

1 Numerische Verfahren für EW-Probleme

$K = \mathbb{R}$ oder $K = \mathbb{C}$

Aufgabe: beide (oder alle) EW einer Matrix $A \in K^{n \times n}$,
 soll auf nicht d. zugeh. EV

Beob: - bei $n \geq 5$ kann es keine endl. Konvergenz der Alg geben,
 weil EW-Bestimmung äquivalent zur Nullstellensuche d.
 char. Polyn. ist \Rightarrow alle Alg., die wir vorkommen sind iterativ

Wir erinnern an einige Eigenschaften von Matrizen:

Satz 1.1 $A \in \mathbb{C}^{n \times n}$. Dann äquivalent:
 (i) A ist normal, d.h. $A^H A = A A^H$ unitär im Fall $K = \mathbb{C}$
 (ii) \exists ONB aus EV von A , d.h. \exists orthog. Q mit $Q^H A Q =$
 Diagonalmatrix

Beim Fischer § 6.6, Golub-Van Loan 7.12

J.a. können Matrizen nicht diagonalisiert werden. Von charakteristischer
 Polynom ist in d. Jordan-Normalform:

Satz 1.2 Sei $A \in K^{n \times n}$. Dann exist $X \in K^{n \times n}$ regulär s.d.

$A = X^{-1} J X$, wobei $J = \begin{pmatrix} J_1 & & \\ & J_2 & \\ & & \ddots \\ & & & J_p \end{pmatrix}$ Blockdiagonal-
 des geteilt \mathbb{R} .
 Jeder Block J_i hat die Form $J_i = \begin{pmatrix} \lambda_i & 1 & & \\ & \lambda_i & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{pmatrix} \in K^{m_i \times m_i}$, wobei

λ_i ein EW von A ist.

Beim 1.3 Die Jordanform bildet in d. Numerik wenig Anwendung,
 da sie nicht stabil unter Störungen ist: Die Menge der Matrizen
 mit paarweise verschiedenen EW (d.h. die Menge d. diagonalisierbaren Matrizen)
 liegt dicht in d. Menge aller Matrizen. (paarweise versch.)

Bsp: $A = \begin{pmatrix} \lambda_1 & 1 & & \\ & \lambda_1 & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_2 \end{pmatrix} \Rightarrow \tilde{A} = \begin{pmatrix} \lambda_1 + \epsilon & 1 & & \\ & \lambda_1 & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_2 \end{pmatrix}$ hat d. EW $\lambda_1, \lambda_1, \lambda_1 + \epsilon$

Numerisch stabil läßt sich jedoch d. sog. Schurform bestimmen

Satz 7.4 Zu jedem $A \in \mathbb{C}^{n \times n}$ \exists orth. $Q \in \mathbb{C}^{n \times n}$ & obere Dreiecksmatrix $R \in \mathbb{C}^{n \times n}$ mit $A = QRQ^H$

Beweis: siehe Fischer §5.4, Golub - van Loan 7.12 □
 - siehe Übung

Satz 7.5 Sei $A = QRQ^H$ eine Schurzerlegung von A . D.h.:

(i) die EW von A (entsprechend ihrer Vielfachheit) sind d. Diagonalelemente von R

(ii) Sei $Q = (q_1, \dots, q_n)$. Für jedes $k \in \{1, \dots, n\}$ ist der Raum $V_k := \langle q_1, \dots, q_k \rangle$ ein invarianter Unterraum von A , d.h. $AV_k \subset V_k$

Beweis: ad (i): A, R sind ähnlich, d.h. haben gleiches char. Polynom

ad (ii): $A = QRQ^H \Rightarrow AQ = QR$. Jedes $x \in V_k$ hat d. Darstellung $x = \sum_{i=1}^k \alpha_i q_i \Rightarrow$ mit $\underline{\alpha} := \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \\ \vdots \\ 0 \end{pmatrix}$ ergibt sich $Ax = A Q \underline{\alpha} = QR \underline{\alpha} = Q(R \underline{\alpha}) \in V_k$
 R obere Dreiecksmatrix □

In der Praxis ist man oft an reellen Matrizen interessiert und will alle Operationen über \mathbb{R} anführen. In dem Zusammenhang existiert eine reelle Schurform:

Satz 7.6 Zu jedem $A \in \mathbb{R}^{n \times n}$ existiert eine orthog. Matrix $Q \in \mathbb{R}^{n \times n}$ und eine "reelle obere Dreiecksmatrix" $R \in \mathbb{R}^{n \times n}$ mit $A = Q^T R Q$

Hier ist $R = \begin{pmatrix} \times & & & \\ & \times & & \\ & & \times & \\ & & & \times \end{pmatrix}$ d.h. auf der Diagonalen sind 2×2 Blöcke zugelassen & 1×1 Blöcke zugelassen

Bsp: Die Matrix $\begin{pmatrix} 2 & x & x \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}$ ist bereits in reeller Schur-Form. Die EW sind $\lambda_1 = 2, \lambda_{2,3} = \pm i$; damit ist (komplexe) Schurform

$$\begin{pmatrix} 2 & x & x \\ 0 & i & x \\ 0 & 0 & -i \end{pmatrix}$$

1.1 Kondition von Eigenwerten

Frage: in welcher Relation stehen $\sigma(A)$ zu $\sigma(A+\Delta A)$, wobei ΔA "klein" sei.

Zwar erwartet man, daß für Elemente λ die Mengen $\sigma(A)$ & $\sigma(A+\Delta A)$ in einem gew. Sinn nicht auseinander rutschen (so sind ja die Nullstellen von char. Polynom, deren Koeff. stetig von den Matrixeinträgen abhängen). Es gilt: $\forall \lambda \in \sigma(A) \exists \mu \in \sigma(A+\Delta A)$

$$(1.1) \quad \forall \mu \in \sigma(A+\Delta A) \quad \min_{\lambda \in \sigma(A)} |\mu - \lambda| \leq C_A \max \{ \|\Delta A\|_2, \|\Delta A\|_2^{1/n} \}$$

wobei $C_A > 0$ von A abhängt (siehe Platz, § 7.2.2, F. von Loan, Thm 7.2.3)

Die größte n -te Wurzel $\|\Delta A\|_2^{1/n}$ (siehe Fall von "Relevanz") heißt was:

$$\forall \mu \in \sigma(A+\Delta A) \quad \min_{\lambda \in \sigma(A)} |\mu - \lambda| \leq C_A \|\Delta A\|_2^{1/n}$$

Diese Aussage ist zu schwach in d. Praxis. Die Abschätzung ist jedoch im Wesentlichen scharf

Bsp 1.7 Für die Matrizen

$$A = \begin{pmatrix} a & 1 & & & \\ & a & & & \\ & & \ddots & & \\ & & & a & \\ & & & & a \end{pmatrix}, \quad \Delta A = \begin{pmatrix} 0 & & & & 0 \\ & \ddots & & & \\ & & \varepsilon & & \\ & & & \ddots & \\ \varepsilon & 0 & & & 0 \end{pmatrix} \quad \text{"klein"}$$

zwei char. Polynome $\chi_A, \chi_{A+\Delta A}$:

$$\chi_A(\lambda) = (a-\lambda)^n, \quad \chi_{A+\Delta A}(\lambda) = (a-\lambda)^n + (-1)^{n-1} \varepsilon, \text{ d.h.}$$

d. EW sind

$$\lambda_1 = \lambda_2 = \dots = \lambda_n = a \quad \text{und} \quad \lambda_k = a + \varepsilon^{1/n} \cdot \omega^k, \quad k=0, \dots, n-1$$

Man beachte: $\|\Delta A\|_2 = \varepsilon$

4
 um eine bessere Anordnung zu erhalten, muß also eine Strukturannahme an A gemacht werden:

Satz 1.8 (Bauer-Feire)

Sei $A \in \mathbb{K}^{n \times n}$ diagonalisierbar, d.h. $T^{-1}AT^{-1} = \text{diag}(\lambda_1, \dots, \lambda_n) =: D$

mit regulärem $T \in \mathbb{K}^{n \times n}$ und EW $\lambda_i, i=1, \dots, n$.

Sei $\|\cdot\|_p$ eine beliebige Norm auf \mathbb{K}^n . d.h.:

$$\forall \mu \in \sigma(A+AA) \quad \min_i |\mu - \lambda_i| \in \text{cond}_p(T) \cdot \|AA\|_p,$$

wobei $\text{cond}_p(T) = \|T\|_p \cdot \|T^{-1}\|_p$. Hier ist

$$\|T\|_p = \sup_{0 \neq x \in \mathbb{K}^n} \frac{\|Tx\|_p}{\|x\|_p}, \quad \|T^{-1}\|_p = \sup_{0 \neq x \in \mathbb{K}^n} \frac{\|T^{-1}x\|_p}{\|x\|_p}$$

$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}$$

Beweis:

OB-DA sei $\mu \in \sigma(A+AA) \setminus \sigma(A)$. Sei v EV zu μ . d.h.:

$$(A+AA) - \mu I) v = 0 \Rightarrow ((A - \mu I) + AA) v = 0 \Rightarrow$$

$$(I + (A - \mu I)^{-1} AA) v = 0$$

$$\Rightarrow \lambda = \frac{\|Av\|_p}{\|v\|_p} = \frac{\|(A - \mu I)^{-1} AA v\|_p}{\|v\|_p} \leq \|(A - \mu I)^{-1} AA\|_p =$$

$$\stackrel{A=TD^{-1}T^{-1}}{=} \|(TDT^{-1} - \mu TT^{-1})^{-1} AA\|_p \leq \|T(D - \mu)^{-1}T^{-1}\|_p \|AA\|_p \leq$$

$$\leq \text{cond}_p(T) \underbrace{\|(D - \mu)^{-1}\|_p}_{\substack{= \max_{1 \leq i \leq n} \\ D \text{ diag.}} \frac{1}{|\lambda_i - \mu|}} \|AA\|_p = \frac{1}{\min_i |\lambda_i - \mu|} \|AA\|_p$$

also $\min_{i \in \{1, \dots, n\}} |\lambda_i - \mu| \in \text{cond}_p(T) \|AA\|_p$

Bem: $\text{cond}_p(T)$ ist typischerweise klein, wenn T fast regulär ist, d.h. einige EV von A sind fast parallel. Falls A selbstadj., dann sind d. EV orthogonal, was im besten Fall ist:

Kor 1.9 $A \in \mathbb{K}^{n \times n}$ selbstadj., $\Delta A \in \mathbb{K}^{n \times n}$...

$$\forall \mu \in \sigma(A + \Delta A) : \min_{\lambda \in \sigma(A)} |\mu - \lambda| \leq \|\Delta A\|_2$$

Beweis: A selbstadj. \Rightarrow ^{Satz 1.1} $A = Q^H D Q$, $Q = \text{orthog.}$, $D = \text{diagonal}$,
 $\text{cond}_2(Q) = \|Q\|_2 \|Q^H\|_2 = 1$

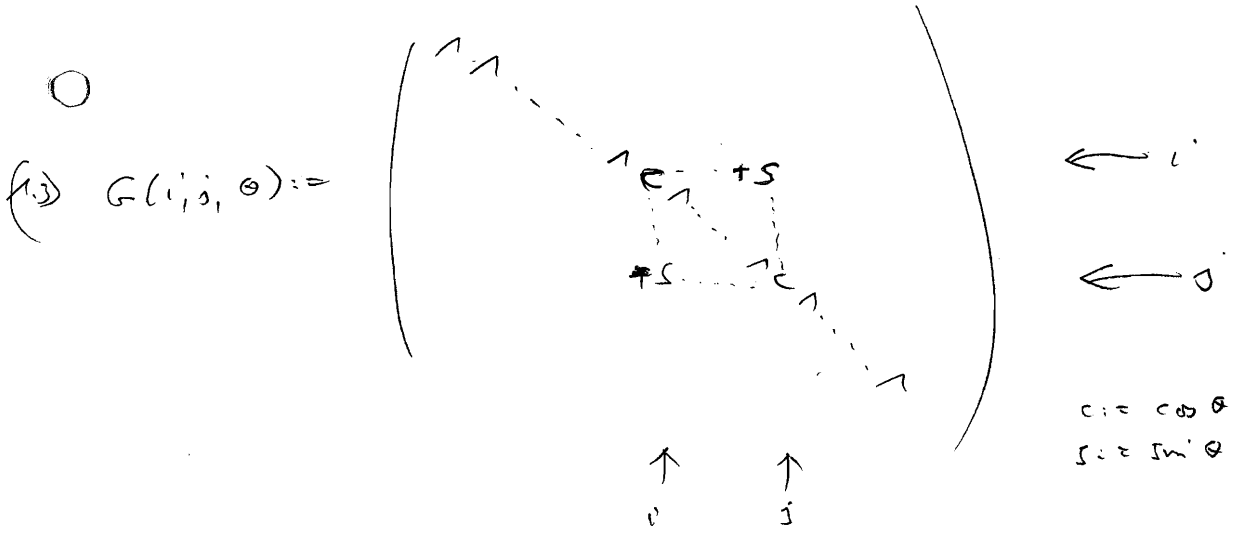
1.2 Jacobiverfahren

Für sym Matrizen $A \in \mathbb{R}^{n \times n}$ ist d. Jacobiverf. die älteste (1841) Methode, EW zu bestimmen. Es ist ein Iterationsverf.

(1.2) $A_{i+1} := Q_i^T A_i Q_i$, $A_0 = A$
 Hier Q_i orthog. Matrizen so bestimmt werden, daß die A_i gegen eine Diagonalmatrix konvergieren. Da in jedem Schritt Ähnlichkeits-Transformationen stattfinden, ist $\sigma(A_i) = \sigma(A)$ $\forall i$. Wenn A_i "hinreichend nahe" an Diagonalforn ist, dann liefern d. Diagonaleinträge von A_i eine (gute) Approx. an $\sigma(A)$.

1.2.1 Givens-Rotationen

Für $\theta \in [0, 2\pi)$ und $i \neq j$ ist



Bsp: $m=2$, d.h. $G(1, 2, \theta) = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}$ ist eine Drehung um Winkel $-\theta$

J.a. ist $G(i, j, \theta)$ eine lin. Abb., die auf d. Unterraum $\langle e_1, \dots, e_{i-1}, e_{i+1}, \dots, e_{j-1}, e_{j+1}, \dots, e_n \rangle$ die Identität ist und auf $\langle e_i, e_j \rangle$ eine Drehung um θ .

Lemma 1.10 Für alle $i \neq j$, $\theta \in [0, 2\pi)$ gilt die $G := G(i, j, \theta)$:
 und $\hat{G} := \begin{pmatrix} c & -s \\ s & c \end{pmatrix}$:

(i) G ist orthogonal

(ii) AG und A unterscheiden sich nur in d. Spalten i, j
 Zudem sind die Spalten i, j von AG Linearkombi. d. der Spalten
 i, j von A . Genauer:

$$(1.4) \quad (AG)(:, [i, j]) = A(:, [i, j]) \cdot \hat{G}$$

(iii) $G^T A$ & A unterscheiden sich nur in d. Zeilen i, j . Genauer:

$$(1.5) \quad (G^T A)([i, j], :) = \hat{G}^T A([i, j], :)$$

(iv) Für $B := G^T A G$, $\hat{B} := B([i, j], [i, j])$, $\hat{A} = A([i, j], [i, j])$ gilt

$$\hat{B} = \hat{G}^T \hat{A} \hat{G}$$

Beweis: nur Nachrechnen. Eine Möglichkeit, (ii) - (iv) zu sehen,
 ist G wie folgt zu schreiben:

$$G = I - P_{ij} P_{ij}^T + P_{ij} \hat{G} P_{ij}^T, \quad \text{wobei } P_{ij} = [e_i, e_j] \in \mathbb{R}^{n \times 2}$$

d.h. (ii) wegen:

$$(1) \text{ Addition: } (AG)(:, k) = AG e_k = A e_k = A(:, k)$$

$$(2) \text{ Addition: } (AG)(:, [i, j]) = AG [e_i, e_j] = AG P_{ij} = A (I - P_{ij} P_{ij}^T + P_{ij} \hat{G} P_{ij}^T) P_{ij} =$$

$$= A P_{ij} - A P_{ij} + \underbrace{A P_{ij}}_{=A} \hat{G} = A(:, [i, j]) \hat{G}$$

d.h. (iv) wegen:

$$\hat{B} = P_{ij}^T B P_{ij} = P_{ij}^T (I - P_{ij} P_{ij}^T + P_{ij} \hat{G} P_{ij}^T A) A (I - P_{ij} P_{ij}^T + P_{ij} \hat{G} P_{ij}^T)$$

$$= \hat{G}^T P_{ij}^T A P_{ij} \hat{G} = \hat{G}^T \hat{A} \hat{G}$$

□

weitere Anwendung von Gyrationsrotationen ist gegeben durch
 einer Matrix zu Null zu setzen. Wir führen im folgenden Lemma ein
 Bsp vor - siehe Übung für weitere Bsp

Lemma 1.11 Sei $A \in \mathbb{R}^{n \times n}$ sym, $i \neq j$. Dann gibt es eine
 Gyrationsrotation $G = G(i, j, \theta)$ derart, daß $B := G^T A G$ die Bed.
 $B_{ij} = B_{ji} = 0$ erfüllt. Für $c = \cos \theta$, $s = \sin \theta$ gilt:

$$(1.6) \quad \begin{cases} s = 0, c = 1 & \text{falls } A_{ij} = 0 \\ c = \frac{1}{\sqrt{1+t^2}}, s = \frac{t}{\sqrt{1+t^2}}, t = \frac{A_{ii} - A_{jj} \pm \sqrt{(A_{ii} - A_{jj})^2 + 4A_{ij}^2}}{+2A_{ij}}, A_{ij} \neq 0 \end{cases}$$

Beweis: Es braucht nur d. Fall $A_{ij} \neq 0$ betrachtet zu werden. Aus

(1) Lemma 1.10, (iv) folgt.

$$\begin{pmatrix} B_{ii} & B_{ij} \\ B_{ji} & B_{jj} \end{pmatrix} = \begin{pmatrix} c-s & \\ +s & c \end{pmatrix} \begin{pmatrix} A_{ii} & A_{ij} \\ A_{ji} & A_{jj} \end{pmatrix} \begin{pmatrix} c & +s \\ -s & c \end{pmatrix}$$

und in bes.

$$(1.7) \quad 0 \stackrel{!}{=} B_{ij} = B_{ji} = -(A_{jj} - A_{ii})sc + A_{ij}(c^2 - s^2)$$

Lösung der Gleichung konstruieren wir mit dem Ansatz
 $s = ct$, $t = \tan \theta$, [Beachte: $c \neq 0$, weil sonst $B_{ij} = -A_{ij} \neq 0$]

Einsetzen ergibt Kürzen mit c ergibt

$$0 \stackrel{!}{=} -(A_{jj} - A_{ii})t + (1-t^2)A_{ij}$$

Auflösen nach t :

$$t = \frac{A_{jj} - A_{ii} \pm \sqrt{(A_{ii} - A_{jj})^2 + 4A_{ij}^2}}{-2A_{ij}}$$

aus $s = ct \rightarrow s^2 = c^2 t^2 \Rightarrow c^2 = \frac{1}{1+t^2}$ ergeben sich die

$$\text{Lösung } c = \frac{1}{\sqrt{1+t^2}}, s = \frac{t}{\sqrt{1+t^2}} \quad \square$$

Bem: Für $A_{ij} \neq 0$ spielt das VZ in (1.6) keine Rolle. Die
 Vorzeichen auf die Wurzeln \pm werden unten eingesehen.

Das Jacobi-Verfahren (A.2) wird in jedem Schritt ein Neben-diagonalelement $B_{ij} \neq 0$ suchen und die Eigenrotation $G = G(i, j, \alpha)$ so wählen, daß $(G^T A G)_{ij} = 0$.
 Da die Eigenrotation natürlich d. anderen Einträge von A ebenfalls verändert, ist nicht klar, daß $G^T A G$ "näher" an Diagonalform ist als A. Um dies zu zeigen, definieren wir für bel. Matrizen $B \in \mathbb{R}^{n \times n}$ die Frobeniusnorm und den Neben-diagonalanteil:

(1.8) $\|B\|_F := \sqrt{\sum_{i,j} |B_{ij}|^2}$
 $off^2(B) := \|B - \text{diag}(B)\|_F^2 = \sum_{i \neq j} |B_{ij}|^2$ ○

Lemma 1.12 Sei $A \in \mathbb{R}^{n \times n}$ sym, $G = G(i, j, \alpha)$ Eigenrotation s.d. $i \neq j$ $B = G^T A G$ die Bed. $B_{ij} = 0 = B_{ji}$ erfüllt. d.g.

$off^2(B) = off^2(A) - 2A_{ij}^2$

Beweis:

• Verteilung: $Q \in \mathbb{R}^{m \times m}$ orthog., $z \in \mathbb{R}^m$ d.g. $\|z\|_2 = \|zQ\|_2$

(1.9) $\|zQ\|_F^2 = \|z\|_F^2 = \|zQ\|_F^2$ ○

Beweis: $\|zQ\|_F^2 = \sum_{k=1}^m \|zQ(:,k)\|_2^2 = \sum_{k=1}^m \|z\|_2^2 \|Q(:,k)\|_2^2 = \|z\|_F^2$

$\|zQ\|_F^2 = \|(zQ)^T\|_F^2 = \|Q^T z^T\|_F^2 = \|z^T\|_F^2 = \|z\|_F^2$

• 1. Schritt aus Lemma 1.10, (iv) ergibt sich

$\|B\|_F^2 = \|G^T \hat{A} G\|_F^2 = \|\hat{A}\|_F^2 = A_{ii}^2 + 2A_{ij}^2 + A_{jj}^2$

$\iff B$ diag. nach Wahl von G!

$B_{ii}^2 + B_{jj}^2$

$\implies A_{ii}^2 + A_{jj}^2 - B_{ii}^2 - B_{jj}^2 = -2A_{ij}^2$

2. Schritt

$$\begin{aligned} \text{off}^2(B) &= \|B\|_F^2 - \sum_{k=1}^n |B_{kk}|^2 = \|B\|_F^2 - \sum_{k \notin \{i,j\}} |B_{kk}|^2 - \sum_{k \in \{i,j\}} |B_{kk}|^2 \\ &= \|A\|_F^2 - \sum_{k \notin \{i,j\}} |A_{kk}|^2 - |B_{ii}|^2 - |B_{jj}|^2 \\ &= \|A\|_F^2 - \sum_{k=1}^n |A_{kk}|^2 + (|A_{ii}|^2 + |A_{jj}|^2 - |B_{ii}|^2 - |B_{jj}|^2) \\ &= \text{off}^2(A) - 2A_{ij}^2 \end{aligned}$$

□

Lemma 1.12 zeigt, daß die Neben-diagonalanteile beim Jacobi-
schritt abnimmt, wenn wir $A_{ij} \neq 0$. Um möglichst schnell
zu konvergieren, wird man (ii) zu wählen, daß $|A_{ij}| = \max_{k \neq l} |A_{kl}|$
Dies ergibt

Alg. 1.13 (Jacobi-Verfahren)

% tol > 0 : Genauigkeitsvorgabe
% A0 input: $A_0 \in \mathbb{R}^{n \times n}$ symmetrisch

- $R = 0$ (wahr (off(A_l) > tol) {
- suche (i,j) mit $i \neq j$ und $|A_{ij}| \geq |A_{rs}| \forall (r,s)$ mit $r \neq s$
 - bestimme Eigenrotation $G \in \mathbb{C}(i,j, \theta)$ aus Lemma 1.11, die $(G^T A_l G)_{ij} = 0$ macht
 - $A_{l+1} := G^T A_l G$
 - $l := l+1$
- }

Offensichtlich konvergiert die Folge (A_l) gegen eine Diagonalmatrix. Wir zeigen nun lineare Konvergenz

Satz 1.14 Sei $A_0 \in \mathbb{R}^{n \times n}$ sym. Sei $(A_l)_{l=1}^{\infty}$ die Folge von Matrizen, die in A_0 1.13 entsteht. d.g.:

$$\text{off}^2(A_{l+1}) \leq \left(1 - \frac{2}{n(n-1)}\right) \text{off}^2(A_l), \quad l=0, 1, \dots$$

Beweis: Schreibe $A = A_l$

Sei (i,j) mit $|A_{ij}| \geq |A_{rs}| \quad \forall (r,s)$ mit $r \neq s$

OBdA. (A sym!) sei $i < j$. d.g.:

$$\begin{aligned} \text{off}^2(A) &= \sum_{\substack{k,l \\ k \neq l}} |A_{kl}|^2 = 2 \sum_{\substack{k \\ k < l}} \sum_{\substack{l \\ k < l}} |A_{kl}|^2 \leq 2 \sum_l \sum_{k \neq l} |A_{ij}|^2 \\ &= 2 |A_{ij}|^2 \sum_{l=1}^n \sum_{k=1, k \neq l}^{l-1} 1 = 2n(n-1) |A_{ij}|^2 \end{aligned}$$

aus Lemma 1.13 damit

$$\begin{aligned} \text{off}^2(A_{l+1}) &= \text{off}^2(A) - 2|A_{ij}|^2 \leq \\ &\leq \text{off}^2(A) - \frac{2}{n(n-1)} \text{off}^2(A) \end{aligned} \quad (*)$$

$$= \left(1 - \frac{2}{n(n-1)}\right) \text{off}^2(A)$$

□

Bezüglich der Konvergenztheorie benötigen wir noch einen Zusammenhang zwischen d. EW der Matrizen A und A Größe $\kappa \text{ off}(A)$. Es gilt

Satz 1.15 Sei $A \in \mathbb{R}^{n \times n}$ sym. Sei $D = \text{diag}(A)$. d.h.:

$$\forall \lambda \in \sigma(A) \exists i \text{ mit } |\text{D}_{ii} - \lambda| \leq \text{off}(A)$$

Beweis: $\Delta A := D - A$. Aus Prop 1.9 folgt dann

$$\forall \mu \in \sigma(A) \min_{\lambda \in \sigma(\underbrace{A + \Delta A}_{= D})} |\lambda - \mu| \leq \|\Delta A\|_2$$

ii) nun ist $\|\Delta A\|_2^2 \leq \|\Delta A\|_F^2 = \text{off}^2(A)$ □

1.2.3 Bemerkungen zum Jacobiverfahren

Zur VZ-Wahl in Lemma 1.11: Für $G = G(i, j, c)$ wie in

Lemma 1.11 gilt für $B = G^T A G$:

$$\|B - A\|_F^2 = 4(1-c) \sum_{\substack{k=1 \\ k \neq \{i, j\}}}^n \left(A_{ki}^2 + A_{kj}^2 \right) + 2 \frac{A_{ij}^2}{c^2}$$

○ wenn wir es also in jedem Jacobi-Schritt die Matrix A nur möglichst wenig ändern wollen, dann sollte c möglichst nahe bei 1 sein, d.h. s möglichst nahe bei 0 sein.
 M.a.W.: der Drehwinkel θ sollte möglichst klein sein.
 Diese Forderung legt den VZ in der Def. von ϵ in Lemma 1.11 fest. Es ergibt sich damit folgender ϵ_{ij} der zudem Ausdrückung klein hält:

Abg 1.16 Funktion $(\tau, \sigma) = \text{sym.schur2}(A, i, j)$

% input sym. Matrix A, i, j mit $i \neq j$ liehert
 % output $\tau \in \mathbb{R}, \sigma \in \mathbb{C}$ von $G(i, j, \tau, \sigma)$, die $B_{ij} = 0$ ✓
 $B = G^T A G$

if $A_{ij} \neq 0$ {

$$\tau := \frac{A_{jj} - A_{ii}}{2 A_{ij}}$$

$$t := (\text{sign}(\tau)) \frac{1}{|\tau| + \sqrt{1 + \tau^2}} ; \quad c := \frac{1}{\sqrt{1 + \tau^2}} ; \quad s := \tau c$$

else { $c := 1; s := 0$ }

end

Bem:

1) Nachrechnen der Eigenschaft:

aus Lemma 1.11 folgt: $t = -\tau \pm \text{sign}(|B_{ij}|) \sqrt{1 + \tau^2} =$
 $= -\tau \pm \sqrt{1 + \tau^2}$

falls $\tau > 0$, dann ist die betragsmäßig kleinere $t_3 =$

$$t = -\tau + \sqrt{1 + \tau^2} = \frac{(\sqrt{1})^2 - \tau^2}{\tau + \sqrt{1 + \tau^2}} = \frac{1}{\tau + \sqrt{1 + \tau^2}}$$

falls $\tau < 0$, dann ist die betragsmäßig kleinere $t_3 =$

$$t = -\tau - \sqrt{1 + \tau^2} = \frac{(-\tau + \sqrt{1 + \tau^2})(\tau + \sqrt{1 + \tau^2})}{\tau + \sqrt{1 + \tau^2}} = \frac{\tau^2 - 1}{\tau + \sqrt{1 + \tau^2}} = -\frac{1}{\tau + \sqrt{1 + \tau^2}}$$

2) Man kann die VZ-Wahl auch so motivieren:

für $A_{ij} = 0$ wählt man $\sigma = 0$, d.h. keine Drehung
 für kleine A_{ij} wird man eine kleine Drehung wählen, i.d.
 |s| sollte klein sein. Dies wird durch die VZ-Wahl
 sichergestellt

zu den Kosten

- in der Praxis operiert man "auf der Matrix A " d.h. man "beschriftet" d. Matrix A in jedem Iterationsschritt mit d. Matrix A_{k+1}
- A_{k+1} unterscheidet sich von A_k nur in den Spalten & Zeilen i, j (siehe Lemma 1.10) \rightarrow Der Rechenaufwand pro Jacobischnitt ist $O(n)$
- Das Suchen des BetragmäÙig größten Neben diagonal- eintrags in jedem schritt kostet $O(n^2)$. In der Praxis wird man deshalb eine Reihenfolge von Indizes (i, j) festlegen und diese in dieser Reihenfolge abarbeiten. z.B. "cyclic by row" - Reihenfolge, z.B. geht man durch die Indizes $(1, 2), (1, 3), \dots, (1, n), (2, 3), (2, 4), \dots, (2, n), \dots, (n-1, n)$. Man kann zeigen, dass diese Variante d. Jacobiverf. konv.
- quadr. Konvergenz: $N = \frac{1}{2}(n(n-1)) = 2$ Anzahl Neben diagonal- einträge. N Schritte des Jacobiverfahrens berechnen man als "sweep". d.h. ist hinreichend große l : $off(A_{l+N}) \leq C \cdot off^2(A_0)$
- Auch d. "cyclic by row" Variante konv. quad.
- Parallelisierbarkeit: Nach Lemma 1.10 ändern sich nur zugehörige Spalten & Zeilen in jedem schritt \rightarrow man kann rel. einfach Jacobischnitte für Diplexpaare $(i, j), (i', j')$ mit $i \neq i', j \neq j'$ weitestgehend unabhängig voneinander ausführen
- das Vernachlässigen kleiner Neben diagonal- elemente ändert nur wenig die ϵ_w (siehe Kor 1.9) \rightarrow man kann sich Strategien überlegen, kleine Neben diagonal- einträge durch Null zu ersetzen um Rech.

1.3 Vektoriteration ("power method")

liefert: betragsmäßig größten EW & zugehörigen EV

Alg 1.17 ("power method")

% input: $A \in K^{n \times n}$, $0 \neq x_0 \in K^n$
 $l := 0$; $x_l := \frac{x_0}{\|x_0\|_2}$; $\tilde{\lambda}_0 := x_0^H A x_0$

repeat {

$x_{l+1} := \frac{A x_l}{\|A x_l\|_2}$ % approximative EV
 $\tilde{\lambda}_{l+1} := x_l^H A x_l$ % approximative EW

} until "genau genug"

Das Alg. konv. gegen d. betragsmäßig größten EW unter geeigneten Vor:

Satz 1.18 Habe $A \in K^{n \times n}$ eine Basis $\{v_1, \dots, v_n\}$ aus EV mit
 zW $\lambda_i, i=1, \dots, n$, die $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|$ erfüllen.

Sei $x_0 = \sum_{i=1}^n \alpha_i v_i$ mit $\alpha_1 \neq 0$. D. g.:

(i) die x_l aus Alg. 1.17 sind vgl. def.
 (ii) $\exists c > 0$ d.h. dass $|\tilde{\lambda}_l - \lambda_1| \leq c \left| \frac{\lambda_2}{\lambda_1} \right|^l$, $l=0, 1, \dots$

Beweis: Aus $x_0 = \sum_{i=1}^n \alpha_i v_i$ folgt $A^l x_0 = \sum_{i=1}^n \alpha_i \lambda_i^l v_i$. Weil
 $\alpha_1 \neq 0$ und $\lambda_1 \neq 0$ folgt also $A^l x_0 \neq 0 \forall l$. Man überlegt sich
 in direkter Weise, dass daraus auch folgt: $x_l \neq 0 \forall l$. Weiter ist:
 $x_l = c_l \cdot A^l x_0 = c_l \left(\alpha_1 \lambda_1^l v_1 + \sum_{i=2}^n \alpha_i \lambda_i^l v_i \right)$ für ein $c_l \neq 0$.

$$\Rightarrow \text{f. } l \rightarrow \infty \quad x_l = c_l \cdot \alpha_1 \lambda_1^l \left(v_1 + \underbrace{\sum_{i=2}^n \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^l v_i}_{=: \varepsilon_l} \right)$$

aus $|\lambda_2| < |\lambda_1|$ für $i=2,3,\dots,n$ folgt:

$$(1.7) \quad \|\epsilon_l\|_2 \leq \sum_{i=2}^n \left| \frac{\alpha_i}{\alpha_1} \right| \left| \frac{\lambda_i}{\lambda_1} \right|^l \|v_i\|_2 \leq C \left| \frac{\lambda_2}{\lambda_1} \right|^l \quad \text{für geeign. } C$$

Wg $\left| \frac{\lambda_2}{\lambda_1} \right| < 1$ können wir (für große l) $\|\epsilon_l\|_2$ als klein ansehen und erhalten:

$$\tilde{\lambda}_{l+1} = x_l^H A x_l = \frac{(v_1 + \epsilon_l)^H A (v_1 + \epsilon_l)}{\|v_1 + \epsilon_l\|_2^2} = \frac{v_1^H A v_1 + v_1^H A \epsilon_l + \epsilon_l^H A v_1 + \epsilon_l^H A \epsilon_l}{\|v_1 + \epsilon_l\|_2^2}$$

$$= \frac{\|v_1\|_2^2 \lambda_1 + O(\|\epsilon_l\|)}{\|v_1\|_2^2 + O(\|\epsilon_l\|)} = \lambda_1 + O(\|\epsilon_l\|)$$

$$\text{also } |\tilde{\lambda}_{l+1} - \lambda_1| \leq C \left| \frac{\lambda_2}{\lambda_1} \right|^l$$

□

Bem 7.18

1) Der EV v_1 ist natürlich in d. Praxis nicht bekannt; die Bedingung $\alpha_1 \neq 0$ ist insofern theoretisch nicht überprüfbar. In d. Praxis ist dies kein Problem: ^{erhält} erstens ist ein zufällig gewählter Startvektor x_0 diese Bedingung mit Wahrscheinlichkeit 1, zweitens erweisen Rundungsungenauigkeiten sofort sofort Komponenten in d. v_1 -Richtung

2) analoges Resultat gilt, falls λ_1 mehrfach d. EV ist

3) Der Alg. konv. nicht, falls es $\lambda_1 \neq \lambda_2$ mit $|\lambda_1| = |\lambda_2|$ gibt. Dies tritt z.B. als Problem auf, wenn $A \in \mathbb{R}^{n \times n}$, $x_0 \in \mathbb{R}^n$ aber A komplexe EW hat!

4) größter Schwachpunkt der Vektoriteration ist jedoch, daß d. Konvergenz langsam ist, falls λ_1 nicht gut vom Rest des

Spektrums separiert ist d.h. falls $\left| \frac{\lambda_2}{\lambda_1} \right| \approx 1$.

5) ...

- 5) die Zahl d. 2-Norm $\|\cdot\|_2$ in Alg 1.17 ist nicht wesentlich (17)
- 6) beliebige Anwendung: bestimme $\|A\|_2 = \sqrt{\lambda_{\max}(A^T A)}$

Alg 1.17 liefert nicht nur Approximationen \tilde{x}_k an d. EW λ_1 sondern auch Approximationen an d. zugehörigen EV v_1 . Um diese Konvergenz zu fassen müssen wir allerdings einen Konvergenz-
Kbegriff auf der Menge der Unterräume des \mathbb{K}^n einführen,
 denn wir können nicht $x_k \rightarrow v_1$ erwarten sondern
 höchstens " $\text{span}\{x_k\} \rightarrow \text{span}\{v_1\}$ ". Wir definieren den
 Abstand $d(S, T)$ zwischen 2 Unterräumen $S, T \subset \mathbb{K}^n$
 wie folgt

Def. 1.20 $S, T \subset \mathbb{K}^n$ Unterräume

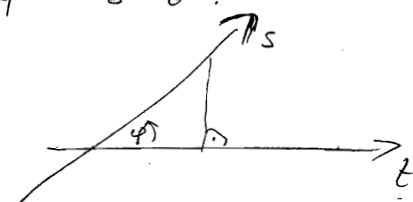
$$d(S, T) := \begin{cases} 1 & \text{falls } \dim(S) \neq \dim(T) \\ \sup_{0 \neq s \in S} \inf_{t \in T} \frac{\|s-t\|_2}{\|s\|_2} & \text{falls } \dim S = \dim T \end{cases}$$

Wir werden später sehen, daß d eine Metrik ist.
 Zum Verständnis der Verifikation benötigen wir nur den
 Fall $\dim S = \dim T = 1$. D.h.:

Bsp 1.21 Seien $s, t \in \mathbb{R}^n \setminus \{0\}$. O.B.d.A sei $\|s\|_2 = \|t\|_2 = 1$

$S := \text{span}\{s\}$, $T := \text{span}\{t\}$, $\cos \varphi = s^T t$

Dann ist $d(S, T) = |\sin \varphi|$



Wir sehen mit b:
 $d(S, T) = 0 \iff S = T$. Dies folgt auch aus Def. 1.20

Weiter zeigen wir die für jedes $s \in S$ gilt: $\frac{\|s - P_T s\|_2}{\|s\|_2} \leq d(S, T)$, wobei P_T die Orthogonalprojektion auf T ist.

Satz 1.22 Gilten d. Voraussetzungen von Satz 1.18. Dann existiert $c > 0$, so dass $d(\text{span}\{x\}, \text{span}\{v\}) \leq c \left| \frac{\lambda_2}{\lambda_1} \right|^k$.

Beweis aus (1.18) folgt $\text{span}\{x\} = \text{span}\{v_1 + \epsilon\}$. Damit folgt aus (1.11)

$$d(\text{span}\{x\}, \text{span}\{v_1\}) \leq \inf_{t \in \mathbb{K}} \frac{\|v_1 + \epsilon - t v_1\|_2}{\|v_1 + \epsilon\|_2} \leq \frac{\|\epsilon\|_2}{\|v_1 + \epsilon\|_2} \leq c \left| \frac{\lambda_2}{\lambda_1} \right|^k$$

(nach Bsp 1.21)

geometrische Interpretation Im Fall $\mathbb{K} = \mathbb{R}$ bedeutet die Konvergenz aus Satz 1.22 gerade, dass für $k \rightarrow \infty$ der Winkel zwischen dem Vektor x und v_1 gegen 0 geht [wir fordern hier, dass d. Winkel im $[0, \pi/2]$ liegt]

~~Bem 1.23 $\text{span}\{v\}$ ist ein invarianter Unterraum von A . Ergänzt man v_1 zu einer ONB $\{\tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_n\}$ von \mathbb{K}^n , so hat die Matrix A in der neuen Basis mit $\tilde{V} = [\tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_n]$ die Form~~

$$\tilde{V}^H A \tilde{V} = \begin{pmatrix} \tilde{v}_1^H A \tilde{v}_1 & \tilde{v}_1^H A \tilde{v}_2 \\ \tilde{v}_2^H A \tilde{v}_1 & \tilde{v}_2^H A \tilde{v}_2 \\ \vdots & \vdots \\ 0 & \tilde{v}_2^H A \tilde{v}_2 \\ \vdots & \vdots \end{pmatrix}$$

~~hier die Approx. x können wir analog vorziehen. Sei $Q = (q_1, \dots, q_m) \in \mathbb{K}^{m \times m}$ orthogonal mit $q_1 = x$. D.h. hier $Q_2 = (q_2, \dots, q_m)$~~

$$Q^H A Q = \begin{pmatrix} q_1^H A q_1 & q_1^H A Q_2 \\ q_2^H A q_1 & q_2^H A Q_2 \end{pmatrix}$$

1.4 inverse Iteration

Die Vektoriteration liefert den betragsmäßig größten EW. Den betragsmäßig kleinsten kann man durch Vektoriteration für d. Matrix A^{-1} erhalten, denn für diagonalisierbares $A = V D V^{-1}$ gilt

$$A^{-1} = V D^{-1} V^{-1} = V \begin{pmatrix} 1/\lambda_1 & & \\ & \dots & \\ & & 1/\lambda_n \end{pmatrix} V^{-1}, \text{ d. h. der betragsmäßig}$$

größte EW von A^{-1} ist d. betragsmäßig kleinste von A . wir erhalten

Alg 1.23 (inverse Iteration)

- $l_i := 0$; $x_0 := \frac{x_0}{\|x_0\|_2}$
repeat {
1) löse LGS $A \tilde{x}_{l+1} = x_l$

- $x_{l+1} := \frac{\tilde{x}_{l+1}}{\|\tilde{x}_{l+1}\|_2}$

- $\tilde{\lambda}_{l+1} := x_{l+1}^H A x_{l+1}$

- $l_i = l+1$

} until "genau genug"

Bem 1.24

1) Falls $0 < |\lambda_n| < |\lambda_{n-1}| \leq |\lambda_{n-2}| \leq \dots \leq |\lambda_1|$ dann folgt
analog zu Satz 1.18: $|\lambda_n - \tilde{\lambda}_l| \leq C \left| \frac{\lambda_n}{\lambda_{n-1}} \right|^l$

Übung: man prüfe dies nach!

2) Für jedem Schritt muß ein LGS $Ax = b$ gelöst werden \rightarrow
es bietet sich an, vorab eine LU-Faktorisierung von A
zu bestimmen.

3. DS

Alg. 1.24 (inverse Iteration mit Shift)

% input: $A \in \mathbb{K}^{n \times n}$, Shift $\lambda \in \mathbb{K}$, $x_0 \in \mathbb{K}^n \setminus \{0\}$

$l := 0$; $x_0 := \frac{x_0}{\|x_0\|_2}$

repeat {

- löse LGS $(A - \lambda) \tilde{x}_{l+1} = x_l$

- $x_{l+1} := \frac{\tilde{x}_{l+1}}{\|\tilde{x}_{l+1}\|_2}$

- $\tilde{\lambda}_{l+1} := x_{l+1}^H A x_{l+1}$

- $l := l+1$

} until "genau genug"

Man kann sich überlegen, daß analog zum Satz 1.18 & analog zur
inverse Iteration folgendes gilt:

Satz 1.25 Sei $A \in \mathbb{K}^{n \times n}$ diagonalisierbar. Sei $\lambda \in \mathbb{K}$ ein gegebenes Sk. A.
Seien die EW von A so nummeriert, daß

$$|\lambda_1 - \lambda| \geq |\lambda_2 - \lambda| \geq |\lambda_3 - \lambda| \geq \dots \geq |\lambda_n - \lambda| > 0 \text{ gilt. } \infty \text{ g. i.}$$

$\exists c > 0$ s. d. die Approximationen $\tilde{\lambda}_l$ aus Alg. 1.24 die

Abchätzung

$$(1.26) |\lambda_n - \tilde{\lambda}_l| \leq c \left| \frac{\lambda_n - \lambda}{\lambda_{n-1} - \lambda} \right|^l, \quad l = 0, 1, \dots$$

schätzen.

Beweis: Analog zum Beweis von Satz 1.18. [Übung]

Bem: relevant für Konvergenz ist das Verhältnis von
Abstand d. zu λ nächstgelegenen EW zum
Abstand d. zu λ zweitnächstgelegenen EW

Vektoriteration

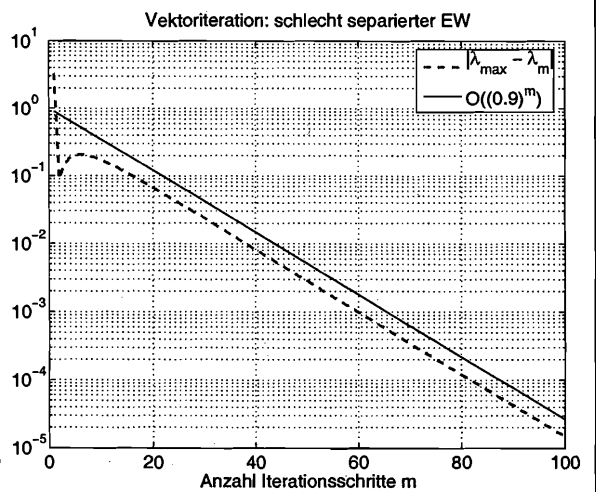
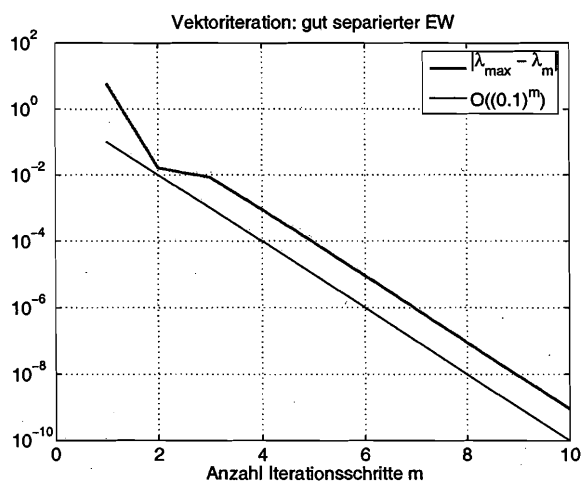
Iterationsvorschrift:

$$x_{l+1} := \frac{Ax_l}{\|x_l\|_2}$$

$$\tilde{\lambda}_{l+1} = x_{l+1}^\top Ax_{l+1}$$

$$A_1 = \begin{pmatrix} 10 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \lambda_1 = 10, \quad \lambda_2 = 1, \quad \lambda_3 = 0, \quad \left| \frac{\lambda_2}{\lambda_1} \right| = 0.1$$

$$A_2 = \begin{pmatrix} 10 & 1 & 0 \\ 0 & 9 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \lambda_1 = 10, \quad \lambda_2 = 9, \quad \lambda_3 = 0, \quad \left| \frac{\lambda_2}{\lambda_1} \right| = 0.9$$



$$A_3 = \begin{pmatrix} c & s & 0 \\ -s & c & 0 \\ 0 & 0 & 0.1 \end{pmatrix},$$

$$c = \cos(\pi/3), \quad s = \sin(\pi/3),$$

$$\lambda = 0.5 \pm 0.5\sqrt{3}, \quad \lambda = 0.1$$

Iterationszahl l	$\tilde{\lambda}_l$
1	0.36666666666667
2	0.49800995024876
3	0.49998000099995
4	0.4999980000010
5	0.4999999800000
6	0.4999999998000
7	0.4999999999980
8	0.5000000000000
9	0.5000000000000
10	0.5000000000000
11	0.5000000000000

In allen Beispielen ist $x_0 = (1, 1, 1)^\top$.

Inverse Iteration und Rayleighquotienteniteration

Inverse Iteration mit Shift λ :

$$\begin{aligned}\tilde{\lambda}_l &= x_l^\top A x_l \\ \tilde{x}_{l+1} &:= (A - \lambda)^{-1} x_l \\ x_{l+1} &:= \frac{\tilde{x}_{l+1}}{\|\tilde{x}_{l+1}\|_2}\end{aligned}$$

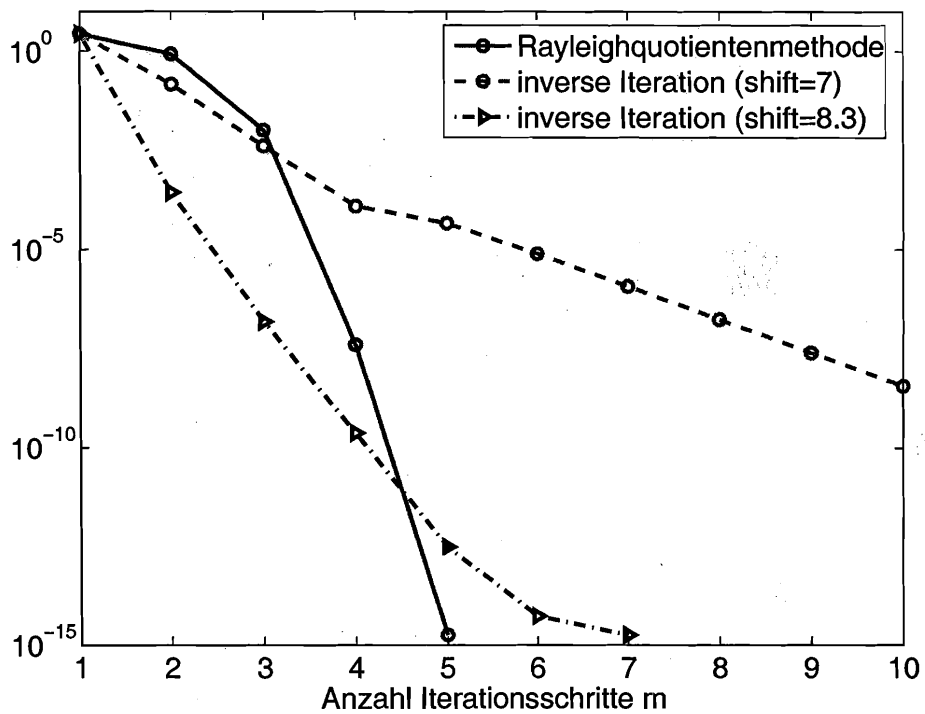
Rayleighquotientenmethode

$$\begin{aligned}\tilde{\lambda}_l &= x_l^\top A x_l \\ \tilde{x}_{l+1} &:= (A - \tilde{\lambda}_l)^{-1} x_l \\ x_{l+1} &:= \frac{\tilde{x}_{l+1}}{\|\tilde{x}_{l+1}\|_2}\end{aligned}$$

$$A = \begin{pmatrix} 10 & 1 & 0 \\ 1 & 9 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \lambda_1 \approx 10.6180, \quad \lambda_2 \approx 8.3820, \quad \lambda_3 = 0,$$

$x_0 = (1, -1, 0)^\top \rightsquigarrow$ Rayleighquotienteniteration konvergiert gegen λ_2

inverse Iteration und Rayleighquotientenmethode



Vorzüge von inverse Iteration mit Shift:

1) Satz 1.25 zeigt, daß die inverse Iteration mit Shift λ gegen den λ -nächstgelegenen EW konvergiert. Insb. ist es so möglich, gezielt EW zu bestimmen

2) geeignete Wahl d. Shifts kann die Konvergenz verbessern, Bsp: Seien die EW $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n < \lambda_{n+1}$, wobei $\lambda_n - \lambda_{n-1} = \delta$ klein ist, d.h. für die Vektornorm, daß die Approximationen $\tilde{\lambda}_k$ um λ_n die Bedingung ρ erfüllen. Für

$$\|\lambda_n - \tilde{\lambda}_k\| \leq C \left| \frac{\lambda_{n-1}}{\lambda_n} \right|^k = C \left(1 - \frac{\delta}{|\lambda_n|}\right)^k$$

kleine δ ist $1 - \frac{\delta}{|\lambda_n|}$ nahe bei 1!! Inverse Iteration einen

Shift $\lambda = \lambda_{n-1} + \frac{2}{3}\delta$ liefert nach (1.12)

$$\|\lambda_n - \tilde{\lambda}_k\| \leq C \left| \frac{\lambda_{n-1} - (\lambda_{n-1} + \frac{2}{3}\delta)}{\lambda_{n-1} + \delta - (\lambda_{n-1} + \frac{2}{3}\delta)} \right|^k = C \left(\frac{1}{2}\right)^k$$

Allg. erkennen wir aus (1.12), daß die inverse Iteration mit Shift desto besser konvergiert, je näher der Shiftparameter λ an dem gesuchten EW ist. Es liegt deshalb nahe, den Shiftparameter im Laufe d. Iteration anzupassen. Die beste verfügbare Approximation

ist der Rayleighquotient $\tilde{\lambda}_k = x_k^H A x_k$. Damit ergibt sich.

Alg 1.26 (Rayleighquotienteniteration)

% input: $A \in \mathbb{K}^{n \times n}$, $0 \neq x_0 \in \mathbb{K}^n$, Startwert in d. Nähe des EV

$l := 0$; $x_0 := \frac{x_0}{\|x_0\|_2}$

repeat {

- $\tilde{\lambda}_l := x_l^H A x_l$
- löse LGS $(A - \tilde{\lambda}_l I) x_{l+1} = x_l$
- $x_{l+1} := \frac{x_{l+1}}{\|x_{l+1}\|_2}$

} until "genau genug"

Man erwartet bessere Konvergenz als im Fall d. unren. Iteration mit
 fester Schrittweite. Wir zeigen nun, dass im Fall von selbstadjungierten
 Matrizen (lokal, d.h. falls d. Startvektor x_0 nicht genügt an einem
 EV ist) sogar kubische Konvergenz erhalten kann

Satz 1.27

Sei $A \in \mathbb{K}^{n \times n}$ selbstadjungiert, $\lambda \in \sigma(A)$ einfacher EW mit zugehörigem
 \mathbb{K} -eindeimensionalen Eigenraum V . Dann existiert $\epsilon_0 > 0$ und $C > 0$, s.d.
 $\forall \epsilon \in (0, \epsilon_0]$ folgendes gilt: Falls $x_0 \in \mathbb{K}^n$ die Bedingung $d(\text{span}\{x_0\}, V) < \epsilon$
 erfüllt, so erhält x_1 , welches sich aus einem Schritt von Alg. 1.26
 ergibt:
 $d(\text{span}\{x_1\}, V) \leq C \epsilon^3$ und $|x_1^H A x_1 - \lambda| \leq C \epsilon^3$

Beweis:

Vorbemerkung: wir nehmen an, dass $d(\text{span}\{x_0\}, V) > 0$, $\Leftrightarrow x_0 \notin V$
 weil sonst die Aufgabe, einen EV und EW zu finden, trivial ist
 $\square x_0 \in V$ impliziert $\frac{x_0^H A x_0}{\|x_0\|_2^2} = \lambda$; das man bereits einen EV
 und EV gefunden hat würde ein direkter Lösungsbeleg
 lösen von $(A - \lambda_0) \tilde{x} = x_0$ durch "Schreiben" der LU-
 Faktorisierung anzeigen \square

1. Schritt:

mit $\sqrt{d(\text{span}\{x\}, V)} = o(\epsilon)$ d.h.: $\exists v \in V$ mit
 $\|v\|_2 = 1$ s.d. $\|x - v\|_2 = o(\epsilon)$

Bew: Sei $P_V: \mathbb{K}^n \rightarrow V$ die Orthogonalprojektion. D.h. $P_V^2 = P_V$

$$\bullet \|x - P_V x\|_2 \leq \frac{\|x - v\|_2}{\|v\|_2} \leq d(\text{span}\{x\}, V) = o(\epsilon)$$

$$\bullet 1 = \|x\|_2 \geq \|P_V x\|_2 \geq \left| \|x\|_2 - \|P_V x - x\|_2 \right| = 1 - o(\epsilon)$$

Daraus folgt: $v := \frac{P_V x}{\|P_V x\|_2}$ erfüllt

$$\|x - v\|_2 \leq \|x - P_V x\|_2 + \left\| P_V x - \frac{P_V x}{\|P_V x\|_2} \right\|_2 =$$

$$= \|x - P_V x\|_2 + \|P_V x\| \left| 1 - \frac{1}{\|P_V x\|_2} \right| = o(\epsilon)$$

$\frac{c \cdot \epsilon - \gamma_1 \epsilon}{5 \epsilon} x_0$ wie im Satz gegeben. Sei $v \in V$ wie im 1. Schritt konstruiert aus EV von A zum EW λ_1 . Seien v_2, \dots, v_m EV von A zu EW $\lambda_i, i=2, \dots, m$, so daß $\{v_1, \dots, v_m\}$ ONB von \mathbb{K}^n bildet [A selbstadjungiert!].

Nach d. 1. Schritt gilt $\|x_0 - v_1\|_2 = O(\epsilon)$. Schreibt man x_0 in d. Basis $\{v_1, \dots, v_m\}$, so hat es d. Form

$$(1.13) \quad x_0 = (1 + \gamma_1) v_1 + \sum_{i=2}^m \gamma_i v_i \quad \text{mit} \quad \|\gamma\|_2 = O(\epsilon).$$

aus $\|x_0\|_2 = 1$ folgt $1 = x_0^H x_0 = |1 + \gamma_1|^2 + \sum_{i=2}^m |\gamma_i|^2$, d. h.

○ $|1 - |1 + \gamma_1|^2| \leq \|\gamma\|_2^2 = O(\epsilon^2)$. Somit gilt für d. Rayleigh-quotienten:

$$\tilde{\lambda}_0 = x_0^H A x_0 = \underbrace{|1 + \gamma_1|^2}_{= 1 + O(\epsilon^2)} \lambda_1 + \underbrace{\sum_{i=2}^m |\gamma_i|^2 \lambda_i}_{= O(\epsilon^2)}, \quad \text{d. h.}$$

$$(1.14) \quad |\lambda_1 - \tilde{\lambda}_0| = O(\epsilon^2)$$

2. Schritt Für d. Lsg \tilde{x}_1 von $(A - \tilde{\lambda}_0) \tilde{x}_1 = x_0$ schreiben wir $\tilde{x}_1 = \sum_{i=1}^m \alpha_i v_i$. Das LGS entkoppelt u. wir bekommen:

$$\begin{aligned} (\lambda_1 - \tilde{\lambda}_0) \alpha_1 &= 1 + \gamma_1 \\ (\lambda_i - \tilde{\lambda}_0) \alpha_i &= \gamma_i \quad i=2, 3, \dots, m \end{aligned}$$

aus (1.14) folgt:

$$|\alpha_1| = \frac{|1 + \gamma_1|}{|\lambda_1 - \tilde{\lambda}_0|} \geq c \epsilon^{-2} \quad \text{für geeignetes } c > 0$$

weiter $|\alpha_i| \leq \frac{|\gamma_i|}{|\lambda_i - \tilde{\lambda}_0|} \leq c |\gamma_i|$ für $i=2, \dots, m$, denn λ_1 ist einfacher EW, d. h. $\min_{i=2, \dots, m} |\lambda_1 - \lambda_i| > 0$

Zu Abbruchkriterien

Sei $x \in \mathbb{R}^n$ mit $\|x\|_2 = 1$ und $R(x) = x^\top Ax$. Sei A diagonalisierbar mit $V^{-1}AV = \Lambda$. Sei $r = Ax - R(x)x$. Dann gilt:

$$\min_i |\lambda_i - R(x)| \leq \text{cond}_2(V) \|r\|_2$$

$$\min_i |\lambda_i - R(x)| \leq \|r\|_2 \quad \text{falls } A \text{ symmetrisch}$$

$$\min_i |\lambda_i - R(x)| \leq C \|r\|_2^2 \quad \text{falls } A \text{ symmetrisch und } R(x) \text{ hinreichend nahe an einem einfachen EV}$$

Beispiel: Vektoriteration (Startvektor: $x_0 = (1, 1, 1)^\top$).

$A_1 = \begin{pmatrix} 10 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$			$A_2 = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 4.5 \end{pmatrix}, \quad \text{cond}_2(V) \approx 26.5.$		
m	$ \lambda_{\max} - R(x_m) $	$\frac{ \lambda_{\max} - R(x_m) }{\ r_m\ _2^2}$	m	$ \lambda_{\max} - R(x_m) $	$\frac{ \lambda_{\max} - R(x_m) }{\ r_m\ _2}$
1	5.78	2.5 ₋₁	1	2.0	1.1
2	4.6 ₋₂	1.09 ₋₁	10	5.3 ₋₂	8.2
3	3.6 ₋₄	1.08 ₋₁	20	2.2 ₋₁	10.8
4	2.8 ₋₆	1.08 ₋₁	30	1.5 ₋₂	11.6
5	2.1 ₋₈	1.08 ₋₁	40	4.6 ₋₃	11.8
6	1.7 ₋₁₀	1.08 ₋₁	50	1.4 ₋₃	11.9
7	1.3 ₋₁₂	1.08 ₋₁	60	4.3 ₋₄	11.9
8	1.0 ₋₁₄	0.9 ₋₁			

somit:

$$|\alpha_1| \geq C \varepsilon^{-2}, \quad \sum_{i=2}^m |\alpha_i|^2 \leq C \|\eta\|_2^2 = O(\varepsilon^2)$$

$$\begin{aligned} \text{Damit für } x_1 &= \frac{\tilde{x}_1}{\|\tilde{x}_1\|_2} \\ d(\text{span}\{x_1\}, V) &= d(\text{span}\{\tilde{x}_1\}, V) = \\ &= \inf_{t \in V} \frac{\|\tilde{x}_1 - t\|_2^2}{\|\tilde{x}_1\|_2^2} = \frac{\|\tilde{x}_1 - P_V \tilde{x}_1\|_2^2}{\|\tilde{x}_1\|_2^2} = \\ &= \frac{\sum_{i=2}^m |\alpha_i|^2}{\sum_{i=1}^m |\alpha_i|^2} = O(\varepsilon^2) \end{aligned}$$

also

$$(1.15) \quad d(\text{span}\{x_1\}, V) = O(\varepsilon^2)$$

4. Schritt: es bleibt z. z.: $|x_1^H A x_1 - \lambda| = O(\varepsilon^3)$. Dies folgt aus dem 3. Schritt, wenn x_0 dort durch x_1 ersetzt wird und das ε dort durch ε^2 ersetzt wird. \square

Bem 1.28

- 1) Im Fall von nicht selbstadj. Matrizen erwartet man quadr. Konvergenz
- 2) Inverse Iteration mit vermitteltem Schritt wird in d. Praxis nicht so oft verwendet, weil die Kosten für d. exakte Faktorisierung in jedem Schritt doch recht hoch sind - bei jedem Schritt können d. Kosten der Faktorisierung (vorab!) mit einer Iterationsschritte amortisiert werden

1.5 orthogonale Iteration

Die Vektoriteration besteht darauf, daß die Folge von Räumen

$(\mathbb{R}^l \text{ span}\{x_i\})_{i=0}^{\infty}$ gegen den Raum $\text{span}\{v_i\}$ konvergiert, wobei v_i die

EV zum betragsmäßig größten EW ist (Satz 1.22). Anstelle d. k -dimensionalen Raums $\text{span}\{x_i\}$ könnte man auch k -dimensional k -dimensionalen Raum verwenden und hoffen, daß man auf diese Weise gegen einen k -dimensionalen Unterraum konvergiert, der von den k dominanten EV (d.h. den EW, die zu den k betragsmäßig größten EW gehören) aufgespannt wird.

Wichtiges daran, daß bei der Vektoriteration in jedem Schritt eine Normalisierung stattfand. Das analoge Vorgehen hier ist, in jedem Schritt eine ONB von $(\mathbb{R}^l \text{ span}\{x_i\})$ zu erzeugen — dies wird mittels einer QR-Zerlegung gemacht. Damit ergibt sich:

Alg 1.29 (Orthogonale Iteration)

% input: $A \in \mathbb{K}^{n \times n}$, $X_0 \in \mathbb{K}^{n \times k}$ mit l.u. Spalten

$l := 0$;

$X_0 = Q_0 R_0$ mit $Q_0 \in \mathbb{K}^{n \times k}$ mit orthogonormalen Spalten; $R \in \mathbb{K}^{k \times k}$ = obere Dreiecksmatrix

repeat {

$$\tilde{X}_{l+1} := A Q_l$$

$$\tilde{X}_{l+1} := Q_{l+1} R_{l+1}$$

% QR-Zerlegung von \tilde{X}_{l+1} ; $Q_{l+1} \in \mathbb{K}^{n \times k}$ hat
% orthogonormal Spalten, $R \in \mathbb{K}^{k \times k}$ = obere Dreiecksmatrix

$l := l+1$

} until "genügend genau"

4. DS

Bem 1.30

1) unter geeigneten Voraussetzungen konvergieren die Räume T_l , die von Spalten der Matrizen Q_l (oder, was dasselbe ist, den Spalten d. \tilde{X}_l) aufgespannt werden, gegen einen k -invarianten Unterraum. Approximation an die entsprechenden EW wird geg. durch die EW der $k \times k$ -Matrix $Q_l^H A Q_l$ — siehe unten bei genauer Diskussion.

2) Die QR-Zerlegung im gleichen Schritt konstruiert mit ONB des Bildes $H^1 S_0$, wobei S_0 der k -dim. Raum ist, d. von 1. Spalten von X_0 aufgespannt ist. Diese Orthogonalisierung ist numerisch wichtig, weil sonst der dominante EV v_1 die anderen Komponenten "überlagert"

Zur Analyse von Alg. 1.29 benötigen wir die Metrik $d(\cdot, \cdot)$ auf der Menge der Unterräume des \mathbb{K}^n , die wir in Def. 1.20 eingeführt haben. Es gilt

Satz 1.31 Sei $d(\cdot, \cdot)$ wie in Def. 1.20 def. D. g.:

(i) d ist eine Metrik; insb. ist d sym. [diese Eigenschaft wird aber
später nicht
genutzt!]

(ii) $d(S, T) \leq 1$ für alle Unterräume S, T

(iii) $d(S^\perp, T^\perp) = d(S, T)$

Weiter gilt für Unterräume S, T mit $\dim S = \dim T$:

(iv) $d(S, T) < 1$ falls $S \cap T^\perp = \{0\}$

(v) $d(S, T) = \|P_{S^\perp} P_S\|_2$, wobei $P_S: \mathbb{K}^n \rightarrow S$ u. $P_{T^\perp}: \mathbb{K}^n \rightarrow T^\perp$ die Orthogonalprojektionen sind

(vi) $d(S, T) = \sin \theta$

Beweis: Der Fall, dass $\dim S \neq \dim T$ ist einfach — wir werden des-
halb nur dem Fall $\dim S = \dim T = k$ betrachten.

ad (i): folgt unmittelbar aus d. Def.

ad (ii): Wir zeigen nur d. Symmetrie — die anderen Eigenschaften sind einfach.

Seien $\underline{S}, \underline{T} \in \mathbb{K}^{n \times k}$ zwei Matrizen mit orthonormalen Spalten,

deren Spalten d. Räume S, T aufspannen. D. g.:

$$d(S, T) = \sup_{0 \neq s \in S} \inf_{t \in T} \frac{\|s - t\|_2}{\|s\|_2} = \sup_{\substack{s \in S \\ \|s\|_2 = 1}} \inf_{\substack{t \in T \\ \|t\|_2 = 1}} \|s - t\|_2$$

jeiles $s \in S$ und $t \in U$ kann geschrieben werden als
 $s = \sum \underline{\sigma}$, $t = \sum \underline{\tau}$ mit $\underline{\sigma} \in \mathbb{K}^l$, $\underline{\tau} \in \mathbb{K}^l$. Wegen Orthogonalität
 d. Spalten von \underline{S} , \underline{T} gilt: $\|\underline{s}\|_2^2 = \underline{\sigma}^H \underline{S}^H \underline{S} \underline{\sigma} = \|\underline{\sigma}\|_2^2$, $\|\underline{t}\|_2^2 = \|\underline{\tau}\|_2^2$

also:

$$\begin{aligned} d^2(S, T) &= \sup_{\substack{\underline{\sigma} \in \mathbb{K}^l \\ \|\underline{\sigma}\|_2 \leq 1}} \sup_{\substack{\underline{\tau} \in \mathbb{K}^l \\ \|\underline{\tau}\|_2 \leq 1}} \|\underline{S}\underline{\sigma} - \underline{T}\underline{\tau}\|_2^2 \\ &= \sup_{\substack{\underline{\sigma} \in \mathbb{K}^l \\ \|\underline{\sigma}\|_2 \leq 1}} \inf_{\substack{\underline{\tau} \in \mathbb{K}^l \\ \|\underline{\tau}\|_2 \leq 1}} \|\underline{\sigma}\|_2^2 + \|\underline{\tau}\|_2^2 - \underline{\sigma}^H \underline{S}^H \underline{T} \underline{\tau} - \underline{\tau}^H \underline{T}^H \underline{S} \underline{\sigma} \end{aligned}$$

Eine einfache Rechnung zeigt, daß d. Infimum für $\underline{\tau}^H = \underline{\sigma}^H \underline{S}^H \underline{T}$ angenommen wird. Also

$$\begin{aligned} d^2(S, T) &= \sup_{\substack{\underline{\sigma} \in \mathbb{K}^l \\ \|\underline{\sigma}\|_2 \leq 1}} \|\underline{\sigma}\|_2^2 + \underline{\sigma}^H \underline{S}^H \underline{T} \underline{T}^H \underline{S} \underline{\sigma} - 2 \underline{\sigma}^H \underline{S}^H \underline{T} \underline{T}^H \underline{S} \underline{\sigma} \\ &= \sup_{\substack{\underline{\sigma} \in \mathbb{K}^l \\ \|\underline{\sigma}\|_2 \leq 1}} \underline{\sigma}^H \left[\underline{I} - \underline{S}^H \underline{T} \underline{T}^H \underline{S} \right] \underline{\sigma} = \max_{\lambda \in \sigma(\underline{I} - \underline{A}\underline{A}^H)} \lambda \\ &\quad \uparrow \\ &\quad \underline{A} := \underline{S}^H \underline{T} \\ &\quad \text{und } \underline{I} - \underline{A}\underline{A}^H \text{ selbstadj.} \end{aligned}$$

völlig analog erhalten wir

$$d^2(T, S) = \max_{\lambda \in \sigma(\underline{I} - \underline{A}^H \underline{A})} \lambda$$

Die gewünschte Symmetrie von $d(\cdot, \cdot)$ folgt nun, weil

$$\sigma(\underline{A}^H \underline{A}) = \sigma(\underline{A} \underline{A}^H) \quad \text{[Wir zeigen nur } \sigma(\underline{A}^H \underline{A}) \subset \sigma(\underline{A} \underline{A}^H)\text{:}$$

Sei $\lambda \in \sigma(\underline{A}^H \underline{A})$. Dann $\exists x \neq 0$ mit $\underline{A}^H \underline{A} x = \lambda x$ ($\underline{A}^H \underline{A}$ selbstadj.!).

Setze $z := \underline{A} x$. Wir unterscheiden 2 Fälle: 1) $z \neq 0$. Dann ist

$$\underline{A} \underline{A}^H z = \underline{A} \underline{A}^H \underline{A} x = \underline{A} \lambda x = \lambda z, \text{ i.R. } \lambda \in \sigma(\underline{A} \underline{A}^H). \quad 2) z = 0. \text{ Dann}$$

ist $\underline{A} x = 0$, i.R. $0 = \underline{A}^H \underline{A} x = \lambda x$, i.R. $\lambda = 0$. Weil 0 auf $\sigma \underline{A}$ von \underline{A} ist, muß es auch $\sigma \underline{A}^H$ sein, und damit auch $\sigma \underline{A} \underline{A}^H$,
 i.R. $0 \in \sigma(\underline{A} \underline{A}^H)$]

ad (v):

$$d^2(S, T) = \sup_{s \in S} \inf_{t \in T} \frac{\|s - t\|_2^2}{\|s\|_2^2} = \sup_{s \in S} \frac{\|s - P_T s\|_2^2}{\|s\|_2^2}$$

$$= \sup_{s \in S} \frac{\|P_{T^\perp} s\|_2^2}{\|s\|_2^2} = \sup_{s \in S} \frac{\|P_{T^\perp} P_S s\|_2^2}{\|P_S s\|_2^2}$$

hiermit $\|P_{T^\perp} P_S\|_2^2 = \sup_{0 \neq x \in \mathbb{K}^n} \frac{\|P_{T^\perp} P_S x\|_2^2}{\|x\|_2^2} = \sup_{0 \neq x} \frac{\|P_{T^\perp} P_S x\|_2^2}{\|P_S x\|_2^2 + \|P_{S^\perp} x\|_2^2}$

und somit $\left\{ \begin{aligned} \bullet \|P_{T^\perp} P_S\|_2^2 &\leq \sup_{0 \neq x \in \mathbb{K}^n} \frac{\|P_{T^\perp} P_S x\|_2^2}{\|P_S x\|_2^2} = \sup_{x \in S} \frac{\|P_{T^\perp} P_S x\|_2^2}{\|P_S x\|_2^2} = d^2(S, T) \\ \bullet \|P_{T^\perp} P_S\|_2^2 &\geq \sup_{x \in S} \frac{\|P_{T^\perp} P_S x\|_2^2}{\|x\|_2^2} = \sup_{x \in S} \frac{\|P_{T^\perp} P_S x\|_2^2}{\|P_S x\|_2^2} = d^2(S, T) \end{aligned} \right.$

ad (iii): Wir denken uns Operatoren P_{T^\perp}, P_S als Matrizen und erinnern uns: $\|A\|_2 = \|A^H\|_2$ für alle Matrizen A . Dann folgt aus (v)

$$d(S, T) = \|P_{T^\perp} P_S\|_2 = \|(P_{T^\perp} P_S)^H\|_2 = \|P_S^H P_{T^\perp}^H\|_2 =$$

orth.-Projektion
und selbstadj.

$$= \|P_S P_{T^\perp}\|_2 \stackrel{(i)}{=} d(T^\perp, S^\perp) \stackrel{(v)}{=} d(S, T)$$

ad (iv): Weil $s \mapsto \|P_{T^\perp} s\|_2$ stetig ist und $\partial B_n(0) \cap S$ kompakt, reicht es wegen $d(S, T) = \sup_{s \in \partial B_n(0) \cap S} \|P_{T^\perp} s\|_2$ zu

zeigen, daß $\|P_{T^\perp} s\|_2 < 1 \quad \forall s \in \partial B_n(0) \cap S$.

Sei $s \in S$ mit $\|s\|_2 = 1$. D.S.: $\|P_{T^\perp} s\|_2^2 = \|s\|_2^2 - \|P_T s\|_2^2$. Aus d.

Voraussetzung $S \cap T^\perp = \{0\}$ folgt $\|P_T s\|_2 > 0$ denn sonst gälte $\|s\|_2^2 = \|P_{T^\perp} s\|_2^2$, was $s \in T^\perp$ impliziert; daraus wiederum $0 \neq s \in S \cap T^\perp = \{0\}$.



... ermöglichen nach der Aussage, was es ergibt, wie sich ... (29)
 stand von Unterräumen unter affinen Abb. verändert:

Lemma 1.32 Sei $V \in \mathbb{K}^{n \times n}$ regulär. D_S Sei Unterräume S, T von \mathbb{K}^n .
 $d(VS, VT) \in \text{cond}_2(V) d(S, T)$ [Hier ist V als lineare Abb. interpretiert und VS, VT die Bilder von S, T unter der Abb.]

Beweis: weil V regulär ist, gilt:

$$d(VS, VT) = \sup_{0 \neq s \in S} \inf_{t \in T} \frac{\|Vs - Vt\|_2}{\|Vs\|_2}$$

(Denn ist $\|V(s-t)\|_2 \leq \|V\|_2 \|s-t\|_2$ und $\|s\|_2 = \|V^{-1}Vs\|_2 \leq \|V^{-1}\|_2 \|Vs\|_2$, was
 woraus sich ergibt: $\frac{\|Vs - Vt\|_2}{\|Vs\|_2} \leq \frac{\|V\|_2 \|s-t\|_2}{\|s\|_2 / \|V^{-1}\|_2} = \|V\|_2 \|V^{-1}\|_2 \frac{\|s-t\|_2}{\|s\|_2}$ □

Die wesentlichen Mechanismen ^{bei} der Konvergenz der orthog. Iteration kann man bereits bei Diagonalisierungen studieren:

Lemma 1.33 Sei $D = \text{diag}(\lambda_1, \dots, \lambda_m)$, spez. $k \in \{1, \dots, m-1\}$. Sei \hat{p} ein Polynom derart, dass $\hat{p}(\lambda_i) \neq 0$ für $i=1, \dots, k$. Sei $D_S = \text{span}\{e_1, \dots, e_k\}$ der D -invariante Unterraum, der von e_1, \dots, e_k auf \mathbb{K}^m gespannt wird. Sei $S \subset \mathbb{K}^m$ ein k -dimensionaler Raum mit

(1.16) $S \cap S^\perp = \{0\}$. d. h.:

(1.17) $d(p(D)S, S) \leq C \frac{\max_{k+1 \leq i \leq m} |\hat{p}(\lambda_i)|}{\min_{1 \leq i \leq k} |\hat{p}(\lambda_i)|}$, wobei die Konstante C nur von S und k abhängt.

Beweis:

1. Schritt: wir zeigen, daß d. Ver. (1.16) zusammen mit $\hat{p}(\lambda_i) \neq 0 \quad \forall i=1, \dots, k$ impliziert: $\dim \hat{p}(D)S = \dim S' = k$

Bew: Annahme: $\dim \hat{p}(D)S < k$. Dann $\exists \neq 0 s \in S$ mit $\hat{p}(D)s = 0$.

Weil sowohl J als auch J^\perp D -invariante Räume sind, gilt

$$P_J \hat{p}(D) P_J s = P_J \hat{p}(D) (s - P_{J^\perp} s) = \underbrace{P_J \hat{p}(D) s}_{=0} - \underbrace{P_J \hat{p}(D) P_{J^\perp} s}_{\substack{\in J^\perp \text{ weil} \\ J^\perp \text{ invar.}}} = 0$$

Aus der Annahme $\hat{p}(\lambda_i) \neq 0, \quad i=1, \dots, k$ folgt damit $P_J s = 0$, d.h. $s \in S \cap J^\perp = \{0\}$ \downarrow

2. Schritt Weil $S \cap J^\perp = \{0\}$, ist $\beta := d(S, J^\perp) < 1$ [Satz 133, (iv)]

Bch.

$$(1.18) \quad \|P_{J^\perp} s\|_2 \leq \frac{\beta}{\sqrt{1-\beta^2}} \|P_J s\|_2 \quad \forall s \in S$$

Bew: $\|P_{J^\perp} s\|_2 = \inf_{t \in J} \|s - t\|_2 \leq \|s\|_2 d(S, J^\perp) = \beta \|s\|_2$

$$\|P_J s\|_2^2 = \|s\|_2^2 - \|P_{J^\perp} s\|_2^2 \geq \|s\|_2^2 - \beta^2 \|s\|_2^2 = (1-\beta^2) \|s\|_2^2$$

$$\text{also } \|P_{J^\perp} s\|_2^2 \leq \beta^2 \|s\|_2^2 \leq \frac{\beta^2}{1-\beta^2} \|P_J s\|_2^2$$

3. Schritt: Weil J und J^\perp D -invariante Räume sind, erkennt man

$$(1.19) \quad \hat{p}(D) = (P_J + P_{J^\perp}) \hat{p}(D) (P_J + P_{J^\perp}) = P_J \hat{p}(D) P_J + P_{J^\perp} \hat{p}(D) P_{J^\perp}$$

$\hat{p}(D)$ in Matrixnotation ist dies nichts anderes als:

$$\hat{p}(D) = \begin{pmatrix} \hat{p}(\lambda_1) & & & \\ & \ddots & & \\ & & \hat{p}(\lambda_k) & \\ & & & \ddots \end{pmatrix} + \begin{pmatrix} & & & 0 \\ & & & \vdots \\ & & & \hat{p}(\lambda_{k+1}) \\ & & & \vdots \\ & & & & \hat{p}(\lambda_m) \end{pmatrix}$$

weiter schätzen wir ab

$$(1.20) \begin{cases} \bullet \|P_J \hat{p}(D) P_J x\|_2^2 \geq \min_{1 \leq i \leq k} |\hat{p}(\lambda_i)|^2 \|P_J x\|_2^2 \quad \forall x \in \mathbb{K}^n \\ \bullet \|P_{J^\perp} \hat{p}(D) P_{J^\perp} x\|_2^2 \leq \max_{k+1 \leq i \leq m} |\hat{p}(\lambda_i)|^2 \|P_{J^\perp} x\|_2^2 \end{cases}$$

4. Schritt:

$$d^2(\hat{p}(D)S, \mathcal{J}) = \sup_{0 \neq s \in S} \inf_t \frac{\|\hat{p}(D)s - t\|_2^2}{\|\hat{p}(D)s\|_2^2} = \sup_{0 \neq s \in S} \frac{\|P_{\mathcal{J}^\perp} \hat{p}(D)s\|_2^2}{\|\hat{p}(D)s\|_2^2}$$

$$(1.19) \quad = \sup_{0 \neq s \in S} \frac{\|P_{\mathcal{J}^\perp} \hat{p}(D)s\|_2^2}{\|P_{\mathcal{J}} \hat{p}(D)s\|_2^2 + \|P_{\mathcal{J}^\perp} \hat{p}(D)s\|_2^2}$$

$$(1.20) \quad \leq \sup_{0 \neq s \in S} \frac{\max_{k+1 \leq i \leq m} |\hat{p}(z_i)|^2 \|P_{\mathcal{J}^\perp} s\|_2^2}{\min_{1 \leq i \leq k} |\hat{p}(z_i)|^2 \|P_{\mathcal{J}} s\|_2^2}$$

$$\leq \frac{\max_{k+1 \leq i \leq m} |\hat{p}(z_i)|^2}{\min_{1 \leq i \leq k} |\hat{p}(z_i)|^2} \sup_{0 \neq s \in S} \frac{\|P_{\mathcal{J}^\perp} s\|_2^2}{\|P_{\mathcal{J}} s\|_2^2} \leq \frac{\beta^2}{1-\beta^2} \text{ nach 2. Schritt}$$

□

Lemma 1.32 impliziert nun auf einfache Weise eine Konvergenzaussage für die orthog. Iteration Alg. 1.29:

Um die Konvergenzaussage kompakt formulieren zu können, definieren wir die Räume $(S^k)^{\infty}$ als die Räume, die durch l Spalten der Matrizen A^k ($l=0,1,\dots$ oder ∞) aufgespannt werden, d.h. $S^k = A^k S^0$. Dann können wir folgendes Konvergenzresultat formulieren:

Satz 1.33 Sei $A \in \mathbb{K}^{n \times n}$ diagonalisierbar; Sei $\{v_1, \dots, v_m\}$ Basis aus EV mit zugehörigen EW $\lambda_1, \dots, \lambda_m$. Es gelte

$$|\lambda_1| > |\lambda_2| > \dots > |\lambda_l| > |\lambda_{l+1}| > \dots > |\lambda_m|. \text{ Sei } S^0 \subset \mathbb{K}^n$$

ein k -dimensionaler Raum mit $S^0 \cap \text{span}\{v_{k+1}, \dots, v_m\} = \{0\}$.

Dann existiert $C > 0$ derart, daß

$$d(S^k, \text{span}\{v_1, \dots, v_k\}) \leq C \left| \frac{\lambda_{k+1}}{\lambda_k} \right|^l, \quad l=0,1,\dots$$

Beweis: Sei $\hat{p}_\ell(x) := x^\ell$. Dann gilt: $S^\ell = \hat{p}_\ell(A) S^0$.

1. Schritt: Beh: denn $S^\ell \subseteq \mathcal{L}$ denn $S^0 = \mathcal{L}$ für alle ℓ .

Annahme: denn $S^\ell \subseteq \mathcal{L}$. Dann existiert $0 \neq s \in S^0$ derart, daß $\hat{p}_\ell(A)s = 0$.
Wir schreiben $s = t + t'$, wobei $t \in \mathcal{J} := \text{span}\{v_1, \dots, v_\ell\}$, $t' \in \mathcal{J}' := \text{span}\{v_{\ell+1}, \dots, v_n\}$.

$$\text{d.h.: } 0 = \hat{p}_\ell(A)s = \underbrace{\hat{p}_\ell(A)t}_{\in \mathcal{J}} + \underbrace{\hat{p}_\ell(A)t'}_{\in \mathcal{J}'}$$

wobei $\mathcal{J}, \mathcal{J}'$ A -invariante Unterräume sind.

$$\text{Also: } \hat{p}_\ell(A)t = 0 \quad \wedge \quad \hat{p}_\ell(A)t' = 0$$

Also $\hat{p}_\ell(\lambda_i) \neq 0$ für $i = 1, \dots, \ell$ folgt $t = 0$. Also ist

$$s = t' \in \mathcal{J}', \text{ d.h. } 0 \neq s \in \mathcal{J}' \cap \text{span}\{v_1, \dots, v_\ell\} = \{0\} \quad \downarrow$$

2. Schritt: Wir betrachten nun die orthog. Iteration in der Basis $\{v_1, \dots, v_n\}$, d.h. wir def. $\tilde{S}^0 := V^{-1}S^0$, $\tilde{\mathcal{J}} := V^{-1}\mathcal{J}$ ($V = [v_1, \dots, v_n]$)

$$d(S^\ell, \mathcal{J}) = d(\hat{p}_\ell(A)S^0, \mathcal{J}) = \sup_{s \in S^0} \inf_{t \in \mathcal{J}} \frac{\|\hat{p}_\ell(A)s - t\|_2}{\|\hat{p}_\ell(A)s\|_2} =$$

$$= \sup_{s \in \tilde{S}^0} \inf_{t \in \tilde{\mathcal{J}}} \frac{\|\hat{p}_\ell(A)Vs - Vt\|_2}{\|\hat{p}_\ell(A)V s\|_2} = \sup_{s \in \tilde{S}^0} \inf_{t \in \tilde{\mathcal{J}}} \frac{\|V\hat{p}_\ell(D)s - Vt\|_2}{\|V\hat{p}_\ell(D)s\|_2}$$

$A = VDV^{-1}$, $D = \text{diag}(\lambda_1, \dots, \lambda_n)$

$$\leq \sup_{s \in \tilde{S}^0} \inf_{t \in \tilde{\mathcal{J}}} \frac{\|V\|_2 \|\hat{p}_\ell(D)s - t\|_2}{\|V\|_2 \|\hat{p}_\ell(D)s\|_2} = \text{cond}_2(V) d(\hat{p}_\ell(D)\tilde{S}^0, \tilde{\mathcal{J}})$$

$\|x\|_2 \leq \|V^{-1}Vx\|_2 \leq \|V^{-1}\|_2 \|Vx\|_2$

man ist $\tilde{\mathcal{J}} = \text{span}\{e_1, \dots, e_\ell\}$. Nach Lemma 1.32 also

$$d(\hat{p}_\ell(D)\tilde{S}^0, \tilde{\mathcal{J}}) \leq C \frac{\max_{1 \leq i \leq \ell} |\hat{p}_\ell(\lambda_i)|}{\min_{1 \leq i \leq \ell} |\hat{p}_\ell(\lambda_i)|} = C \left| \frac{\lambda_{\ell+1}}{\lambda_\ell} \right|^\ell$$

Satz 1.33 zeigt, daß die orthog. Iteration effektiv eine Folge von k -dim. Räumen erzeugt, die gegen einen k -dimensionalen invarianten Unterraum konvergiert. Um den Zusammenhang mit der Schurform (siehe Satz 1.5) und Eigenwertapproximationen klären, formulieren wir:

Satz 1.34 Sei $A \in \mathbb{K}^{n \times n}$ und $J \subset \mathbb{K}^n$ ein k -dimensionaler A -invarianter Unterraum. Sei $Q = (Q_1, Q_2)$ eine orthogonale Matrix mit $Q_1 \in \mathbb{K}^{n \times k}$, $Q_2 \in \mathbb{K}^{n \times (n-k)}$. Sei S der k -dim. Raum, der von d. Spalten von Q_1 aufgespannt wird. D.h. für d. Matrix

$$Q^H A Q = \begin{pmatrix} Q_1^H A Q_1 & Q_1^H A Q_2 \\ Q_2^H A Q_1 & Q_2^H A Q_2 \end{pmatrix} : \|Q_2^H A Q_1\|_2 \leq 2d(S, J) \|A\|_2$$

Beweis: Def. $T_1 := P_J Q_1$ [wende Proj. P_J spaltenweise an]

$$T_2 := P_{J^\perp} Q_2$$

d.h. $Q_2^H A Q_1 = (Q_2 - T_2)^H A Q_1 + T_2^H A (Q_1 - T_1) + T_2^H A T_1$
 ≤ 0 , weil J invariant unter A
 $= 0$, weil Spalten von T_2 in J^\perp

also: $\|Q_2^H A Q_1\|_2 \leq \|Q_2 - T_2\|_2 \|A\|_2 \|Q_1\|_2 + \|T_2\|_2 \|A\|_2 \|Q_1 - T_1\|_2$
 ≤ 1 , weil Q_1 orthog. Spalten

$$\leq \|A\|_2 \left[\|Q_2 - T_2\|_2 + \|T_2\|_2 \|Q_1 - T_1\|_2 \right]$$

$$\|T_2\|_2 = \sup_x \frac{\|T_2 x\|_2}{\|x\|_2} = \sup_x \frac{\|P_{J^\perp} Q_2 x\|_2}{\|x\|_2} \leq \sup_x \frac{\|Q_2 x\|_2}{\|x\|_2} \leq 1, \text{ weil } Q_2 \text{ orthog. Spalten hat.}$$

weiter schätzen wir ab:

$$d(S, J) = \sup_{x \in \mathbb{K}^k} \inf_{t \in J} \frac{\|Q_1 x - t\|_2}{\|Q_1 x\|_2} = \sup_{x \in \mathbb{K}^k} \frac{\|Q_1 x - P_J Q_1 x\|_2}{\|Q_1 x\|_2}$$

$$39 = \sup_{x \in \mathbb{K}^2} \frac{\|Ax\|_2}{\|x\|_2} = \sup_{x \in \mathbb{K}^2} \frac{\|Q_1^{-1}Tx\|_2}{\|x\|_2} = \|Q_1^{-1}T\|_2$$

Q_1 hat orthom. Spalten

Wählt in Spalten von Q_2 den Raum S^\perp aufspannen
 $\|Q = (Q_1, Q_2)$ ist orthogonal! $\| \cdot \|_2$ ergibt sich analog:

$$d(S^\perp, T^\perp) \geq \|Q_2 - T_2\|_2$$

Wir erhalten somit

$$\|Q_2^H A Q_2\| \leq \|A\|_2 \left[\|Q_1 - T_1\|_2 + \|Q_2 - T_2\|_2 \right] \leq \|A\|_2 \left[d(S, T) + d(S^\perp, T^\perp) \right]$$

Aus Satz 1.37, (iii) folgt $d(S^\perp, T^\perp) = d(S, T)$ Somit ist $\| \cdot \|_2$ Satz bewiesen □

Sätze 1.33, 1.34 klären somit die Eigenwertkonvergenz der orthogonalen Iteration:

Kor 1.35: Es mögen d. Voraussetzungen von Satz 1.33 gelten. Seien $\lambda_1^p, \dots, \lambda_b^p \in \mathbb{C}$ die EW der Matrix $Q_p^H A Q_p \in \mathbb{K}^{k \times k}$, wobei die $Q_p \in \mathbb{K}^{m \times k}$ in Alg 1.29 bestimmt werden. Dann gilt: $\exists C_{p,k} > 0$ derart, daß für

$$i=1, \dots, b: \min_{\lambda \in \sigma(A)} |\lambda_i^p - \lambda| \leq C_{p,k} \left| \frac{\lambda_{k+1}^p}{\lambda_k^p} \right|^p, \quad p=0, 1, \dots$$

Beweis: Sei $Q_p \in \mathbb{K}^{m \times k}$ wie in Alg 1.29. Ergänze Q_p zu einer Orthogonalmatrix $Q = (Q_p, \tilde{Q}_p) \in \mathbb{K}^{m \times m}$. Dann ist $Q^H A Q$ ähnlich zu $Q_p^H A Q_p$

$$Q^H A Q = \begin{pmatrix} Q_p^H A Q_p & Q_p^H A \tilde{Q}_p \\ \tilde{Q}_p^H A Q_p & \tilde{Q}_p^H A \tilde{Q}_p \end{pmatrix}$$

||

$\sigma(A) \subseteq \sigma(Q^H A Q)$. Betrachtet man $\tilde{A} := \begin{pmatrix} Q_p^H A Q_p & Q_p^H A \tilde{Q}_p \\ \cdot & \tilde{Q}_p^H A \tilde{Q}_p \end{pmatrix}$ (35)

so folgt für $\Delta A := A^H A - \tilde{A}$:

$$\|\Delta A\|_2 = \|Q_p^H A \tilde{Q}_p\|_2 \quad \text{Sind aus Satz 1.8:}$$

$$\forall \mu \in \sigma(A) \quad \min_{\lambda \in \sigma(A)} |\mu - \lambda| \leq C \cdot \|\Delta A\|_2 \leq C d(A^{\circ}, \text{spektrale})$$

Satz 1.35

$$\leq C \left| \frac{\lambda_{k+1}}{\lambda_k} \right|^l$$

↑
Satz 1.33

Bem: Man kann sich überlegen, daß die EW $\lambda^{\pm k} = \lambda, \dots, \lambda$ Approximationen an die EW $\lambda_1, \dots, \lambda_k$ darstellen (Übung!)
 daß die EW $\lambda_1, \dots, \lambda_k$ darstellen (Übung!)

Bem 1.36

Kor 1.35 kann auch als ein Schritt in Richtung Triangulierung auf Schurform verstanden werden: Vernachlässigt man den Block $Q_p^H A \tilde{Q}_p$ (der ja klein ist!), so erhält man, daß

$$Q^H A Q = \begin{pmatrix} Q_p^H A Q_p & Q_p^H A \tilde{Q}_p \\ \approx 0 & \tilde{Q}_p^H A \tilde{Q}_p \end{pmatrix} \quad \text{Block-Dreiecksform hat.}$$

Jeder der Blöcke $Q_p^H A Q_p, \tilde{Q}_p^H A \tilde{Q}_p$ kann nun separat weiterbehandelt werden, um schließlich Schurform zu erhalten. Man beachte auch, daß nach Übungsaufz. 2

$$\sigma \begin{pmatrix} Q_p^H A Q_p & \times \\ 0 & \tilde{Q}_p^H A \tilde{Q}_p \end{pmatrix} = \sigma(Q_p^H A Q_p) \cup \sigma(\tilde{Q}_p^H A \tilde{Q}_p).$$

Bem 1.37 Wie im Satz 1.33 stellen wir Forderung $S^{\circ} \cap \text{span}\{v_1, \dots, v_n\} = \emptyset$. Dies ist das Analog zur Forderung $\alpha_1 \neq 0$ im Satz 1.28. Wählt man S° den k -dim. Raum S° zufällig, so ist diese Forderung fast sicher erfüllt.

1.6 QR-Algorithmus

Der QR-Algorithmus ist der schnell rechenfähigste Alg., um alle EW einer Matrix zu erhalten. Der Alg. erzeugt eine Folge $(A^p)_{p=0}^{\infty}$ von Matrizen, die zu $A_0 = A$ orthogonal ähnlich sind und gegen (Block-) Dreiecksform konvergieren.

1.6.1 einfacher QR-Algorithmus

Wir werden sehen, daß durch Wahl von $X_0 = I_m$ in der orthogonalen Iteration effektiv nur orthogonale Iterationen mit Startmatrizen $X_0 = E_k = [e_1, \dots, e_k]$ für $k = 1, \dots, m$ durchgeführt werden.

Starten wir Alg. 1.23 (orthog. Iteration) mit $X_0 = E_m$, so ergibt sich

$$\begin{aligned}
 A &= A E_m = \hat{Q}_1 R_1 \\
 A \hat{Q}_1 &= \hat{Q}_2 R_2 \\
 A \hat{Q}_2 &= \hat{Q}_3 R_3
 \end{aligned}$$

und damit:

$$(1.21) \quad A^p = \underbrace{A \dots A}_{p \text{ mal}} = \underbrace{A \dots A}_{p-1 \text{ mal}} \hat{Q}_1 R_1 = \dots = \hat{Q}_p \underbrace{R_p R_{p-1} \dots R_1}_{=: \tilde{R}_p}$$

ist obere Dreiecksmatrix

Wegen \tilde{R}_p obere Dreiecksgestalt hat, ist für jedes $k \in \{1, \dots, m\}$:

$$A^p E_k = \hat{Q}_p \left(\tilde{R}_p E_k \right) = \left(\hat{Q}_p E_k \right) \tilde{R}_p$$

$$\underbrace{\begin{pmatrix} \times & & & \\ & \times & & \\ & & \times & \\ & & & \times \end{pmatrix}}_k = \begin{pmatrix} \tilde{R}_p \\ 0 \end{pmatrix}$$

Mit anderen Worten: die ersten k Spalten von \hat{Q}_p spannen das Bild von $\text{span}\{e_1, \dots, e_k\}$ unter A^p auf (wir nehmen vereinfachend an, daß A regulär ist, wobei \tilde{R}_p regulär ist).

definieren wir

(1.22) $A_k := \hat{Q}_k^H A \hat{Q}_k$ so können wir für jedes beliebig $k \in \{1, \dots, n\}$ die orthog. Matrix \hat{Q}_k als $\hat{Q}_k = (\hat{Q}_{k,1}, \hat{Q}_{k,2})$

partitionieren. Die Matrix A_k nimmt dann folgende Form an:

$$(1.23) A_k = \left(\begin{array}{c|c} \hat{Q}_{k,1}^H A \hat{Q}_{k,1} & \hat{Q}_{k,1}^H A \hat{Q}_{k,2} \\ \hline \hat{Q}_{k,2}^H A \hat{Q}_{k,1} & \hat{Q}_{k,2}^H A \hat{Q}_{k,2} \end{array} \right)$$

Die Sätze 1.34, 1.35 zeigen uns nun, daß wir darauf hoffen können, daß für jedes $k \in \{1, \dots, n\}$ der Block $\hat{Q}_{k,1}^H A \hat{Q}_{k,2}$ gegen Null geht:

Satz 1.38

Sei $A \in K^{n \times n}$ diagonalisierbar mit Eigenbasis $\{v_1, \dots, v_n\}$ und zugehörigen $\lambda \in W$ $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$. Es gelte für alle $k \in \{1, \dots, n\}$

(1.24) $\text{span}\{e_1, \dots, e_k\} \cap \text{span}\{v_{k+1}, \dots, v_n\} = \{0\}$

Seien die Orthogonalen Matrizen $\hat{Q}_l, l=0, 1, \dots$ diejenigen, die bei der orthogonalen Iteration (Abg. 1.29) mit Startmatrix $X_0 = I_n$ entstehen. Dann gilt für die Matrizen $A_l := \hat{Q}_l^H A \hat{Q}_l$

Die A_l konvergieren gegen oben Dreiecksform. Insb. gilt für jedes $k \in \{1, \dots, n\}$: $\|A_l([k+1:n], [1:k])\|_2 \leq c_{k,l} \left| \frac{\lambda_{k+1}}{\lambda_k} \right|^l$

Beweis: Nach obigen sind die ersten k Spalten von \hat{Q}_l diejenigen, die bei orthogonalen Iteration mit $X_0 = E_k$ entstehen. Aus

den Sätzen 1.34, 1.35 folgt dann (siehe (1.21)):

$$\| \hat{Q}_{l,2}^H A \hat{Q}_{l,1} \|_2 \leq c \left| \frac{\lambda_{k+1}}{\lambda_k} \right|^l, \quad l=0, \dots$$

Wir besetzen: $\hat{Q}_{l,2}^H A \hat{Q}_{l,1} = A_l([k+1:n], [1:k])$ □

G.D.P

Bem 1.40 genau wie in Satz 1.35 erhält man auch Aus-
 sagen über die Konvergenz d. Diagonaleinträge von A_l
 gegen die EW von A . 13

Wir wenden uns nun der Frage zu, wie die Matrizen A_l erzeugt
 werden können. Dies leistet die sog. "QR-Iteration".

Alg 1.41 (einfacher QR-Algorithmus)

% input: $A_0 \in \mathbb{K}^{n \times n}$

$l := 0$

repeat { % bestimme QR-Zerlegung von A_l

- $A_l := Q_{l+1} R_{l+1}$

- $A_{l+1} := R_{l+1} Q_{l+1}$

- $l := l + 1$

} until "genau genug"

Beweis wir zeigen, daß Alg. 1.41 tatsächlich die Matrizen A_l aus
 (1.22) erzeugt, schreiben es wir ^{hier} etwas anders auf:

○

Alg 1.42 (QR ohne Shift)

% input: $A_0 \in \mathbb{K}^{n \times n}$, def: $p_l(x) = x \quad \forall l \in \mathbb{N}_0$

$l := 0$

repeat {

- $p_{l+1}(A_l) := Q_{l+1} R_{l+1}$

- $A_{l+1} := Q_{l+1}^H A_l Q_{l+1}$

- $l := l + 1$

} until "genau genug"

% QR-Zerlegung von $p_l(A_l)$

Bew: $p_l(x) = x \Rightarrow$ der Alg. ist:

$A_l := Q_{l+1} R_{l+1}$

$A_{l+1} := Q_{l+1}^H A_l Q_{l+1} = Q_{l+1}^H Q_{l+1} R_{l+1} Q_{l+1}$
 $= R_{l+1} Q_{l+1}$

Wir $p(x) = x$, stimmen beide Alg. überein. Bevor wir zeigen, daß die Matrizen A_k aus Alg. 1.42 mit den aus (1.25) übereinstimmend definieren wir

$$(1.25) \quad \left\{ \begin{array}{l} \hat{p}_k(x) := p_k(x) p_{k-1}(x) \cdots p_1(x) \\ \hat{Q}_k := Q_1 \cdots Q_k \\ \hat{R}_k := R_k R_{k-1} \cdots R_1 \end{array} \right. \quad \begin{array}{l} \prod Q_i \text{ aus Alg. 1.42} \\ \prod R_i \text{ aus Alg. 1.42} \end{array}$$

und beweisen

$$(1.26) \quad \hat{p}_k(A_0) = \hat{Q}_k \hat{R}_k$$

Beweis von (1.26): Anstatt eines formalen Induktionsbeweises beobachte:

$$(1.27) \quad A_{k+1} = Q_{k+1}^H A_k Q_{k+1} = \begin{matrix} Q_{k+1}^H & & & \\ & Q_k^H & & \\ & & \ddots & \\ & & & Q_1^H \end{matrix} A_{k+1} Q_{k+1} = \cdots = (Q_1 \cdots Q_{k+1})^H A_0 (Q_1 \cdots Q_{k+1}) = \hat{Q}_{k+1}^H A_0 \hat{Q}_{k+1}$$

• $p_{k+1}(A_k) = Q_{k+1} R_{k+1}$ impliziert

$$\begin{aligned} Q_{k+1} R_{k+1} &= p_{k+1}(A_k) = p_{k+1} \left((Q_1 \cdots Q_k)^H A_0 (Q_1 \cdots Q_k) \right) \\ &= (Q_1 \cdots Q_k)^H p_{k+1}(A_0) (Q_1 \cdots Q_k) \\ &\quad \uparrow \\ &\quad p_{k+1} \text{ Polynom} \end{aligned} \quad (1)$$

$$\Rightarrow p_{k+1}(A_0) = Q_1 \cdots Q_{k+1} R_{k+1} Q_k^H \cdots Q_1^H$$

$$\begin{aligned} \Rightarrow p_{k+1}(A_0) p_k(A_0) \cdots p_1(A_0) &= \\ &= \left(Q_1 \cdots Q_{k+1} R_{k+1} Q_k^H \cdots Q_1^H \right) \left(Q_1 \cdots Q_k R_k Q_{k-1}^H \cdots Q_1^H \right) \cdots \\ &= \left(Q_1 \cdots Q_{k-1} R_{k-1} Q_{k-2}^H \cdots Q_1^H \right) \cdots \left(Q_1 R_1 \right) \\ &= \hat{Q}_{k+1} R_{k+1} \hat{R}_k \cdots R_1 = \hat{Q}_{k+1} \hat{R}_{k+1} \end{aligned} \quad (3)$$

Bem 1.43 Die spezielle Form $p(x) = x$ wurde in (1.26) nicht ausgenutzt — (1.26) gilt für beliebige Polynome, wenn ~~die~~ nur die Matrizen A, Q, R mit Alg 1.42 bestimmt werden — diese Best. ist relevant für den QR-Alg. mit Schritt (siehe unten). (41)

Lemma 1.44 Sei $A \in \mathbb{K}^{n \times n}$ regulär. Dann stimmen die Matrizen Q und R , die in Alg 1.42 erzeugt werden, mit denen überein, die bei Orthogonalen Iteration in (1.21), (1.22) erzeugt werden.

Beweis: Weil A regulär ist, ist auch jedes A^k regulär. Wegen der Eindeutigkeit der QR-Zerlegung einer regulären Matrix, folgt aus der QR -Zerlegung von A^k , die in (1.21), (1.22) angegeben werden, die Gleichheit der Matrizen Q , aus (1.22), (1.21) folgt dann auch die Gleichheit der Matrizen R . (1.27)

1.6.2 Implementierungspaspalte

Der QR-Alg. 1.44 ist teuer: jeder Schritt involviert ein QR-Faktorisierung (Kosten: $O(n^3)$); selbst wenn pro Schritt ein "EW"-Faktor "abgeworfen" wird, führen diese n Schritte dann auf Kosten $O(n^4)$. Wir zeigen nun, daß, wenn A zuerst auf (obere) Hessenbergform gebracht wird, die Kosten pro Schritt nur $O(n^2)$ sind. Kern dieser Kostenreduktion ist, daß die Matrizen A_k , die bei der QR-Iteration entstehen alle (obere) Hessenbergform haben, wenn nur A_0 (obere) Hessenbergform hat:

Lemma 1.45 Sei $A_0 \in \mathbb{K}^{n \times n}$ (obere) Hessenbergmatrix. Sei $p_{k+1}(x) = x - \sigma_{k+1}$

$Q_{k+1} \in \mathbb{K}^n$ o.g. ...

von $p_{k+1}(A_k)$

(i) eine QR-Faktorisierung $Q_{k+1} R_{k+1} = p_{k+1}(A_k)$ läßt sich mit $O(n^2)$ Aufwand bestimmen

(ii) $A_{k+1} := R_{k+1} Q_{k+1} + \sigma_{k+1} \text{Id} = Q_{k+1}^H A_k Q_{k+1}$ läßt sich mit Aufwand $O(n^2)$ bestimmen und A_{k+1} hat (obere) Hessenbergform.

(iii) falls A_k Hermitesch, (also A_k triagonal!), dann ist A_{k+1} ebenfalls

Beweis: [wir erlauben $a_{pp} = -\sigma_2$ um später QR-Iteration mit Schritt behandeln zu können]

① wir konstruieren Q_p mit Eigenrotationen. Im Fall $K = \mathbb{C}$ haben Eigenrotationen die in (1.3) angegebene Form, wobei nun die Matrix G wie folgt aussieht:

$$G = \begin{pmatrix} c & -s e^{it} \\ s e^{it} & c \end{pmatrix} \quad \text{mit } t \in \mathbb{R}, \text{ analog zu Lemma 1.10}$$

und zu Übungsaufg. 3 gilt:

- $G(i,j, \theta, t)^H A$ ändert nur die Zeilen i, j von A ; die Zeilen i, j entstehen als Linearkombination von $A(i, :), A(j, :)$
- $A G(i,j, \theta, t)$ ändert nur die Spalten i, j von A ; Spalten i, j entstehen als Linearkombination von $A(:, i), A(:, j)$
- für $i \neq j$ und $A_{ij} \neq 0$ kann eine Eigenrotation $G(i,j, \theta, t)$ gefunden werden, für die gilt: $(G(i,j, \theta, t)^H A)_{ij} = 0$:

[Sei $\hat{A} = \begin{pmatrix} A_{ii} & A_{ij} \\ A_{ji} & A_{jj} \end{pmatrix}$. Dann ist $(G^H A) \begin{bmatrix} i \\ j \end{bmatrix} = \begin{pmatrix} c & -s e^{-it} \\ s e^{it} & c \end{pmatrix} \begin{pmatrix} A_{ii} & A_{ij} \\ A_{ji} & A_{jj} \end{pmatrix} = \begin{pmatrix} x & A_{ij}c - s e^{-it} A_{jj} \\ s A_{ii} e^{it} + A_{ji}c & x \end{pmatrix}$

Fordert man $(G^H A)_{ij} = 0$, so heißt dies $A_{ij}c - s e^{-it} A_{jj} = 0$

Aus $A_{ij} \neq 0$ und den Darstellungen $A_{ij} = |A_{ij}| e^{i\varphi}$, $A_{jj} = |A_{jj}| e^{i\psi}$ gilt

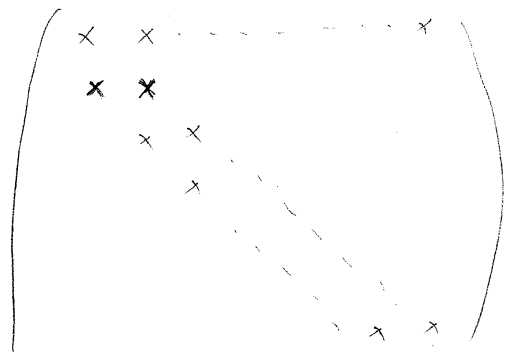
$$|A_{ij}| e^{i\varphi} c - s e^{-it} |A_{jj}| e^{i\psi} = 0, \text{ d.h. } t = \varphi - \psi \text{ und } s |A_{jj}| = c |A_{ij}|$$

$$\Rightarrow \cos \theta = \frac{|A_{jj}|}{|A_{ij}|}]$$

Plant R_p (obere) Hessenbergform, so hat auch $\text{pe}_1(A_p) = R_p - \sigma_{2+1}$ obere Hessenbergform. Wir wenden nun schrittweise Eigenrotationen an, um $\text{pe}_1(A_p)$ auf obere Dreiecksform zu bringen. Hierfür verwenden wir $G(i,j, \theta, t)$ bei folgender Folge von p :

Spaltenpaaren i, j :

geeign.



- (1,2) perm. (1,2) zu annullieren
- (2,3) " (3,3) " "
- (3,4) " (3,4) " "
- ...
- (m-1,m) " (m,m) " " R

Auf diese Weise erhält man obere Dreiecksgestalt. Anschließend werden die Givensrotationen $G(1,2), G(2,3), \dots, G(m-1,m)$ von rechts an R multipliziert. Multiplikation mit



- $G(1,2)$ komb. Spalten 1 & 2
 \rightarrow erzeugt Eintrag bei (1,2)
- $G(2,3)$ kombiniert Spalten 2,3
" " 3,4
- $G(3,4)$ " " " "

was Hessenbergform ergibt. Wichtig Anwendung dieser Givensrotation $O(m)$ Aufwand hat, sind die Spalten von A_{k+1} . Eine alternative Möglichkeit, die obere Hessenbergform zu erhalten ergibt sich wie folgt: Falls $P_{k+1}(A_k)$ regulär ist, dann ist

$$\begin{aligned}
 P_{k+1}(A_k) &= Q_{k+1} R_{k+1} \text{ und} \\
 A_{k+1} &= Q_{k+1}^H A_k Q_{k+1} = R_{k+1} \underbrace{P_{k+1}(A_k)^{-1}}_{A_k^{-\sigma_{k+1}}} A_k \underbrace{P_{k+1}(A_k)}_{A_k^{-\sigma_{k+1}}} R_{k+1}^{-1} \\
 &= R_{k+1} (A_k^{-\sigma_{k+1}})^{-1} (A_k^{-\sigma_{k+1}} + \sigma_{k+1}) (A_k^{-\sigma_{k+1}}) R_{k+1}^{-1} = \\
 &= R_{k+1} (A_k^{-\sigma_{k+1}} + \sigma_{k+1}) R_{k+1}^{-1} = P_{k+1} A_k R_{k+1}^{-1}
 \end{aligned}$$

Man rechnet nun nach, daß dieses Produkt einer oberen Dreiecksmatrix, der Hessenbergmatrix A_k sowie einer weiteren oberen Dreiecksmatrix eine obere Dreiecksmatrix ergibt. □

ad (iii): $P_{k+1}(A_k)$ hat Triidiagonalgestalt. Die Anwendung der Givensrotation Q_{k+1} , um auf Dreiecksgestalt zu kommen liefert eine Matrix

Es bleibt, die ~~Matrix~~ ~~Jung~~ ~~Normal~~ ~~form~~ ~~matrix~~ ~~A~~ in einem Vorabschritt auf obere Hessenbergform zu bringen. Dies kostet $O(n^3)$. Man könnte die mit Givensrotationen durchführen in folgender Reihenfolge:

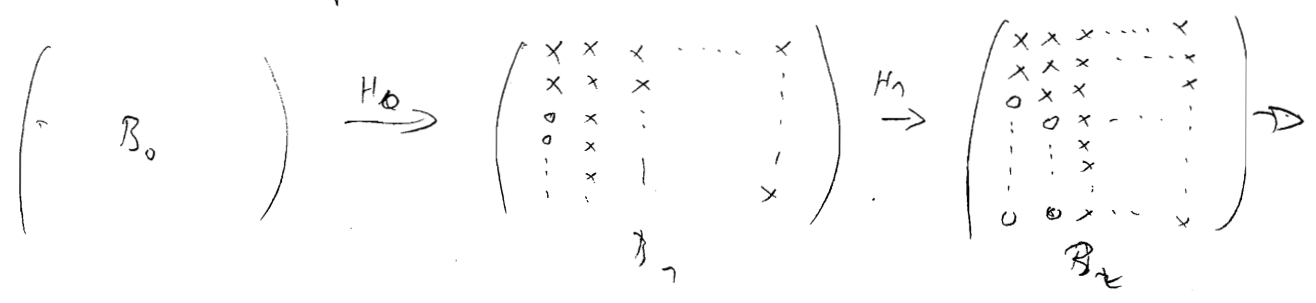
1. Spalte: $(2,3), (3,4), (4,5), \dots, (n-2, n)$ [um $A_{31}, A_{41}, A_{51}, \dots, A_{n1}$ zu annihilieren - beachte $G^H(i, i+1)$ operiert auf Zeile $i, i+1$ (um zu annihilieren) und $G(i, i+1)$ auf Spalte $i, i+1$]

2. Spalte: $(3,4), (4,5), \dots, (n-2, n)$ [um $A_{42}, A_{52}, \dots, A_{n,2}$ zu annihilieren]

3. Spalte: $(4,5), (5,6), \dots, (n-2, n)$

⋮
(n-2). Spalte: (n-1, n)

Die Reduktion auf Hessenbergform kann auch mit Householdermatrizen gemacht werden. Der Vorteil ist, daß dies nur ca. $1/2$ der Rechenoperationen benötigt (aber natürlich immer noch $O(n^3)$). Das Vorgehen ist wie folgt, um eine Matrix B_0 auf Hessenbergform zu bringen:



Jeder Schritt ist von der Form $B_{k+1} = H_k^H B_k H_k$, wobei H_k folgende Form hat:

$$H_k = \left(\begin{array}{c|c} \text{Id}_{k+1} & 0 \\ \hline 0 & \tilde{H}_k \end{array} \right) \text{ und } \tilde{H}_k \text{ ist eine "klassische" Householdertransformation, die einen (unter zu wählenden) Vektor } \dots \text{ annihiliert.}$$

Sei B_p von der Form

$$B_p = \left(\begin{array}{c|c} R & B' \\ \hline 0 & \tilde{B} \end{array} \right)$$

$\underbrace{\hspace{2cm}}_{r \text{ Spalten}} \quad \uparrow \quad \underbrace{\hspace{1cm}}_{1 \text{ Spalte}}$

, $R \in K^{(p+1) \times (p+1)}$ bereits
 oben Hessenbergform hat

z. B.:

$$H_p^H B_p H_p = \left(\begin{array}{c|c} \text{Id} & 0 \\ \hline 0 & \tilde{H}_p \end{array} \right) \left(\begin{array}{c|c} R & B' \\ \hline 0 & \tilde{B} \end{array} \right) \left(\begin{array}{c|c} \text{Id} & 0 \\ \hline 0 & \tilde{H}_p \end{array} \right)$$

H_p selbstadj.

$$= \left(\begin{array}{c|c} \text{Id} & 0 \\ \hline 0 & \tilde{H}_p \end{array} \right) \left(\begin{array}{c|c} R & B' \tilde{H}_p \\ \hline 0 & \tilde{B} \tilde{H}_p \end{array} \right) = \left(\begin{array}{c|c} R & B' \tilde{H}_p \\ \hline 0 & \tilde{B} \tilde{H}_p \end{array} \right)$$

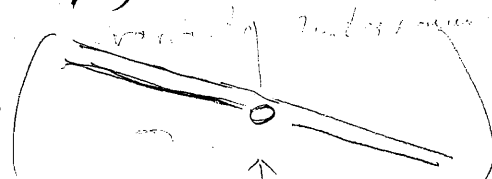
man können wir \tilde{H}_p so wählen, daß $\tilde{H}_p b$ auf dem Vielfachen
 des ~~Einheits~~ des $e_1 \in K^{n-1}$ abgebildet wird.

○

1.6.2 Deflation und Shift-Strategien

Aus Kostengründen liest man den QR-Algorithmus mit Hessenbergmatrizen durch. Zur Effizienzsteigerung verwendet man 2 weitere ~~von~~ Techniken, um die Konvergenz zu beschleunigen und 1. Kosten zu senken:

1) Deflation: Falls $A_{ii}(i+1, i) \neq 0$ für $i < n-1$, so σ "zerfällt" A_{ii} :



$\sigma(A_{ii}) = \sigma(A_{ii}([1:i], [1:i])) \cup \sigma(A_{ii}([i+1:n], [i+1:n]))$. Man wird also dem QR-Alg. auf jede dieser beiden Teilmatrizen anwenden. In der Praxis tritt natürlich

2) Shift: $A_{ii}(i+1, i) = 0$ nicht auf — i ein oft verwendetes Ersatzkriterium ist: $|A_{ii}(i+1, i)| \leq c \cdot \epsilon [|A_{ii}(i, i)| + |A_{ii}(i+1, i+1)|]$ wobei c moderat und ϵ = Maschinengenauigkeit

2) Shift: Wenn Deflation nicht "von alleine" passiert, verwendet man Shift-Strategien, um die Konvergenz zu beschleunigen. Dies bietet dann die Möglichkeit der Deflation.

Der QR-Alg. mit Shift ist

(47)

Alg. 1.46 (QR-mit "single shift")

% input: $A_0 \in K^{n \times n}$; def. $P_\ell(x) := x - \sigma_\ell$ mit zu wählende $\sigma_\ell \in K$

$\ell := 0$

repeat {

- wähle σ_ℓ

- $P_{\ell+1}(A_\ell) := Q_{\ell+1} R_{\ell+1}$

- $A_{\ell+1} := Q_{\ell+1}^H A_\ell Q_{\ell+1}$

% oder, was das gleiche ist:

$$A_{\ell+1} = R_{\ell+1} Q_{\ell+1} + \sigma_{\ell+1} \text{Id}$$

○ - $\ell := \ell + 1$

} until genau genug

Auf die Form

Wie die Shiftparameter σ_ℓ zu wählen sind, gibt das

folgende Konvergenzresultat einen Hinweis:

Satz 1.47 Sei $A_0 \in K^{n \times n}$ diagonalisierbar mit EV v_1, \dots, v_m

und zugehörigen EW $\lambda_1, \dots, \lambda_m$. Def. hier jedes ℓ das Polynom

○ $P_\ell(x) := P_\ell(x) P_{\ell+1}(x) \dots P_s(x)$. Es gelte für ein $k \in \{1, \dots, m-1\}$

• $P_\ell(\lambda_j) \neq 0$ für $j = 1, \dots, k$

• $\text{span}\{\tau_1, \dots, \tau_k\} \cap \text{span}\{v_{k+1}, \dots, v_m\} = \{0\}$.

D.h.: der Matrixblock $A_\ell([\ell+1:n], [1:k])$, der in Alg. 1.46 erzeugt wird, erfüllt

$$(1.28) \quad \|A_\ell([\ell+1:n], [1:k])\|_2 \leq C_{A,\ell} \frac{\max_{k+1 \leq i \leq m} |P_\ell(\lambda_i)|}{\min_{1 \leq i \leq k} |P_\ell(\lambda_i)|} =: \tau_\ell,$$

wobei $C_{A,\ell} > 0$ nur von A und k abhängt.

Beweis: Der Beweis verläuft analog zum Fall ohne Shift: Die Matrizen Q_1, \dots, Q_ℓ , die in Alg 1.46 erzeugt werden def. $Q_\ell := Q_1 \dots Q_\ell$ (cf. (1.25)). Weil Alg. 1.46 & Alg. 1.42 eigentlich übereinstimmen, folgen wir (cf. Bem. 1.43), daß (1.26), (1.27) gelten, d.h. $A_\ell = Q_\ell^H A_0 Q_\ell$ und $\hat{P}_\ell(A_0) = Q_\ell \hat{P}_\ell(A_0)$. Als nächstes überzeugt man sich davon, daß $S^\ell = \hat{P}_\ell(A_0)$ (spann $\{e_1, \dots, e_k\}$) gerade der Spann der ersten k Spalten von $-Q_\ell$ ist. Hierzu reicht es, zu sehen, daß dem $S^\ell = \ell$ \hat{P}_ℓ hat obere Dreiecksform! Sei $V D V^{-1} = A$ (Diagonalisierung), $V = (v_1, \dots, v_n)$, $V^{-1} = (v_1', \dots, v_n')$. Dann ist $\hat{P}_\ell(A) E_k = V \hat{P}_\ell(D) V^{-1} E_k = \begin{pmatrix} \hat{P}_\ell(\lambda_1) & & \\ & \dots & \\ & & \hat{P}_\ell(\lambda_k) \end{pmatrix} (v_1', \dots, v_k')$
 $= (\hat{P}_\ell(\lambda_1) v_1', \hat{P}_\ell(\lambda_2) v_2', \dots, \hat{P}_\ell(\lambda_k) v_k')$
 folgt, daß diese Matrix vollen Rang hat. Wörtlich wie im Beweis von Satz 1.34 ergibt sich dann: $d(S^\ell, J) \leq \frac{\max_{1 \leq i \leq k} |\hat{P}_\ell(\lambda_i)|}{\min_{1 \leq i \leq k} |\hat{P}_\ell(\lambda_i)|}$, wobei $J = \text{span}\{v_1, \dots, v_k\}$. Aus Satz 1.35 folgt

R.DS dann die Behauptung laut (1.28) ist die Konvergenz des QR-Verfahrens mit Shift
 das desto besser, je kleiner τ_ℓ ist. Das können wir dadurch erreichen, daß ein EW von A (fast) Nullstelle von \hat{P}_ℓ ist. Dazu benötigt man (gute) Schätzungen eines EW. Wir verwenden den "Rayleighquotient-Shift": $\sigma_{\ell+1} = A_\ell(m, m)$. Dies motiviert sich wie folgt:
 Falls das Verfahren konvergiert, so konvergieren die Diagonalelemente von A_ℓ gegen EW von A_0 . Insbesondere ist $A_\ell(m, m)$ eine Approx. an einen EW von A_0 .
 In der Praxis wird man QR-Iteration mit Shift mit Deflation koppeln. Dies ergibt:

Alg 1.48 (QR- mit Rayleighquotient Shift und Deflation)
 Funktion QR-Shift(A_0)
 % input: $A_0 \in K^{n \times n}$ auf Hessenbergform
 % output: EW von A_0
 $k := 0$
 while $\{ |A_\ell(m, m)| \text{ zu groß} \}$ % Deflation noch nicht möglich
 - $\sigma_{\ell+1} = A_\ell(m, m)$; $P_{\ell+1}(x) := x - \sigma_{\ell+1}$
 - $P_{\ell+1}(A_\ell) =: Q_{\ell+1} R_{\ell+1}$
 - $A_{\ell+1} := Q_{\ell+1}^H A_\ell Q_{\ell+1} + \sigma_{\ell+1} I$
 }
 return ($A_\ell(m, m)$ U QR-Shift($A_\ell([1:n-1], [1:n-1])$))

1.6.3 abschließende Bemerkungen

- Shiftstrategien funktionieren in d. Praxis recht gut: wie bei der Rayleigh-quotienteniteration erreicht man lokal quadratisch (bei sym. Problemen sogar oft kubisch) Konvergenz gegen einen EW \rightarrow nach wenigen Schritten kann man Deflation machen
- In d. Praxis verwendet man Mehrfachshifts, d.h. die pp in $\# 1.4$ sind Polynome höherer Ordnung, z.B. quadratisch quadratisch (mit Nullstellen als d. EW ~~von~~ des 2×2 Blocks $A_2 \in \mathbb{C}^{(n-1) \times (n-1)}$). Dies beschleunigt zum einen d. Konvergenz, zum anderen ermöglicht es bei reellen Matrizen mit ^{reiner} reeller Ähnlichkeit, daß die (A_2) gegen ~~den~~ reelle Schurform konvergiert

- einfache Shiftstrategien schlagen manchmal fehl: z.B. für

$$A = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \text{ konvergiert d. QR-} A_3 \text{ und d. QR-} A_3$$

mit Rayleighquotientenshift nicht \parallel Das ist konsistent mit dem R-Satz 1.34: Die EW von A sind $1, \frac{1}{2}(-1 + \sqrt{5}i), \frac{1}{2}(-1 - \sqrt{5}i)$ und haben alle Betrag 1 \parallel . Beobachtet man, daß QR-Iter. mit festen Shift \rightarrow einer QR-Iteration (ohne Shift) für $A-A$ entspricht, dann stellt man, daß das Konvergenzverhalten von A mit festem Shift λ durch $|\lambda_1 - \lambda|, |\lambda_2 - \lambda|, |\lambda_3 - \lambda|$ beschrieben wird. Falls diese 3 Zahlen getrennt sind, dann erhalten

○ wir Konvergenz.

- effiziente Implementierungen sparen noch mehr bei der QR-Faktorisierung durch Verwendung d. "impliziten Q-Theorems"
- leistungsfähige Implementierungen brauchen im Schritt 2-3 Iteration pro EW \rightarrow Gesamtkosten $W = \underbrace{O(n^3)}_{\text{Hessenberg-Form von } A_0} + n \underbrace{O(m^2)}_{\text{Kosten pro Schritt}}$

- Falls Schurform $Q^H A Q = R$ vorliegt, sind d. EV von A einfach zu bestimmen: x ist EV von A (\Leftrightarrow) $y := Qx$ ist EV von R . Sei $\lambda = R_{ii}$ einfacher EW von R . Dann können wir y wie folgt bestimmen: $Ry = \lambda y \Leftrightarrow (R - \lambda) y = 0$

$\begin{pmatrix} R_{ii} - \lambda & & x \\ 0 & 0 & \tilde{x} \\ 0 & 0 & R - \lambda \end{pmatrix} y = 0$. Wobei λ einfacher EW ist, sind $R - \lambda, \tilde{R} - \lambda$ regulär. $\tilde{R} - \lambda$ hat obere Dreiecksform!

1.6.4 SVD

$$A \in \mathbb{K}^{m \times m}, \quad m \times m \text{ -iges: } SVD: A = U \Sigma V^T \quad \left[\begin{array}{l} U \in \mathbb{K}^{m \times m}, \\ \Sigma \in \mathbb{R}^{m \times m} \\ V \in \mathbb{K}^{m \times m} \end{array} \right]$$

1. Idee: Wäre die Σ_{ii} gerade d. ö.w der Hermiteschen Matrix $A^H A$ sind, könnte man 1. QR-Alg. auf $A^H A$ anwenden. $A^H A$ symmetrisch, macht man das nicht, wie folgendes Bsp zeigt:

$$A = \begin{pmatrix} 1 & 1 \\ 0 & \sqrt{\epsilon} \\ \sqrt{\epsilon} & 0 \end{pmatrix} \rightarrow A^H A = \begin{pmatrix} 1+\epsilon & 1 \\ 1 & 1+\epsilon \end{pmatrix} \quad \text{d.h. Singulärwerte von } A \text{ sind } \sqrt{\epsilon}, \sqrt{2+\epsilon}$$

Anderssatz sind die Singulärwerte von $\tilde{A} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$ die Zahlen $0, \sqrt{2}$. Würde befehlen: Das Berechnen von $A^H A = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$ und anschließende Störungen um $O(\epsilon)$ auf die Form $A^H A$ führt zu ~~den~~ Störungen d. Singulärwerte von A um $\sqrt{\epsilon}$. \rightarrow schlecht konditioniert

2. Idee: Setze $C = \begin{pmatrix} 0 & A \\ A^H & 0 \end{pmatrix}$ Dann ist C Hermitesch

und

$$C = Z^H \begin{pmatrix} -\Sigma & 0 & 0 \\ 0 & \Sigma & 0 \\ 0 & 0 & 0 \end{pmatrix} Z, \quad Z = \frac{1}{\sqrt{2}} \begin{pmatrix} u & v \\ -v & u \end{pmatrix}$$

unitär

Also sind 1. BW von C gerade \pm Singulärwerte von A .

QR-Algorithm ohne und mit Shift

QR-Iteration ohne Shift

$$A =: QR$$

$$A := RQ$$

QR-Iteration mit single Shift

$$\sigma := A_{nn}$$

$$A - \sigma I =: QR$$

$$A := RQ + \sigma I$$

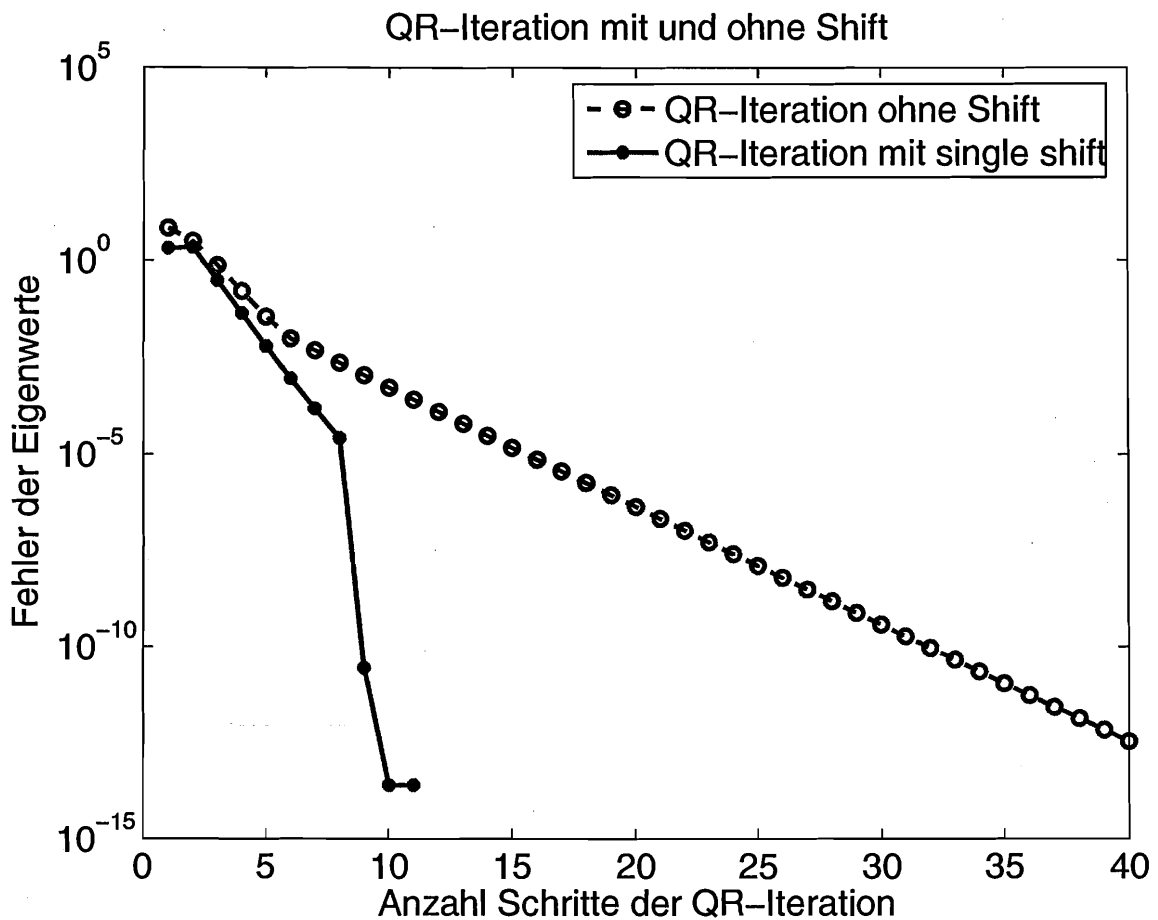
prüfe, ob A zerfällt. Falls ja:

wende QR-Alg. mit Shift

auf oberen Teilblock an.

Wir erwarten:

1. *lineare Konvergenz* beim QR-Algorithmus *ohne* Shift (genauer: wenn die EW die Bedingung $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$ erfüllen).
2. *schnelle (asymptotisch quadratische) Konvergenz* beim QR-Algorithmus mit Shift.



Ausgangsmatrix

$$A = \begin{pmatrix} 3.5488 & 15.593 & 8.5775 & -4.0123 \\ 2.3595 & 24.524 & 14.596 & -5.8157 \\ 0.0899 & 27.599 & 21.438 & -5.8415 \\ 1.9227 & 55.667 & 39.717 & -10.558 \end{pmatrix},$$

Hessenbergform

$$Q^T A Q = H = \begin{pmatrix} 3.5488 & -9.8025 & 4.1046 & -14.8282 \\ -3.0450 & 36.5229 & -8.3593 & 55.8301 \\ 0 & 25.8343 & -2.5900 & 41.15901 \\ 0 & 0 & -0.5372 & 1.4710 \end{pmatrix},$$

QR-Iteration ohne Shift

$$A_1 = \begin{pmatrix} 23.9 & -6.6 & 39.2 & 34.5 \\ -21.8 & 12.8 & -42.1 & -38.6 \\ 0 & 1.0 & 0.63 & -0.6 \\ 0 & 0 & 0.72 & 1.6 \end{pmatrix}$$

$$A_5 = \begin{pmatrix} 29.99 & 32.66 & 71.6 & -10.6 \\ -2.1_{-2} & 6.0 & 1.7 & -0.44 \\ 0 & 1.5_{-2} & 2.02 & -0.21 \\ 0 & 0 & 6.4_{-2} & 0.97 \end{pmatrix}$$

$$A_{10} = \begin{pmatrix} 29.97 & 32.9 & 70.68 & 14.9 \\ -6.6_{-6} & 6.0 & 1.79 & 0.56 \\ 0 & 6.2_{-5} & 2.0 & 0.27 \\ 0 & 0 & -1.8_{-3} & 0.99 \end{pmatrix}$$

$$A_{15} = \begin{pmatrix} 29.9 & 32.9 & 70.65 & -15.05 \\ -2.1_{-9} & 6.0 & 1.79 & -0.56 \\ 0 & 2.6_{-7} & 2.0 & -0.28 \\ 0 & 0 & 5.3_{-5} & 0.99 \end{pmatrix}$$

$$A_{20} = \begin{pmatrix} 29.97 & 32.9 & 70.65 & 15.0529 \\ -6.8_{-13} & 6.0 & 1.79 & 0.56 \\ 0 & 1.1_{-9} & 2.0 & 0.28 \\ 0 & 0 & -1.6_{-6} & 0.99 \end{pmatrix}$$

QR-Iteration mit Shift und Deflation

$$A_1 = \begin{pmatrix} 32.0 & 1.66 & -28.9 & 49.9 \\ -23.4 & 4.17 & 23.6 & -42.1 \\ 0 & -0.65 & 0.99 & 0.03 \\ 0 & 0 & -0.33 & 1.76 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 32.1 & 30.8 & 8.1 & -71.9 \\ -1.8 & 3.8 & -0.05 & 3.2 \\ 0 & 0.14 & 0.98 & 0.085 \\ 0 & 0 & -0.16 & 2.02 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} 30.3 & 32.7 & -4.17 & 72.15 \\ -0.2 & 5.70 & 0.08 & 1.21 \\ 0 & -4.0_{-2} & 0.99 & -0.29 \\ 0 & 0 & 2.98_{-3} & 2.0 \end{pmatrix}$$

$$A_4 = \begin{pmatrix} 30.1 & 32.9 & -4.06 & -72.12 \\ -0.03 & 6.0 & 0.089 & -1.78 \\ 0 & 1.01_{-2} & 0.99 & 0.27 \\ 0 & 0 & -2.3_{-6} & 2.0 \end{pmatrix}$$

$$A_5 = \begin{pmatrix} 30.0 & 32.9 & -4.14 & 72.12 \\ -4.5_{-3} & 6.0 & 0.07 & 1.86 \\ 0 & -2.56_{-3} & 0.99 & -0.28 \\ 0 & 0 & 1.5_{-12} & 2.0 \end{pmatrix}$$

$$A_6 = \begin{pmatrix} 30.0 & 32.9 & -4.12 & -72.12 \\ -6.4_{-4} & 6.0 & 0.07 & -1.87 \\ 0 & 6.5_{-4} & 0.99 & 0.28 \\ 0 & 0 & 0.0 & 2.0 \end{pmatrix}$$

$$A_7 = \begin{pmatrix} 30.0 & 32.9 & 4.12 \\ -1.1_{-5} & 6.0 & -0.07 \\ 0 & 1.2_{-9} & 0.99 \end{pmatrix}$$

$$A_8 = \begin{pmatrix} 30.0 & 32.9 & -4.12 \\ -2.0_{-5} & 6.0 & 0.073 \\ 0 & 0.0 & 0.99 \end{pmatrix}$$

$$A_9 = \begin{pmatrix} 30.0 & -32.9 \\ 2.1_{-11} & 6.0 \end{pmatrix}$$

$$A_{10} = \begin{pmatrix} 30.0 & 32.9 \\ -2.5_{-23} & 6.0 \end{pmatrix}$$

$$R = \begin{pmatrix} 30.0 & 32.94 & -4.12 & -72.1 \\ 0.0 & 6.0 & 0.073 & -1.87 \\ 0 & 0.0 & 0.99 & 0.28 \\ 0 & 0 & 0.0 & 2.0 \end{pmatrix}$$

Orthogonale Iteration

Iterationsvorschrift:

input: $\tilde{Q}_0 \in \mathbb{K}^{n \times k}$ mit orthonormalen Spalten

$$\begin{aligned} \tilde{X}_{l+1} &:= A Q_l \\ [Q_l, R_l] &= \text{qr}(\tilde{Q}_{l+1}) \end{aligned}$$

Satz: Falls die k orthonormalen Spalten von Q_l zu einer ONB von \mathbb{K}^n ergänzt werden (ergänzenden Spalten: \hat{Q}_l), dann gilt mit der Matrix $Q = (Q_l, \hat{Q}_l)$, daß

$$Q^H A Q = \begin{pmatrix} Q_l^H A Q_l & Q_l^H A \hat{Q}_l \\ \hat{Q}_l^H A Q_l & \hat{Q}_l^H A \hat{Q}_l \end{pmatrix} \quad \text{die Abschätzung} \quad \|\hat{Q}_l^H A Q_l\|_2 \leq C \delta^l, \quad \text{erfüllt,}$$

wobei $\delta < 1$, falls für die EW von A gilt: $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_k| > |\lambda_{k+1}| \geq |\lambda_{k+2}| \geq \dots \geq |\lambda_n|$.

orthogonale Iteration mit $Q_0 = e_1$:

$$\begin{aligned} A &= \begin{pmatrix} 3.5488 & 15.593 & 8.5775 & -4.0123 \\ 2.3595 & 24.524 & 14.596 & -5.8157 \\ 0.0899 & 27.599 & 21.438 & -5.8415 \\ 1.9227 & 55.667 & 39.717 & -10.558 \end{pmatrix} \\ Q_1^T A Q_1 &= \begin{pmatrix} 23.88 & 7.4511 & -18.7348 & 48.5777 \\ 21.81 & 13.8262 & -19.4457 & 53.4859 \\ 0.05 & 0.9388 & 0.7410 & 1.1440 \\ -0.38 & -0.6794 & 0.0344 & 0.5050 \end{pmatrix} \\ Q_5^T A Q_5 &= \begin{pmatrix} 2.9995_{+01} & -3.2632_{+01} & -7.2229_{+01} & -4.5374_{+00} \\ 2.0674_{-02} & 5.9635_{+00} & 1.8037_{+00} & 2.9067_{-01} \\ 6.7202_{-06} & 1.6060_{-02} & 2.0253_{+00} & 1.2553_{-01} \\ -9.5156_{-07} & -1.4322_{-03} & -1.5312_{-01} & 9.6943_{-01} \end{pmatrix} \\ Q_{10}^T A Q_{10} &= \begin{pmatrix} 2.9966_{+01} & -3.2941_{+01} & -7.0716_{+01} & 1.4757_{+01} \\ 6.6205_{-06} & 5.9990_{+00} & 1.7890_{+00} & -5.5720_{-01} \\ 8.9954_{-12} & 6.7581_{-05} & 2.0006_{+00} & -2.7247_{-01} \\ 3.6005_{-14} & 1.6941_{-07} & 4.2406_{-03} & 9.8680_{-01} \end{pmatrix} \\ Q_{15}^T A Q_{15} &= \begin{pmatrix} 2.9966_{+01} & -3.2942_{+01} & -7.0655_{+01} & -1.5044_{+01} \\ 2.1287_{-09} & 5.9990_{+00} & 1.7866_{+00} & 5.6447_{-01} \\ 1.1896_{-17} & 2.7806_{-07} & 1.9994_{+00} & 2.7658_{-01} \\ -1.4008_{-21} & -2.0498_{-11} & -1.2463_{-04} & 9.8791_{-01} \end{pmatrix} \end{aligned}$$

orthogonale Iteration mit $Q_0 = [e_1, e_2]$:

$$\begin{aligned}
 A &= \begin{pmatrix} 3.5488 & 15.593 & 8.5775 & -4.0123 \\ 2.3595 & 24.524 & 14.596 & -5.8157 \\ 0.0899 & 27.599 & 21.438 & -5.8415 \\ 1.9227 & 55.667 & 39.717 & -10.558 \end{pmatrix} \\
 Q_1^T A Q_1 &= \begin{pmatrix} 23.88 & 7.4511 & -18.7348 & 48.5777 \\ 21.81 & 13.8262 & -19.4457 & 53.4859 \\ 0.05 & 0.9388 & 0.7410 & 1.1440 \\ -0.38 & -0.6794 & 0.0344 & 0.5050 \end{pmatrix} \\
 Q_5^T A Q_5 &= \begin{pmatrix} 2.9995_{+01} & -3.2632_{+01} & -7.2229_{+01} & -4.5374_{+00} \\ 2.0674_{-02} & 5.9635_{+00} & 1.8037_{+00} & 2.9067_{-01} \\ 6.7202_{-06} & 1.6060_{-02} & 2.0253_{+00} & 1.2553_{-01} \\ -9.5156_{-07} & -1.4322_{-03} & -1.5312_{-01} & 9.6943_{-01} \end{pmatrix} \\
 Q_{10}^T A Q_{10} &= \begin{pmatrix} 2.9966_{+01} & -3.2941_{+01} & -7.0716_{+01} & 1.4757_{+01} \\ 6.6205_{-06} & 5.9990_{+00} & 1.7890_{+00} & -5.5720_{-01} \\ 8.9954_{-12} & 6.7581_{-05} & 2.0006_{+00} & -2.7247_{-01} \\ 3.6005_{-14} & 1.6941_{-07} & 4.2406_{-03} & 9.8680_{-01} \end{pmatrix} \\
 Q_{15}^T A Q_{15} &= \begin{pmatrix} 2.9966_{+01} & -3.2942_{+01} & -7.0655_{+01} & -1.5044_{+01} \\ 2.1287_{-09} & 5.9990_{+00} & 1.7866_{+00} & 5.6447_{-01} \\ 1.1896_{-17} & 2.7806_{-07} & 1.9994_{+00} & 2.7658_{-01} \\ -1.4008_{-21} & -2.0498_{-11} & -1.2463_{-04} & 9.8791_{-01} \end{pmatrix}
 \end{aligned}$$

orthogonale Iteration mit $Q_0 = [e_1, e_2, e_3]$:

$$\begin{aligned}
 A &= \begin{pmatrix} 3.5488 & 15.593 & 8.5775 & -4.0123 \\ 2.3595 & 24.524 & 14.596 & -5.8157 \\ 0.0899 & 27.599 & 21.438 & -5.8415 \\ 1.9227 & 55.667 & 39.717 & -10.558 \end{pmatrix} \\
 Q_1^T A Q_1 &= \begin{pmatrix} 23.88 & 7.4511 & -18.7348 & 48.5777 \\ 21.81 & 13.8262 & -19.4457 & 53.4859 \\ 0.05 & 0.9388 & 0.7410 & 1.1440 \\ -0.38 & -0.6794 & 0.0344 & 0.5050 \end{pmatrix} \\
 Q_5^T A Q_5 &= \begin{pmatrix} 2.9995_{+01} & -3.2632_{+01} & -7.2229_{+01} & -4.5374_{+00} \\ 2.0674_{-02} & 5.9635_{+00} & 1.8037_{+00} & 2.9067_{-01} \\ 6.7202_{-06} & 1.6060_{-02} & 2.0253_{+00} & 1.2553_{-01} \\ -9.5156_{-07} & -1.4322_{-03} & -1.5312_{-01} & 9.6943_{-01} \end{pmatrix} \\
 Q_{10}^T A Q_{10} &= \begin{pmatrix} 2.9966_{+01} & -3.2941_{+01} & -7.0716_{+01} & 1.4757_{+01} \\ 6.6205_{-06} & 5.9990_{+00} & 1.7890_{+00} & -5.5720_{-01} \\ 8.9954_{-12} & 6.7581_{-05} & 2.0006_{+00} & -2.7247_{-01} \\ 3.6005_{-14} & 1.6941_{-07} & 4.2406_{-03} & 9.8680_{-01} \end{pmatrix} \\
 Q_{15}^T A Q_{15} &= \begin{pmatrix} 2.9966_{+01} & -3.2942_{+01} & -7.0655_{+01} & -1.5044_{+01} \\ 2.1287_{-09} & 5.9990_{+00} & 1.7866_{+00} & 5.6447_{-01} \\ 1.1896_{-17} & 2.7806_{-07} & 1.9994_{+00} & 2.7658_{-01} \\ -1.4008_{-21} & -2.0498_{-11} & -1.2463_{-04} & 9.8791_{-01} \end{pmatrix}
 \end{aligned}$$