Simon King, FSU Jena
Fakultät für Mathematik und Informatik
Henicke, Kraume, Lafeld, Max, Rump

Lineare Algebra für *-Informatik
FMI-MA0022

Wintersemester 2020/21 Übungsblatt 9

Liveaufgaben (20.–21.01.2021)

Präsenzaufgabe 9.1: *Wie man $\chi_A(X)$ __nicht__ berechnet*
Sei $A := \begin{pmatrix} 2 & 1 & 1 \\ 0 & 4 & 2 \\ -3 & 0 & 0 \end{pmatrix} \in M_3(\mathbb{R})$.

a) Berechnen Sie $\chi_A(X)$. **Tipp:** Laplace 3. Zeile.

b) Dozent K. hat schlecht geschlafen und will daher die Rechnung wie folgt vereinfachen: Wende Gauß an, nämlich $A \rightsquigarrow \begin{pmatrix} 2 & 1 & 1 \\ 0 & 4 & 2 \\ 0 & \frac{3}{2} & \frac{3}{2} \end{pmatrix} \rightsquigarrow \begin{pmatrix} 2 & 1 & 1 \\ 0 & 4 & 2 \\ 0 & 0 & \frac{3}{4} \end{pmatrix}$, und an der Dreiecksgestalt erkennt man $\chi_A(X) = (X-2)\cdot(X-4)\cdot(X-3/4)$.
Hat K. recht, oder sollte man seine Methode in der Klausur vermeiden?

Präsenzaufgabe 9.2: *Ein numerischer Algorithmus*
Sei $n \in \mathbb{N}^*$, $A \in M_n(\mathbb{R})$, $\vec{v}_1, ..., \vec{v}_r \in \mathbb{R}^n$ Eigenvektoren von A zu Eigenwerten $\lambda_1, ..., \lambda_r \in \mathbb{R}$, und $\vec{v} := \alpha_1\vec{v}_1 + ... + \alpha_r\vec{v}_r$.

a) Drücken Sie $A^k \cdot \vec{v}$ ($k \in \mathbb{N}$) durch k, $\alpha_1, ..., \alpha_r$, $\vec{v}_1, ..., \vec{v}_r$ und $\lambda_1, ..., \lambda_r$ aus.

b) Sei zudem $\alpha_1 \neq 0$ und $\lambda_1 > \lambda_2 \geq \lambda_3 \geq ... \geq \lambda_r \geq 0$.
Wie kann man einen beliebig guten Näherungswert für $\alpha_1\vec{v}_1$ berechnen, wenn λ_1 bekannt ist? Verwenden Sie die Formel, die Sie in der vorigen Teilaufgabe fanden.

c) *Praktisches Beispiel, falls Sie online auf ein CAS zugreifen können:*
Sei $A := \begin{pmatrix} 0.4 & 0.3 & 0.2 \\ 0.4 & 0.2 & 0.6 \\ 0.2 & 0.5 & 0.2 \end{pmatrix} \in M_3(\mathbb{R})$ und $\vec{v} := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$. Sie dürfen verwenden, dass die Voraussetzungen aus der vorigen Teilaufgabe mit $\lambda_1 = 1$ erfüllt sind. Berechnen Sie $A^k\vec{v}$ für $k \in \{5, 10, 15\}$. Wird durch $A^{15}\vec{v}$ tatsächlich ein Eigenvektor von A zum Eigenwert $\lambda_1 = 1$ sinnvoll genähert?

d) Die Voraussetzungen aus b) seien erfüllt, aber diesmal sei λ_1 unbekannt. Wie kann man den Algorithmus modifizieren, so dass man näherungsweise sowohl λ_1 als auch $\vec{v}_1$ (bis auf Skalierung) berechnen kann? **Anmerkung:** Der hier untersuchte Algorithmus ist sogar für große n praktikabel, falls die betrachtete Matrix nur wenige von Null verschiedene Einträge hat (wieso?); schon im 19. Jahrhundert wurde bewiesen, dass es sogar für $n = 5$ im Allgemeinen unmöglich ist, Eigenwerte und Eigenvektoren exakt zu berechnen.