frac{h_{i+1}^2}{2} - \frac{\gamma_{i+1} - \gamma_i}{h_{i+1}} \frac{h_{i+1}^3}{6} \\
&= s_{i+1} - s'_{i+1} h_{i+1} + \frac{1}{3} \gamma_{i+1} h_{i+1}^2 + \frac{1}{6} \gamma_i h_{i+1}^2 \\
\lim_{x \to x_i^+} s'(x) &= s'_{i+1} - \gamma_{i+1} h_{i+1} + \frac{\gamma_{i+1} - \gamma_i}{h_{i+1}} \frac{h_{i+1}^2}{2} = s'_{i+1} - \frac{1}{2} \gamma_{i+1} h_{i+1} - \frac{1}{2} \gamma_i h_{i+1} \\
\lim_{x \to x_i^+} s''(x) &= \gamma_{i+1} - \frac{\gamma_{i+1} - \gamma_i}{h_{i+1}} h_{i+1} = \gamma_i.
\end{aligned}$$

Durch Verwendung dieser Formeln für $i \in \{1, \dots, m-1\}$ folgt, dass eine abschnittsweise durch (5.1) definierte Funktion genau dann ein kubischer Spline ist, falls (5.2) und (5.3) für alle $i \in \{1, \dots, m-1\}$ gelten. Dieser erfüllt dann offenbar auch für alle $i = 1, \dots, m-1$

$$s(x_i) = s_i, \quad s'(x_i) = s'_i, \quad \text{und} \quad s''(x_i) = \gamma_i. \tag{5.4}$$

Durch Verwendung der obigen Formeln für $i = m$ bzw. $i = 0$ folgt außerdem, dass (5.4) immer auch für $i = m$ erfüllt ist, und dass (5.4) für den Fall $i = 0$ äquivalent ist zur Gültigkeit von (5.2) und (5.3) für $i = 0$. □

Wir zeigen wie sich damit konstruktiv die eindeutige Existenz eines interpolierenden natürlichen kubischen Splines zeigen lässt. Die Untersuchung der anderen Varianten geht ähnlich, siehe z.B. [Hanke].

Satz 5.6

Seien $s_i \in \mathbb{R}$ $(i = 0, \dots, m)$. Es existiert genau ein die Werte (x_i, s_i) interpolierender natürlicher kubischer Spline, also

$$s \in S_{\Delta,3} \quad \textit{mit } s(x_i) = s_i,\ i = 0, \dots, m, \quad s''(x_0) = 0 = s''(x_m).$$

$s : [a,b] \to \mathbb{R}$ ist gegeben durch

$$s(x) := s_i + s'_i (x - x_i) + \gamma_i \frac{(x - x_i)^2}{2} + \frac{\gamma_i - \gamma_{i-1}}{h_i} \frac{(x - x_i)^3}{6} \quad \forall x \in [x_{i-1}, x_i).$$

Dabei ist $\gamma_0 = 0 = \gamma_m$ und der Vektor $(\gamma_1, \dots, \gamma_{m-1}) \in \mathbb{R}^{m-1}$ löst das eindeutig lösbare LGS

$$\frac{1}{6} \begin{pmatrix} 2(h_1 + h_2) & h_2 & 0 & & \\ h_2 & 2(h_2 + h_3) & \ddots & & \\ & \ddots & \ddots & h_{m-1} & \\ & 0 & h_{m-1} & 2(h_{m-1} + h_m) \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_{m-1} \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{m-1} \end{pmatrix}, \tag{5.5}$$

wobei

$$d_i = \frac{s_{i+1}-s_i}{h_{i+1}} - \frac{s_i - s_{i-1}}{h_i}, \qquad i = 1,\dots,m-1.$$

Die s_i' sind gegeben durch

$$s_i' = \frac{s_i - s_{i-1}}{h_i} + \gamma_{i-1}\frac{h_i}{6} + \gamma_i \frac{h_i}{3}, \qquad i = 1,\dots,m. \tag{5.6}$$

Beweis: Die Matrix des LGS ist offensichtlich strikt diagonaldominant und damit das LGS nach Satz 3.7 eindeutig lösbar.

Sei $s \in S_{\Delta,3}$ ein die Werte (x_i, s_i) interpolierender natürlicher kubischer Spline und wie in Lemma 5.5 bezeichne $s_i = s(x_i)$, $s_i' := s'(x_i)$ und $\gamma_i := s''(x_i)$ $(i = 0,\dots,m)$. Aus (5.2) in Lemma 5.5 folgt für $i = 1,\dots,m-1$

$$\frac{s_{i+1}-s_i}{h_{i+1}} - \frac{s_i - s_{i-1}}{h_i} = s_{i+1}' - \frac{h_{i+1}}{6}\gamma_i - \frac{h_{i+1}}{3}\gamma_{i+1} - s_i' + \frac{h_i}{6}\gamma_{i-1} + \frac{h_i}{3}\gamma_i$$

und zusammen mit (5.3) folgt

$$\begin{aligned} d_i &:= \frac{s_{i+1}-s_i}{h_{i+1}} - \frac{s_i - s_{i-1}}{h_i} \\ &= \frac{h_{i+1}}{2}\gamma_i + \frac{h_{i+1}}{2}\gamma_{i+1} - \frac{h_{i+1}}{6}\gamma_i - \frac{h_{i+1}}{3}\gamma_{i+1} + \frac{h_i}{6}\gamma_{i-1} + \frac{h_i}{3}\gamma_i \\ &= \frac{1}{6}\left(h_i\gamma_{i-1} + 2(h_i + h_{i+1})\gamma_i + h_{i+1}\gamma_{i+1}\right). \end{aligned}$$

Die γ_i, $i = 1,\dots,m-1$, lösen also das eindeutige lösbare LGS (5.5). Die behauptete Gestalt der s_i' folgt aus (5.2). Damit ist auch gezeigt jeder interpolierender natürliche kubischer Spline dieselben γ_i, s_i und s_i' besitzen muss, es kann also nach Lemma 5.5 höchstens einen interpolierenden natürlichen kubischen Spline geben.

Durch die Interpolationsbedingungen zusammen mit den beiden Randbedingungen $s''(x_0) = 0 = s''(x_m)$ sind $m+3$ lineare Gleichungen für s aus dem $m+3$-dimensionalen Vektorraum $S_{\Delta,3}$ gegeben. Für ein lineares Gleichungsystem mit gleicher Anzahl Unbekannte wie Gleichungen folgt die Lösbarkeit aus der Eindeutigkeit, so dass damit auch die Existenz eines interpolierenden natürlichen kubischen Splines bewiesen ist. □

Wir zeigen noch, dass der natürliche kubische Spline die zu interpolierenden Punkte durch eine Funktion mit (in gewissem Sinne) geringster Krümmung verbindet:

Satz 5.7
Seien $y_i \in \mathbb{R}$, $i = 0,\dots,m$. Sei $s \in S_{\Delta,3}$ der die Werte (x_i, y_i) interpolierende natürliche kubische Spline und sei $f \in C^2([a,b])$ irgendeine andere Funktion die ebenfalls die Interpolationsaufgabe $f(x_i) = y_i$, $i = 0,\dots,m$ erfüllt. Dann gilt

$$\|s''\|_{L^2([a,b])} \leq \|f''\|_{L^2([a,b])},$$

Dabei bezeichnet für eine Funktion $g \in C([a,b])$ (hier: $g = s''$ und $g = f''$)

$$\|g\|_{L^2([a,b])} := \left(\int_a^b |g(t)|^2 \,\mathrm{d}t\right)^{1/2}$$

die zum Skalarprodukt aus Bemerkung 1.18 und Blatt 4, Aufgabe 2 (mit Gewichtsfunktion $w(x) = 1$) zugehörige Norm.

Beweis: Im diesem Beweis bezeichne $\|\cdot\|$ stets die L^2-Norm $\|\cdot\|_{L^2([a,b])}$. Wir werden zeigen, dass bzgl. des L^2-Skalarproduktes $s'' \perp f'' - s''$ gilt, also

$$\int_a^b s''(f'' - s'') \,\mathrm{d}t = 0.$$

Damit folgt dann die Behauptung aus

$$\begin{aligned}\|f''\|^2 &= \|f'' - s'' + s''\|^2 = \|f'' - s''\|^2 + 2\int_a^b (f'' - s'')s'' \,\mathrm{d}t + \|s''\|^2 \\ &= \|f'' - s''\|^2 + \|s''\|^2 \geq \|s''\|.\end{aligned}$$

Durch partielle Integration erhalten wir, dass

$$\begin{aligned}\int_a^b s''(f'' - s'') \,\mathrm{d}t &= \sum_{i=1}^m \int_{x_{i-1}}^{x_i} s''(f'' - s'') \,\mathrm{d}t \\ &= \sum_{i=1}^m s''(f' - s')|_{x_{i-1}}^{x_i} - \sum_{i=1}^m \int_{x_{i-1}}^{x_i} s'''(f' - s') \,\mathrm{d}t.\end{aligned}$$

Ein Teleskopsummenargument zeigt, dass für den ersten Summanden gilt

$$\sum_{i=1}^m s''(f' - s')|_{x_{i-1}}^{x_i} = s''(b)(f'(b) - s'(b)) - s''(a)(f'(a) - s'(a)) = 0,$$

wobei wir $s''(b) = 0 = s''(a)$ ausgenutzt haben. Für den zweiten Summanden verwenden wir dass s''' auf (x_{i-1}, x_i) konstant ist und erhalten aus $f(x_i) = s(x_i)$, dass

$$\int_{x_{i-1}}^{x_i} s'''(f' - s') \,\mathrm{d}t = s'''|_{(x_{i-1},x_i)}(f - s)|_{x_{i-1}}^{x_i} = 0.$$

Insgesamt ist also $\int_a^b s''(f'' - s'') \,\mathrm{d}t = 0$ gezeigt und damit die Behauptung bewiesen. □

Literaturverzeichnis

[Hanke] M. Hanke-Bourgeois: *Grundlagen der Numerischen Mathematik und des Wissenschaftlichen Rechnens*, Teubner Verlag, Wiesbaden, 2009.

[Fischer] G. Fischer, B. Springborn: *Lineare Algebra: Eine Einführung für Studienanfänger*, Springer-Verlag, 2020.

[Heuser] H. Heuser: *Lehrbuch der Analysis Teil 1*, Teubner Verlag, Wiesbaden, 2009.