PROF. DR. BERND SIMEON
CHRISTIAN GOBERT
THOMAS MÄRZ

TECHNISCHE UNIVERSITÄT MÜNCHEN

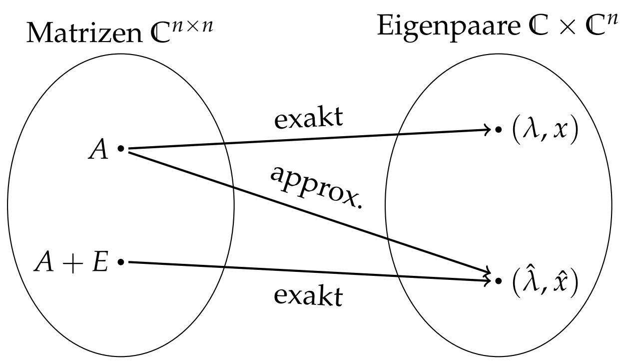
Numerik

SS 2009

Lösungsvorschläge zu Übungsblatt 5

Aufgabe 1: *(Rückwertsfehler von Eigenpaaren)*

Die Berechnung eines Rechtseigenpaares $(\lambda, x) \in \mathbb{C} \times \mathbb{C}^n$ für die Matrix $A \in \mathbb{C}^{n\times n}$ ist eine Abbildung $A \to (\lambda, x)$. Sei nun $(\hat\lambda, \hat x) \in \mathbb{C} \times \mathbb{C}^n$ ein approximatives Rechtseigenpaar, z.B. durch Vektoriteration ermittelt. Im Rückwärtsfehlerkonzept wird der Fehler, der in $(\hat\lambda, \hat x)$ steckt, zurückgespielt zu einem Eigabefehler E in der Eingabe A und zwar so , dass $(\hat\lambda, \hat x)$ ein exaktes Rechtseigenpaar von $A + E$ ist (siehe Skizze).



Damit sind die Rückwärtsfehlermatrizen E durch

$$(A + E)\hat x = \hat\lambda \, \hat x$$

charakterisiert.

a) Bestimmen Sie ein solches E. Versuchen Sie es mit einer Rang 1 Matrix. Benutzen Sie dazu das Residuum r

$$r := \hat\lambda \, \hat x - A\hat x.$$

Sie können ohne Einschränkung davon ausgehen, dass $\|\hat x\|_2 = 1$ gilt.

b) Berechnen Sie $\|E\|_2$ für die Matrix E aus Teil a).

c) Benutzen Sie die so gewonnene Abschätzung des Rückwärtsfehlers, um ein Abbruchkriterium für die Vektoriteration (und seine Verwandten) zu entwerfen.

d) Formulieren Sie den Algorithmus für die Vektoriteration mit dem Abbruchkriterium aus Teil c).

Lösung Aufgabe 1:

a) Ansatz, der das Residuum einflechtet:

$$A\hat{x} + \hat{\lambda}\ \hat{x} - A\hat{x} = \hat{\lambda}\ \hat{x} \qquad \Leftrightarrow \qquad A\hat{x} + r = \hat{\lambda}\ \hat{x}\ .$$

Wir nutzen, dass $1 = \|\hat{x}\|_2^2 = \hat{x}^H\hat{x}$ gilt und erhalten

$$\begin{aligned} A\hat{x} + r \cdot 1 &= \hat{\lambda}\ \hat{x} \\ A\hat{x} + r \cdot (\hat{x}^H\hat{x}) &= \hat{\lambda}\ \hat{x} \\ (A + r\hat{x}^H)\hat{x} &= \hat{\lambda}\ \hat{x} \\ (A + E)\hat{x} &= \hat{\lambda}\ \hat{x}\ . \end{aligned}$$

Also erfüllt die Rang 1 Matrix $E = r\hat{x}^H$ den Zweck.

b) Hier die Berechnung mit der Definition der Matrix-2-Norm als Operatornorm:

$$\|E\|_2 = \max_{\|h\|_2=1} \|Eh\|_2 = \max_{\|h\|_2=1} \|r\|_2|\hat{x}^H h| \le \|r\|_2\|\hat{x}\|_2 \max_{\|h\|_2=1} \|h\|_2 = \|r\|_2$$

Wobei Gleicheit für die speziellen Wahlen $h = \pm\hat{x}$ des Testvektors gilt.

c) Man wird abbrechen, wenn der Rückwärtsfehler $\|r\|_2 = \|E\|_2$ klein genug ist, somit hätte man die Möglichkeiten

$$\begin{aligned} \text{absolut:} &\quad \|r\|_2 \le \text{TOL} \\ \text{relativ:} &\quad \|r\|_2 \le \|A\| \cdot \text{TOL} \end{aligned}$$

Wobei für $\|A\|$ nicht die Matrix-2-Norm nimmt. Warum wohl ?

Eine andere Möglichkeit, die ohne die Vorgabe einer Toleranz auskommt, wäre

$$\|r\|_2 \le \|A\| \cdot C \cdot K_{\text{iter}} \cdot \text{eps}\ ,$$

wobei K_{iter} die Kosten einer Iteration bedeutet und C eine geeignete Konstante ist. Interpretation: Man bricht dann ab, wenn das approximative Eigenpaar der aktuellen Iteration als **rückwärtsstabil berechnet** gilt.

d) Algorithmus für die Vektoriteration mit dem Abbruchkriterium:

```
function [lambda,x] = vec_iter(A,x0,TOL)
   TOL = norm(A,inf) * TOL;

   h = A*x0;
   while true
      x = h;
      x = x/norm(x,2);

      h=A*x;
      lambda = x'*h;

      if norm( lambda*x - h ,2) <= TOL
         break;
      end
   end
```

Aufgabe 2: *(QR-Iteration)*

Die QR-Iteration zur Eigenwertbestimmung ist für *allgemeine* Matrizen $A \in \mathbb{R}^{n\times n}$ sehr teuer. Dieser Nachteil lässt sich dadurch beheben, dass A zunächst mittels einer Ähnlichkeitstransformation auf spezielle Gestalt gebracht wird. Sei dafür $A^{(0)} = A$ und für $k = 0, ..., n-3$

$$A^{(k+1)} = (Q^{(k)})^T A^{(k)} Q^{(k)},$$

wobei $Q^{(k)}$ die Householder-Reflexion zur Eliminierung der Einträge $A^{(k)}_{(k+2):n,(k+1)}$ bezeichne. Zeigen Sie:

a) $A^{(k)}$ ist ähnlich zu A,

b) $A^{(n-2)}$ ist eine obere Hessenberg-Matrix.

c) Welche Gestalt besitzt $A^{(n-2)}$, wenn A symmetrisch ist?

Auf $A_0 = A^{(n-2)}$ werde nun die QR-Iteration folgendermaßen angewandt: Für $i \geq 0$ wird mittels geeigneter *Givensrotationen* eine Q_iR_i-Zerlegung von A_i und anschließend $A_{i+1} = R_iQ_i$ berechnet.

d) Zeigen Sie, dass A_i für $i \geq 0$ die gleiche Struktur wie $A^{(n-2)}$ in b) bzw. c) besitzt.

e) Bestimmen Sie den Aufwand zur Berechnung von A_{i+1} aus A_i für den allgemeinen und den symmetrischen Fall. Es genügt jeweils die Form $\mathcal{O}(n^s)$ mit einem geeigneten s.

Lösung Aufgabe 2:

a) Mittels vollständiger Induktion wird gezeigt, dass alle Matrizen $A^{(k)}$ die gleichen Eigenwerte besitzen. Sei dazu $A^{(k)}x = \lambda x$. Dann gilt mit $y = (Q^{(k)})^T x$

$$A^{(k+1)}y = (Q^{(k)})^T A^{(k)} Q^{(k)} y = (Q^{(k)})^T A^{(k)} x = (Q^{(k)})^T \lambda x = \lambda y.$$

Also ist $A^{(k+1)}$ ähnlich zu $A^{(k)}$ und per Induktion auch ähnlich zu $A^{(0)} = A$.

b) Wir zeigen per Induktion, dass $A^{(k)}$ und $Q^{(k)}$ die folgende Gestalt haben:

$$A^{(k)} = \left(\begin{array}{c|c} H^{(k)} & * \\ \hline 0\ldots0\ a_{k+1} & \tilde{A}^{(k)} \end{array}\right), Q^{(k)} = \left(\begin{array}{c|c} I & 0 \\ \hline 0 & \tilde{Q}^{(k)} \end{array}\right)$$

mit der oberen Hessenberg-Matrix $H^{(k)} \in \mathbb{R}^{(k+1)\times(k+1)}$, $a_{k+1} \in \mathbb{R}^{n-k-1}$, $\tilde{A}^{(k)} \in \mathbb{R}^{(n-k-1)\times(n-k-1)}$ und insbesondere $(\tilde{Q}^{(k)})^T a_{k+1} = (\alpha, 0, \ldots, 0)^T$. Sei diese Annahme für ein k mit $1 \leq k < n$ erfüllt. Dann ergibt sich für die nächste Iterierte

$$\begin{aligned}
A^{(k+1)} &= (Q^{(k)})^T A^{(k)} Q^{(k)} = \left(\begin{array}{c|c} I & 0 \\ \hline 0 & (\tilde{Q}^{(k)})^T \end{array}\right) \left(\begin{array}{c|c} H^{(k)} & * \\ \hline 0\ldots0\ a_{k+1} & \tilde{A}^{(k)} \end{array}\right) \left(\begin{array}{c|c} I & 0 \\ \hline 0 & \tilde{Q}^{(k)} \end{array}\right) \\
&= \left(\begin{array}{cccc|c} \multicolumn{4}{c|}{H^{(k)}} & * \\ \hline 0 & \ldots & 0 & \alpha & \\ 0 & \ldots & 0 & 0 & (\tilde{Q}^{(k)})^T \tilde{A}^{(k)} \\ \vdots & & \vdots & \vdots & \\ 0 & \ldots & 0 & 0 & \end{array}\right) \left(\begin{array}{c|c} I & 0 \\ \hline 0 & \tilde{Q}^{(k)} \end{array}\right) \\
&= \left(\begin{array}{cccc|c} \multicolumn{4}{c|}{H^{(k)}} & * \\ \hline 0 & \ldots & 0 & \alpha & \\ 0 & \ldots & 0 & 0 & (\tilde{Q}^{(k)})^T \tilde{A}^{(k)} (\tilde{Q}^{(k)}) \\ \vdots & & \vdots & \vdots & \\ 0 & \ldots & 0 & 0 & \end{array}\right) = \left(\begin{array}{c|c} H^{(k+1)} & * \\ \hline 0\ldots0\ a_{k+2} & \tilde{A}^{(k+1)} \end{array}\right)
\end{aligned}$$

mit der oberen Hessenbergmatrix $H^{(k+1)} \in \mathbb{R}^{(k+2)\times(k+2)}$, so dass der Induktionsschritt gezeigt wurde. Für $k = n-2$ ergibt sich dann nach Induktion die obere Hessenbergmatrix

$$A^{(n-2)} = \left(\begin{array}{c|c} H^{(n-2)} & * \\ \hline 0 \dots 0 \; a_{n-1} & \tilde{A}^{(n-2)} \end{array}\right)$$

mit $H^{(n-2)} \in \mathbb{R}^{(n-1)\times(n-1)}$ und $a_{n-1}, \tilde{A}^{(n-2)} \in \mathbb{R}$.

c) Wiederum mit vollständiger Induktion wird gezeigt, dass alle $A^{(k)}$ symmetrisch sind. Sei $A^{(k)}$ symmetrisch. Dann ist

$$(A^{(k+1)})^T = \left((Q^{(k)})^T A^{(k)} Q^{(k)}\right)^T = (Q^{(k)})^T (A^{(k)})^T Q^{(k)} = (Q^{(k)})^T A^{(k)} Q^{(k)} = A^{(k+1)},$$

d.h. $A^{(k+1)}$ ist ebenfalls symmetrisch. Also ist auch $A^{(n-2)}$ symmetrisch und gleichzeitig eine obere Hessenbergmatrix, also Tridiagonalmatrix.

d) Zunächst werden mittels geeigneter Givensrotationen die Subdiagonalelemente zu Null reduziert,

$$R_i = G_{n-1}^{(i)} \cdots G_1^{(i)} A_i = \begin{pmatrix} * & \cdots & \cdots & * \\ & \ddots & & \vdots \\ & & \ddots & \vdots \\ & & & * \end{pmatrix}$$

wobei R_i für die Tridiagonalmatrix A_i (mit Bandbreite 1) höchstens die obere Bandbreite[1] 2 haben kann. Anschließend berechnet man A_{i+1} als

$$A_{i+1} = R_i Q_i = R_i (G_1^{(i)})^T \cdots (G_{n-1}^{(i)})^T.$$

Die Matrix Q_i hat die untere Bandbreite 1, so dass auch A_{i+1} die untere Bandbreite 1 aufweist und damit eine obere Hessenberg-Matrix ist.

e) Es werden $\mathcal{O}(n)$ Givensrotationen angewendet. Im allgemeinen Fall kostet eine Givensrotation $\mathcal{O}(n)$ Operationen, im Fall einer Tridiagonalmatrix $\mathcal{O}(1)$ Operationen. Insgesamt ergibt sich also ein Aufwand von $\mathcal{O}(n^2)$ für den allgemeinen und von $\mathcal{O}(n)$ für den symmetrischen Fall.

[1] obere bzw. untere Bandbreite bezieht sich auf die erste und letzte Zeile der Matrix, d.h. obere Bandbreite p heisst, dass $a_{1,q} = 0$ für $q \geq p+2$