Prof. Dr. Reinhard Klein
Alexander Schier

Universität Bonn
Institut für Informatik II
Wintersemester 2023/2024 17.01.2024

Übungen zu Angewandte Mathematik: Numerik

Blatt 13: Wiederholung

Die Lösungen für die praktischen Aufgaben müssen bis Mittwoch, den 24.01.2024, um 12:00 im eCampus hochgeladen werden. Die Lösungen zu Theorieaufgaben müssen bis 12:00 in die Postfächer im Raum 0.004 im Hörsaalgebäude eingeworfen oder digital im eCampus abgegeben werden. Bei digitaler Abgabe werden werden keine Scans, Fotos, etc. gewertet.

Aufgabe 1 (Permutationsmatrizen, *4 Punkte*)

Sei $P \in \{0,1\}^{n\times n}$ eine allgemeine Permutationsmatrix $P = P_n \cdots P_1$, wobei die P_i elementare Permutationsmatrizen sind, welche jeweils 2 Zeilen vertauschen.

Zeige:

a) Das Produkt von zwei Permutationsmatrizen ist wieder eine Permutationsmatrix.
b) Die Multiplikation mit einer Permutationsmatrix von links vertauscht Zeilen. Zeige, dass die Multiplikation mit einer Permutationsmatrix von rechts Spalten vertauscht.
c) Eine elementare Permutationsmatrix, die zwei Zeilen vertauscht, ist symmetrisch.
d) Für eine elementare Permutationsmatrix, die zwei Zeilen vertauscht, gilt $P_i^2 = I$.
e) Für Permutationsmatrizen gilt im Allgemeinen nicht $P^2 = I$.
f) Eine Permutationsmatrix $P = P_n \cdots P_1$ ist orthonormal.
g) Eine Permutation heißt gerade, wenn sie eine gerade Anzahl Vertauschungen vornimmt und sonst ungerade. Zeige, dass wenn $P = P_k \cdots P_1$ ist mit k paarweise verschiedenen Permutationsmatrizen P_i, die jeweils zwei Zeilen vertauschen, gilt $\det(P) = \begin{cases} 1 & \text{wenn } k \text{ gerade} \\ -1 & \text{wenn } k \text{ ungerade} \end{cases}$

Aufgabe 2 (Eigenwertzerlegung, *4 Punkte*)

Sei $n \in \mathbb{N}$ und $A \in \mathbb{C}^{n\times n}$ eine quadratische Matrix. Eine Zahl $\lambda \in \mathbb{C}$ heißt *Eigenwert* von A zum *Eigenvektor* $v \in \mathbb{C}^n \setminus \{0\}$, wenn gilt $\lambda \cdot v = A \cdot v$.

a) Zur Berechnung von Eigenwerten kann man das charakteristische Polynom nutzen:

$$\begin{aligned} p_A : \mathbb{C} &\to \mathbb{C} \\ x &\mapsto \det(x \cdot I - A) \end{aligned}$$

Dabei bezeichnet $I \in \mathbb{C}^{n\times n}$ wie üblich die Einheitsmatrix. Beweise, dass $\lambda \in \mathbb{C}$ genau dann ein Eigenwert von A ist, wenn gilt $\chi_A(\lambda) = 0$.

Hinweis: Es darf ohne Beweis verwendet werden, dass $\det(A) = 0$ genau dann, wenn

$$\ker A := \{w \in \mathbb{C}^n \mid A \cdot w = 0\} \neq \{0\}.$$

b) A heißt diagonalisierbar genau dann, wenn es eine Basis des $\mathbb{C}^n$ aus Eigenvektoren gibt. Sei $v_1, \ldots, v_n \in \mathbb{C}^n$ eine solche Basis mit Eigenwerten $\lambda_1, \ldots, \lambda_n \in \mathbb{C}$. Sei $V := (v_1, \ldots, v_n) \in \mathbb{C}^{n \times n}$ eine Matrix mit $v_1, \ldots, v_n$ als Spalten. Beweise, dass

$$\Lambda := V^{-1} \cdot A \cdot V \in \mathbb{C}^{n \times n}$$

eine Diagonalmatrix ist. Gebe dabei Λ explizit an.

Bemerkung: $A = V \cdot \Lambda \cdot V^{-1}$ bezeichnet man als *Eigenwertzerlegung* oder auf Englisch als "Eigenvalue decomposition" (EVD).

Aufgabe 3 (Kondition, *4 Punkte*)

Sei $\mathbf{A}$ eine 202×202 Matrix mit $\|A\|_2 = 100$ und $\|A\|_F = 101$. Gib die schärfste mögliche Abschätzung der 2-Norm Kondition $\kappa(A)$ nach unten an.

Hinweis: Betrachte dazu die Singulärwertzerlgung von $\mathbf{A}$ und nutze aus, dass sowohl die induzierte 2-Norm $\|\cdot\|_2$, als auch die Frobeniusnorm $\|\cdot\|_F$ unter unitären Transformationen invariant sind und für die Frobeniusnorm gilt $\|A\|_F = \sqrt{\mathrm{Spur}(A^* A)}$. Weiterhin lässt sich ausnutzen, dass die 2-Norm dem größten Singulärwert einer Matrix entspricht.

Aufgabe 4 (Fliesskommazahlen implementieren, *4 Punkte*)

In der Vorlesung haben wir Fließkommazahlen gemäß dem Standard IEEE 754 kennengelernt. Dieser definiert Fließkommazahlen mit 64 bit (double precision), 32 bit (single precision) und 16 bit (half precision). Während für Berechnungen fast immer double oder single precision verwendet wird sind half precision Werte vor allem zum Speichern großer Datenmengen nützlich, z.B. bei High Dynamic Range Fotos.

Um die Eigenschaften von Fließkommazahlen besser kennenzulernen betrachten wir nun analog definierte *8 bit quarter precision* (Minifloat) Fließkommazahlen. Wir verwenden ein Bit für das Vorzeichen $s \in \{0, 1\}$, vier bit für den Exponenten $e \in \{0, \ldots, 2^4 - 1\}$ und drei bit für die Mantisse $m \in \{0, \ldots, 2^3 - 1\}$. Die zugehörige Fließkommazahl ist dann gegeben durch:

$$f(s, e, m) = (-1)^s \cdot \begin{cases} m \cdot 2^{-3} & \text{, falls } e = 0 \text{ (denormalisierte Zahl)} \\ (2^3 + m) \cdot 2^{e-4} & \text{, falls } e \neq 0 \text{ (normalisierte Zahl)} \end{cases}$$

Dabei sind die denormalisierten Zahlen dazu da den Bereich um 0 herum abzudecken wohingegen die normalisierten Zahlen die restliche Zahlengerade abdecken. Auf die IEEE 754 Sonderwerte $\pm\infty$ und NaN (not a number) verzichten wir hier.

a) Schreibe eine Funktion `GetQuarterFloat(Sign,Exponent,Mantissa)`, die $f(s, e, m)$ evaluiert.
b) Generiere eine Liste aller quarter precision Zahlen und plotte sie als Punkte auf einer Geraden.
c) Was ist die größte Zahl und was die kleinste darstellbare positive Zahl?

   Hinweis: Den Plot kann man mit `matplotlib.pyplot.scatter` erstellen wobei man als y-Koordinaten Nullen übergibt. In Python muss man `matplotlib.pyplot.show` aufrufen sobald alle Plots fertig sind.

d) Generiere eine logarithmische Abtastung der positiven Zahlen $x_n := 10^{\frac{n-5000}{5000} \cdot 4{,}6}$ mit $n \in \{0, \ldots, 10000\}$. Runde jede Zahl x_n auf die nächstgelegene quarter precision Zahl y_n. Plotte dann den relativen Rundungsfehler δ_n gegen x_n, wobei

$$\delta_n := \frac{|y_n - x_n|}{x_n}.$$

Hinweis: Die Abtastung kann man mit `numpy.logspace` erstellen. Zum Runden auf den nächsten gültigen Wert kann `numpy.searchsorted` verwendet werden.

e) Erläutere die Ergebnisse des Plots aus Aufgabe d). Wie verhalten sich die relativen Rundungsfehler für sehr kleine oder sehr große Zahlen? Wie verhalten sie sich im Bereich der normalisierten Zahlen?