Heinrich Heine Universität Düsseldorf

Mathematisches Institut

Prof. Dr. Achim Schädle
Marina Fischer

23.11.2021

27	28	29	30	Σ

Name: ____________________

Numerik II – 7. Übungsblatt

Aufgabe 27:

(a) Es sei $A = xy^H$, wobei x und y Vektoren in $\mathbb{C}^n$, $n \geq 2$, sind. Zeigen Sie, dass 0 ein Eigenwert von A mit Vielfachheit mindestens $n-1$ ist und dass der verbleibende Eigenwert $\lambda = y^H x$ ist.

(b) Bestimmen Sie alle Eigenwerte der Householder-Matrix $P = I - 2uu^H$ mit $\|u\|_2 = 1$.

Aufgabe 28:

(a) Transformieren Sie die Matrix

$$A = \begin{bmatrix} 3 & 3 \\ 4 & 4 \\ 0 & 3 \end{bmatrix}$$

mit Householder-Transformationen von links auf obere Dreiecksform.

(b) Geben Sie eine QR-Zerlegung von A an.

Aufgabe 29:

Gegeben sei eine beliebige nicht-singuläre (3×3)-Matrix, in der wir Nullelemente durch Multiplikation mit unitären Matrizen U_j (etwa Householder-Matrizen) von links oder rechts erzeugen möchten. Betrachten Sie die folgenden Matrixstrukturen:

$$\begin{bmatrix} \times & \times & 0 \\ 0 & \times & \times \\ 0 & 0 & \times \end{bmatrix}, \qquad \begin{bmatrix} \times & 0 & 0 \\ \times & \times & 0 \\ \times & \times & \times \end{bmatrix}, \qquad \begin{bmatrix} \times & \times & 0 \\ 0 & 0 & \times \\ 0 & 0 & \times \end{bmatrix}$$

Ordnen Sie jeder der drei Matrixstrukturen eine der folgenden Situationen zu und rechtfertigen Sie Ihre Aussage:

(a) Kann durch eine Folge von Multiplikationen von links mit Matrizen U_j erzeugt werden.

(b) Nicht (a), kann durch eine Folge von Multiplikationen von links und rechts mit Matrizen U_j erzeugt werden. Dabei müssen die Transformationen von links und rechts *nicht* dieselben sein.

(c) Kann nicht durch eine Folge von Multiplikationen von links und rechts mit Matrizen U_j erzeugt werden.

b.w.

Aufgabe 30: (QR Algorithmus ohne Shifts)

Implementieren Sie den QR Algorithmus ohne Shifts aus der Vorlesung. Gehen Sie dabei in folgenden Schritten vor.

(a) Schreiben Sie eine Funktion `HH = qralg(H,tol)`, die den ungeshifteten QR Algorithmus auf eine obere Hessenbergmatrix $H \in \mathbb{C}^{n \times n}$ anwendet. Brechen Sie ab, wenn der betragsmäßig größte Eintrag der unteren Nebendiagonalen kleiner als `tol` ist.

(b) Schreiben Sie eine Funktion `ew = findeeigenwerte(A,tol)`, die zunächst die obere Hessenbergform einer Matrix `A` berechnet und dann `qralg` verwendet um alle Eigenwerte von `A` zu berechnen. Speichern Sie die berechneten Eigenwerte in einem geeigneten Vektor `ew`.

(c) Modifizieren Sie `qralg` derart, dass Sie neben `HH` einen Vektor zurückliefern, der die Werte von $|H_{n,n-1}|$ in jeder Iteration enthält. Plotten Sie dann die aneinandergereihte Konvergenzgeschichte aller Eigenwerte in einem logarithmischen Plot gegen die Anzahl an QR-Zerlegungen.

(d) Testen Sie Ihre Implementierung mit Matrizen mit reellen Eigenwerten und der Hilbertmatrix (`hilbert` in Python bzw. `hilb` in Matlab).

Hinweis: Die Pythonfunktionen `hessenberg` und `qr` aus dem Modul `scipy.linalg` bzw. die Matlabfunktionen `hess` und `qr` dürfen Sie verwenden.

Abgabe der Übungsaufgaben am Dienstag, 30.11.2021 zu Beginn der Vorlesung.
Abgabe der Programmierübungen per E-Mail bis Dienstag, 30.11.2021, 8:30 Uhr an ***marina.fischer@uni-duesseldorf.de.***