Fachbereich Mathematik & Informatik
Freie Universität Berlin
Prof. Dr. Ralf Kornhuber, Prof. Dr. Christof Schütte, Lasse Hinrichsen

11. Übung zur Vorlesung

# Computerorientierte Mathematik I

WS 2020/2021

http://numerik.mi.fu-berlin.de/wiki/WS_2020/CoMaI.php

**Abgabe: Do., 25. Februar 2021, 12:15 Uhr**

**1. Aufgabe** (6 Bonus TP)

a) Zeigen Sie, dass die Kondition einer Diagonalmatrix $M \in \mathbb{R}^{n,n}$ mit

$$M_{ij} = \begin{cases} \lambda_i & \text{für } i = j \\ 0 & \text{sonst} \end{cases} \qquad \text{und } \lambda_i \neq 0 \text{ für } i = 1, \ldots, n$$

bezüglich der Norm $\|\cdot\|_\infty$ auf $\mathbb{R}^n$ durch $\kappa_\infty(M) = \frac{\max_{i=1\ldots,n} |\lambda_i|}{\min_{i=1\ldots,n} |\lambda_i|}$ gegeben ist.

b) Bestimmen Sie explizit Vektoren $b, \tilde{b} \in \mathbb{R}^n \setminus \{0\}$, so dass für $x := M^{-1}b$ und $\tilde{x} := M^{-1}\tilde{b}$ die Gleichung

$$\frac{\|x - \tilde{x}\|_\infty}{\|x\|_\infty} = \kappa_\infty(M) \frac{\|b - \tilde{b}\|_\infty}{\|b\|_\infty}$$

gilt.

**2. Aufgabe** (6 Bonus TP)
Seien $A, \tilde{A} \in \mathbb{R}^{n \times n}$ sowie $x, \tilde{x}, b, \tilde{b} \in \mathbb{R}^n$ gegeben. Gelte weiter $b \neq 0$, $Ax = b$, $\tilde{A}\tilde{x} = \tilde{b}$ und $\|A - \tilde{A}\| < 1/\|A^{-1}\|$. Beweisen Sie die Abschätzung

$$\frac{\|x - \tilde{x}\|}{\|x\|} \leq \kappa(A) \left( \frac{\|A - \tilde{A}\|}{\|A\|} + \frac{\|b - \tilde{b}\|}{\|b\|} \right) + o(\|A - \tilde{A}\| + \|b - \tilde{b}\|)$$

für $\|A - \tilde{A}\| + \|b - \tilde{b}\| \to 0$. Hierbei ist $\kappa(A)$ die Kondition der Matrix $A$ bezüglich der $\|\cdot\|$-Norm.

**3. Aufgabe** (10 Bonus PP)

a) Implementieren Sie den Gaußschen Algorithmus ($LR$–Zerlegung ohne Spaltenpivoting). Stellen Sie dazu drei Routinen zur Verfügung, und zwar die eigentliche $LR$–Zerlegung, die Vorwärtselimination und die Rückwärtssubstitution:

- `def lr(A)`

  Erhält eine Matrix (in Form eines `numpy.ndarrays`) `A` und soll ein Tupel `(L, R)` zurückgeben, wobei $L$ und $R$ jeweils Matrizen sind, so dass $LR = A$.

- `def vorwaerts_elim(L, b)`

  Erhält eine untere Dreiecksmatrix `L` und einen Vektor `b` und berechnet $z$, so dass $Lz = b$. Rückgabewert dieser Funktion soll dieses $z$ sein.

- `def rueckwaerts_subst(R, z)`

  Erhält eine obere Dreiecksmatrix `R` und einen Vektor `z` und berechnet $x$, so dass $Rx = z$. Rückgabewert dieser Funktion soll dieses $x$ sein.

Achten Sie darauf, dass die Funktionen die *Eingabedaten* nicht modifziert! Sollte der Gaußsche Algorithmus nicht durchführbar sein, sollte Ihre Funktion eine Fehlermeldung produzieren!

b) Testen Sie Ihre Implementierung, indem Sie Systeme verschiedener Größen lösen. Genauer, für $n = 2, 3, 5, 10, 50$, erzeugen Sie eine zufällig Matrix $A$ (beispielsweise mit `numpy.random.rand(n,n)`) und eine rechte Seite $b$ (ebenfalls mit `numpy.random.rand(n)`). Stellen Sie dabei sicher, dass die Matrix vollen Rank hat (beispielsweise indem Sie `numpy.linalg.det(A)!=0` überprüfen).

Lösen Sie das System $Ax = b$ mittels der von Ihnen geschriebenen Funktionen und betrachten Sie den Fehler mittels

```
np.linalg.norm(A.dot(x) - b).
```

Welche Ausgabe erwarten Sie für eine korrekte Lösung? Wie verhält sich das zu Ihrem Ergebnis?

## Allgemeine Hinweise

Die Punkte unterteilen sich in Theoriepunkte (TP) und Programmierpunkte (PP). Bitte beachten Sie die auf der Vorlesungshomepage angegebenen Hinweise zur Bearbeitung und Abgabe der Übungszettel, insbesondere der Programmieraufgaben.