

Beispiel 3.2. Gesucht $u(x)$, das eine *Differentialgleichung* vom Typ

$$-u''(x) + \lambda(x)u(x) = f(x), \quad x \in [0, 1],$$

mit den *Randbedingungen*

$$u(0) = u(1) = 0$$

erfüllt (Randwertaufgabe). Gitterpunkte:

$$x_j = jh, \quad j = 0, \dots, n, \quad h = \frac{1}{n},$$

Ableitungen \rightarrow *Differenzenquotienten* (Taylor):

$$\begin{aligned}u(x_j + h) &= u(x_j) + hu'(x_j) + \frac{h^2}{2}u''(x_j) + \frac{h^3}{6}u'''(x_j) + \mathcal{O}(h^4), \\u(x_j - h) &= u(x_j) - hu'(x_j) + \frac{h^2}{2}u''(x_j) - \frac{h^3}{6}u'''(x_j) + \mathcal{O}(h^4).\end{aligned}$$

Daraus folgt sofort

$$u(x_j + h) - 2u(x_j) + u(x_j - h) = h^2u''(x_j) + \mathcal{O}(h^4),$$

und somit

$$u''(x_j) = \frac{1}{h^2}[u(x_{j+1}) - 2u(x_j) + u(x_{j-1}))] + \mathcal{O}(h^2).$$

D. h., bis auf Terme 2. Ordnung in der Schrittweite h entspricht die 2. Ableitung einem **Differenzenquotient**.

Diskretisierung:

$$-\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} + \lambda(x_j)u_j = f(x_j), \quad j = 1, \dots, n-1,$$

zusammen mit den Randbedingungen $u_0 = u_n = 0$.

Für kleines h , also großes n : $u_j \approx u(x_j)$.

Gleichungssystem in Matrixform:

$$\begin{pmatrix} 2 + h^2\lambda_1 & -1 & 0 & \cdots & 0 \\ -1 & 2 + h^2\lambda_2 & -1 & \cdots & \vdots \\ 0 & -1 & 2 + h^2\lambda_3 & \cdots & 0 \\ \vdots & \cdots & \cdots & \cdots & -1 \\ 0 & \cdots & 0 & -1 & 2 + h^2\lambda_{n-1} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n-2} \\ u_{n-1} \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-2} \\ f_{n-1} \end{pmatrix}.$$

△

Beispiel 3.3. Gesucht $u(x)$, das die *Integralgleichung*

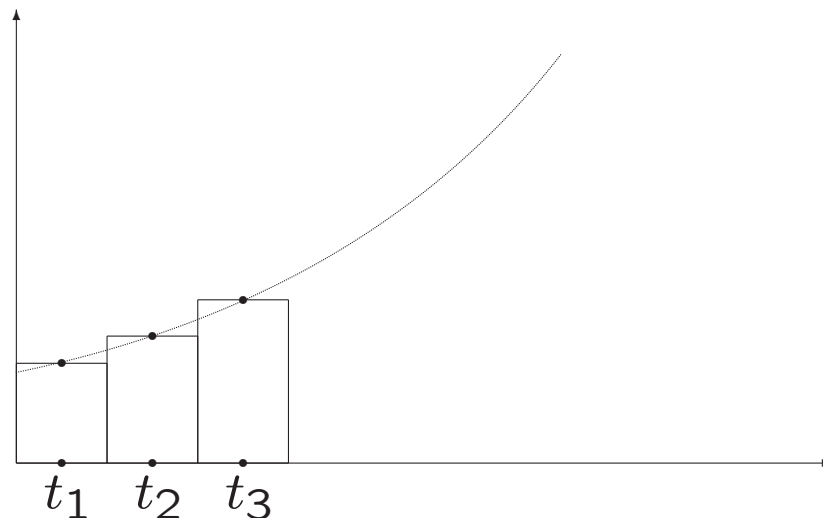
$$u(x) + 2 \int_0^1 \cos(xt)u(t) dt = 2, \quad x \in [0, 1]$$

erfüllt. Gitterpunkte

$$t_j = \left(j - \frac{1}{2}\right)h, \quad j = 1, \dots, n, \quad h = \frac{1}{n}.$$

Annäherung des Integrals (Mittelpunktsregel):

$$\int_0^1 f(t) dt \approx h \sum_{j=1}^n f(t_j)$$



Es gilt:

$$\left| \int_0^1 f(t) dt - h \sum_{j=1}^n f(t_j) \right| \leq \frac{h^2}{24} \max_{x \in [0,1]} |f''(x)|$$

Weitere Annäherung: Gleichung nur in den Punkten $x = t_i$ betrachten.
Gleichungssystem für $u_i \approx u(t_i)$:

$$u_i + 2h \sum_{j=1}^n \cos(t_i t_j) u_j = 2, \quad i = 1, 2, \dots, n.$$

In Matrixform ergibt sich

$$\begin{pmatrix} \frac{1}{2h} + \cos(t_1 t_1) & \cos(t_1 t_2) & \cdots & \cos(t_1 t_n) \\ \cos(t_2 t_1) & \frac{1}{2h} + \cos(t_2 t_2) & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \cos(t_n t_1) & \cdots & \cdots & \frac{1}{2h} + \cos(t_n t_n) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = \frac{1}{h} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}.$$

△

Beispiel 3.4.

In der linearen Ausgleichsrechnung (siehe Kapitel 4) entsteht auf natürliche Weise ein lineares Gleichungssystem (die sogenannten *Normalgleichungen*). △

Beispiel 3.5.

Zur Lösung eines *nichtlinearen* Gleichungssystems werden oft Linearisierungsverfahren, wie z.B. das Newton-Verfahren eingesetzt (siehe Kapitel 5). Bei so einem Verfahren ergibt sich eine *Reihe* von *linearen* Gleichungssystemen. △

Notation: $\mathbb{R}^{m \times n}$ ist die Menge der Matrizen

$$A = \begin{pmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{pmatrix}$$

mit Einträgen $a_{i,j} \in \mathbb{R}$.

In diesem Kapitel geht es um folgende zentrale

Aufgabe:

Zu $A \in \mathbb{R}^{n \times n}$ und $b = (b_1, \dots, b_n)^T \in \mathbb{R}^n$ bestimme ein $x = (x_1, \dots, x_n)^T \in \mathbb{R}^n$, das

$$\begin{array}{rcccc} a_{1,1}x_1 & + \dots + & a_{1,n}x_n & = & b_1 \\ \vdots & & \vdots & & \vdots \\ a_{n,1}x_1 & + \dots + & a_{n,n}x_n & = & b_n \end{array}$$

bzw. kurz

$$Ax = b$$

erfüllt.

Bemerkung 3.6

Folgende Aussagen sind äquivalent für das System $Ax = b$:

- (i) Das System hat für jedes $b \in \mathbb{R}^n$ eine eindeutige Lösung $x \in \mathbb{R}^n$.
- (ii) Die Matrix A hat vollen Rang n .
- (iii) Das *homogene* System $Ax = 0$ nur die triviale Lösung $x = 0$.
- (iv) Es gilt $\det A \neq 0$.

A heißt *regulär* oder *nichtsingulär*, wenn $\det A \neq 0$.

△

Annahme:

In den Abschnitten 3.2-3.8 wird stets angenommen, daß $\det A \neq 0$ gilt.

3.2 Kondition und Störungssätze

Wir betrachten zunächst den einfachsten Fall, daß *nur* die rechte Seite b gestört ist.

Satz 3.7. Sei $x + \Delta x$ die Lösung von $A(x + \Delta x) = b + \Delta b$.
Dann gilt:

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

Der relative Fehler der Lösung läßt sich also durch das Vielfache

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

des relativen Fehlers der rechten Seite (Eingabedaten) abschätzen. Die Größe $\kappa(A)$ heißt (*relative*) *Konditionszahl* (bzgl. der Norm $\|\cdot\|$) der Matrix A .

$\kappa(A)$ beschreibt auch maßgeblich die Störungen in den übrigen Eingabedaten.

Satz 3.9. Es gelte $\kappa(A) \frac{\|\Delta A\|}{\|A\|} < 1$. Sei $x + \Delta x$ die Lösung von

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

Dann gilt:

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

Bemerkung 3.10.

In einer Maschine mit Maschinengenauigkeit eps sind die Daten A, b mit relativen Fehlern $\leq \text{eps}$ behaftet. Aufgrund des Satzes 3.9 sagt man, daß es wegen der Kondition des Problems $(A, b) \rightarrow x = A^{-1}b$ einen für die Bestimmung von x *unvermeidlichen Fehler* in der Größenordnung

$$\kappa(A) \text{ eps}$$

gibt.



Beispiel 3.11.

Sei

$$A = \begin{pmatrix} 3 & 1.001 \\ 6 & 1.997 \end{pmatrix}, \quad b = \begin{pmatrix} 1.999 \\ 4.003 \end{pmatrix} .$$

Dann gilt

$$A^{-1} = \frac{1}{-0.015} \begin{pmatrix} 1.997 & -1.001 \\ -6 & 3 \end{pmatrix}, \quad \|A^{-1}\|_{\infty} = 600 ,$$

$$\|A\|_{\infty} = 7.997, \quad \kappa_{\infty}(A) = 4798.2 .$$

Für

$$\tilde{A} = \begin{pmatrix} 3 & 1 \\ 6 & 1.997 \end{pmatrix}, \quad \tilde{b} = \begin{pmatrix} 2.002 \\ 4 \end{pmatrix}$$

hat man

$$\Delta A = \begin{pmatrix} 0 & -0.001 \\ 0 & 0 \end{pmatrix}, \quad \|\Delta A\|_\infty = 0.001,$$
$$\Delta b = \begin{pmatrix} 0.003 \\ -0.003 \end{pmatrix}, \quad \|\Delta b\|_\infty = 0.003.$$

Aus Satz 3.9 ergibt sich somit

$$\frac{\|\Delta x\|_\infty}{\|x\|_\infty} \leq 10.49$$

als Schranke für den relativen Fehler der Lösung \tilde{x} von $\tilde{A}\tilde{x} = \tilde{b}$.

Da die exakte Lösung $x = (1, -1)^T$ das System $Ax = b$ und $\tilde{x} = (0.2229, 1.3333)^T$ das System $\tilde{A}\tilde{x} = \tilde{b}$ löst, ist der tatsächliche relative Fehler

$$\frac{\|\Delta x\|_\infty}{\|x\|_\infty} \approx 2.333$$

△

Gleichungssystem $Ax = b$.

Annäherung der Lösung: \tilde{x} .

Residuum $\tilde{r} = b - A\tilde{x}$, ist ohne Kenntnis der Lösung x berechenbar.

Beachte: $\tilde{r} = 0 \Leftrightarrow \tilde{x} = x$.

Wie aussagekräftig ist die Größe des Residuums in Bezug auf den tatsächlichen Fehler ?

Hängt wieder von der Kondition ab.

Für die Norm des Residuums im Vergleich zu der des Fehlers gilt:

$$\kappa(A)^{-1} \frac{\|\tilde{r}\|}{\|b\|} \leq \frac{\|x - \tilde{x}\|}{\|x\|} \leq \kappa(A) \frac{\|\tilde{r}\|}{\|b\|} .$$

Beispiel 3.12.

Sei A die Matrix aus Beispiel 3.11 und $b = (3, 6)^T$.

Exakte Lösung: $x = (1, 0)^T$.

Für die Annäherungen

$$\tilde{x} = \begin{pmatrix} 0.99684 \\ 0.00949 \end{pmatrix}, \quad \hat{x} = \begin{pmatrix} 1.000045 \\ 0.000089 \end{pmatrix},$$

gilt

$$\|\tilde{r}\|_{\infty} = \|b - A\tilde{x}\|_{\infty} = 1.95 \cdot 10^{-5}$$

$$\|\hat{r}\|_{\infty} = \|b - A\hat{x}\|_{\infty} = 4.48 \cdot 10^{-4}.$$

Die Norm des Residuums für \tilde{x} ist also viel kleiner als für \hat{x} :

$$\|\tilde{r}\|_{\infty} \ll \|\hat{r}\|_{\infty}.$$

Der Fehler in \tilde{x} ist aber viel größer als in \hat{x} :

$$\|\tilde{x} - x\|_{\infty} = 9.49 \cdot 10^{-3} \gg \|\hat{x} - x\|_{\infty} = 8.90 \cdot 10^{-5}.$$

△

3.2.1 Zeilenskalierung

Eine einfache Methode, die **Kondition** günstig zu beeinflussen, ist die *Zeilenskalierung* (Zeilenäquilibrierung).

Darunter versteht man die Multiplikation der i -ten Zeile mit einer Zahl $d_i \neq 0$ ($1 \leq i \leq n$).

Das Gleichungssystem $Ax = b$ wird dadurch in ein äquivalentes System

$$D_z Ax = D_z b$$

umgeformt, wobei D_z die Diagonalmatrix $D_z = \text{diag}(d_1, \dots, d_n)$ bezeichnet.

Man kann nun versuchen, die Skalierung D_z so zu wählen, daß die Kondition des Gleichungssystems (wesentlich) verbessert wird.

Sei D_z die Diagonalmatrix mit Diagonaleinträgen definiert durch

$$d_i = \left(\sum_{j=1}^n |a_{i,j}| \right)^{-1} .$$

Für die skalierte Matrix gilt

$$\sum_{j=1}^n |(D_z A)_{i,j}| = 1 \quad \text{für alle } i,$$

also sind die *Betragssummen aller Zeilen gleich eins*. Eine Matrix mit dieser Eigenschaft heißt *zeilenweise äquilibriert*.

Die Skalierung mit D_z hat folgende Optimalitätseigenschaft:

$$\kappa_{\infty}(D_z A) \leq \kappa_{\infty}(DA) \quad \text{für jede reguläre Diagonalmatrix } D ,$$

d.h., diese Zeilenskalierung liefert die minimale Konditionszahl bezüglich der Maximumnorm.

Beispiel 3.14.

Für

$$A = \begin{pmatrix} 8 & 10000 \\ 50 & -60 \end{pmatrix}$$

erhält man $\kappa_\infty(A) = 201.2$ und

$$D_z A = \begin{pmatrix} 0.799 \cdot 10^{-3} & 0.999 \\ 0.455 & -0.545 \end{pmatrix}, \quad \kappa_\infty(D_z A) = 3.40 .$$

Hierbei wurde die erste Zeile mit $\frac{1}{1008}$, die zweite Zeile mit $\frac{1}{110}$ multipliziert. △

3.4 Dreiecksmatrizen, Rückwärtseinsetzen

Dreiecksmatrizen ergeben leicht lösbare Systeme.

Eine Matrix $R = (r_{i,j})_{i,j=1}^n \in \mathbb{R}^{n \times n}$ heißt *obere Dreiecksmatrix*, falls

$$r_{i,j} = 0 \quad \text{für } i > j$$

gilt. Die avisierte „leichte Lösbarkeit“ läßt sich am Beispiel einer oberen Dreiecksmatrix illustrieren:

$$\begin{array}{rcccccc} r_{1,1}x_1 & + & r_{1,2}x_2 & + & \cdots & + & r_{1,n}x_n & = & b_1 \\ & & r_{2,2}x_2 & + & \cdots & + & r_{2,n}x_n & = & b_2 \\ & & & & \cdots & & & & \vdots \\ & & & & & & r_{n-1,n-1}x_{n-1} & + & r_{n-1,n}x_n & = & b_{n-1} \\ & & & & & & & & r_{n,n}x_n & = & b_n. \end{array} \quad (1)$$

Da bekanntlich

$$\det R = r_{1,1}r_{2,2} \cdots r_{n,n},$$

ist (1) genau dann stets eindeutig lösbar, wenn alle Diagonaleinträge $r_{j,j}$, $j = 1, \dots, n$, von Null verschieden sind.

Dies erlaubt folgendes Vorgehen: Aus der letzten Gleichung in (1) erhält man sofort

$$x_n = b_n / r_{n,n}.$$

Einsetzen von x_n in die zweitletzte Gleichung erlaubt die Auflösung nach x_{n-1} :

$$x_{n-1} = (b_{n-1} - r_{n-1,n}x_n) / r_{n-1,n-1}.$$

Allgemein sieht dies wie folgt aus:

Rückwärtseinsetzen:

Für $j = n, n - 1, \dots, 2, 1$ berechne

$$x_j = \left(b_j - \sum_{k=j+1}^n r_{j,k}x_k \right) / r_{j,j},$$

wobei die für $j = n$ leer ist und als Null interpretiert wird.

Beispiel 3.16.

$$R = \begin{pmatrix} 3 & -1 & 2 \\ 0 & 1 & 3 \\ 0 & 0 & 2 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 1 \\ 4 \end{pmatrix}$$

$$x_3 = 4/2 = 2,$$

$$x_2 = 1 - 3 \cdot 2 = -5,$$

$$x_1 = \frac{1}{3}(0 - (-1)(-5) - 2 \cdot 2) = -3$$

△

Rechenaufwand 3.17.

Für jedes $j = n - 1, \dots, 1$:

$n - j$ Multiplikationen / Additionen,
eine Division,
und für $j = n$ eine Division. Also insgesamt

- $\sum_{j=1}^{n-1} (n - j) = \frac{n(n - 1)}{2}$ Additionen / Multiplikationen,
- n Divisionen.

*Der Rechenaufwand für Rückwärtseinsetzen ist also ca.
 $\frac{1}{2}n^2$ Operationen.*

Operation = Multiplikation oder Division.

Eigenschaften 3.18.

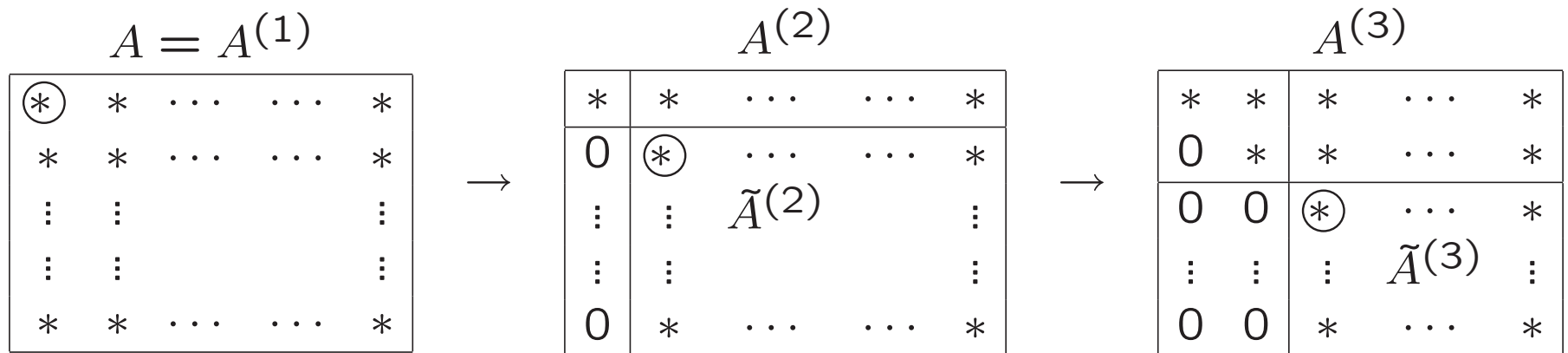
- Das Produkt von oberen (unteren) Dreiecksmatrizen ist wieder eine obere (untere) Dreiecksmatrix.
- Die Inverse einer oberen (unteren) nichtsingulären Dreiecksmatrix ist wieder eine obere (untere) Dreiecksmatrix.
- Die Determinante einer Dreiecksmatrix ist gerade das Produkt aller Diagonaleinträge.

3.5 Gauß-Elimination, LR-Zerlegung

Die bekannteste Methode, das System

$$Ax = b \quad (\det A \neq 0)$$

auf Dreiecksgestalt zu bringen, ist die *Gauß-Elimination*.



Die Einträge der Matrix $A^{(k)}$ werden mit $a_{i,j}^{(k)}$ notiert.

Der Eintrag $a_{k,k}^{(k)}$ ($\textcircled{*}$ oben) heißt *Pivotelement*.

In entsprechender Weise ist natürlich auch die rechte Seite b umzuformen.

Es ergibt sich das folgende Verfahren:

Gauß-Elimination

Gegeben $A \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n,$
($\det A \neq 0$).

Für $j = 1, 2, \dots, n - 1$

und falls $a_{j,j}^{(j)} \neq 0$ ist

für $i = j + 1, j + 2, \dots, n$

subtrahiere in $(A \ b)$ Zeile j mit

Faktor $\ell_{i,j} := \frac{a_{i,j}^{(j)}}{a_{j,j}^{(j)}}$ von Zeile i .

Das Resultat hat die Form

$$(R \ c),$$

wobei R eine obere Dreiecksmatrix ist.

Der Algorithmus versagt, falls ein $a_{j,j}^{(j)} = 0$.

Beispiel 3.19.

$$\begin{pmatrix} 2 & -1 & -3 & 3 \\ 4 & 0 & -3 & 1 \\ 6 & 1 & -1 & 6 \\ -2 & -5 & 4 & 1 \end{pmatrix} x = \begin{pmatrix} 1 \\ -8 \\ -16 \\ -12 \end{pmatrix}, \rightarrow (A \ b) = \begin{pmatrix} 2 & -1 & -3 & 3 & 1 \\ 4 & 0 & -3 & 1 & -8 \\ 6 & 1 & -1 & 6 & -16 \\ -2 & -5 & 4 & 1 & -12 \end{pmatrix}$$

Gauß-Elimination:

$$\begin{array}{l} j = 1 \\ \longrightarrow \\ l_{2,1} = \frac{4}{2} \\ l_{3,1} = \frac{6}{2} \\ l_{4,1} = \frac{-2}{2} \end{array} \begin{pmatrix} 2 & -1 & -3 & 3 & 1 \\ 0 & 2 & 3 & -5 & -10 \\ 0 & 4 & 8 & -3 & -19 \\ 0 & -6 & 1 & 4 & -11 \end{pmatrix} \begin{array}{l} j = 2 \\ \longrightarrow \\ l_{3,2} = \frac{4}{2} \\ l_{4,2} = \frac{-6}{2} \end{array} \begin{pmatrix} 2 & -1 & -3 & 3 & 1 \\ 0 & 2 & 3 & -5 & -10 \\ 0 & 0 & 2 & 7 & 1 \\ 0 & 0 & 10 & -11 & -41 \end{pmatrix}$$

$$\begin{array}{l} j = 3 \\ \longrightarrow \\ l_{4,3} = \frac{10}{2} \end{array} \begin{pmatrix} 2 & -1 & -3 & 3 & 1 \\ 0 & 2 & 3 & -5 & -10 \\ 0 & 0 & 2 & 7 & 1 \\ 0 & 0 & 0 & -46 & -46 \end{pmatrix} = (R \ c).$$

Wegen $Ax = b \Leftrightarrow Rx = c$ liefert Rückwärtseinsetzen die Lösung $x = (-4\frac{1}{2}, 2, -3, 1)^T$.

△

Der Prozeß

- Bestimme $(A \ b) \rightarrow (R \ c)$ gemäß (3.37), (3.38).
- Löse $Rx = c$.

heißt *Gauß-Elimination ohne Pivotisierung*.
Letzterer Zusatz wird später erläutert.

Andere Darstellung der Gauß-Elimination

Der Übergang von $A^{(j)} \rightarrow A^{(j+1)}$, lässt sich als Multiplikation des Systems mit einer regulären Matrix interpretieren.

Eine Matrix der Form

$$L_k = \begin{pmatrix} 1 & & & & & & \\ & \dots & & & & & \\ & & 1 & & & & \\ & & -\ell_{k+1,k} & 1 & & & \\ & \emptyset & \vdots & & \dots & & \\ & & -\ell_{n,k} & \emptyset & & & 1 \end{pmatrix},$$

mit $\ell_{k+1,k}, \dots, \ell_{n,k} \in \mathbb{R}$ beliebig, heißt *Frobenius-Matrix*.

Wählt man $\ell_{i,j} := \frac{a_{i,j}^{(j)}}{a_{j,j}^{(j)}}$ (vorausgesetzt $a_{j,j}^{(j)} \neq 0$), dann gilt

$$L_j A^{(j)} = A^{(j+1)}.$$

Iteriert man dies, ergibt sich

$$\underbrace{L_{n-1} L_{n-2} \cdots L_1 A}_{=: R} x = \underbrace{L_{n-1} L_{n-2} \cdots L_1 b}_{=: c},$$

Mit $e^k = (0, \dots, 0, 1, 0, \dots)^T$ gilt

$$\begin{aligned}L_k &= I - (0, \dots, 0, \ell_{k+1,k}, \dots, \ell_{n,k})^T (e^k)^T \\ \det L_k &= 1 \\ L_k^{-1} &= I + (0, \dots, 0, \ell_{k+1,k}, \dots, \ell_{n,k})^T (e^k)^T.\end{aligned}$$

Da die L_k invertierbar sind, ergibt sich

$$A = L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1} R =: LR,$$

wobei $L := L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1}$ wieder eine untere Dreiecksmatrix ist. Mehr noch: **L läßt sich sofort explizit angeben.** Dazu überprüft man wiederum per Rechnung, daß

$$L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1} = \begin{pmatrix} 1 & & & \\ \ell_{2,1} & \cdots & \emptyset & \\ \vdots & \cdots & \cdots & \\ \ell_{n,1} & \cdots & \ell_{n,n-1} & 1 \end{pmatrix}.$$

Ein bemerkenswertes „Nebenprodukt“ der Gauß-Elimination ist also eine *Faktorisierung* von A in ein Produkt einer normierten unteren Dreiecksmatrix L und einer oberen Dreiecksmatrix R .

Satz 3.21. Gilt im Gauß-Algorithmus stets (3.37), dann erhält man

$$A = LR,$$

wobei R durch (3.39) definiert ist und L die durch (3.38) definierte normierte untere Dreiecksmatrix ist.

Beispiel 3.22. Sei

$$A = \begin{pmatrix} 2 & -1 & -3 & 3 \\ 4 & 0 & -3 & 1 \\ 6 & 1 & -1 & 6 \\ -2 & -5 & 4 & 1 \end{pmatrix}$$

die Matrix aus Beispiel 3.19 und

$$L = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 2 & 1 & 0 \\ -1 & -3 & 5 & 1 \end{pmatrix}, \quad R = \begin{pmatrix} 2 & -1 & -3 & 3 \\ 0 & 2 & 3 & -5 \\ 0 & 0 & 2 & 7 \\ 0 & 0 & 0 & -46 \end{pmatrix}$$

die bei der Gauß-Elimination berechneten Dreiecksmatrizen.
Es gilt $A = LR$.

△

3.5.1 Gauß-Elimination mit Spaltenpivotisierung

Vertauschung von Zeilen ist sicherlich notwendig, wenn man ein verschwindendes Pivotelement trifft. Das folgende Beispiel zeigt, daß das Vertauschen von Zeilen sehr vorteilhaft sein kann, auch wenn die Pivotelemente nicht Null sind, wenn man Rundungsfehler mit einbezieht.

Beispiel 3.23.

$$\begin{pmatrix} 0.00031 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -3 \\ -7 \end{pmatrix}$$

Mit $\ell_{2,1} = 1/0.00031$ ergibt Gauß-Elimination

$$(R \ c) = \left(\begin{array}{cc|c} 0.00031 & 1 & -3 \\ 0 & 1 - \frac{1}{0.00031} & -7 - \frac{-3}{0.00031} \end{array} \right).$$

Bei 4-stelliger Rechnung ergibt sich daraus

$$\left(\begin{array}{cc|c} 0.00031 & 1 & -3 \\ 0 & -3225 & 9670 \end{array} \right)$$

Rückwärtseinsetzen liefert dann

$$\tilde{x}_1 \approx -6.452, \quad \tilde{x}_2 \approx -2.998.$$

Exakte Rechnung ergibt allerdings

$$x_1 = -4.00124\dots, \quad x_2 = -2.998759\dots,$$

d.h., \tilde{x}_1 ist auf keiner Stelle korrekt.

Dieses Ergebnis ist *unakzeptabel*, weil die Kondition des Problems sehr gut ist: $\kappa_\infty(A) = 4.00$.

Vertauscht man die Zeilen der Matrix und führt dann einen Eliminationsschritt durch, so ergibt sich ungefähr

$$(R \ c) = \left(\begin{array}{cc|c} 1 & 1 & -7 \\ 0 & 0.9997 & -2.998 \end{array} \right)$$

und bei 4-stelliger Arithmetik schließlich

$$x_1 \approx -4.001, \quad x_2 \approx -2.999,$$

also völlig akzeptable Werte.

Beachte:

Obwohl $a_{1,1} \neq 0$ gilt, ist hier offensichtlich eine Vertauschung von Zeilen angebracht. Es liegt also nahe, als Pivotelement stets das *betragsgrößte* Element der jeweiligen Spalte zu nehmen.

Nach obigen Überlegungen ist es also nicht sinnvoll, nur dann Gleichungen (Zeilen in der Matrix) zu vertauschen, wenn ein Nulleintrag dies erzwingt. Stattdessen setzt man *stets* bei der Berechnung der Matrix $A^{(j+1)}$ aus $A^{(j)}$ ein *betragsgrößtes* Element in der ersten Spalte der verbleibenden Untermatrix an die Pivotposition.

Da man das j -te Pivotelement in der j -ten Spalte sucht, nennt man diesen Vertauschungsvorgang *Spaltenpivotisierung*. Die Matrix sollte zuvor äquilibriert werden.

Analyse der Gauß-Elimination mit Spaltenpivotisierung

S_n : alle Permutationen auf der Menge $\{1, \dots, n\}$.

Beispiel: $\pi : \{1, 2, 3\} \rightarrow \{1, 2, 3\}$ mit $\pi(1) = 3, \pi(2) = 1, \pi(3) = 2$ ist eine Permutation in S_3 .

Bezeichnet man wieder mit e^i den i -ten Basisvektor, so läßt sich jeder Permutation $\pi \in S_n$ folgendermaßen eine Matrix zuordnen

$$P_\pi := \left(e^{\pi(1)} \ e^{\pi(2)} \ \dots \ e^{\pi(n)} \right)^T.$$

$P_{i,k}$: Vertauschung der i -ten und k -ten Zeile von I .

Zum Beispiel, für $n = 4, i = 2, k = 4$ erhält man:

$$P_{2,4} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

$$\det P_{i,k} = 1 \text{ für } i = k, \quad \det P_{i,k} = -1 \text{ für } i \neq k.$$

Bemerkung 3.24. Für $A \in \mathbb{R}^{n \times m}$ und $\pi \in S_n$ ist das Produkt $P_\pi A$ diejenige Matrix, die man aus A durch Vertauschen der Zeilen gemäß π erhält. △

Gauß-Elimination *mit* Spaltenpivotisierung ist für *jede* nichtsinguläre Matrix durchführbar und es gilt folgende Verallgemeinerung von Satz 3.21.

Satz 3.25. Zu jeder nichtsingulären Matrix A existiert eine Permutationsmatrix P , eine (dazu) eindeutige untere normierte Dreiecksmatrix L , deren Einträge sämtlich betragsmäßig durch eins beschränkt sind, und eine eindeutige obere Dreiecksmatrix R , so daß

$$PA = LR.$$

Die Matrizen P, L und R ergeben sich aus der Gauß-Elimination mit Spaltenpivotisierung.

Der Beweis dieses Satzes ist konstruktiv und verläuft im Prinzip genauso wie bei Satz 3.21. Der einzige Unterschied liegt darin, daß etwa im Schritt von $A^{(j)}$ nach $A^{(j+1)}$ vor der Elimination gegebenenfalls die j -te Zeile mit einer Zeile $i > j$ vertauscht wird. Sei $A^{(j)}$ die sich nach $j - 1$ Schritten der Gauß-Elimination mit Spaltenpivotisierung ergebende Matrix ($A^{(1)} = A$). Diese Matrix hat die Struktur

$$A^{(j)} = \left(\begin{array}{ccc|c} * & & * & \\ & \dots & & * \\ \hline \emptyset & & * & \\ & & & \\ & \emptyset & & \tilde{A}^{(j)} \end{array} \right), \quad \tilde{A}^{(j)} \in \mathbb{R}^{(n-j+1) \times (n-j+1)}$$

Insgesamt erhält man für die Gauß-Elimination mit Spaltenpivotisierung die Umformungskette

$$\underbrace{L_{n-1}P_{\tau_{n-1}}L_{n-2}\cdots P_{\tau_2}L_1P_{\tau_1}A}_{=:R}x = \underbrace{L_{n-1}P_{\tau_{n-1}}L_{n-2}\cdots P_{\tau_2}L_1P_{\tau_1}b}_{=:c},$$

Nach Konstruktion ist

$$R = L_{n-1}P_{\tau_{n-1}}L_{n-2}P_{\tau_{n-2}}\cdots P_{\tau_2}L_1P_{\tau_1}A.$$

eine obere Dreiecksmatrix.

Es gilt:

$$P_{\pi}L_kP_{\pi}^{-1} = P_{\pi}L_kP_{\pi}^T = \begin{pmatrix} 1 & & & & & & & \\ & \cdots & & & & & & \\ & & 1 & & & & & \\ & & -l_{\pi(k+1),k} & 1 & & & & \\ & \emptyset & \vdots & & \cdots & & & \\ & & -l_{\pi(n),k} & \emptyset & & \cdots & & 1 \end{pmatrix} =: \hat{L}_k,$$

Hieraus folgt $P_\pi L_k = \hat{L}_k P_\pi$, also kann man P_π in diesem Sinne „durch L_k ziehen“. In (3.52) kann man P_{τ_2} durch L_1 ziehen und erhält

$$R = L_{n-1} P_{\tau_{n-1}} L_{n-2} P_{\tau_{n-2}} \cdots P_{\tau_3} L_2 \hat{L}_1 P_{\tau_2} P_{\tau_1} A, \quad \hat{L}_1 := P_{\tau_2} L_1 P_{\tau_2}^T$$

Jetzt kann man P_{τ_3} durch L_2 und \hat{L}_1 ziehen, also

$$\begin{aligned} R &= L_{n-1} P_{\tau_{n-1}} L_{n-2} P_{\tau_{n-2}} \cdots L_3 \hat{L}_2 P_{\tau_3} \hat{L}_1 P_{\tau_2} P_{\tau_1} A \\ &= L_{n-1} P_{\tau_{n-1}} L_{n-2} P_{\tau_{n-2}} \cdots L_3 \hat{L}_2 \hat{\hat{L}}_1 P_{\tau_3} P_{\tau_2} P_{\tau_1} A \\ &\quad \text{mit } \hat{L}_2 := P_{\tau_3} L_2 P_{\tau_3}^T, \quad \hat{\hat{L}}_1 := P_{\tau_3} P_{\tau_2} L_1 P_{\tau_2}^T P_{\tau_3}^T. \end{aligned}$$

Man sieht nun das Muster, wie man durch analoge Wiederholung das Produkt der Transpositionen vor A sammelt. Man erhält schließlich

$$R = \tilde{L}_{n-1} \cdots \tilde{L}_1 P_{\pi_0} A, \quad \text{mit } \tilde{L}_k = P_{\pi_k} L_k P_{\pi_k}^T,$$

Hierbei ist

$$\pi_{n-1} = I, \quad \pi_k = \tau_{n-1} \cdots \tau_{k+1}, \quad k = 0, \dots, n-2.$$

Insgesamt erhält man

$$\tilde{L}_1^{-1} \cdots \tilde{L}_{n-1}^{-1} R = P_{\pi_0} A.$$

(! Fehler in (3.56) im Buch).

Da die \tilde{L}_k wieder Frobenius-Matrizen sind, folgt nun die behauptete Faktorisierung.

Skalierung und Gauß-Elimination mit Spaltenpivotisierung

- Bestimme die Diagonalmatrix

$$D = \text{diag}(d_1, \dots, d_n),$$

so daß DA zeilenweise äquilibriert ist, d.h.

$$d_i = \left(\sum_{k=1}^n |a_{i,k}| \right)^{-1}, \quad i = 1, \dots, n.$$

- Wende Gauß-Elimination mit Spaltenpivotisierung auf DA an. Im j -ten Schritt der Gauß-Elimination wählt man diejenige Zeile als Pivotzeile, die das betragsmäßig größte Element in der ersten Spalte der $(n + 1 - j) \times (n + 1 - j)$ rechten unteren Restmatrix hat. Falls diese Pivotzeile und die j -te Zeile verschieden sind, werden sie vertauscht.

***LR*-Zerlegung mit Skalierung und Spaltenpivotisierung**

Für $i = 1, 2, \dots, n$:

$$d_i \leftarrow 1 / \left(\sum_{k=1}^n |a_{i,k}| \right);$$

Für $j = 1, 2, \dots, n$:

$$a_{i,j} \leftarrow d_i a_{i,j}; \quad (\text{Skalierung})$$

Für $j = 1, 2, \dots, n - 1$:

Berechne p mit $j \leq p \leq n$, so daß $|a_{p,j}| = \max_{j \leq i \leq n} |a_{i,j}|$;

$$r_j = p;$$

Vertausche Zeile j mit Zeile p ; (Spaltenpivotisierung)

Für $i = j + 1, j + 2, \dots, n$:

$$a_{i,j} \leftarrow \frac{a_{i,j}}{a_{j,j}} \quad (\text{neue Einträge in } L)$$

Für $k = j + 1, j + 2, \dots, n$:

$$a_{i,k} \leftarrow a_{i,k} - a_{i,j} a_{j,k}; \quad (\text{neue Einträge in } R)$$

Aufwand.

- Zeilensummenberechnung: $n(n - 1)$ Additionen;
- Berechnung der Skalierung: n Divisionen;
- Für $j = 1, 2, \dots, n - 1$
 - Berechnung der neuen Einträge in L : $(n - j)$ Divisionen;
 - Berechnung der neuen Einträge in R : $(n - j)^2$ Multiplik./Additionen

Dominierender Aufwand: $\sum_{j=1}^{n-1} (n - j)^2 = \sum_{j=1}^{n-1} j^2 \sim n^3/3$.

Rechenaufwand 3.29. LR -Zerlegung über Gauß-Elimination mit Spaltenpivotisierung kostet ca.

$$\frac{1}{3}n^3 \text{ Operationen.}$$

Die Skalierung (falls nötig) kostet nur $\mathcal{O}(n^2)$ Operationen.

Beispiel 3.30.

$$A = \begin{pmatrix} 1 & 5 & 0 \\ 2 & 2 & 2 \\ -2 & 0 & 2 \end{pmatrix}, \text{ dann } D = \begin{pmatrix} \frac{1}{6} & 0 & 0 \\ 0 & \frac{1}{6} & 0 \\ 0 & 0 & \frac{1}{4} \end{pmatrix} \text{ und } DA = \begin{pmatrix} \frac{1}{6} & \frac{5}{6} & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ -\frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \quad (\text{Skalierung})$$

Gauß-Elimination mit Spaltenpivotisierung:

$j = 1 :$

$$DA \xrightarrow{\text{Vertauschung}} \begin{array}{|ccc|} \hline -\frac{1}{2} & 0 & \frac{1}{2} \\ \hline \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \hline \frac{1}{6} & \frac{5}{6} & 0 \\ \hline \end{array} \xrightarrow{\text{Elimination}} \begin{array}{|ccc|} \hline -\frac{1}{2} & 0 & \frac{1}{2} \\ \hline -\frac{2}{3} & \frac{1}{3} & \frac{2}{3} \\ \hline -\frac{1}{3} & \frac{5}{6} & \frac{1}{6} \\ \hline \end{array},$$

$j = 2 :$

$$\xrightarrow{\text{Vertauschung}} \begin{array}{|ccc|} \hline -\frac{1}{2} & 0 & \frac{1}{2} \\ \hline -\frac{1}{3} & \frac{5}{6} & \frac{1}{6} \\ \hline -\frac{2}{3} & \frac{1}{3} & \frac{2}{3} \\ \hline \end{array} \xrightarrow{\text{Elimination}} \begin{array}{|ccc|} \hline -\frac{1}{2} & 0 & \frac{1}{2} \\ \hline -\frac{1}{3} & \frac{5}{6} & \frac{1}{6} \\ \hline -\frac{2}{3} & \frac{2}{5} & \frac{3}{5} \\ \hline \end{array}.$$

$$\text{Also } L = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{3} & 1 & 0 \\ -\frac{2}{3} & \frac{2}{5} & 1 \end{pmatrix}, \quad R = \begin{pmatrix} -\frac{1}{2} & 0 & \frac{1}{5} \\ 0 & \frac{5}{6} & \frac{1}{3} \\ 0 & 0 & \frac{3}{5} \end{pmatrix}.$$

Man rechnet einfach nach, daß

$$LR = PDA$$

gilt, wobei $P = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}.$

Die Matrix P ist das Produkt von $P_{1,3}$ und $P_{2,3}$. △

Merke:

- Skalierung/Äquilibrierung verbessert die *Kondition* des Problems.
- Pivotisierung verbessert die *Stabilität* der Gauß-Elimination/*LR*-Zerlegung (vgl. Abschnitt 3.8)

Sei $A \in \mathbb{R}^{n \times n}$ eine Matrix die schon zeilenweise äquilibriert ist. Sei für diese Matrix die LR -Zerlegung $PA = LR$ bekannt.

Lösen eines Gleichungssystems

Die Lösung von

$$Ax = b$$

ergibt sich über die Lösung zweier Dreieckssysteme

$$\begin{aligned} Ax = b &\iff PAx = Pb &\iff LRx = Pb \\ & &\iff Ly = Pb \text{ und } Rx = y. \end{aligned}$$

Zuerst bestimmt man also y durch Vorwärtseinsetzen aus $Ly = Pb$, um danach x aus $Rx = y$ durch Rückwärtseinsetzen zu berechnen.

Mehrere rechte Seiten

Hat man mehrere Gleichungssysteme mit derselben Matrix A , aber verschiedenen rechten Seiten b , so benötigt man nur *einmal* den dominierenden Aufwand zur Bestimmung der LR -Zerlegung ($\sim \frac{1}{3}n^3$).

Für jede rechte Seite fällt dann nur die Lösung von $Ly = Pb$, $Rx = y$ für die verschiedenen rechten Seiten b an.

Die dazu benötigte Anzahl der Operationen ist jeweils $\sim n^2$, also von geringerer Ordnung.

Berechnung der Inversen

In den meisten Fällen ist es weder notwendig noch angebracht, die Inverse einer Matrix explizit zu berechnen. Dennoch gibt es Situationen, wo dies sinnvoll ist. Man kann dann folgendermaßen vorgehen. Sei $x^i \in \mathbb{R}^n$ die i -te Spalte der Inverse von A :

$$A^{-1} = \begin{pmatrix} x^1 & x^2 & \dots & x^n \end{pmatrix} .$$

Aus $AA^{-1} = I$ folgt

$$Ax^i = e^i, \quad i = 1, \dots, n .$$

Zur Berechnung der Inverse bietet sich folgende Strategie an:

- Bestimme die LR -Zerlegung $PA = LR$ über Gauß-Elimination mit Spaltenpivotisierung,
- Löse die Gleichungssysteme

$$LRx^i = Pe^i, \quad i = 1, \dots, n .$$

Gesamtaufwand: etwa $\frac{4}{3}n^3$ Operationen.

Berechnung von Determinanten

Aus $PA = LR$ folgt

$$\det P \det A = \det L \det R = \det R.$$

Wegen

$\det P = \det P_{n,r_n} \det P_{n-1,r_{n-1}} \cdots \det P_{1,r_1} = (-1)^{\#\text{Zeilenvertauschungen}}$,
folgt

$$\det A = (-1)^{\#\text{Zeilenvertauschungen}} \prod_{j=1}^n r_{j,j} .$$

Rechnerarithmetik: statt L und R erhält man Näherungen \tilde{L}, \tilde{R} .

Entsprechend ist der aus (3.59) berechnete Vektor \tilde{x} *nicht* die exakte Lösung von $Ax = b$.

Das Ziel der *Nachiteration* ist, diese Näherung *iterativ* zu verbessern.

Sei $x^0 := \tilde{x}$, $r = r^0 := b - Ax^0$. Der Fehler $\delta^0 := x - x^0$ ist gerade die Lösung des *Defektsystems*

$$A\delta^0 = Ax - Ax^0 = b - Ax^0 = r^0.$$

Verfahren:

Für $k = 0, 1, 2, \dots$, gegeben r^0 , berechne:

$$\tilde{L}y^k = r^k, \quad \tilde{R}\delta^k = y^k;$$

$$x^{k+1} := x^k + \delta^k;$$

$$r^{k+1} := b - Ax^{k+1};$$

Um eine Verbesserung zu erzielen, wird r^k in der Praxis oft mit doppelter Genauigkeit berechnet.

3.6 Cholesky-Zerlegung

Definition 3.31. $A \in \mathbb{R}^{n \times n}$ heißt symmetrisch positiv definit (s.p.d.), falls

$$A^T = A \quad (\text{Symmetrie})$$

und

$$x^T A x > 0 \quad (\text{positiv definit})$$

für alle $x \in \mathbb{R}^n$, $x \neq 0$, gilt.

Beispiel 3.32.

1. $A = I$ (Identität) ist s.p.d. Die Symmetrie ist trivial und

$$x^T I x = x^T x = \|x\|_2^2 > 0,$$

falls $x \neq 0$.

2. Sei $B \in \mathbb{R}^{m \times n}$, $m \geq n$, und B habe vollen Rang. Dann ist $A := B^T B \in \mathbb{R}^{n \times n}$ s.p.d., denn:

$$A^T = (B^T B)^T = B^T (B^T)^T = B^T B = A.$$

Sei $x \in \mathbb{R}^n$, $x \neq 0$. Dann gilt

$$x^T A x = x^T B^T B x = (Bx)^T (Bx) = \|Bx\|_2^2 \geq 0.$$

Es gilt $x^T A x = \|Bx\|_2^2 = 0$ nur falls $Bx = 0$ gilt.

Da B vollen Rang hat, muß daher $x = 0$ sein.

Satz 3.33.

$A \in \mathbb{R}^{n \times n}$ sei s.p.d. Dann gelten folgende Aussagen:

- (i) A ist invertierbar, und A^{-1} ist s.p.d.
- (ii) A hat nur strikt positive (insbesondere reelle) Eigenwerte.
- (iii) Jede Hauptuntermatrix von A ist s.p.d.
- (iv) Die Determinante von A ist positiv (und damit die Determinante aller Hauptuntermatrizen von A).
- (v) A hat nur strikt positive Diagonaleinträge und der betragsgrößte Eintrag von A liegt auf der Diagonalen.
- (vi) Bei der Gauß-Elimination ohne Pivotisierung sind alle Pivotelemente strikt positiv.

Satz 3.34 Jede s.p.d. Matrix $A \in \mathbb{R}^{n \times n}$ besitzt eine eindeutige Zerlegung

$$A = LDL^T,$$

wobei L eine normierte untere Dreiecksmatrix und D eine Diagonalmatrix mit Diagonaleinträgen

$$d_{i,i} > 0, \quad i = 1, \dots, n,$$

ist. Umgekehrt ist jede Matrix der Form LDL^T , wobei D eine Diagonalmatrix ist, die $d_{i,i} > 0$ erfüllt, und L eine normierte untere Dreiecksmatrix ist, symmetrisch positiv definit.

Diese Zerlegung heißt *Cholesky-Zerlegung*.

Aufgrund von Satz 3.33 (vi) ist bei s.p.d. Matrizen Gauß-Elimination *ohne Pivotisierung* durchführbar.

Beispiel 3.35.

$$A = \begin{pmatrix} 2 & 6 & -2 \\ 6 & 21 & 0 \\ -2 & 0 & 16 \end{pmatrix}, L = \begin{pmatrix} 1 & 0 & 0 \\ \ell_{2,1} & 1 & 0 \\ \ell_{3,1} & \ell_{3,2} & 1 \end{pmatrix}, D = \begin{pmatrix} d_{1,1} & 0 & 0 \\ 0 & d_{2,2} & 0 \\ 0 & 0 & d_{3,3} \end{pmatrix}.$$

Es gilt

$$\begin{aligned} LDL^T &= \begin{pmatrix} 1 & 0 & 0 \\ \ell_{2,1} & 1 & 0 \\ \ell_{3,1} & \ell_{3,2} & 1 \end{pmatrix} \begin{pmatrix} d_{1,1} & 0 & 0 \\ 0 & d_{2,2} & 0 \\ 0 & 0 & d_{3,3} \end{pmatrix} \begin{pmatrix} 1 & \ell_{2,1} & \ell_{3,1} \\ 0 & 1 & \ell_{3,2} \\ 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} d_{1,1} & 0 & 0 \\ \ell_{2,1}d_{1,1} & d_{2,2} & 0 \\ \ell_{3,1}d_{1,1} & \ell_{3,2}d_{2,2} & d_{3,3} \end{pmatrix} \begin{pmatrix} 1 & \ell_{2,1} & \ell_{3,1} \\ 0 & 1 & \ell_{3,2} \\ 0 & 0 & 1 \end{pmatrix}. \end{aligned}$$

Die elementweise Auswertung der Gleichung $LDL^T = A$, die man aufgrund der Symmetrie auf den unteren Dreiecksteil beschränken kann, ergibt

$$\begin{aligned}
j = 1: \quad (1,1)\text{-Element: } d_{1,1} = a_{1,1} = 2 &\implies \boxed{d_{1,1} = 2} \\
(2,1)\text{-Element: } l_{2,1}d_{1,1} = a_{2,1} = 6 &\implies l_{2,1} = 6/2 \implies \boxed{l_{2,1} = 3} \\
(3,1)\text{-Element: } l_{3,1}d_{1,1} = a_{3,1} = -2 &\implies l_{3,1} = -2/2 \\
&\implies \boxed{l_{3,1} = -1} \\
j = 2: \quad (2,2)\text{-Element: } l_{2,1}^2 d_{1,1} + d_{2,2} = a_{2,2} = 21 &\implies d_{2,2} = 21 - 9 * 2 \\
&\implies \boxed{d_{2,2} = 3} \\
(3,2)\text{-Element: } l_{3,1}d_{1,1}l_{2,1} + l_{3,2}d_{2,2} = a_{3,2} = 0 & \\
&\implies l_{3,2} = -(-1) * 2 * 3/3 \implies \boxed{l_{3,2} = 2} \\
j = 3: \quad (3,3)\text{-Element: } l_{3,1}^2 d_{1,1} + l_{3,2}^2 d_{2,2} + d_{3,3} = a_{3,3} = 16 & \\
&\implies d_{3,3} = 16 - (-1)^2 * 2 - 2^2 * 3 \implies \boxed{d_{3,3} = 2}
\end{aligned}$$

$$\implies L = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ -1 & 2 & 1 \end{pmatrix}, \quad D = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

△

Für die aufeinanderfolgenden Spalten, $k = 1, 2, \dots, n$, hat man explizite Formeln für $d_{k,k}$ und $l_{i,k}$ ($i > k$):

$$d_{k,k} = a_{k,k} - \sum_{j=1}^{k-1} l_{k,j}^2 d_{j,j} ,$$

$$l_{i,k} = \left(a_{i,k} - \sum_{j=1}^{k-1} l_{i,j} d_{j,j} l_{k,j} \right) / d_{k,k} .$$

Für $k = 1, 2, \dots, n$:

$$\text{diag} \leftarrow a_{k,k} - \sum_{j < k} a_{k,j}^2 a_{j,j};$$

falls $\text{diag} < 10^{-5} a_{k,k}$ Abbruch

$$a_{k,k} \leftarrow \text{diag},$$

für $i = k + 1, \dots, n$

$$a_{i,k} \leftarrow \left(a_{i,k} - \sum_{j < k} a_{i,j} a_{j,j} a_{k,j} \right) / a_{k,k};$$

Rechenaufwand 3.36. Man kann das Cholesky-Verfahren mit ca. $\frac{1}{6}n^3$ Multiplikationen und etwa ebenso vielen Additionen realisieren. Der Rechenaufwand beträgt also etwa die Hälfte des Aufwands der Standard-*LR*-Zerlegung.

Bemerkung 3.37.

- LDL^T entspricht der LR -Zerlegung für $R = DL^T$. Bei s.p.d. Matrizen ist Pivotisierung weder nötig noch sinnvoll. Beachte, daß Pivotisierung die Symmetrie der Matrix zerstören würde.

- Die Lösung des Problems $Ax = b$ reduziert sich wieder auf

$$Ly = b, \quad DL^T x = y, \quad \text{d.h. } L^T x = D^{-1}y.$$

- In obiger Version enthält das Verfahren die Abfrage

$$\text{diag} < 10^{-5} a_{k,k}.$$

Falls dies gilt, kann nicht mehr gewährleistet werden, daß das entsprechende Pivotelement strikt positiv ist. In diesem Sinne *testet* das Verfahren Positiv-Definitheit.

$$A = \begin{pmatrix} a_{1,1} & \cdots & a_{1,p} & 0 & \cdots & \cdots & 0 \\ \vdots & \cdots & & \cdots & 0 & & \vdots \\ a_{q,1} & & \cdots & & \cdots & 0 & \vdots \\ 0 & \cdots & & \cdots & & \cdots & 0 \\ \vdots & 0 & \cdots & & \cdots & & a_{n-p+1,n} \\ \vdots & & 0 & \cdots & & \cdots & \vdots \\ 0 & \cdots & \cdots & 0 & a_{n,n-q+1} & \cdots & a_{n,n} \end{pmatrix}.$$

Man sagt dann, A hat Bandbreite $p + q - 1$. Unter gewissen Voraussetzungen erlaubt die Bandstruktur eine erhebliche Reduktion des Rechen- und Speicheraufwands:

- Bei Gauß-Elimination *ohne* Pivotisierung bleibt die Bandbreite erhalten.
- Sei L, R die entsprechende LR -Zerlegung. Dann hat L die Bandbreite q und R die Bandbreite p .
- Der Rechenaufwand bei der LR -Zerlegung ist von der Ordnung pqn , beim Vor- und Rückwärtseinsetzen von der Ordnung $(p + q)n$.

Falls die Bandbreiten kleine Konstanten sind, wird der Aufwand also im wesentlichen durch n bestimmt, so daß sich insgesamt ein Rechenaufwand ergibt, der *proportional* zur Anzahl n der Unbekannten ist.

Ein wichtiger Spezialfall ergibt sich mit $p = q = 2$. Die allgemeine Form ist

$$\begin{pmatrix} a_{1,1} & a_{1,2} & 0 & \cdots & 0 \\ a_{2,1} & a_{2,2} & a_{2,3} & \ddots & \vdots \\ 0 & a_{3,2} & a_{3,3} & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & a_{n-1,n} \\ 0 & \cdots & 0 & a_{n,n-1} & a_{n,n} \end{pmatrix}.$$

Algorithmus zur Bestimmung der LR -Zerlegung:

Setze $r_{1,1} := a_{1,1}$.

Für $j = 2, 3, \dots, n$:

$$\ell_{j,j-1} = a_{j,j-1}/r_{j-1,j-1}$$

$$r_{j-1,j} = a_{j-1,j}$$

$$r_{j,j} = a_{j,j} - \ell_{j,j-1}r_{j-1,j}.$$

Rechenaufwand 3.39. Ca. $5n$ Operationen.

3.8 Stabilitätsanalyse bei der LR- und Cholesky-Zerlegung

Nach dem Prinzip der *Rückwärtsanalyse* wird das Ergebnis der Rechnung als Ergebnis *exakter* Rechnung zu *gestörten Eingabedaten* $A + \Delta A$ interpretiert.

Kann man ΔA abschätzen, so liefert der Störungssatz 3.9 über die Kondition Abschätzungen für den Fehler $\|\Delta x\|/\|x\|$.

Sei

$$Ax = b$$

mit A symmetrisch positiv definit. Das Cholesky-Verfahren wird eingesetzt.

Man kann zeigen, daß die *berechnete* Lösung \tilde{x} die *exakte* Lösung eines *gestörten* Systems

$$(A + \Delta A)\tilde{x} = b$$

ist, wobei man die Störung durch

$$\frac{\|\Delta A\|_\infty}{\|A\|_\infty} \lesssim c_n \text{ eps}$$

abschätzen kann.

Hierbei ist eps die Maschinengenauigkeit und c_n eine „kleine“ Zahl, welche nur von der Dimension n der Matrix A abhängt.

Die Störung ΔA ist in der Größenordnung der Datenrundungsfehler.

Deshalb ist das Resultat \tilde{x} mit einem Fehler behaftet, der in der Größenordnung des durch die Kondition des Problems bedingten *unvermeidbaren* Fehlers bleibt:

$$\frac{\|x - \tilde{x}\|_\infty}{\|x\|_\infty} \leq \frac{\kappa_\infty(A) \frac{\|\Delta A\|_\infty}{\|A\|_\infty}}{1 - \kappa_\infty(A) \frac{\|\Delta A\|_\infty}{\|A\|_\infty}} \lesssim \frac{\kappa_\infty(A) c_n \text{eps}}{1 - \kappa_\infty(A) c_n \text{eps}}$$

$$\approx \kappa_\infty(A) c_n \text{eps} \quad \text{wenn } \kappa_\infty(A) c_n \text{eps} \ll 1 .$$

Damit ist das Lösen eines s.p.d. Systems über das Cholesky-Verfahren **stabil**.

Falls A nicht symmetrisch positiv definit ist, kann man zur Lösung des Problems $Ax = b$ auf eine LR -Zerlegung der Matrix A zurückgreifen. Dazu wird die Gauß-Elimination eingesetzt.

Falls Gauß-Elimination **mit Spaltenpivotisierung** angewendet wird, kann man Resultate wie beim Cholesky-Verfahren zeigen.

Damit kann man die Lösung eines linearen Gleichungssystems über die Gauß-Elimination *mit Spaltenpivotisierung* als ein **stabiles** Verfahren einstufen.

Die Lösung über die Gauß-Elimination *ohne* Pivotisierung ist im allgemeinen *nicht* stabil.

Beispiel 3.40.

Betrachte

$$Ax = b,$$

wobei $A \in \mathbb{R}^{n \times n}$ eine sogenannte *Hilbertmatrix* ist:

$$A = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \cdots & \frac{1}{2n-1} \end{pmatrix}.$$

Wir wählen $b = \left(\frac{1}{n}, \frac{1}{n+1}, \dots, \frac{1}{2n-1}\right)^T$, d.h., $x = (0, 0, \dots, 0, 1)^T$.

A ist s.p.d. Sei $n = 12$. Cholesky-Verfahren auf einer Maschine mit $\text{eps} \approx 10^{-16}$. Das berechnete Resultat \tilde{x} ist mit einem Fehler

$$\frac{\|x - \tilde{x}\|_\infty}{\|x\|_\infty} \approx 1.6 * 10^{-2}$$

behaftet! Erklärung: $\kappa_\infty(A) = \|A\|_\infty \|A^{-1}\|_\infty \approx 10^{16}$.

Eine Alternative zur LR -Zerlegung bietet die QR -Zerlegung einer Matrix.

Die Matrix A wird zerlegt in eine Dreiecksmatrix und eine *orthogonale* matrix.

Orthogonale Matrizen

Eine Matrix $Q \in \mathbb{R}^{n \times n}$ heißt *orthogonal*, falls

$$Q^T Q = I,$$

d.h., falls die Spalten von Q eine Orthonormalbasis des \mathbb{R}^n bilden.

Die Inverse einer solchen Matrix ist also einfach zu bestimmen:

$$Q^{-1} = Q^T.$$

Satz 3.41.

Sei $Q \in \mathbb{R}^{n \times n}$ orthogonal. Dann gilt:

- (i) Q^T ist orthogonal.
- (ii) $\|Qx\|_2 = \|x\|_2$ für alle $x \in \mathbb{R}^n$.
- (iii) $\kappa_2(Q) = 1$.
- (iv) Für beliebiges $A \in \mathbb{R}^{n \times m}$ bzw. $A \in \mathbb{R}^{m \times n}$, $m \in \mathbb{N}$ beliebig, gilt $\|A\|_2 = \|QA\|_2 = \|AQ\|_2$.
- (v) Es gilt (für A wie vorhin) $\kappa_2(A) = \kappa_2(QA) = \kappa_2(AQ)$.
- (vi) Sei $\tilde{Q} \in \mathbb{R}^{n \times n}$ orthogonal, dann ist $Q\tilde{Q}$ orthogonal.

Bemerkungen

Grundlage ist folgende einfache Beobachtung: Gelingt es, $A \in \mathbb{R}^{n \times n}$ als Produkt

$$A = QR$$

zu schreiben, wobei Q orthogonal und R eine obere Dreiecksmatrix ist, so gilt

$$Ax = b \iff QRx = b \iff Rx = Q^T b,$$

d.h., das Problem reduziert sich wieder auf Rückwärtseinsetzen, falls R , also A , invertierbar ist.

Die QR -Zerlegung kann auch für allgemeine rechteckige $(m \times n)$ -Matrizen konstruiert werden.

Zwei Methoden: Givens-Rotationen und Householder-Transformationen.

3.9.1 Givens-Rotationen

Grundaufgabe:

Gegeben sei $(a, b)^T \in \mathbb{R}^2 \setminus \{0\}$. Finde $c, s \in \mathbb{R}$ mit

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} r \\ 0 \end{pmatrix}$$

und

$$c^2 + s^2 = 1.$$

Offensichtlich ist die obige Matrix dann orthogonal.

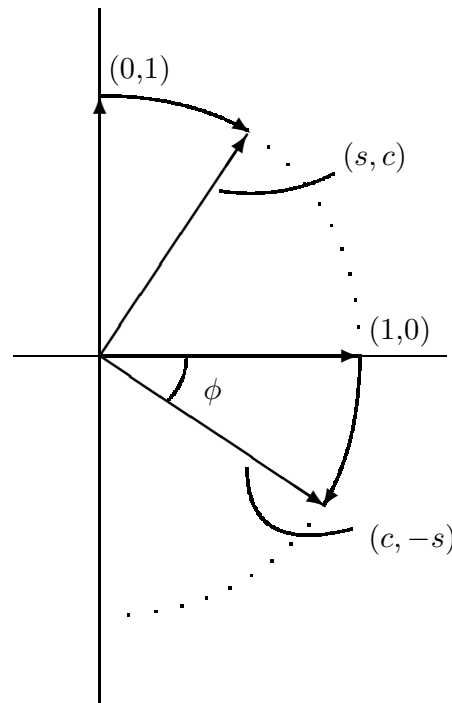
Bemerkung 3.42. Man kann wegen $c^2 + s^2 = 1$

$$c = \cos \phi, \quad s = \sin \phi$$

für ein $\phi \in [0, 2\pi]$ setzen, d.h., die obige Matrix stellt eine Drehung im \mathbb{R}^2 um den Winkel ϕ dar, vgl. Abb.3.3. Für die tatsächliche Rechnung wird ϕ allerdings *nie* benötigt. △

Da eine Drehung die Euklidische Länge eines Vektors unverändert läßt, gilt natürlich

$$\|(r, 0)^T\|_2 = |r| = \sqrt{a^2 + b^2} = \|(a, b)^T\|_2.$$



Die Lösung der Grundaufgabe ist:

$$r := \pm\sqrt{a^2 + b^2}, \quad c := \frac{a}{r}, \quad s := \frac{b}{r}.$$

$$G_{i,k} \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_{i-1} \\ r \\ x_{i+1} \\ \vdots \\ x_{k-1} \\ 0 \\ x_{k+1} \\ \vdots \\ x_m \end{pmatrix}$$

$i \rightarrow$ (points to the row containing r)
 $k \rightarrow$ (points to the row containing 0)

für

$$r = \pm \sqrt{x_i^2 + x_k^2}, \quad c = \frac{x_i}{r}, \quad s = \frac{x_k}{r}.$$

Beispiel 3.43.

$$\begin{pmatrix} 4 \\ -3 \\ 1 \end{pmatrix} \xrightarrow{G_{1,2}} \begin{pmatrix} 5 \\ 0 \\ 1 \end{pmatrix} \xrightarrow{G_{1,3}} \begin{pmatrix} \sqrt{26} \\ 0 \\ 0 \end{pmatrix}$$

$$\text{mit } G_{1,2} = \begin{pmatrix} \frac{4}{\sqrt{26}} & -\frac{3}{\sqrt{26}} & 0 \\ 0 & \frac{4}{\sqrt{26}} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad G_{1,3} = \begin{pmatrix} \frac{5}{\sqrt{26}} & 0 & \frac{1}{\sqrt{26}} \\ 0 & 1 & 0 \\ -\frac{1}{\sqrt{26}} & 0 & \frac{5}{\sqrt{26}} \end{pmatrix}. \quad \triangle$$

Beispiel 3.45.

$$\begin{pmatrix} * & * & * \\ * & * & * \\ * & * & * \\ * & * & * \end{pmatrix} \xrightarrow{\widetilde{G}_{1,2}} \begin{pmatrix} \circledast & \circledast & \circledast \\ 0 & \circledast & \circledast \\ * & * & * \\ * & * & * \end{pmatrix} \xrightarrow{\widetilde{G}_{1,3}} \begin{pmatrix} \circledast & \circledast & \circledast \\ 0 & * & * \\ 0 & \circledast & \circledast \\ * & * & * \end{pmatrix} \xrightarrow{\widetilde{G}_{2,3}} \begin{pmatrix} * & * & * \\ 0 & \circledast & \circledast \\ 0 & 0 & \circledast \\ * & * & * \end{pmatrix} \\
 \xrightarrow{\widetilde{G}_{1,4}} \begin{pmatrix} \circledast & \circledast & \circledast \\ 0 & * & * \\ 0 & 0 & * \\ 0 & \circledast & \circledast \end{pmatrix} \xrightarrow{\widetilde{G}_{2,4}} \begin{pmatrix} * & * & * \\ 0 & \circledast & \circledast \\ 0 & 0 & * \\ 0 & 0 & \circledast \end{pmatrix} \xrightarrow{\widetilde{G}_{3,4}} \begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & \circledast \\ 0 & 0 & 0 \end{pmatrix}.$$

Mit \circledast werden die Einträge angedeutet, die bei der Anwendung von $G_{i,k}$ neu berechnet werden müssen.

Die Reihenfolge $G_{1,2}, G_{1,3}, G_{1,4}, G_{2,3}, G_{2,4}, G_{3,4}$ wäre auch möglich.

△

Beispiel 3.46.

$$\begin{pmatrix} 3 & 5 \\ 0 & 2 \\ 0 & 0 \\ 4 & 5 \end{pmatrix} \xrightarrow{\widetilde{G}_{1,4}} \begin{pmatrix} 5 & 7 \\ 0 & 2 \\ 0 & 0 \\ 0 & -1 \end{pmatrix} \xrightarrow{\widetilde{G}_{2,4}} \begin{pmatrix} 5 & 7 \\ 0 & \sqrt{5} \\ 0 & 0 \\ 0 & 0 \end{pmatrix},$$

wobei

$$G_{1,4} = \begin{pmatrix} \frac{3}{5} & 0 & 0 & \frac{4}{5} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{4}{5} & 0 & 0 & \frac{3}{5} \end{pmatrix}, \quad G_{2,4} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{2}{\sqrt{5}} & 0 & -\frac{1}{\sqrt{5}} \\ 0 & 0 & 1 & 0 \\ 0 & \frac{1}{\sqrt{5}} & 0 & \frac{2}{\sqrt{5}} \end{pmatrix}. \quad \triangle$$

Die obige Konstruktion mit Givens-Rotationen zeigt, daß für *jede* Matrix $A \in \mathbb{R}^{m \times n}$ eine QR -Zerlegung existiert:

Satz 3.47. Sei $A \in \mathbb{R}^{m \times n}$. Dann existiert ein $Q \in \mathcal{O}_m(\mathbb{R})$ und eine obere Dreiecksmatrix $R \in \mathbb{R}^{m \times n}$ mit

$$A = QR,$$

wobei R im Sinne von (3.89), (3.90) zu verstehen ist.

Hinsichtlich der Implementierung der QR -Zerlegung über Givens-Rotationen bemerken wir, daß die Matrizen $G_{i,k}$ *nie explizit berechnet werden*.

***QR*-Zerlegung über Givens-Rotationen:**

- Das Verfahren *sehr stabil* (siehe auch [GL]). Pivotisierung ist *nicht* erforderlich.
- Durch Berücksichtigung von schon vorhandenen 0-Einträgen bei dünnbesetzten Matrizen läßt sich das Verfahren flexibel an die Struktur einer Matrix anpassen.
- Der Aufwand für die *QR*-Zerlegung einer vollbesetzten $m \times n$ -Matrix über Givens-Rotationen beträgt etwa $\frac{4}{3}n^3$ Operationen, falls $m \approx n$, und etwa $2mn^2$ Operationen, falls $m \gg n$. Zu beachten ist aber, daß für dünnbesetzte Matrizen der Aufwand wesentlich niedriger ist.
- Bei sogenannten *schnellen* Givens-Rotationen wird der Aufwand etwa halbiert ($\sim \frac{2}{3}n^3$, falls $n \approx m$; $\sim mn^2$, falls $m \gg n$).

Householder-Transformationen

Zu $v = (v_1, \dots, v_n)^T \in \mathbb{R}^n$, $v \neq 0$, definiert man die *Dyade*

$$vv^T := \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} (v_1, \dots, v_n) = \begin{pmatrix} v_1v_1 & \cdots & v_1v_n \\ \vdots & & \vdots \\ v_nv_1 & \cdots & v_nv_n \end{pmatrix}.$$

Man definiert nun die *Householder-Transformation*

$$Q_v = I - 2 \frac{vv^T}{v^T v}.$$

Eigenschaften 3.50.

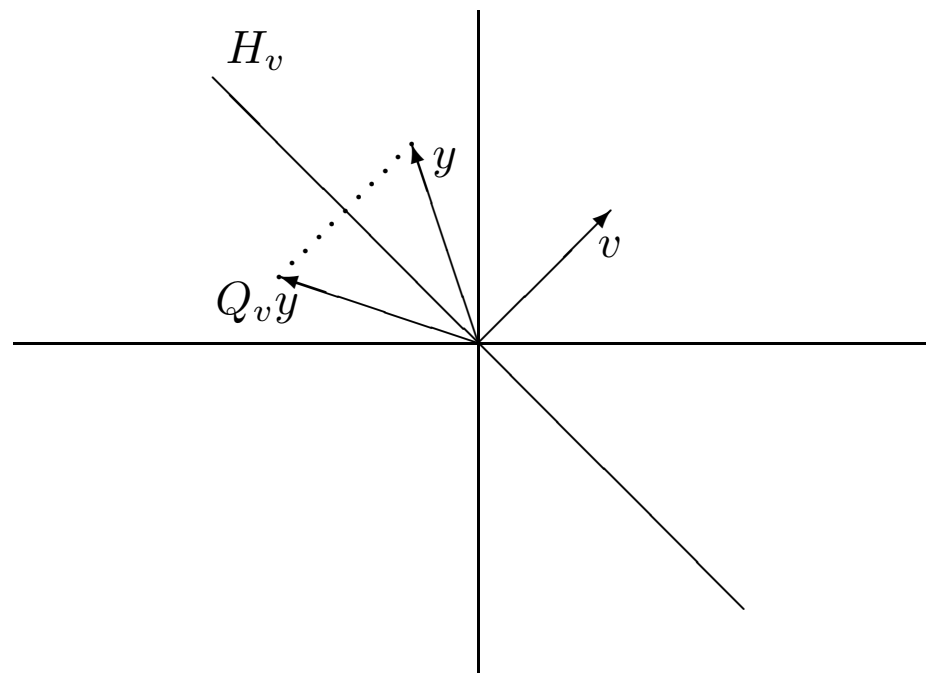
- $Q_v = Q_v^T$.
- $Q_v^2 = I$
- $Q_{\alpha v} = Q_v$, $\alpha \in \mathbb{R}$, $\alpha \neq 0$
- $Q_v y = y \iff y^T v = 0$.
- $Q_v v = -v$.

Die Householder Transformation Q_v ist orthogonal:

$$Q_v^{-1} = Q_v^T.$$

Geometrische Interpretation: Spiegelung.

Sei $H_v = \{x \in \mathbb{R}^n \mid x^T v = 0\}$.



Grundaufgabe

Zu $y \in \mathbb{R}^n$, $y \notin \text{span}(e^1)$, finde $v \in \mathbb{R}^n$, so daß

$$Q_v y = \pm \|y\|_2 e^1$$

gilt. Man erhält als Lösung der Grundaufgabe:

$$v = y \pm \|y\|_2 e^1 .$$

Um Auslöschung bei der Berechnung von

$$v = (y_1 \pm \|y\|_2, y_2, \dots, y_n)^T$$

zu vermeiden, wählt man: $v = y + \text{sign}(y_1) \|y\|_2 e^1$, mit $\text{sign}(0) := 1$.

Mit dieser Wahl erhält man insgesamt

$$\begin{aligned} \alpha &= \text{sign}(y_1) \|y\|_2 \\ v &= y + \alpha e^1 \\ Q_v y &= -\alpha e^1. \end{aligned}$$

Beispiel 3.51

Zu $y = (2, 2, 1)^T$ wird $v \in \mathbb{R}^3$ gesucht, so daß

$$Q_v y = \pm \|y\|_2 e^1 = \pm 3 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

gilt. Aufgrund von (3.94) ergibt sich $\alpha = 3$ und $v = y + \alpha e^1 = (5, 2, 1)^T$, so daß

$$Q_v y = \begin{pmatrix} -3 \\ 0 \\ 0 \end{pmatrix}.$$

Beachte: zur Berechnung von $Q_v y$ die explizite Form von Q_v

$$Q_v = I - 2 \frac{v v^T}{v^T v} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} - 2 \frac{\begin{pmatrix} 5 \\ 2 \\ 1 \end{pmatrix} (5, 2, 1)}{(5, 2, 1) \begin{pmatrix} 5 \\ 2 \\ 1 \end{pmatrix}} = \frac{1}{15} \begin{pmatrix} -10 & -10 & -5 \\ -10 & 11 & -2 \\ -5 & -2 & 14 \end{pmatrix}$$

nicht benötigt wird.

△

Reduktion auf obere Dreiecksform

Sei a^1 die erste Spalte der Matrix $A \in \mathbb{R}^{m \times n}$, $m \geq n$.

Man wendet die Grundaufgabe mit $y = a^1$ an, d.h., man setzt

$$v^1 = a^1 + \text{sign}(a_{1,1}) \|a^1\|_2 e^1, \quad Q_1 := Q_{v^1} \in \mathcal{O}_m(\mathbb{R}), \quad (6)$$

so daß sich

$$Q_1 A = \begin{array}{c|ccc} * & * & \cdots & * \\ 0 & & & \\ \vdots & & & \\ 0 & & & \end{array} \tilde{A}^{(2)} = A^{(2)}$$

mit $\tilde{A}^{(2)} \in \mathbb{R}^{(m-1) \times (n-1)}$ ergibt. Dann erhält man

$$Q_2 Q_1 A = \begin{pmatrix} * & * & * & \cdots & * \\ 0 & * & * & \cdots & * \\ 0 & 0 & & & \\ \vdots & \vdots & & & \\ 0 & 0 & & & \end{pmatrix} \tilde{A}^{(3)} = A^{(3)}.$$

$$Q_{n-1} \cdots Q_2 Q_1 A = R, \quad \text{also} \quad A = Q_1 Q_2 \cdots Q_{n-1} R = QR.$$

Beispiel 3.52.

$$A = \begin{pmatrix} 1 & 1 \\ 2 & 0 \\ 2 & 0 \end{pmatrix}, \quad v^1 = \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} + 3e^1 = \begin{pmatrix} 4 \\ 2 \\ 2 \end{pmatrix}.$$

$$Q_1 := Q_{v^1}, \quad Q_1 A = \left(Q_1 \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} \quad Q_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \right).$$

Für die zwei Spalten der Matrix $Q_1 A$ ergibt sich

$$Q_1 \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} = \begin{pmatrix} -3 \\ 0 \\ 0 \end{pmatrix} \quad (\text{siehe (3.94)}),$$

$$Q_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} - \frac{2}{(v^1)^T v^1} v^1 (v^1)^T \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} - \frac{1}{3} v^1 = \begin{pmatrix} -\frac{1}{3} \\ \frac{2}{3} \\ \frac{2}{3} \end{pmatrix}.$$

Daraus folgt

$$Q_1 A = \begin{pmatrix} -3 & -\frac{1}{3} \\ 0 & -\frac{2}{3} \\ 0 & -\frac{2}{3} \end{pmatrix}.$$

$$v^2 = \begin{pmatrix} -\frac{2}{3} \\ -\frac{2}{3} \end{pmatrix} - \frac{2}{3}\sqrt{2} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -\frac{2}{3}(1 + \sqrt{2}) \\ -\frac{2}{3} \end{pmatrix}.$$

$$\tilde{Q}_2 = \tilde{Q}_{v^2}; \quad \tilde{Q}_2 \begin{pmatrix} -\frac{2}{3} \\ -\frac{2}{3} \end{pmatrix} = \begin{pmatrix} \frac{2}{3}\sqrt{2} \\ 0 \end{pmatrix} \quad (\text{siehe (3.94)}).$$

Insgesamt erhält man

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & & \\ 0 & \tilde{Q}_2 & \end{pmatrix} Q_1 A = \begin{pmatrix} -3 & -\frac{1}{3} \\ 0 & \frac{2}{3}\sqrt{2} \\ 0 & 0 \end{pmatrix}.$$

***QR*-Zerlegung über Householder-Spiegelung:**

- Dieses Verfahren ist ebenfalls *sehr stabil*. Gesonderte Pivottisierung ist wiederum *nicht* erforderlich.
- Der Aufwand für die *QR*-Zerlegung einer vollbesetzten $m \times n$ -Matrix über Householder-Transformationen ist etwa $\frac{2}{3}n^3$ Operationen, falls $m \approx n$, und etwa mn^2 Operationen, falls $m \gg n$.

Wichtige Anwendungen der *QR*-Zerlegung:

Ausgleichsrechnung, Berechnung von Eigenwerten.