

Skript Numerik I

bei Prof. Dr. Blank im WS14/15

Gesina Schwalbe

2. November 2014

Inhaltsverzeichnis

1	Einführung	1
2	Lineare Gleichungssysteme: Direkte Methoden	3
2.1	Gaußsches Eliminationsverfahren	3
2.1.1	Vorwärtselimination	3
2.1.2	Rückwärtselimination	5
2.1.3	Vorsicht	5
2.1.4	Weitere algorithmische Anmerkungen	5
2.1.5	Dreieckszerlegung	5
2.1.6	Vorwärtssubstitution	6
2.1.7	Gauß-Elementor zur Lösung von $Ax = b$	6
2.1.8	Rechenaufwand gezählt in „flops“	6
2.1.9	Definition: Landau-Symbole	6
2.1.10	Allgemeines zur Aufwandsbetrachtung	6
2.1.11	Formalisieren des Gauß-Algorithmus	6
2.1.12	Lemma (Eigenschaften der L_k -Matrizen)	6
2.1.13	Satz (LR- oder LU-Zerlegung)	6
2.2	Gaußsches Eliminationsverfahren mit Pivotisierung	6
2.2.1	Spaltenpivotisierung (=partielle/ halbmaximale Pivotisierung) . .	7
2.2.2	Bemerkungen	7
2.2.3	Satz: Dreieckszerlegung mit Permutationsmatrix	7
2.2.4	Lösen eines Gleichungssystems $Ax = b$	8
2.2.5	Bemerkungen	8
2.2.6	Beispiel zur Pivotisierung	8
3	Fehleranalyse	9
3.1	Zahlendarstellung und Rundungsfehler	9
3.1.1	Definition: Gleitkommazahl	10
3.1.2	Bemerkung	10
3.1.3	Beispiel	10
3.1.4	Verteilung der Maschinenzahlen	11
3.1.5	Bezeichnungen	11
3.1.6	Rundungsfehler	12
3.1.7	Bemerkung	13
3.1.8	Auslöschung von signifikanten Stellen	13

3.2	Kondition eines Problems	14
3.2.1	Definition: Problem	14
3.2.2	Definition: absoluter und relativer Fehler	14
3.2.3	Wiederholung: Normen	14
3.2.4	Definition: Matrixnorm	15
3.2.5	Definition: Frobeniusnorm, p-Norm, Verträglichkeit	15
3.2.6	Bemerkungen	15
3.2.7	Definition: absolute und relative normweise Kondition	16
3.2.8	Lemma	16
3.2.9	Beispiel: Kondition der Addition	17
3.2.10	Beispiel: Lösen eines Gleichungssystems	18
3.2.11	Definition: Kondition einer Matrix	20
3.2.12	Lemma (Neumannsche Reihe)	21
3.2.13	Bemerkung	22
3.2.14	Beispiel: Kondition eines nichtlinearen Gleichungssystems	22
3.2.15	Beispiel	23
3.2.16	Definition: Komponentenweise Kondition	23
3.2.17	Lemma	23
3.2.18	Beispiel	24
3.3	Stabilität von Algorithmen	25
3.3.1	Bemerkung	26
3.3.2	Beispiel	26
3.3.3	Definition:	27
3.3.4	Definition:	27
3.3.5	Lemma (Fehlerfortpflanzung)	27
3.3.6	Korollar	29
3.3.7	Bemerkung	29
3.3.8	Bemerkung zur Sprechweise	29
3.3.9	Beispiel	30
3.3.10	Rückwärtsanalyse	30

1 Einführung

06.10.2014

2 Lineare Gleichungssysteme: Direkte Methoden

Sei $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$. Gesucht ist $x \in \mathbb{R}^n$ mit

$$A \cdot x = b$$

Weitere Voraussetzungen sind die Existenz und Eindeutigkeit einer Lösung. Bemerkung:

- Ein verlässlicher Lösungsalgorithmus überprüft dies und behandelt alle Fälle.
- Die Cramersche Regel ist ineffizient (s. Einführung).
- Das Inverse für $x = A^{-1} \cdot b$ aufzustellen ist ebenso ineffizient, denn es ist keine Lösung für alle $b \in \mathbb{R}^n$ verlangt und der Algorithmus wird evtl. instabil aufgrund vieler Operationen.

\Rightarrow Invertieren von Matrizen vermeiden!!

\Rightarrow Lösen des Linearen Gleichungssystems!!

2.1 Gaußsches Eliminationsverfahren

Das Verfahren wurde 1809 von Friedrich Gauß, 1759 von Joseph Louis Lagrange beschrieben und war seit dem 1. Jhd. v. Chr. in China bekannt.

2.1.1 Vorwärtselimination

Das Gaußverfahren gilt der Lösung eines linearen Gleichungssystems der Form

$$Ax = b$$

mit $A = (a_{ij})_{i,j \leq n} \in K^{n \times n}$ Matrix und $b = (b_i)_{i \leq n} \in K^n$ Vektor.

Der zugehörige Algorithmus sieht folgendermaßen aus:

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

||

2 Lineare Gleichungssysteme: Direkte Methoden

$$(i\text{-te Zeile}) - (1. \text{ Zeile}) \cdot \frac{a_{i1}}{a_{11}} \Rightarrow a_{i1} = 0$$

\Downarrow

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

\Downarrow

\vdots

mit

$$a_{ij}^{(1)} = a_{ij} - a_{1j} \cdot \frac{a_{i1}}{a_{11}} \quad \text{für } i, j = 2, \dots, n$$

$$b_i^{(1)} = b_i - b_1 \cdot \frac{a_{i1}}{a_{11}} \quad \text{für } i = 2, \dots, n$$

In jedem Schritt werden die Einträge der k -ten Spalte analog unterhalb der Diagonalen (also $k = 1, \dots, n-1$) eliminiert:

$$(i\text{-te Zeile}) - (k\text{-te Zeile}) \cdot \frac{a_{ik}}{a_{kk}} \quad \text{für } i = k+1, \dots, n$$

Die Reihe

$$A \rightarrow A^{(1)} \rightarrow A^{(2)} \rightarrow \dots \rightarrow A^{(n-1)}$$

wird bis zum n -ten Schritt fortgeführt, d.h. bis eine obere Dreiecksgestalt eintritt:

$$\underbrace{\begin{pmatrix} a_{11} & \cdots & \cdots & a_{1n} \\ & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ & & \ddots & \vdots \\ 0 & & & a_{nn}^{(n-1)} \end{pmatrix}}_{:=R} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \underbrace{\begin{pmatrix} b_1 \\ b_2^{(1)} \\ \vdots \\ b_n^{(n-1)} \end{pmatrix}}_{:=z}$$

$$Rx = z \quad (2.1.1)$$

wobei für $i = k+1, \dots, n$ die Einträge wie folgt aussehen:

$$l_{ik} := \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} \quad (2.1.2)$$

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{kj}^{(k-1)} \cdot l_{ik} \quad \text{für } j = k+1, \dots, n \quad (2.1.3)$$

$$b_i^{(k)} = b_i^{(k-1)} - b_k^{(k-1)} \cdot l_{ik} \quad (2.1.4)$$

Dieser Prozess wird **Vorwärtselimination** genannt.

2.1.2 Rückwärtselimination

Für die Lösung des Gleichungssystems ist dann noch die **Rückwärtssubstitution** nötig:

$$x_1 = \frac{b_1^{(n-1)}}{a_{nn}^{(n-1)}} \quad (2.1.5)$$

$$x_{n-1} = \frac{b_{n-1}^{(n-2)} - a_{n-1,n}^{(n-1)} \cdot x_n}{a_{(n-1)(n-1)}^{(n-2)}} \quad (2.1.6)$$

$$x_k = \frac{b_k^{(k-1)} - \sum_{j=k+1}^n a_{kj}^{(k-1)} x_j}{a_{kk}^{(k-1)}} \quad (2.1.7)$$

2.1.3 Vorsicht

Algorithmen 2.1.1 und 2.1.2 sind nur ausführbar, falls für die sog. **Pivotelemente** $a_{kk}^{(k-1)}$ gilt:

$$a_{kk}^{(k-1)} \neq 0 \quad \text{für } k = 1, \dots, n$$

Dies ist auch für invertierbare Matrizen nicht immer gewährleistet.

2.1.4 Weitere algorithmische Anmerkungen

Matrix A und Vektor b sollten möglichst **nie** überschrieben werden! (Stattdessen kann eine Kopie überschrieben werden.)

Das Aufstellen von A und b ist bei manchen Anwendungen das teuerste, sie gehen sonst verloren. In 2.1.1 wird das obere Dreieck von A überschrieben. Dies ist möglich, da in (2.1.3) nur die Zeilen $k+1, \dots, n$ mithilfe der k -ten bearbeitet werden. Am Ende steht R im oberen Dreieck von A und z in b .

Die l_{ik} werden spaltenweise berechnet und können daher anstelle der entsprechenden Nullen (in der Kopie) von A gespeichert werden, d.h.:

$$\tilde{L} := (l_{ik}) \quad (2.1.8)$$

und R werden sukzessive in A geschrieben.

IMAGE MISSING

Der Vektor z und anschließend der Lösungsvektor x kann in (eine Kopie von) b geschrieben werden. Wird eine neue rechte Seite b betrachtet, muss 2.1.1 nicht komplett neu ausgeführt werden, da sich \tilde{L} nicht ändert. Es reicht 2.1.4 zu wiederholen.

IMAGE MISSING

2.1.5 Dreieckszerlegung

Die Dreieckszerlegung einer Matrix A entspricht dem Verfahren aus 2.1.1, nur ohne die Zeile (2.1.4).

2 Lineare Gleichungssysteme: Direkte Methoden

2.1.6 Vorwärtssubstitution

Die Vorwärtssubstitution entspricht der in 2.1.4 bzw. dem Verfahren aus 2.1.1 ohne die Bestimmung von l_{ik} und R , also nur Schritt 2.1.4.

2.1.7 Gauß-Eliminator zur Lösung von $Ax = b$

1 Dreieckszerlegung

2 Vorwärtssubstitution $b_i^{(k)} = b_i^{(k-1)} - b_k^{(k-1)} \cdot l_{ik}$

3 Rückwärtssubstitution $x_k = \frac{b_k^{(k-1)} - \sum_{j=k+1}^n a_{kj}^{(k-1)} x_j}{a_{kk}^{(k-1)}}$

2.1.8 Rechenaufwand gezählt in „flops“

„flops“ = floating point operations
MISSING

2.1.9 Definition: Landau-Symbole

MISSING

2.1.10 Allgemeines zur Aufwandsbetrachtung

MISSING

2.1.11 Formalisieren des Gauß-Algorithmus

MISSING

2.1.12 Lemma (Eigenschaften der L_k -Matrizen)

MISSING

2.1.13 Satz (LR- oder LU-Zerlegung)

MISSING

2.2 Gaußsches Eliminationsverfahren mit Pivotisierung

MISSING

13.10.2014

2.2.1 Spaltenpivotisierung (=partielle/ halbmaximale Pivotisierung)**2.2.2 Bemerkungen****2.2.3 Satz: Dreieckszerlegung mit Permutationsmatrix****Beweis**15.10.2014
(Fortsetzung)

$$\begin{aligned}
 PA &= LR \\
 R &= A^{(n-1)} \\
 &= L_{n-1} P_{\tau_{n-1}} \dots L_1 P_{\tau_1} A
 \end{aligned}$$

Da τ_i nur zwei Zahlen $\geq i$ vertauscht, ist

15.10.2014

$$\Pi_i := \tau_{n-1} \circ \dots \circ \tau_i \quad \text{für } i = 1, \dots, (n-1)$$

eine Permutation der Zahlen $\{i, \dots, n\}$, d.h. insbesondere gilt:

$$\begin{aligned}
 \Pi_i(j) &= j && \text{für } j = 1, \dots, (i-1) \\
 \Pi_i(j) &\in \{i, \dots, n\} && \text{für } j = i, \dots, n. \\
 P_{\Pi_{i+1}} &= (e_1, \dots, e_i, e_{\Pi_{i+1}(i+1)}, \dots, e_{\Pi_{i+1}(n)}) \\
 &= \begin{pmatrix} I_i & 0 \\ 0 & P_\sigma \end{pmatrix}
 \end{aligned}$$

Damit folgt:

$$\begin{aligned}
 P_{\Pi_{i+1}} \cdot P_{\Pi_{i+1}}^{-1} &= P_{\Pi_{i+1}} \cdot \left(\begin{array}{c|c} I_i & 0 \\ \hline 0 & I_{n-i} \end{array} \begin{array}{c} -l_{i+1,i} \\ \vdots \\ -l_{n,i} \end{array} \right) \cdot \begin{pmatrix} I_i & 0 \\ 0 & P_\sigma^{-1} \end{pmatrix} \\
 &= \begin{pmatrix} I_i & 0 \\ 0 & P_\sigma \end{pmatrix} \cdot \left(\begin{array}{c|c} I_i & 0 \\ \hline 0 & P_\sigma^{-1} \end{array} \begin{array}{c} -l_{i+1,i} \\ \vdots \\ -l_{n,i} \end{array} \right) \\
 &= \left(\begin{array}{c|c} I_i & 0 \\ \hline 0 & I_{n-i} \end{array} \begin{array}{c} -l_{\Pi_{i+1}(i+1),i} \\ \vdots \\ -l_{\Pi_{i+1}(n),i} \end{array} \right) \\
 &= I - (P_{\Pi_{i+1}} l_i) e_i^T \\
 &=: \hat{L}_i
 \end{aligned}$$

und

$$R = L_{n-1}$$

2 Lineare Gleichungssysteme: Direkte Methoden

$$\begin{aligned}
& \cdot (P_{\tau_{n-1}} L_{n-2} P_{\tau_{n-1}}^{-1}) \\
& \cdot (P_{\tau_{n-1}} P_{\tau_{n-2}} L_{n-2} P_{\tau_{n-2}}^{-1} P_{\tau_{n-1}}^{-1}) \\
& \vdots \\
& \cdot (P_{\tau_{n-1}} \cdots P_{\tau_1} L_1 P_{\tau_1} \cdots P_{\tau_{n-1}}) \cdot A \\
& = L_{n-1} \hat{L}_{n-2} \cdots \hat{L}_1 P_{\Pi_1} \cdot A
\end{aligned}$$

Nach Lemma 2.1.12 gilt daher, es existiert eine Permutation Π_1 mit

$$P_{\Pi_1} \cdot A = LR,$$

wobei R obere Dreiecksgestalt hat und

$$L = \begin{pmatrix} 1 & & & 0 \\ l_{\Pi_2(2),1} & \ddots & & \\ \vdots & \ddots & 1 & \\ l_{\Pi_n(n),1} & \cdots & l_{\Pi_n(n),n-1} & 1 \end{pmatrix} \quad \text{mit } |l_{ij}| \leq 1$$

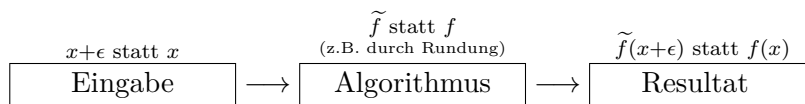
gilt.

2.2.4 Lösen eines Gleichungssystems $Ax = b$

2.2.5 Bemerkungen

2.2.6 Beispiel zur Pivotisierung

3 Fehleranalyse



Bei der Fehleranalyse liegt das Hauptaugenmerk auf

Eingabefehler

z.B. Rundungsfehler, Fehler in Messdaten, Fehler im Modell (falsche Parameter)

Fehler im Algorithmus

z.B. Rundungsfehler durch Rechenoperationen, Approximationen

(z.B. Ableitung durch Differenzenquotient oder die Berechnung von Sinus durch abgebrochene Reihenentwicklung)

1. *Frage* Wie wirken sich Eingabefehler auf das Resultat unabhängig vom gewählten Algorithmus aus?
2. *Frage* Wie wirken sich (Rundungs-)Fehler des Algorithmus aus?
Und wie verstärkt der Algorithmus Eingabefehler?

3.1 Zahlendarstellung und Rundungsfehler

Auf (Digital-)Rechnern können nur endlich viele Zahlen realisiert werden.
Die wichtigsten Typen sind:

- **ganze Zahlen** (integer):

$$z = \pm \sum_{i=0}^m z_i \beta_i \quad \text{mit} \quad \begin{array}{l} \beta = \text{Basis des Zahlensystems (oft } \beta = 2) \\ z_i \in \{0, \dots, \beta - 1\} \end{array}$$

- **Gleitpunktzahlen** (floating point)

3.1.1 Definition: Gleitkommazahl

Eine Zahl $x \in \mathbb{Q}$ mit einer Darstellung

$$x = \sigma \cdot (a_1.a_2 \cdots a_t)_\beta \cdot \beta^e = \sigma \beta^e \cdot \sum_{\nu=1}^t a_\nu \beta^{-\nu+1}$$

$\beta \in \mathbb{N}$	Basis des Zahlensystems
$\sigma \in \{\pm 1\}$	Vorzeichen
$m = (a_1.a_2 \cdots a_t)_\beta$ $= \sum_{\nu=1}^t a_\nu \beta^{-\nu+1}$	Mantisse
$a_i \in \{0, \dots, \beta - 1\}$	Ziffern der Mantisse
$t \in \mathbb{N}$	Mantissenlänge
$e \in \mathbb{Z}$	mit $e_{\min} \leq e \leq e_{\max}$ Exponent

heißt **Gleitkommazahl** mit t Stellen und Exponent e zur Basis b .

Ist $a_1 \neq 0$, so heißt x **normalisierte Gleitkommazahl**

3.1.2 Bemerkung

- a) 0 ist keine normalisierte Gleitkommazahl, da $a_1 = 0$ ist.
- b) $a_1 \neq 0$ stellt sicher, dass die Gleitkommadarstellung eindeutig ist.
- c) In der Praxis werden auch nicht-normalisierte Darstellungen verwendet.
- d) Heutige Rechner verwenden meist $\beta = 2$, aber auch $\beta = 8, \beta = 16$.

3.1.3 Beispiel

bit-Darstellung nach IEEE-Standard 754 von floating point numbers

Sei die Basis $\beta = 2$.

	Speicherplatz	t	e_{\min}	e_{\max}
einfache Genauigkeit (float)	32bits = 4Bytes	24	-126	127
doppelte Genauigkeit (double)	64bits = 8Bytes	52	-1022	1023

Darstellung im Rechner (Bitmuster) für float:

$$\boxed{s \mid b_0 \cdots b_7 \mid a_2 \cdots a_{24}}$$

(Da $a_1 \neq 0$, also $a_1 = 1$ gilt, wird a_1 nicht gespeichert)

Interpretation ($s, b, a_i \in \{0, 1\} \forall i$)

- s Vorzeichenbit: $\sigma = (-1)^s \Rightarrow \begin{matrix} \sigma(0) = 1 \\ \sigma(1) = -1 \end{matrix}$

- $b = \sum_{i=0}^7 b_i \cdot 2^i \in \{1, \dots, 254\}$ speichert den Exponenten mit
 $e = b - \underbrace{127}_{\text{Basiswert}}$ (kein Vorzeichen nötig)

Beachte: $b_0 = \dots = b_7 = 0$ sowie $b_0 = \dots = b_7 = 0$ sind bis auf Ausnahmen keine gültigen Exponenten

- $m = (a_1.a_2 \dots a_{24}) = 1 + \sum_{\nu=2}^{24} a_\nu 2^{1-\nu}$ stellt die Mantisse dar, $a_1 = 1$ wird nicht abgespeichert.

- Besondere Zahlen per Konvention:

$x = 0$: s bel., $b = 0$, $m = 1$

s	$0 \dots 0$	$0 \dots 0$
-----	-------------	-------------

$x = \pm\infty$: s bel., $b = 255$, $m = 1$

s	$1 \dots 1$	$0 \dots 0$
-----	-------------	-------------

$x = \text{NaN}$ s bel., $b = 255$, $m \neq 1$

$x = (-1)^s$ s bel., $b = 0$, $m \neq 1$ und x hat die Form $x = (0 + \sum_{\nu=2}^{24} a_\nu \cdot 2^{1-\nu}) \cdot 2^{126}$
 („denormalized“ number)

20.10.2014

Betragsmäßig **größte Zahl**:

0	$01 \dots 1$	$1 \dots \dots 1$
---	--------------	-------------------

$$x_{\max} = (2 - 2^{-23}) \cdot 2^{127} \approx 3,4 \cdot 10^{38}$$

Betragsmäßig **kleinste Zahl**:

0	$0 \dots 0$	$0 \dots \dots 01$
---	-------------	--------------------

$$x_{\min} = (2 - 2^{-23}) \cdot 2^{-126} = 2^{-149} \approx 1,4 \cdot 10^{-45}$$

3.1.4 Verteilung der Maschinenzahlen

ungleichmäßig im Dezimalsystem, z. B.

$$x = \pm a_1.a_2a_3 \cdot 2^e \quad -2 \leq e \leq 1 \quad a_i \in \{0,1\}$$

IMAGE MISSING

ist im Dualsystem gleichmäßig verteilt.

3.1.5 Bezeichnungen

overflow es ergibt sich eine Zahl, die betragsmäßig größer ist als die größte maschinendarstellbare Zahl

underflow entsprechend, betragsmäßig kleiner als die kleinste positive Zahl

Bsp.: overflow beim integer $b = e + 127$

$$\begin{array}{rcl} b & = & 254 \quad 11111110 \\ & + & 3 \quad 00000011 \\ b + 3 = 257 \bmod 2^8 & = & 1 \quad 100000001 \end{array}$$

3.1.6 Rundungsfehler

Habe $x \in \mathbb{R}$ die normalisierte Darstellung

$$\begin{aligned} x &= \sigma \cdot \beta^e \left(\sum_{\nu=1}^t a_{\nu} \beta^{1-\nu} + \sum_{\nu=t+1}^{\infty} a_{\nu} \beta^{1-\nu} \right) \\ &= \sigma \cdot \beta^e \left(\sum_{\nu=1}^t a_{\nu} \beta^{1-\nu} + \beta^{1-t} \sum_{l=1}^{\infty} a_{t+l} \beta^{-l} \right) \end{aligned}$$

mit $e_{\min} \leq e \leq e_{\max}$, dann wird mit $fl(x)$ die gerundete Zahl bezeichnet, wobei $fl(x)$ eindeutig gegeben ist durch die Schranke an den **absoluten Rundungsfehler**

$$|fl(x) - x| \leq \begin{cases} \frac{1}{2} \beta^{e+1+t} & \text{bei symmetrischem Runden} \\ \beta^{e+1+t} & \text{bei Abschneiden} \end{cases}.$$

Für die **relative Rechengenauigkeit** folgt somit

$$\frac{|fl(x) - x|}{|x|} \leq \begin{cases} \frac{1}{2} \beta^{1-t} & \text{bei symmetrischem Runden} \\ \beta^{1-t} & \text{bei Abschneiden} \end{cases}.$$

Die **Maschinengenauigkeit** des Rechners ist daher durch

$$eps = \beta^{1-t} \quad (\text{für float} \approx 10^{-7}, \text{ für double} \approx 10^{-16})$$

gegeben.

Die Mantissenlänge bestimmt also die Maschinengenauigkeit. Bei einfacher Genauigkeit ist $fl(x)$ bis auf ungefähr 7 signifikante Stellen genau.

Im Folgenden betrachten wir symmetrisches Runden und definieren daher

$$\tau := \frac{1}{2} eps$$

Weiterhin gilt:

- a) Die kleinste Zahl am Rechner, welche größer als 1 ist, ist

$$1 + eps$$

- b) Eine Maschinenzahl x repräsentiert eine Eingabemenge

$$E(x) = \{\tilde{x} \in \mathbb{R} : |\tilde{x} - x| \leq \tau |x|\}$$

IMAGE MISSING

3.1.7 Bemerkung

Gesetze der arithmetischen Operationen gelten i.A. nicht, z.B.

- x Maschinenzahl $\Rightarrow fl(x + \nu) = x$ für $|\nu| < \tau|x|$
- Assoziativ- und Distributivgesetze gelten nicht, z.B. für $\beta = 10$, $t = 3$, $a = 0,1$, $b = 105$, $c = -104$ gilt:

$$\begin{aligned} fl(a + fl(a + c)) &= 1,1 \\ fl(fl(a + b) + c) &= fl(fl(105, 1) + (-104)) \\ &= fl(105 - 104) \\ &= 1 \quad \nabla \end{aligned}$$

\Rightarrow Für einen Algorithmus ist die Reihenfolge der Operationen wesentlich! Mathematisch äquivalente Formulierungen können zu verschiedenen Ergebnissen führen.

3.1.8 Auslöschung von signifikanten Stellen

Sei $x = 9,995 \cdot 10^{-1}$, $y = 9,984 \cdot 10^{-1}$. Runde auf drei signifikante Stellen und berechne $x - y$:

$$\begin{aligned} \tilde{f}(x, y) &:= fl(fl(x) - fl(y)) = fl(1,00 \cdot 10^0 - 9,98 \cdot 10^{-1}) \\ &= fl(0,02 \cdot 10^{-1}) \\ &= fl(2,00 \cdot 10^{-3}) \\ f(x, y) &:= x - y \\ &:= 0,0011 = 1,1 \cdot 10^{-3} \end{aligned}$$

Daraus ergibt sich der relative Fehler

$$\frac{|\tilde{f}(x, y) - f(x, y)|}{|f(x, y)|} = \frac{|2 \cdot 10^{-3} - 1,1 \cdot 10^{-3}|}{|1,1 \cdot 10^{-3}|} = 82\%$$

Der Grund hierfür ist, dass das Problem der Subtraktion zweier annähernd gleich großer Zahlen schlecht konditioniert ist.

Zwei Regeln:

- 1) Umgehbare Subtraktion annähernd gleich großer Zahlen vermeiden!
- 2) Unumgängliche Subtraktion möglichst an den Anfang des Algorithmus stellen! (siehe später)

3.2 Kondition eines Problems

Es wird das Verhältnis

$$\frac{\text{Ausgabefehler}}{\text{Eingabefehler}}$$

untersucht.

3.2.1 Definition: Problem

Sei $f : U \subseteq \mathbb{R}^n \mapsto \mathbb{R}^m$ mit U offen und sei $x \in U$. Dann bezeichne (f, x) das Problem, zu einem gegebenen x die Lösung $f(x)$ zu finden.

3.2.2 Definition: absoluter und relativer Fehler

Sei $x \in \mathbb{R}^n$ und $\tilde{x} \in \mathbb{R}^n$ eine Näherung an x . Weiterhin sei $\|\cdot\|$ eine Norm auf \mathbb{R}^n .

a) $\|\tilde{x} - x\|$ heißt **absoluter Fehler**

b) $\frac{\|\tilde{x} - x\|}{\|x\|}$ heißt **relativer Fehler**

Da der relative Fehler skalierungsinvariant ist, d.h. unabhängig von der Wahl von x ist, ist dieser i.d.R. von größerem Interesse. Beide Fehler hängen von der Wahl der Norm ab! Häufig werden Fehler auch komponentenweise gemessen:

$$\begin{array}{lll} \text{Für } i = 1, \dots, n : & |\tilde{x}_i - x_i| \leq \delta & (\text{absolut}) \\ & |\tilde{x}_i - x_i| \leq \delta |x_i| & (\text{relativ}) \end{array}$$

3.2.3 Wiederholung: Normen

Euklidische Norm (l_2 -Norm):	$\ x\ _2 := \sqrt{\sum_{i=1}^n x_i ^2}$
<i>IMAGE MISSING</i>	
Maximumsnorm (l_∞ -Norm):	$\ x\ _\infty := \max\{ x_i : i = 1, \dots, n\}$
<i>IMAGE MISSING</i>	
Summennorm (l_1 -Norm):	$\ x\ _1 := \sum_{i=1}^n x_i $
<i>IMAGE MISSING</i>	
Hölder-Norm (l_p -Norm):	$\ x\ _p := \left(\sum_{i=1}^n x_i ^p \right)^{\frac{1}{p}}$

3.2.4 Definition: Matrixnorm

Auf dem \mathbb{R}^n sei die Norm $\|\cdot\|_a$ und auf dem \mathbb{R}^m die Norm $\|\cdot\|_b$ gegeben. Dann ist die zugehörige **Matrixnorm** gegeben durch:

$$\begin{aligned}\|A\|_{a,b} &:= \sup_{x \neq 0} \frac{\|Ax\|_b}{\|x\|_a} \\ &= \sup_{\|x\|_a=1} \|Ax\|_b\end{aligned}\tag{3.2.1}$$

Also ist $\|A\|_{a,b}$ die kleinste Zahl $c > 0$ mit

$$\|Ax\|_b \leq c \|x\|_a \quad \forall x \in \mathbb{R}^n$$

3.2.5 Definition: Frobeniusnorm, p-Norm, Verträglichkeit

Sei $A \in \mathbb{R}^{m \times n}$.

- a) **Frobeniusnorm** (Schurnorm): $\|A\|_F := \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}^2|}$
- b) **p-Norm**: $\|A\|_p := \|A\|_{p,p}$
- c) Eine Matrixnorm heißt **verträglich** mit den Vektornormen $\|\cdot\|_a, \|\cdot\|_b$, falls gilt ¹:

$$\|Ax\|_b \leq \|A\| \cdot \|x\|_a \quad \forall x \in \mathbb{R}^n$$

3.2.6 Bemerkungen

- a) Die Normen $\|\cdot\|_F$ und $\|\cdot\|_p$ sind **submultiplikativ**, d.h.

$$\|A \cdot B\| \leq \|A\| \cdot \|B\|$$

- b) Die Norm $\|\cdot\|_{1,1}$ wird auch **Spaltensummennorm** genannt:

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^m |a_{ij}|$$

Sie ist das Maximum der Spaltensummen².

- c) Die Norm $\|\cdot\|_{\infty,\infty}$ wird auch **Zeilensummennorm** genannt³:

$$\|A\|_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^n |a_{ij}|$$

¹ Beachte: $\|A\|_{a,b}$ ist die kleinste Norm im Gegensatz zu $\|A\|$, welche hier beliebig ist.

²Beweis: siehe Übungsblatt 3

³Beweis: siehe Übungsblatt 3

3 Fehleranalyse

- d) Die Frobeniusnorm $\|\cdot\|_F$ ist verträglich mit der euklidischen Norm $\|\cdot\|_2$
- e) Die Wurzeln aus den Eigenwerten von $A^T A$ heißen **Singulärwerte** σ_i von A. Mit ihnen kann die $\|\cdot\|_{2,2}$ Norm dargestellt werden⁴:

$$\begin{aligned}\|A\|_2 &= \max\{\sqrt{\mu} : A^T A \cdot x = \mu x \text{ für ein } x \neq 0\} \\ &= \sigma_{\max}\end{aligned}$$

3.2 a) Normweise Konditionsanalyse

3.2.7 Definition: absolute und relative normweise Kondition

Sei (f, x) ein Problem mit $f : U \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$ und $\|\cdot\|_a$ auf \mathbb{R}^n und $\|\cdot\|_b$ auf \mathbb{R}^m eine Norm.

- a) Die **absolute normweise Kondition** eines Problems (f, x) ist die kleinste Zahl $\kappa_{abs} > 0$ mit

$$\begin{aligned}\|f(\tilde{x}) - f(x)\|_b &\leq \kappa_{abs}(f, x) \|\tilde{x} - x\|_a + o(\|\tilde{x} - x\|_a) \\ \left(f(\tilde{x}) - f(x) = \underbrace{f'(x)(\tilde{x} - x)}_{\text{Taylorentwicklung}} \pm o(\|\tilde{x} - x\|) \right) &\text{ für } \tilde{x} \rightarrow x\end{aligned}\tag{3.2.2}$$

- b) Die **relative normweise Kondition** eines Problems (f, x) mit $x \neq 0, f(x) \neq 0$ ist die kleinste Zahl $\kappa_{rel} > 0$ mit

$$\frac{\|f(\tilde{x}) - f(x)\|_b}{\|f(x)\|_b} \leq \kappa_{rel}(f, x) \frac{\|\tilde{x} - x\|_a}{\|x\|_a} + o\left(\frac{\|\tilde{x} - x\|_a}{\|x\|_a}\right) \quad \text{für } \tilde{x} \rightarrow x\tag{3.2.3}$$

- c) Sprechweise:

- falls κ „klein“ ist, ist das Problem „gut konditioniert“
- falls κ „groß“ ist, ist das Problem „schlecht konditioniert“

3.2.8 Lemma

Falls f differenzierbar ist, gilt

$$\kappa_{abs}(f, x) = \|Df(x)\|_{a,b}\tag{3.2.4}$$

und für $f(x) \neq 0$

$$\kappa_{rel}(f, x) = \frac{\|x\|_a}{\|f(x)\|_b} \cdot \|Df(x)\|_{a,b}\tag{3.2.5}$$

wobei $Df(x)$ die Jakobi-Matrix bezeichnet.

⁴Beweis: siehe Übungsblatt 3

3.2.9 Beispiel: Kondition der Addition

$f(x_1, x_2) := x_1 + x_2, f : \mathbb{R}^2 \rightarrow \mathbb{R}.$

Wähle l_1 -Norm auf \mathbb{R}^2 (und \mathbb{R})

$$\begin{aligned} Df(x_1, x_2) &= (\nabla f^T) = \left(\frac{\partial}{\partial x_1} f, \frac{\partial}{\partial x_2} f \right) \\ &= (1, 1) \end{aligned} \quad (\text{Matrix!})$$

damit

$$\begin{aligned} \kappa_{abs}(f, x) &= \|Df(x)\|_{1,1} && (\text{Matrix-Norm!!}) \\ &= \|Df(x)\|_1 \\ &= 1 \\ \kappa_{rel}(f, x) &= \frac{\|x\|_1}{\|f(x)\|_1} \cdot \|Df(x)\|_1 \\ &= \frac{|x_1| + |x_2|}{|x_1 + x_2|} \end{aligned}$$

Daraus folgt: Die Addition zweier Zahlen mit gleichem Vorzeichen ergibt

$$\kappa_{rel} = 1$$

Die Subtraktion zweier annähernd gleich großer Zahlen ergibt eine sehr schlechte relative Konditionierung:

$$\kappa_{rel} \gg 1$$

Zum Beispiel in 3.1.8: Es ist

$$\begin{aligned} x &= \begin{pmatrix} 9,995 \\ -9,984 \end{pmatrix} \cdot 10^{-1} \\ \tilde{x} = fl(x) &= \begin{pmatrix} 1 \\ -9,98 \cdot 10^{-1} \end{pmatrix} \end{aligned}$$

also

$$\begin{aligned} \frac{|f(\tilde{x}) - f(x)|}{|f(x)|} &= \frac{0,9}{1,1} = 0,8\overline{1} \\ &\leq \kappa_{rel}(f, x) \cdot \frac{\|\tilde{x} - x\|_1}{\|x\|_1} \\ &= \kappa_{rel}(f, x) \cdot 4,6 \cdot 10^{-4} \end{aligned}$$

3.2.10 Beispiel: Lösen eines Gleichungssystems

Sei $A \in \mathbb{R}^{n \times n}$ invertierbar und $b \in \mathbb{R}^n$. Es soll

$$Ax = b$$

gelöst werden. Die möglichen Lösungen in A und in b lassen sich folgendermaßen ermitteln:

- a) Betrachte die Störungen in b :
Sei hierzu

$$f : b \mapsto x = A^{-1}b$$

Berechne dann $\kappa(f, b)$ und löse

$$\begin{aligned} A(x + \Delta x) &= b + \Delta b \\ f(b + \Delta b) - f(b) &= \Delta x \\ &= A^{-1} \cdot \Delta b && \text{da } x = A^{-1}b \\ \Rightarrow \|\Delta x\|_b &= \|A^{-1} \Delta b\|_b \\ &\leq \|A^{-1}\|_{a,b} \cdot \|\Delta b\|_b && \forall b, \Delta b \end{aligned}$$

wobei $\|\cdot\|$ auf $\mathbb{R}^{n \times n}$ die dem \mathbb{R}^n zugeordnete Matrix-Norm sei.

Die Abschätzung ist **scharf**, d.h. es gibt ein $\Delta b \in \mathbb{R}^n$, so dass „ $=$ “ gilt, nach Definition 3.2.4.

Also gilt⁵:

$$\kappa_{abs}(f, b) = \|A^{-1}\|_{a,b} \quad (3.2.6)$$

unabhängig von b . ($x \mapsto Ax \quad \kappa_{abs}$)
Ebenso folgt die scharfe Abschätzung

$$\begin{aligned} \frac{\|f(b + \Delta b) - f(b)\|}{\|f(b)\|} &= \frac{\|\Delta x\|}{\|x\|} \\ &= \frac{\|A^{-1} \Delta b\|}{\|x\|} \\ &\leq \frac{\|A^{-1}\| \cdot \|b\|}{\|x\|} \cdot \frac{\|\Delta b\|}{\|b\|} \end{aligned}$$

Damit

$$\kappa_{rel}(f, b) = \|A^{-1}\| \cdot \frac{\|b\|}{\|A^{-1} \cdot b\|} \quad (3.2.7)$$

⁵vgl. auch Lemma 3.2.8: $\kappa_{abs}(f, b) = \|Df(b)\|_{a,b} = \|A^{-1}\|_{a,b}$

Da $\|b\| \leq \|A\| \cdot \|x\| = \|A\| \cdot \|A^{-1}b\|$ folgt:

$$\kappa_{rel}(f, b) \leq \|A\| \cdot \|A^{-1}\| \quad (3.2.8)$$

für alle (möglichen rechten Seiten) b .

3.2.8 ist scharf in dem Sinne, dass es ein $\hat{b} \in \mathbb{R}^n$ gibt mit

$$\|\hat{b}\| = \|A\| \cdot \|\hat{x}\|$$

und somit

$$\kappa_{rel}(f, \hat{b}) = \|A\| \cdot \|A^{-1}\| \quad (3.2.9)$$

b) Betrachte die Störungen in A :

Löse also

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

Sei hierzu

$$\begin{aligned} f : A &\mapsto x = A^{-1}b \\ \mathbb{R}^{n \times n} &\rightarrow \mathbb{R}^n \end{aligned}$$

und berechne $\kappa(f, A)$ mittels Ableitung $Df(A) : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^n$:

$$\begin{aligned} C \mapsto Df(A)C &= \frac{d}{dt} \left((A + tC)^{-1} \cdot b \right) \Big|_{t=0} \\ &= \frac{d}{dt} \left((A + tC)^{-1} \right) \Big|_{t=0} \cdot b \end{aligned}$$

Weiterhin gilt

$$\frac{d}{dt} \left((A + tC)^{-1} \right) \Big|_{t=0} = -A^{-1}CA^{-1},$$

da

$$\begin{aligned} 0 &= \frac{d}{dt} I \\ &= \frac{d}{dt} \left((A + tC)(A + tC)^{-1} \right) \\ &= C(A + tC)^{-1} + (A + tC) \cdot \frac{d}{dt} (A + tC)^{-1} \\ \Leftrightarrow \frac{d}{dt} (A + tC)^{-1} &= -(A + tC)^{-1} \cdot C(A + tC)^{-1}, \end{aligned}$$

falls $(A + tC)$ invertierbar ist. Für ein genügend kleines t ist das gewährleistet, da A invertierbar ist (s. Lemma 3.2.12).

$$\Rightarrow Df(A)C = -A^{-1}CA^{-1}b$$

3 Fehleranalyse

Somit folgt

$$\begin{aligned}
\kappa_{abs}(f, A) &= \|Df(A)\| \\
&= \sup_{\substack{C \neq 0 \\ C \in \mathbb{R}^{n \times n}}} \frac{\|A^{-1}CA^{-1}b\|}{\|C\|} \\
&\leq \sup_{\substack{C \neq 0 \\ C \in \mathbb{R}^{n \times n}}} \frac{\|A^{-1}\| \cdot \|C\| \cdot \|A^{-1}b\|}{\|C\|} \\
&= \|A^{-1}\| \cdot \|x\| \\
&\leq \|A^{-1}\|^2 \cdot \|b\| \\
\kappa_{rel}(f, A) &= \frac{\|A\|}{\|f(A)\|} \cdot \|Df(A)\| \\
&\leq \|A\| \cdot \|A^{-1}\|
\end{aligned} \tag{3.2.10}$$

c) betrachte Störungen in A und b :

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

Für κ müsste $\|(A, b)\|$ festgelegt werden. Dies wird jedoch nicht betrachtet. Es gilt aber folgende Abschätzung für invertierbare Matrizen $A \in \mathbb{R}^{n \times n}$ und Störungen $\Delta A \in \mathbb{R}^{n \times n}$ mit $\|A^{-1}\| \cdot \|\Delta A\| < 1$:

$$\frac{\|\Delta x\|}{\|x\|} \leq \|A\| \cdot \|A^{-1}\| \cdot (1 - \|A^{-1}\| \cdot \|\Delta A\|) \cdot \underbrace{\left(\frac{\|\Delta b\|}{\|b\|} + \frac{\|\Delta A\|}{\|A\|} \right)}_{\neq \frac{\|(\Delta A, \Delta b)\|}{\|(A, b)\|}} \tag{3.2.11}$$

Beweis: s. Übungsblatt

3.2.11 Definition: Kondition einer Matrix

Sei $\|\cdot\|$ eine Norm auf $\mathbb{R}^{n \times n}$ und $A \in \mathbb{R}^{n \times n}$ eine reguläre Matrix. Die Größe

$$\kappa_{\|\cdot\|}(A) = \text{cond}_{\|\cdot\|}(A) := \|A\| \cdot \|A^{-1}\|$$

heißt **Kondition der Matrix** bzgl. der Norm $\|\cdot\|$.

Ist $\|\cdot\|$ von einer Vektor-Norm $\|\cdot\|_p$ induziert, bezeichnet $\text{cond}_p(A)$ die $\text{cond}_{\|\cdot\|_p}(A)$. Wir schreiben $\text{cond}(A)$ für $\text{cond}_2(A)$.

$\text{cond}_{\|\cdot\|}(A)$ schätzt die relative Kondition eines linearen GLS $Ax = b$ für alle möglichen Störungen in b oder in A ab und diese Abschätzung ist scharf.

Es stellt sich nun die Frage:

Wann existiert die Inverse der gestörten invertierbaren Matrix A ?

Hierzu werden wir die Relationen benötigen:

$$A + \Delta A = A(I + A^{-1}\Delta A)$$

und mit $C \in \mathbb{R}^{n \times n}$, $\|C\| < 1$

$$(I - C)^{-1} = \sum_{k=0}^{\infty} C^k$$

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

27.10.2014

3.2.12 Lemma (Neumannsche Reihe)

Sei $C \in \mathbb{R}^{n \times n}$ mit $\|C\| < 1$ und mit einer submultiplikativen Norm $\|\cdot\|$, so ist $(I - C)$ invertierbar und es gilt:

$$(I - C)^{-1} = \sum_{k=0}^{\infty} C^k$$

Weiterhin gilt:

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

Beweis Es gilt zu zeigen, dass $\sum_{k=1}^{\infty} C^k$ existiert:

Sei $q := \|C\| < 1$, dann gilt:

$$\begin{aligned} \left\| \sum_{k=0}^m C^k \right\| &\leq \sum_{k=0}^m \|C^k\| && \text{Dreiecksungleichung} \\ &\leq \sum_{k=0}^m \|C\|^k && \text{da } \|\cdot\| \text{ submultiplikativ} \\ &= \sum_{k=0}^m q^k \\ &= \frac{1 - q^{m+1}}{1 - q} \\ &\leq \frac{1}{1 - \|C\|} && \forall m \in \mathbb{N}, \text{ da } q < 1 \text{ (geometr. Reihe)} \end{aligned}$$

Daraus folgt bereits, dass $\sum_{k=1}^{\infty} C^k$ existiert (nach Majorantenkriterium).

Weiter gilt dann:

$$\begin{aligned} (I - C) \sum_{k=1}^{\infty} C^k &= \lim_{m \rightarrow \infty} (I - C) \sum_{k=1}^m C^k \\ &= \lim_{m \rightarrow \infty} (C^0 - C^{m+1}) \\ &= I \end{aligned} \quad \square$$

3.2.13 Bemerkung

a) Für symmetrische, positiv definite Matrix $A \in \mathbb{R}^{n \times n}$ gilt⁶:

$$\kappa_2(A) = \frac{\lambda_{max}}{\lambda_{min}} \quad (3.2.13)$$

b) Eine andere Darstellung von $\kappa(A)$ ist

$$\kappa(A) := \frac{\max_{\|x\|=1} \|Ax\|}{\min_{\|x\|=1} \|Ax\|} \in [0, \infty] \quad (3.2.14)$$

Diese ist auch für nicht invertierbare und rechteckige Matrizen wohldefiniert.
Dann gilt offensichtlich:

c) $\kappa(A) \geq 1$

d) $\kappa(\alpha A) = \kappa(A)$ für $0 \neq \alpha \in \mathbb{R}$ (skalierungsinvariant)

e) $A \neq 0$ und $A \in \mathbb{R}^{n \times n}$ ist genau dann singular, wenn $\kappa(A) = \infty$.

Wegen der Skalierungsinvarianz ist die Kondition zur Überprüfung der Regularität von A besser geeignet als die Determinante.

3.2.14 Beispiel: Kondition eines nichtlinearen Gleichungssystems

Sei $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ stetig differenzierbar und $y \in \mathbb{R}^n$ gegeben.

Löse

$$f(x) = y$$

Gesucht:

$$\kappa(f^{-1}, y)$$

mit f^{-1} Ausgabe und y Eingabe.

Sei $Df(x)$ invertierbar, dann existiert aufgrund des Satzes für implizite Funktionen die inverse Funktion f^{-1} lokal in einer Umgebung von y mit $f^{-1}(y) = x$, sowie

$$D(f^{-1})(y) = (Df(x))^{-1}$$

Hiermit folgt:

$$\begin{aligned} \kappa_{abs}(f^{-1}, y) &= \|(Df(x))^{-1}\| \\ \kappa_{rel}(f^{-1}, y) &= \frac{\|f(x)\|}{\|x\|} \cdot \|(Df(x))^{-1}\| \end{aligned} \quad (3.2.15)$$

⁶Beweis: siehe Übungsblatt 3

Für skalare Funktionen $f : \mathbb{R} \rightarrow \mathbb{R}$ folgt somit:

$$\kappa_{rel}(f^{-1}, y) = \frac{|f(x)|}{|x|} \cdot \frac{1}{|f'(x)|}$$

Falls $|f'(x)| \rightarrow 0$ ist es eine schlechte absolute Kondition.

Für $|f'(x)| \gg 0$ ist es eine gute absolute Kondition.

IMAGE MISSING

Damit bedeutet eine kleine Störung in y eine große Störung in x .

3.2 a) Komponentenweise Konditionsanalyse

3.2.15 Beispiel

Falls A Diagonalmatrix hat, sind die Gleichungen unabhängig voneinander (entkoppelt). Die erwartete relative Kondition wäre dann – wie bei skalaren Gleichungen – stets gleich 1. Ebenso sind Störungen nur in der Diagonale zu erwarten. Jedoch:

$$\begin{aligned} A &= \begin{pmatrix} 1 & 0 \\ 0 & \epsilon \end{pmatrix} \\ \Rightarrow A^{-1} &= \begin{pmatrix} 1 & 0 \\ 0 & \epsilon^{-1} \end{pmatrix} \\ \Rightarrow \kappa_\infty = \kappa_2 &= \frac{1}{\epsilon} \quad \text{für } 0 < \epsilon \leq 1 \end{aligned}$$

3.2.16 Definition: Komponentenweise Kondition

Sei (f, x) ein Problem mit $f(x) \neq 0$ und $x = (x_i)_{i=1, \dots, n}$ mit $x_i \neq 0$ für alle $i = 1, \dots, n$. Die **komponentenweise Kondition** von (f, x) ist die kleinste Zahl $\kappa_{rel} \geq 0$, so dass:

$$\frac{\|f(\tilde{x}) - f(x)\|_\infty}{\|f(x)\|_\infty} \leq \kappa_{rel} \cdot \max_i \frac{|\tilde{x}_i - x_i|}{|x_i|} + o\left(\max_i \frac{|\tilde{x}_i - x_i|}{|x_i|}\right) \quad \text{für } \tilde{x} \rightarrow x$$

Vorsicht:

$$\frac{\|\tilde{x} - x\|_\infty}{\|x\|_\infty} \neq \max_i \frac{|\tilde{x}_i - x_i|}{|x_i|}$$

3.2.17 Lemma

Sei f differenzierbar und fasse $|\cdot|$ komponentenweise auf, d.h. $|x| = \begin{pmatrix} |x_1| \\ \vdots \\ |x_n| \end{pmatrix}$. Dann gilt:

$$\kappa_{rel} = \frac{\| \|Df(x)\| \cdot |x| \|_\infty}{\|f(x)\|_\infty} \quad (3.2.16)$$

3 Fehleranalyse

Beweis Vergleiche seien ebenfalls komponentenweise zu verstehen.
Nach dem Satz von Taylor gilt:

$$\begin{aligned}
 f_i(\tilde{x}) - f_i(x) &= \left(\frac{\partial f_i}{\partial x_1}(x), \dots, \frac{\partial f_i}{\partial x_n}(x) \right) \cdot \begin{pmatrix} \tilde{x}_1 - x_1 \\ \vdots \\ \tilde{x}_n - x_n \end{pmatrix} + o(\|\tilde{x} - x\|) \\
 \Rightarrow |f_i(\tilde{x}) - f_i(x)| &\leq |Df(x)| \cdot \begin{pmatrix} |x_1| \cdot \frac{\tilde{x}_1 - x_1}{|x_1|} \\ \vdots \\ |x_n| \cdot \frac{\tilde{x}_n - x_n}{|x_n|} \end{pmatrix} + o\left(\max_i \frac{\tilde{x}_i - x_i}{|x_i|}\right) \quad \text{da } x_i \text{ fest und } \tilde{x}_i \rightarrow x_i \\
 &\leq |Df(x)| \cdot |x| \cdot \max_i \frac{\tilde{x}_i - x_i}{|x_i|} + o\left(\max_i \frac{\tilde{x}_i - x_i}{|x_i|}\right) \\
 \Rightarrow \frac{\|f(\tilde{x}) - f(x)\|_\infty}{\|f(x)\|_\infty} &\leq \frac{\| |Df(x)| \cdot |x| \|_\infty}{\|f(x)\|_\infty} \cdot \max_i \frac{\tilde{x}_i - x_i}{|x_i|} + o\left(\max_i \frac{\tilde{x}_i - x_i}{|x_i|}\right)
 \end{aligned}$$

Wähle $\tilde{x}_i = x_j + h \cdot \text{sign} \frac{\partial f_i}{\partial x_j}(x)$ mit $h > 0$, dann gilt:

$$|Df_i(x)(\tilde{x} - x)| = Df_i(x)(\tilde{x} - x)$$

und in obiger Rechnung gilt Gleichheit.

Also folgt, dass

$$\frac{\| |Df(x)| \cdot |x| \|_\infty}{\|f(x)\|_\infty} = \kappa_{rel}$$

□

3.2.18 Beispiel

a) Komponentenweise Kondition der Multiplikation

$$\begin{aligned}
 f: \mathbb{R}^2 &\rightarrow \mathbb{R}, f(x, y) := x \cdot y \\
 \Rightarrow Df(x, y) &= (y, x) \\
 \Rightarrow \kappa_{rel}(x, y) &= \frac{\left\| (|y|, |x|) \cdot \begin{pmatrix} |x| \\ |y| \end{pmatrix} \right\|_\infty}{|x \cdot y|} \\
 &= \frac{2 \cdot |x| \cdot |y|}{|x \cdot y|} \\
 &= 2
 \end{aligned}$$

b) Komponentenweise Kondition eines linearen Gleichungssystems:

Löse $Ax = b$ mit möglichen Störungen in b , also zu

$$f: b \mapsto A^{-1}b$$

3.3 Stabilität von Algorithmen

$$\kappa_{rel} = \frac{\| |A^{-1}| \cdot |b| \|_{\infty}}{\|A^{-1}b\|_{\infty}}$$

Falls A eine Diagonalmatrix ist, folgt:

$$\kappa_{rel} = 1$$

c) Komponentenweise Kondition des Skalarproduktes:

$$\begin{aligned} \langle x, y \rangle &:= \sum_{i=1}^n x_i y_i = x^T y \\ f: \mathbb{R}^2 &\rightarrow \mathbb{R}, f(x, y) = \langle x, y \rangle \\ \Rightarrow Df(x, y) &= (y^T, x^T) \\ \kappa_{rel} &= \frac{\left\| \left| (y^T, x^T) \right| \cdot \left| \begin{pmatrix} x \\ y \end{pmatrix} \right| \right\|_{\infty}}{\|\langle x, y \rangle\|_{\infty}} \\ &= \frac{2 \cdot |y^T| \cdot |x|}{|\langle x, y \rangle|} \\ &= 2 \cdot \frac{\langle |x|, |y| \rangle}{|\langle x, y \rangle|} \\ &= 2 \cdot \frac{\cos(|x|, |y|)}{\cos(x, y)} \end{aligned}$$

$$\text{da } \cos(x, y) = \frac{\langle y, x \rangle}{\|x\|_2 \cdot \|y\|_2}.$$

Falls x und y nahezu senkrecht aufeinander stehen, kann das Skalarprodukt sehr schlecht konditioniert sein.

Zum Beispiel für $x = \tilde{x} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ und $y = \begin{pmatrix} 1 + 10^{-10} \\ -1 \end{pmatrix}$, $\tilde{y} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$.

IMAGE MISSING

3.3 Stabilität von Algorithmen

Bislang: Kondition eines gegebenen Problems (f, x) .

Nun stellt sich die Frage: *Was passiert durch das Implementieren am Rechner?*

Ein „stabiler“ Algorithmus sollte ein gut konditioniertes Problem nicht „kaputt machen“.

IMAGE MISSING

3.3 a) Vorwärtsanalyse

Die Fehlerfortpflanzung durch die einzelnen Rechenschritte, aus denen die Implementierung aufgebaut ist, wird abgeschätzt.

3.3.1 Bemerkung

Für die Rechenoperationen $+$, $-$, \cdot , $/$, kurz ∇ , gilt:

$$\begin{aligned} fl(a \nabla b) &= (a \nabla b) \cdot (1 + \epsilon) \\ &= (a \nabla b) \cdot \frac{1}{1 + \mu} \end{aligned} \quad (3.3.17)$$

29.10.2014

mit $|\epsilon|, |\mu| \leq eps$.

3.3.2 Beispiel

Sei $f(x_1, x_2, x_3) := \frac{x_1 x_2}{x_3}$ mit Maschinenzahlen x_i und $x_3 \neq 0$ und sei der Algorithmus durch

$$f(x_1, x_2, x_3) = (f^{(2)} \circ f^{(1)})(x_1, x_2, x_3)$$

gegeben mit

$$\begin{aligned} f^{(1)}(x_1, x_2, x_3) &= (x_1 \cot x_2, x_3) \quad \text{und} \\ f^{(2)}(y, z) &= \frac{y}{z} \end{aligned}$$

Die Implementierung \tilde{f} von f beinhaltet Rundungsfehler.

Sei $x = (x_1, x_2, x_3)$. Daraus folgt:

$$\begin{aligned} \tilde{f}^{(1)}(x) &= (fl(x_1 \cdot x_2), x_3) \\ &= (x_1 x_2 (1 + \epsilon_1), x_3) \end{aligned}$$

mit $|\epsilon_1| \leq eps$:

$$\begin{aligned} \tilde{f}(x) &= \tilde{f}^{(2)}(\tilde{f}^{(1)}(x)) \\ &= fl(f^{(2)}(x_1 x_2 (1 + \epsilon_1), x_3)) \\ &= \frac{x_1 x_2 (1 + \epsilon_1)}{x_3} \cdot (1 + \epsilon_2) \\ &= f(x) \cdot (1 + \epsilon_1)(1 + \epsilon_2) \end{aligned}$$

mit $|\epsilon_2| \leq eps$:

$$\begin{aligned} \frac{|\tilde{f}(x) - f(x)|}{|f(x)|} &= |\epsilon_1 + \epsilon_2 + \epsilon_1 \cdot \epsilon_2| \\ &\leq 2eps + eps^2 \end{aligned}$$

Dies ist eine „worst case“ Analyse, da immer der maximale Fehler angenommen wird, und gibt i.d.R. eine starke Überschätzung des Fehlers an. Für qualitative Aussagen sind sie jedoch unnützlich.

In Computersystemen stehen mehr Operationen wie ∇ zur Verfügung, die mit einer relativen Genauigkeit eps realisiert werden können.

Daher:

3.3.3 Definition:

Eine Abbildung $\phi : U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ heißt **elementar ausführbar**, falls es eine elementare Operation $\tilde{\phi} : \mathbb{F}^n \rightarrow \mathbb{F}^m$ gibt, wobei \mathbb{F} die Menge der Maschinenzahlen bezeichne mit

$$|\tilde{\phi}_i(x) - \phi_i(x)| \leq \text{eps} |\phi_i(x)| \quad \forall x \in \mathbb{F}^n \text{ und } i = 1, \dots, m. \quad (3.3.18)$$

$\tilde{\phi}$ heißt dann **Realisierung** von ϕ .

Bemerkung: aus (3.3.18) folgt für $1 \leq p \leq \infty$:

$$\left\| \tilde{\phi}(x) - \phi(x) \right\|_p \leq \text{eps} \|\phi(x)\|_n \quad \forall x \in \mathbb{F}^n \quad (3.3.19)$$

3.3.4 Definition:

Sei $f : E \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ gegeben.

Ein Tupel $(f^{(1)}, \dots, f^{(l)})$ mit $l \in \mathbb{N}$ von elementar ausführbaren Abbildungen

$$f^{(i)} : U_i \subseteq \mathbb{R}^{k_i} \rightarrow U_{i+1} \subseteq \mathbb{R}^{k_{i+1}}$$

mit $k_1 = n$ und $k_{l+1} = m$ heißt **Algorithmus** von f , falls

$$f = f^{(l)} \circ \dots \circ f^{(1)}$$

Das Tupel $(\tilde{f}^{(1)}, \dots, \tilde{f}^{(l)})$ mit Abbildungen $\tilde{f}^{(i)}$, welche Realisierungen der $f^{(i)}$ sind, heißt **Implementation** von $(f^{(1)}, \dots, f^{(l)})$. Die Komposition

$$\tilde{f} = \tilde{f}^{(l)} \circ \dots \circ \tilde{f}^{(1)}$$

heißt Implementation von f .

Im Allgemeinen gibt es verschiedene Implementierungen einer Abbildung f .

3.3.5 Lemma (Fehlerfortpflanzung)

Sei $x \in \mathbb{R}^n$ und $\tilde{x} \in \mathbb{F}^n$ mit $|\tilde{x}_i - x_i| \leq \text{eps} |x_i|$ für alle $i = 1, \dots, n$. Sei $(f^{(1)}, \dots, f^{(l)})$ ein Algorithmus für f und $(\tilde{f}^{(1)}, \dots, \tilde{f}^{(l)})$ eine zugehörige Implementation. Mit den Abkürzungen

$$\begin{aligned} x^{(j+1)} &:= f^{(j)} \circ \dots \circ f^{(1)} \\ x^{(1)} &:= x \end{aligned}$$

und entsprechend mit $\tilde{x}^{(j+1)}$ gilt, falls $x^{(j+1)} \neq 0$ für alle $j = 0, \dots, (l-1)$ und $\|\cdot\|$ eine beliebige p-Norm ist:

$$\frac{\left\| \tilde{x}^{(j+1)} - x^{(j+1)} \right\|}{\left\| x^{(j+1)} \right\|} \leq \text{eps} \cdot \mathcal{K} + o(\text{eps}) \quad (3.3.20)$$

3 Fehleranalyse

$$\mathcal{K}^{(j)} = (1 + \kappa^{(j)} + \kappa^{(j)} \cdot \kappa^{(j-1)} + \dots + \kappa^{(j)} \cdot \dots \cdot \kappa^{(1)})$$

wobei $\mathcal{K}^{(j)} := \kappa_{rel}(f^{(j)}, x^{(j)})$ die Kondition der elementar ausführbaren Operationen $f^{(j)}$ ist.

Beweis

$$\begin{aligned} \frac{\|\tilde{x}^{(j+1)} - x^{(j+1)}\|}{\|x^{(j+1)}\|} &= \frac{\|\tilde{f}^{(j)}(\tilde{x})^{(j)} - f^{(j)}(x^{(j)})\|}{\|f^{(j)}(x^{(j)})\|} \\ &\leq \frac{\|\tilde{f}(\tilde{x}) - f(\tilde{x})\|}{\|f(\tilde{x})\|} \cdot \frac{\|f(\tilde{x})\|}{\|f(x)\|} + \frac{\|f(\tilde{x}) - f(x)\|}{\|f(x)\|} \quad (\text{Index } j \text{ vernachlässigt}) \\ &\leq eps \left(1 + \frac{\|f(\tilde{x}) - f(x)\|}{\|f(x)\|}\right) + \frac{\|f(\tilde{x}) - f(x)\|}{\|f(x)\|} \\ &\stackrel{\text{nach 3.3.19}}{=} eps + (eps + 1) \cdot \left(\kappa^{(j)} \cdot \frac{\|\tilde{x}^{(j)} - x^{(j)}\|}{\|x^{(j)}\|}\right) + o\left(\frac{\|\tilde{x}^{(j)} - x^{(j)}\|}{\|x^{(j)}\|}\right) \end{aligned}$$

Nach Voraussetzung gilt Gleichung (3.3.20) mit $\mathcal{K}^{(0)} = 1$ für $j = 0$.
Für $j = 1$ folgt nach Voraussetzung mit Gleichung (3.3.19)

$$\begin{aligned} \frac{\|\tilde{x}^{(2)} - x^{(2)}\|}{\|x^{(2)}\|} &\leq eps + (eps + 1) \cdot (\kappa^{(1)} eps + o(eps)) \\ &= eps(1 + \kappa^{(1)}) + o(eps) \\ &= eps\mathcal{K}^{(1)} + o(eps) \end{aligned}$$

Womit der Induktionsanfang gezeigt ist.

Für den Induktionsschritt von $j - 1$ zu j :

$$\begin{aligned} \frac{\|\tilde{x}^{(j+1)} - x^{(j+1)}\|}{\|x^{(j+1)}\|} &\leq eps + (1 + eps)\kappa^{(j)} [eps\mathcal{K}^{(j-1)} + o(eps)] \\ &\quad + (1 + eps) \cdot o(eps \cdot \mathcal{K}^{(j-1)} + o(eps)) \\ &= eps \left(1 + \kappa^{(j)} \cdot \mathcal{K}^{(j-1)}\right) + o(eps) \end{aligned}$$

Mit $\mathcal{K}^{(j)} = 1 + \kappa^{(j)} \cdot \mathcal{K}^{(j-1)}$ folgt die Behauptung. □

Hiermit folgt:

3.3.6 Korollar

Unter der Voraussetzung von Lemma 3.3.5 gilt:

$$\frac{\|\tilde{f}(\tilde{x}) - f(x)\|}{\|f(x)\|} \leq eps \cdot (1 + \kappa(l) + \kappa(l) \cdot \kappa(l-1) + \dots + \kappa(l) \cdot \dots \cdot \kappa(1)) + o(eps) \quad (3.3.21)$$

3.3.7 Bemerkung

Mit Korollar 3.3.6 ist offensichtlich, dass schlecht konditionierte Probleme zu elementar ausführbaren Abbildungen so früh wie möglich ausgeführt werden sollten.

Nach Beispiel 3.2.9 ist die Subtraktion zweier annähernd gleicher Zahlen schlecht konditioniert. Deshalb sollte man unvermeidbare Subtraktionen möglichst früh durchführen. Allerdings hängt $\kappa^{(j)}$ nicht nur von $f^{(j)}$, sondern auch vom Zwischenergebnis $x^{(j)}$ ab, welches a priori unbekannt ist.

3.3.8 Bemerkung zur Sprechweise

Der Quotient

$$\frac{\overbrace{\|\tilde{f}(\tilde{x}) - f(x)\|}^{\text{Gesamtfehler}}}{\|f(x)\|} = \frac{\underbrace{\|\tilde{f}(\tilde{x})\|}_{\substack{\text{Fehler} \\ \text{durch} \\ \text{Problem}}}}{\underbrace{\|f(x)\|}_{\substack{\text{Fehler} \\ \text{durch} \\ \text{Problem}}}} \cdot \frac{\underbrace{\|\tilde{x} - x\|}_{\substack{\text{Eingabe-} \\ \text{fehler}}}}{\underbrace{\|x\|}_{\substack{\text{Eingabe-} \\ \text{fehler}}}} \quad (3.3.22)$$

gibt die **Güte des Algorithmus** an. Als Stabilitätsindikator kann also

$$o(f, \tilde{f}, x) := \frac{\mathcal{K}}{\kappa_{rel}(f, x)} \quad (3.3.23)$$

verwendet werden und es gilt

$$\frac{\|\tilde{f}(\tilde{x}) - f(x)\|}{\|f(x)\|} < \underbrace{o(f, \tilde{f}, x)}_{\substack{\text{Beitrag} \\ \text{des} \\ \text{Algorithmus}}} \cdot \underbrace{\kappa_{rel}(f, x)}_{\substack{\text{Beitrag} \\ \text{des} \\ \text{Problems}}} \cdot \underbrace{eps}_{\substack{\text{Rundungs-} \\ \text{fehler}}} + o(eps)$$

Falls $o(f, \tilde{f}, x) < 1$, dämpft der Algorithmus die Fehlerfortpflanzung der Eingabe- und Rundungsfehler und heißt **stabil**.

Für $o(f, \tilde{f}, x) \gg 1$ heißt der Algorithmus **instabil**.

3.3.9 Beispiel

Nach Gleichung (3.3.19) gilt für die Elementaroperationen $\mathcal{K} \leq 1$. Da für die Subtraktion zweier annähernd gleich großer Zahlen $\kappa_{rel} \gg 1$ gilt, ist der Stabilitätsfaktor zweier annähernd gleich großer Zahlen sehr klein und der Algorithmus also stabil, Falls es sich jedoch bei einer zusammengesetzten Abbildung $f = h \circ g$ bei der zweiten Abbildung h um eine Subtraktion handelt, gilt

$$\mathcal{K} = (1 + \kappa(sub) + \kappa(sub) \cdot \kappa(g))$$

und die Stabilität ist gefährdet. Genauere Abschätzungen und damit genauere Indikatoren können durch komponentenweise Betrachtungen erhalten werden.

3.3.10 Rückwärtsanalyse

Die Fragestellung ist nun:

Kann $\tilde{f}(\hat{x})$ als exaktes Ergebnis von einer gestörten Eingabe \hat{x} unter der exakten Abbildung f aufgefasst werden?

Das würde heißen

$$\exists \hat{x} \in \mathbb{R}^n : f(\hat{x}) = \tilde{f}(\tilde{x}).$$

Dann schätze den Fehler

$$\|\hat{x} - x\|$$

bzw. für nicht injektive f

$$\min_{\hat{x} \in \mathbb{R}^n} \left\{ \|\hat{x} - x\| \mid f(\hat{x}) = \tilde{f}(\tilde{x}) \right\}$$

ab.

IMAGE MISSING

Index

- Algorithmus, 27
- Basis, 10
- double, 10
- Dreieckszerlegung, 3, 5
- elementar ausführbar, 27
- entkoppelt, 23
- Fehler, 9, 14
 - absoluter, 14
 - absoluter Rundungsfehler, 12
 - Fortpflanzung, 27
 - relativer, 14
- floating point, 9, 10
- floating point operations, 6
- flops, 6
- Frobeniusmatrix, 6
- Güte
 - Algorithmus, 29
- Gauß-Eliminator, 6
- Gaußsches Eliminationsverfahren, 3, 6
- Genauigkeit
 - Maschinengenauigkeit, 12
 - relative Rechengenauigkeit, 12
- Gleitkommazahl, 10
- Implementation, 27
- integer, 9
- Kondition
 - absolute normweise, 16
 - gut/schlecht konditioniert, 16
 - komponentenweise, 23
 - Kondition der Addition, 17
 - Matrix, 20
 - Konditionsanalyse
 - komponentenweise, 23
 - Landau-Symbole, 6
 - LR-Zerlegung, 6
 - LU-Zerlegung, 6
 - Mantisse, 10
 - MISSING, 27
 - Neumannsche Reihe, 21
 - Norm, 14
 - Euklidische Norm, 14
 - Frobeniusnorm, 15
 - Hölder-Norm, 14
 - Matrixnorm, 15
 - Maximumsnorm, 14
 - Spaltensummennorm, 15
 - submultiplikative, 15
 - Summennorm, 14
 - verträglich, 15
 - normalisierte Gleitkommazahl, 10
 - normweise Kondition, 16
 - p-Norm, 15
 - Permutationsmatrix, 7
 - Pivotelement, 5
 - Pivotisierung, 7
 - halbmaximale, 7
 - partielle, 7
 - Spalten-, 7
 - vollständige, 7
 - Zeilen-, 7
 - Problem, 14
 - Rückwärtsanalyse, 30

Index

Rückwärtssubstitution, 5
Realisierung, 27
Rechenaufwand, 6
Rundungsfehler, 9

scharf, 18
Singularwert, 16
Stabilität, 25, 29
Stabilität des Algorithmus, 9

Verfahren von Crout, 6
Vorwärtselimination, 3, 7
Vorwärtssubstitution, 3, 4, 6

Zahlendarstellung, 9
Zeilensummennorm, 15