6. Lineare Gleichungssysteme

Wir betrachten lineare Gleichungssysteme der Form:

$$\begin{matrix} a_{11}x_1 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ \vdots & & & & \vdots & & \vdots \\ a_{n1}x_1 & + & \cdots & + & a_{nn}x_n & = & b_n \end{matrix}$$

Matrixschreibweise:

$$\boxed{Ax = b} \tag{6.1}$$

mit $A \in \mathbb{K}^{(n,n)}$, $b \in \mathbb{K}^n$, $x \in \mathbb{K}^n$ $(\mathbb{K} = \mathbb{R}/\mathbb{C})$

Generelle Voraussetzung:

A reguläre Matrix!

($\Longleftrightarrow$ die Zeilen von A sind linear unabhängig
$\Longleftrightarrow$ die Spalten von A sind linear unabhängig
$\Longleftrightarrow$ $\det(A) \neq 0$)

Bem.: Lösung über $x = A^{-1}b$ (inverse Matrix) ist i. Allg. zu aufwendig.

A^{-1}: Lösung von $AX = I_n$

Ausnahme: Falls A^{-1} leicht zu berechnen ist; etwa $H = I_n - 2ww^T$, $\|w\| = 1 \Rightarrow H^{-1} = H$.

6.1 Gestaffelte Systeme

- Für ein gestaffeltes System mit einer oberen Dreiecksmatrix

$$\boxed{Rx = y,} \qquad R = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ 0 & & r_{nn} \end{pmatrix} \tag{6.2}$$

läßt sich die Lösung x durch Rückwärts-Substitution berechnen:

$$\begin{aligned} & x_n := y_n / r_{nn} \\ & \text{für } i = n-1, \dots, 1 \\ & \quad \Big| \; x_i := \Big(y_i - \sum_{j=i+1}^{n} r_{ij} x_j \Big) / r_{ii} \end{aligned} \tag{6.3}$$

Wegen $\det(R) = \prod_{i=1}^{n} r_{ii} \neq 0$ ist der Algorithmus durchführbar.

Aufwand: $\approx \frac{1}{2} n^2$ wesentliche Operationen

- Für ein gestaffeltes System mit einer unteren Dreiecksmatrix

$$\boxed{Ly = b,} \qquad L = \begin{pmatrix} l_{11} & & 0 \\ \vdots & \ddots & \\ l_{n1} & \cdots & l_{nn} \end{pmatrix} \tag{6.4}$$

läßt sich die Lösung y durch Vorwärts-

Substitution berechnen:

$$
\begin{aligned}
&y_1 := b_1 / l_{11} \\
&\text{für } i = 2, \dots, n \\
&\quad \Big| \; y_i := \Big(b_i - \sum_{j=1}^{i-1} l_{ij}\, y_j \Big) / l_{ii}
\end{aligned}
\qquad \underline{(6.5)}
$$

Die obigen Anmerkungen gelten analog.

6.2 Gauß-Elimination

Ausgangssystem:

$$Ax = b \quad : \quad \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ \vdots & & \vdots & \vdots \\ a_{n1} & \dots & a_{nn} & b_n \end{array} \right)$$

1. Eliminationsschritt:

a) Durch ev. Vertauschen der ersten Zeile mit einer anderen lässt sich $a_{11} \neq 0$ erreichen.

a_{11}: Pivotelement (Pivot: Drehzapfen, Angelpunkt)

b) Für $i = 2, \dots, n$: Subtrahiere das

$$l_{i1} := a_{i1} / a_{11}$$

Fache der 1. Zeile von der Zeile Nr. i:

für $i = 2, \dots, n$

$$\left|\begin{array}{l} l_{i1} := a_{i1} / a_{11} \\ \text{für } j = 1, \dots, n \\ \quad \left| \; a_{ij}^{(2)} := a_{ij} - l_{i1} \cdot a_{1j} \right. \\ b_i^{(2)} := b_i - l_{i1} b_1 \end{array}\right. \tag{6.6}$$

Resultat:

$$A^{(2)} x = b^{(2)} : \quad \left(\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} & b_n^{(2)} \end{array}\right) \tag{6.7}$$

Bei dieser Transformation bleibt die Lösungsmenge unverändert! Ebenso gilt:

$$\det A^{(2)} = \pm \det A$$

(„−" $\Longleftrightarrow$ Es gab eine Zeilenvertauschung)

Das gleiche Verfahren wird nun auf die Gleichungen Nr. $2, \dots, n$ von (6.7) angewendet.
[Die entsprechende Teilmatrix ist aufgrund des Laplaceschen Entwicklungssatzes regulär!]

$\Longrightarrow$

Nach $n-1$ Eliminationsschritten erhält man ein <u>gestaffeltes lin. Gleichungssystem</u>:

$$A^{(n)}x = b^{(n)} \quad : \quad \left[\begin{array}{cccc|c} a_{11} & \cdots & \cdots & a_{1n} & b_1 \\ & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ & & \ddots & \vdots & \vdots \\ 0 & & & a_{nn}^{(n)} & b_n^{(n)} \end{array}\right] \qquad \underline{(6.8)}$$

Matrixdarstellung:

Annahme: Alle Zeilenvertauschungen sind bereits vor dem Eliminationsprozess durchgeführt worden!

Der k. Eliminationsschritt

$$A^{(k)}x = b^{(k)} \longrightarrow A^{(k+1)}x = b^{(k+1)}$$

lässt sich als Matrixmultiplikation schreiben:

$$A^{(k+1)} = L_k \cdot A^{(k)}, \quad b^{(k+1)} = L_k b^{(k)} \qquad \underline{(6.9)}$$

$$L_k := \begin{bmatrix} 1 & & & & 0 \\ & \ddots & & & \\ & & 1 & & \\ 0 & & -l_{k+1,k} & \ddots & \\ & & \vdots & & \\ & & -l_{n,k} & 0 & 1 \end{bmatrix}, \quad l_{ik} := \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$$

$$\underline{(6.10)}$$

(Eliminationsmatrix)

Speicherung:

$$\left[\begin{array}{ccccc|c}
a_{11} & a_{12} & \cdots & \cdots & a_{1n} & b_1 \\
l_{21} & a_{22}^{(2)} & \cdots & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\
l_{31} & l_{32} & \ddots & & \vdots & \vdots \\
\vdots & \vdots & & \ddots & \vdots & \vdots \\
l_{n1} & l_{n2} & \cdots & l_{n,n-1} & a_{nn}^{(n)} & b_n^{(n)}
\end{array}\right]$$

Algorithmus (6.11) (ohne Zeilenvertauschung)

für $k = 1, \ldots, n-1$
- für $i = k+1, \ldots, n$
  - $a_{ik} := a_{ik} / a_{kk}$
  - für $j = k+1, \ldots, n$
    - $\boxed{a_{ij} := a_{ij} - a_{ik} \cdot a_{kj}}$
  - $b_i := b_i - a_{ik} b_k$

Aufwand: $\approx \frac{1}{3} n^3$ wesentl. Operationen

Beachte: Die l_{ik} können auf a_{ik} gespeichert werden!

Beispiel (6.12)

$$\begin{aligned} x_1 - 5x_2 + x_3 &= 2 \\ 2x_1 + 4x_2 + x_3 &= 1 \\ x_1 + x_2 + x_3 &= 0 \end{aligned}$$

$$\left(\begin{array}{ccc|c} 1 & -5 & 1 & 2 \\ 2 & 4 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{array}\right) \to \left(\begin{array}{ccc|c} 1 & -5 & 1 & 2 \\ 2 & 14 & -1 & -3 \\ 1 & 6 & 0 & -2 \end{array}\right)$$

$$\to \left(\begin{array}{ccc|c} 1 & -5 & 1 & 2 \\ 2 & 14 & -1 & -3 \\ 1 & 6/14 & 6/14 & -5/7 \end{array}\right)$$

Rückwärts-Substitution:

$$x_3 = -\frac{5}{7} \Big/ \frac{6}{14} = -\frac{5}{3}$$

$$x_2 = (-3 + x_3)/14 = -\frac{1}{3}$$

$$x_1 = (2 - x_3 + 5 \cdot x_2)/1 = 2$$

Lösung: $x = \left(2, -\frac{1}{3}, -\frac{5}{3}\right)^T$

Berücksichtigung von Zeilenvertauschungen

Man arbeitet mit einem Permutationsvektor

$$p = (p_1, \dots, p_n)^T \in \mathbb{N}^n,$$

der die jeweils aktuelle Reihenfolge der Zeilen (d.h. der Gleichungen!) angibt.

Startvektor ist $p = (1,2,\dots,n)^T$.

Bei Vertauschung von Zeilen k und j ist lediglich p_k und p_j zu vertauschen.

Ferner ist in Algorithmus (6.11) mit den Zeilenindizes p_i bzw. p_k anstelle von i bzw. k zu arbeiten!

6.3 LR-Zerlegung

Wir betrachten die Gauß-Elimination ohne Zeilenvertauschung:

$\boxed{R := A^{(n)}}$ ist eine obere Δ-Matrix und es gilt nach (6.9):

$$R = L_{n-1} A^{(n-1)} = \dots = L_{n-1} L_{n-2} \dots \cdot L_1 \cdot A$$

Algorithmus (6.11)' (mit Zeilenvertauschung)

$p := (1, \dots, n)$

für $k = 1, \dots, n-1$

  if ($a_{p_k,k} = 0$) then

    Suche Index p_j, $j > k$, mit $a_{p_j,k} \neq 0$

    Vertausche p_j und p_k

  endif

  für $i = k+1, \dots, n$

    $a_{p_i,k} := a_{p_i,k} / a_{p_k,k}$

    für $j = k+1, \dots, n$

      $a_{p_i,j} := a_{p_i,j} - a_{p_i,k} \cdot a_{p_k,j}$

    $b_{p_i} := b_{p_i} - a_{p_i,k} \cdot b_{p_k}$

$\Longrightarrow$ (alle L_i sind regulär!)

$$A = \underbrace{(L_1^{-1} L_2^{-1} \cdot \ldots \cdot L_{n-1}^{-1})}_{=:\, L} \cdot R$$

Satz (6.13) (über inverse Eliminationsmatrizen)

a) $$L_k^{-1} = \begin{bmatrix} 1 & & & & & 0 \\ & \ddots & & & & \\ & & 1 & & & \\ & & l_{k+1,k} & \ddots & & \\ & & \vdots & & \ddots & \\ & & l_{n,k} & & & 1 \end{bmatrix}$$

b) $$L = L_1^{-1} L_2^{-1} \cdot \ldots \cdot L_{n-1}^{-1} = \begin{bmatrix} 1 & & & & 0 \\ l_{21} & 1 & & & \\ \vdots & \ddots & \ddots & & \\ \vdots & & \ddots & \ddots & \\ l_{n1} & \cdots & \cdots & l_{n,n-1} & 1 \end{bmatrix}$$

(normierte untere Δ-Matrix!)

Beweis:

zu a) Matrixdarstellung:

$$L_k = I - l_k e_k^T, \quad l_k := \begin{bmatrix} 0 \\ \vdots \\ 0 \\ l_{k+1,k} \\ \vdots \\ l_{n,k} \end{bmatrix}, \quad e_k := \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Damit rechnet man nach:

$$(I - l_k e_k^T)(I + l_k e_k^T) =$$

$$= I + l_k e_k^T - l_k e_k^T - l_k \underbrace{\left[e_k^T l_k\right]}_{=0} e_k^T = I$$

$$\Longrightarrow \quad L_k^{-1} = I + l_k e_k^T .$$

b) $L = L_1^{-1} L_2^{-1} \dots L_{n-1}^{-1}$

$$= (I + l_1 e_1^T)(I + l_2 e_2^T) \dots (I + l_{n-1} e_{n-1}^T)$$

$$= I + l_1 e_1^T + l_2 e_2^T + \dots + l_{n-1} e_{n-1}^T$$

↑ Alle "gemischten" Terme „$l_i e_i^T l_j e_j^T$" mit $i<j$ verschwinden!

∎

Anmerkungen:

1.) Die LR-Zerlegung einer Matrix A wird durch Gauß-Elimination (ohne rechte Seite) berechnet!

Beispiel: (→ (6.12))

$$\underbrace{\begin{pmatrix} 1 & & 0 \\ 2 & 1 & \\ 1 & 6/14 & 1 \end{pmatrix}}_{L} \cdot \underbrace{\begin{pmatrix} 1 & -5 & 1 \\ & 14 & -1 \\ 0 & & 6/14 \end{pmatrix}}_{R} = \underbrace{\begin{pmatrix} 2 & 4 & 1 \\ 1 & 1 & 1 \end{pmatrix}}_{A}$$

2.) Nicht jede reguläre Matrix besitzt eine LR-Zerlegung:

$$\begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ * & 1 \end{pmatrix} \begin{pmatrix} \overset{0}{*} & * \\ 0 & * \end{pmatrix} \quad ?$$

ABER:

Jede reguläre Matrix A besitzt nach geeigneter Zeilenvertauschung eine LR-Zerlegung, d.h. es gibt eine Permutationsmatrix

$$P = \begin{pmatrix} e_{p_1}^T \\ \vdots \\ e_{p_n}^T \end{pmatrix}, \quad p = (p_1, \dots, p_n) \text{ Permutation,}$$

so dass $\boxed{PA = L \cdot R}$, L: normierte untere Δ-Matrix, R: reguläre obere Δ-Matrix!

3.) Mittels Berechnung der LR-Zerlegung

$$PAx = LRx = Pb$$

kann man das lin. Gleichungssystem $Ax=b$ folgendermaßen lösen:

(i) Berechne P, L, R
(Kenntnis von b wird nicht benötigt!)

(ii) Löse $Ly = Pb$ mittels Vorwärts-Substitution

(iii) Löse $Rx = y$ mittels Rückwärts-Substitution.

Dieser Weg ist zu empfehlen, falls mehrere lineare Gl.systeme mit gleicher Koeffizienten-matrix zu lösen sind.

Satz (6.14) (Existenz einer LR-Zerlegung)

gilt für alle $i = 1, \dots, n-1$:

$$\det \begin{bmatrix} a_{11} & \cdots & a_{1i} \\ \vdots & & \vdots \\ a_{i1} & \cdots & a_{ii} \end{bmatrix} \neq 0,$$

so besitzt A eine LR-Zerlegung und diese ist eindeutig bestimmt.

Beweis: (vollst. Induktion über n)

$n = 1$: klar

$\underline{n \Rightarrow n+1}$: Man zerlege die Matrizen:

$$A = \left[\begin{array}{c:c} A_n & c \\ \hdashline r^T & a_{n+1,n+1} \end{array}\right] \overset{!}{=} \left[\begin{array}{c:c} L_n & 0 \\ \hdashline \ell^T & 1 \end{array}\right] \left[\begin{array}{c:c} R_n & u \\ \hdashline 0^T & r_{n+1,n+1} \end{array}\right]$$

$$\Longleftrightarrow \quad \begin{cases} A_n = L_n \cdot R_n \\ c = L_n \cdot u \\ r^T = \ell^T R_n \\ a_{n+1,n+1} = \ell^T u + r_{n+1,n+1} \end{cases}$$

Induktionsvor. $\Rightarrow$ L_n, R_n sind aus der ersten Gleichung eindeutig bestimmt.

Nach Voraus. $\det(A_n) \neq 0 \Rightarrow$ L_n und R_n sind reguläre Matrizen.

$\Rightarrow$ u, ℓ lassen sich aus der zweiten und dritten Gleichung eindeutig bestimmen!

Die vierte Gleichung legt dann $r_{n+1,n+1}$ eindeutig fest! ∎

Der Satz liefert eine Bedingung dafür, wann die Gauß-Elimination ohne Zeilenvertauschung durchführbar ist.

Beispiel:

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 2 & 1 \end{pmatrix}$$ erfüllt die Voraus. von Satz (6.14); die LR-Zerlegung lautet

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

Beachte, dass A selbst nicht regulär sein muss!

6.4 Pivoting, Skalierung

Die Stabilität des Eliminationsverfahrens hängt davon ab,

- welche Elemente als Pivotelemente gewählt werden und
- wie die Gleichungen skaliert sind.

Beispiel:

$$\begin{aligned} 10^{-4} x_1 + x_2 &= 1 \\ x_1 + x_2 &= 2 \end{aligned}$$

Lösung: $x_1 = \frac{10^4}{10^4 - 1} \doteq 1.0001$, $x_2 = \frac{10^4 - 2}{10^4 - 1} \doteq 0.9999$

Dreistellige Rechnung mit Pivotelement 10^{-4} liefert jedoch:

$$fl(x_1) = 0.00\,, \quad fl(x_2) = 1.00$$

Grund für Versagen: Das Pivotelement 10^{-4} ist zu klein!

Aber: Multipl. der 1. Gleichung mit 10^4 liefert gleiches numer. Ergebnis

Grund für Versagen: Die Gleichungen sind unterschiedlich skaliert!

Gebräuchliche Pivotstrategien:

a) Spaltenpivoting: Wähle das betragsgrößte Element der aktuellen Spalte, also im k. Eliminationsschritt:

Wähle $j \in \{k, \dots, n\}$ mit

$$|a_{jk}^{(k)}| = \max_{k \le i \le n} |a_{ik}^{(k)}|$$

und vertausche die Zeilen k und j.

b) Vollständige Pivotsuche:

Man bestimme das betragsgrößte Element der gesamten Restmatrix und vertausche Zeilen und Spalten!

Bestimme $i,j \in \{k,..,n\}$ mit

$$|a_{ij}^{(k)}| = \max\{|a_{\nu\mu}^{(k)}| : k \le \nu,\mu \le n\}$$

und vertausche Zeilen Nr. i und k und Spalten Nr. j und k.

Skalierungsstrategien:

- **Äquilibrierung:** Bestimme reguläre Diagonalmatrizen D und $\tilde{D}$, so dass für die skalierte Matrix $A' := DA\tilde{D}$ gilt:

$$\forall i,j: \quad \sum_{k=1}^{n} |a'_{ik}| = \sum_{k=1}^{n} |a'_{kj}|$$

Es ist i. Allg. aufwendig, eine solche Skalierung zu bestimmen.

- **Einfachere Variante:** $A' = DA$ mit

$$\forall i,j: \quad \sum_{k=1}^{n} |a'_{ik}| = \sum_{k=1}^{n} |a'_{jk}| = 1$$

6.5 Nachiteration

Die numerische Lösung $\tilde{x}$ eines linearen Gleichungssystems $Ax = b$ wird dieses i. Allg. nur näherungsweise erfüllen.

Setzt man $x = \tilde{x} + \Delta x$, so folgt aus $Ax = b$:

$$A \Delta x = b - A\tilde{x} =: r \quad \text{Residuum}$$

Ist $r \neq 0$, so kann man versuchen, dieses lin. Gleichungssystem für Δx zu lösen um per $x := \tilde{x} + \Delta x$ die Lösung zu verbessern.

Algorithmus: (6.15)

$x^{(1)} := \tilde{x}$, TOL: Genauigkeitsparameter

für $j = 1, 2, \dots$

- $r^{(j)} := b - Ax^{(j)}$
- Falls $\left(\|r^{(j)}\| \leq \text{TOL} \cdot \|x^{(j)}\| \right)$: Stop
- Löse $A \cdot \Delta x^{(j)} = r^{(j)}$
- $x^{(j+1)} := x^{(j)} + \Delta x^{(j)}$

Beachte: Es wird nur eine LR-Zerlegung von A benötigt! r doppeltgenau berechnen!

6.6 Cholesky-Zerlegung

Definition:

a) Zu $A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \in \mathbb{K}^{(m,n)}$ heißt

$A^T := \begin{pmatrix} a_{11} & \cdots & a_{m1} \\ \vdots & & \vdots \\ a_{1n} & \cdots & a_{mn} \end{pmatrix} \in \mathbb{K}^{(n,m)}$ die transponierte Matrix.

b) Eine Matrix $A \in \mathbb{R}^{(n,n)}$ heißt symmetrisch, falls $A = A^T$.

Bemerkung: Einige elementare Eigenschaften:

- $(\alpha A + \beta B)^T = \alpha A^T + \beta B^T$
- $(AB)^T = B^T A^T$
- $A \in \mathbb{R}^{(n,n)}$ regulär $\Rightarrow (A^T)^{-1} = (A^{-1})^T$.

Wir untersuchen die numer. Lösung linearer Gleichungssysteme $Ax = b$ mit regulärer, symmetrischer Koeffizientenmatrix A.

Ziel: Halber Aufwand gegenüber dem nichtsymmetrischen Fall!

Satz (6.16) (Cholesky-Zerlegung)

Sei $A \in \mathbb{R}^{(n,n)}$ symmetrisch und es gelte

$$\forall k = 1, 2, \ldots, n: \det(A_k) \neq 0, \quad A_k := \begin{bmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} \end{bmatrix}$$

Dann gibt es eind. bestimmte Matrizen L und D, L: normierte untere Δ-Matrix, $D = \operatorname{diag}(d_1, \ldots, d_n)$, $d_j \neq 0$ mit

$$\boxed{A = L\,D\,L^T}$$

(rationale Cholesky-Zerlegung)

Beweis: Nach Satz (6.14) besitzt A eine eindeutig bestimmte LR-Zerlegung $A = L \cdot R$.

Da A regulär ist $\Rightarrow \det A = \prod_{i=1}^{n} r_{ii} \neq 0$

und damit: $\forall i = 1, \ldots, n: \; r_{ii} \neq 0$.

Damit lässt sich die Diagonale $D := \operatorname{diag}(r_{11}, \ldots, r_{nn})$ aus der Matrix R herausziehen:

$$R = D\,\tilde{L}^T, \quad \tilde{L} \text{ normierte untere } \Delta\text{-Matrix}$$

Aus der Symmetrie von A folgt:

$$A = A^T = (L D \tilde{L}^T)^T = \tilde{L}(D\,L^T)$$

Dies ist eine weitere LR-Zerlegung von A

Eindeutigkeit der LR-Zerlegung

$$\Longrightarrow \quad \tilde{L} = L, \text{ also } A = LDL^T \qquad \blacksquare$$

Berechnung der Cholesky-Zerlegung:

$$A = L \cdot R \quad \Longrightarrow \quad a_{ik} = \sum_{j=1}^{\min(i,k)} l_{ij} r_{jk}$$

$$1 \le i \le k \le n \quad \Longrightarrow \quad a_{ik} = \sum_{j=1}^{i-1} l_{ij} r_{jk} + r_{ik}$$

$$\Longrightarrow \quad \left.\begin{array}{ll} r_{ik} = a_{ik} - \sum_{j=1}^{i-1} l_{ij} r_{jk}, & i = 1, .., k \\ l_{ki} = r_{ik} / r_{ii}, & i = 1, .., k-1 \end{array}\right\} k = 1, \dots, n$$

Reihenfolge: r_{ik}, l_{ki} $(i = 1, .., k-1)$, r_{kk}

Algorithmus (6.17)

für $k = 1, \dots, n$

- $d_k := a_{kk}$
- $r_{ik} := a_{ki} - \sum_{j=1}^{i-1} l_{ij} r_{jk} \quad (i = 1, .., k-1)$
- für $i = 1, .., k-1$
  - $t := r_{ik}$
  - $l_{ki} := r_{ik} / d_i$
  - $d_k := d_k - l_{ki} \cdot t$

Bemerkungen:

a) In Algorithmus (6.17) lassen sich sowohl die r_{ik} wie auch die l_{ki} auf a_{ki} $(i<k)$ abspeichern.

Die Information über A bleibt dann ober=halb der Diagonalen von A erhalten.

b) Der Algorithmus ist stabil, er benötigt weder Pivotsuche noch Skalierung!

Aufwand: $\approx \frac{1}{6} n^3$ wesentliche Operationen

c) Der Algorithmus kann dazu dienen, auf einfache Weise die „Definitheit" einer symmetrischen Matrix zu überprüfen.

Definition (6.18)

Eine symmetrische Matrix $A \in \mathbb{R}^{(n,n)}$ heißt

positiv semidefinit $:\Longleftrightarrow \forall x \in \mathbb{R}^n: \; x^T A x \geq 0$

positiv definit $:\Longleftrightarrow \forall x \in \mathbb{R}^n \setminus \{0\}: \; x^T A x > 0$

analog: negativ semidefinit, negativ definit

indefinit $:\Longleftrightarrow \exists x, y \in \mathbb{R}^n: \; x^T A x < 0 < y^T A y$

Bemerkung:

Jede quadratische Funktion (quadratische Form) $q: \mathbb{R}^n \longrightarrow \mathbb{R}$ lässt sich beschreiben durch

$$q(x) = \frac{1}{2} x^T A x + b^T x + c$$

mit $A \in \mathbb{R}^{(n,n)}$ symmetrisch, $b \in \mathbb{R}^n$, $c \in \mathbb{R}$.

Beispiel: $(n=2)$

$$q(x,y) = 2x^2 - 5xy + y^2 - 2x + 3y + 10$$

$$\Longrightarrow \quad q(x,y) = \frac{1}{2}(x,y)\begin{pmatrix} 4 & -5 \\ -5 & 2 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + (-2,3)\begin{pmatrix} x \\ y \end{pmatrix} + 10.$$

Die Nullstellenmenge einer quadratischen Form $Q := q^{-1}(0)$ heißt eine Quadrik. Ihre geometrische Gestalt lässt sich mit Hilfe der Definitheit von A bestimmen. So gilt im Fall $n=2$:

A definit $\Longleftrightarrow$ Q Ellipse

A semidefinit (aber nicht definit) $\Longleftrightarrow$ Q Parabel

A indefinit $\Longleftrightarrow$ Q Hyperbel

Satz (6.19)

Sei $A \in \mathbb{R}^{(n,n)}$ eine symmetrische Matrix.

a) A positiv definit $\iff$

$\iff \quad \forall k=1,..,n: \quad \det(A_k) > 0$

$\iff \quad A = LDL^T \wedge \forall k=1,..,n: \ d_k > 0$

b) A negativ definit $\iff$

$\iff \quad \forall k=1,..,n: \quad (-1)^k \cdot \det(A_k) > 0$

$\iff \quad A = LDL^T \wedge \forall k=1,..,n: \ d_k < 0$

Beweis: (zu a)

$\Longrightarrow$ Sei A positiv definit!

Wäre für ein $k \in \{1,..,n\}$ $\det(A_k) = 0$, so gibt es $\tilde{x} \in \mathbb{R}^k \setminus \{0\}$ mit $A_k \tilde{x} = 0$.

Für $x := \begin{bmatrix} \tilde{x} \\ 0 \end{bmatrix} \in \mathbb{R}^n$ folgt dann: $\quad x \neq 0$ und $x^T A x = 0$, Widerspruch!

Also gilt dann: $\forall k \in \{1,..,n\} \ \det(A_k) \neq 0$, d.h. A besitzt eine Cholesky-Zerlegung $A = LDL^T$. Mit $y := L^T x$ folgt:

$$x^T A x = x^T L D L^T x = y^T D y = \sum_{k=1}^{n} d_k y_k^2$$

und damit (L regulär!)

$$\forall x \neq 0: x^T A x > 0 \iff \forall k=1,..,n: \ d_k > 0$$

Schließlich: $\det(A_k) = \prod_{j=1}^{k} d_j$ und damit

$$\forall k=1,..,n : (d_k > 0) \iff \forall k=1,..,n : \det(A_k) > 0$$

$\underline{\Leftarrow :}$ Umgekehrt: gilt $A = LDL^T$, $\forall k : d_k > 0$, so folgt mit $x \in \mathbb{R}^n \setminus \{0\}$ auch $y := L^T x \neq 0$ und damit $x^T A x = \sum_{k=1}^{n} d_k y_k^2 > 0$ ∎

Beispiel (6.20)

Ist die folgende Matrix positiv definit?

$$A = \begin{bmatrix} 1 & -2 & -1 & 1 \\ -2 & 8 & 10 & 10 \\ -1 & 10 & 26 & 41 \\ 1 & 10 & 41 & 89 \end{bmatrix}$$

Wir berechnen die LR-Zerlegung:

$$A \longrightarrow \begin{bmatrix} 1 & -2 & -1 & 1 \\ -2 & 4 & 8 & 12 \\ -1 & 2 & 9 & 18 \\ 1 & 3 & 2 & 16 \end{bmatrix} \Longrightarrow$$

$$L = \begin{bmatrix} 1 & & & 0 \\ -2 & 1 & & \\ -1 & 2 & 1 & \\ 1 & 3 & 2 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 1 & & & 0 \\ & 4 & & \\ & & 9 & \\ 0 & & & 16 \end{bmatrix}$$

Da alle $d_k > 0$ sind, ist A nach (6.19) positiv definit!

Überzeugen Sie sich, dass tatsächlich $DL^T = R$.

6.7 Fehleranalyse

Wir wollen die Kondition eines linearen Gleichungssystems $Ax = b$ untersuchen: Wie wirken sich Fehler in den Eingangsdaten (A, b) auf die Lösung x aus?

Das Konzept aus Kap. 1 "komponentenweise Fehleranalyse" ist wegen $(A,b) \in \mathbb{R}^{n^2+n}$, $x \in \mathbb{R}^n$ für größere n kaum brauchbar. (Man hätte $n(n^2+n)$ Konditionszahlen!)

Abhilfe: „Normweise Kondition" !

Definition (6.21)

Zu $n \in \mathbb{N}$ heißt eine Abbildung $\|.\| : \mathbb{R}^n \to \mathbb{R}$ eine Norm (Vektornorm), falls für alle $x, y \in \mathbb{R}^n$, $\alpha \in \mathbb{R}$ gelten:

- $\|x\| \geq 0, \quad \|x\| = 0 \iff x = 0$
- $\|\alpha x\| = |\alpha| \, \|x\|$
- $\|x+y\| \leq \|x\| + \|y\|$

Gebräuchliche Normen sind: $(x \in \mathbb{R}^n)$

$$\|x\|_1 := \sum_{i=1}^{n} |x_i|$$

$$\|x\|_2 := \sqrt{\sum_{i=1}^{n} x_i^2}$$

$$\|x\|_\infty := \max\{|x_i| \mid i = 1, \dots, n\}$$

Definition (6.22)

Sind $\|.\|$ und $\|.\|'$ Normen im $\mathbb{R}^n$ und $\mathbb{R}^m$, so wird durch

$$\|A\| := \max\{\|Ax\|' \mid \|x\| = 1\}$$

die zugehörige Matrixnorm (Operatornorm) für $A \in \mathbb{R}^{(m,n)}$ definiert.

Bemerkungen:

a) Matrixnormen sind Normen, d.h. sie erfüllen die Eigenschaften aus (6.21).
Darüber hinaus sind sie submultiplikativ, d.h.

$$\|Ax\|' \leq \|A\| \, \|x\| \tag{6.23}$$

b) Im Fall $m = n$ wählt man i. Allg. die gleiche Vektornorm im Bildraum wie im Urbildraum.

Gebräuchliche Matrixnormen $(n=m)$: (6.24)

Bzgl. $\|.\|_1$: $\|A\|_1 = \max\limits_{1\le j\le n} \sum\limits_{i=1}^{n} |a_{ij}|$

(Spaltensummennorm)

Bzgl. $\|.\|_2$: $\|A\|_2 = \sqrt{\lambda_{max}(A^T A)}$

(Spektralnorm)

Bzgl. $\|.\|_\infty$: $\|A\|_\infty = \max\limits_{1\le i\le n} \sum\limits_{j=1}^{n} |a_{ij}|$

(Zeilensummennorm)

λ_{max} bezeichnet dabei den größten Eigenwert der (symmetrischen und positiv semidefiniten) Matrix $A^T A$. Dieser lässt sich effizient numerisch berechnen. Die obigen Matrixnormen lassen sich in MATLAB mit den Befehlen

$$\boxed{\texttt{norm(A,p)}, \quad p = 1, 2, \texttt{inf}}$$

auswerten.

Fehler in der rechten Seite b :

Wir betrachten ein lineares Gleichungssystem $Ax=b$ mit $A\in\mathbb{R}^{(n,n)}$ regulär, $b\in\mathbb{R}^n$.
Damit ist $x = A^{-1}b$ und bei einem Fehler Δb in der rechten Seite folgt: $\Delta x = A^{-1}\Delta b$.

Normweise Abschätzung:

$$\|\Delta x\| = \|A^{-1} \cdot \Delta b\| \leq \|A^{-1}\| \cdot \|\Delta b\|$$

und für den relativen Fehler:

$$\frac{\|\Delta x\|}{\|x\|} = \frac{\|A^{-1}\Delta b\|}{\|x\|}$$

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

$$= \|A^{-1}\| \frac{\|Ax\|}{\|x\|} \cdot \frac{\|\Delta b\|}{\|b\|}$$

$$\leq \left(\|A^{-1}\| \cdot \|A\|\right) \frac{\|\Delta b\|}{\|b\|}$$

Damit ist $\kappa_{abs} := \|A^{-1}\|$ bzw. $\kappa_{rel} := \|A^{-1}\| \|A\|$ eine Abschätzung für die absolute bzw. relative Kondition des Problems $x = A^{-1}b$ bzw. Fehler in b.

Definition (6.25)

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

heißt die Kondition der Matrix $A \in \mathbb{R}^{(n,n)}$ (bezüglich der Norm $\|.\|$).

MATLAB-Befehl:

$$\boxed{\text{cond}(A,p)\ ,\quad p = 1, 2, \text{inf}}$$

Satz (6.26) Gegeben $Ax = b$, A regulär

a) Für Fehler in b gilt:

$$\frac{\|\Delta x\|}{\|x\|} \leq \kappa(A) \frac{\|\Delta b\|}{\|b\|}$$

b) Für Fehler in A gilt (bei $\|A^{-1} \Delta A\| < 1$):

$$\frac{\|\Delta x\|}{\|x\|} \leq \left(\frac{\kappa(A)}{1 - \kappa(A) \frac{\|\Delta A\|}{\|A\|}} \right) \frac{\|\Delta A\|}{\|A\|}$$

Beweis:

zu a): Oben gezeigt

zu b): $\longrightarrow$ Stoer: Numerische Mathematik I

Der folgende Satz zeigt, dass aus der Kleinheit des Residuums $r := b - A\tilde{x}$ auf die Güte der Näherungslösung $\tilde{x}$ im Sinn der Rückwärtsanalyse geschlossen werden kann:

Satz (6.27) (Prager, Oettli 1964)

Sei $\tilde{x}$ Näherungslösung von $Ax = b$, $\Delta A \geq 0$, $\Delta b \geq 0$.
Es sind äquivalent

a) $\exists\, \tilde{A}, \tilde{b} : |\tilde{A} - A| \leq \Delta A$, $|\tilde{b} - b| \leq \Delta b$ und $\tilde{A}\tilde{x} = \tilde{b}$

b) $|b - A\tilde{x}| \leq \Delta A \cdot |\tilde{x}| + \Delta b$

Erläuterungen

- Für $A=(a_{ij}) \in \mathbb{R}^{(n,n)}$, $b=(b_i) \in \mathbb{R}^n$ wird definiert $|A| := (|a_{ij}|)$ (Matrix) und $|b| := (|b_i|)$ (Vektor!).

- Zu einer Näherungslösung $\tilde{x}$ des linearen Gleichungssystems $Ax=b$ berechne man (möglichst genau!) den Residuenvektor $r := b - A\tilde{x}$. Dann wähle man $\Delta A \geq 0$, $\Delta b \geq 0$ (möglichst klein), so dass b) erfüllt ist.
  Nach dem Satz v. Prager und Oettli existieren dann innerhalb dieser Fehlermargen $\tilde{A}$ u. $\tilde{b}$, so dass $\tilde{x}$ exakte Lösung des modifizierten Gleichungssystems $\tilde{A}x=\tilde{b}$ ist.

Beweis zu (6.27):

a) $\Rightarrow$ b) Mit $\delta A := \tilde{A} - A$, $\delta b := \tilde{b} - b$ gilt

$$\begin{aligned} |b - A\tilde{x}| &= |\tilde{b} - \delta b - (\tilde{A} - \delta A)\tilde{x}| \\ &= |-\delta b + \delta A\,\tilde{x}| \\ &\leq |\delta b| + |\delta A|\,|\tilde{x}| \\ &\leq \Delta b + \Delta A\,|\tilde{x}| . \end{aligned}$$

b) $\Rightarrow$ a) Man setze

$$r := b - A\tilde{x},$$

$$s := \Delta b + \Delta A |\tilde{x}| \geq 0,$$

$$k_i := \begin{cases} r_i / s_i, & \text{falls } s_i > 0 \\ 0, & \text{falls } s_i = 0, \end{cases}$$

$$\delta a_{ij} := \operatorname{sign}(\tilde{x}_j) \cdot k_i \cdot \Delta a_{ij},$$

$$\delta b_i := -k_i \Delta b_i .$$

Nach b) ist $|k_i| \leq 1$, daher $|\delta A| \leq \Delta A$, $|\delta b| \leq \Delta b$.

Mit $\tilde{A} := A + \delta A$, $\tilde{b} := b + \delta b$ folgt:

$$\begin{aligned}
[\tilde{A}\tilde{x} - \tilde{b}]_i &= [A\tilde{x} + \delta A \tilde{x} - b - \delta b]_i \\
&= -r_i + \sum_j \operatorname{sign}(\tilde{x}_j)\, k_i \,\Delta a_{ij}\, \tilde{x}_j + k_i \Delta b_i \\
&= -r_i + \underbrace{\Big(\sum_j \Delta a_{ij} |\tilde{x}_j| + \Delta b_i\Big)}_{s_i} k_i \\
&= 0
\end{aligned}$$

(beachte $|r_i| \leq s_i$ nach b)!) $\blacksquare$