Prof. Dr. Dirk Lebiedz
M.Sc. Pascal Heiter
Dipl.-Math. oec. Klaus Stolle

Universität Ulm
Institut für Numerische Mathematik
Wintersemester 2013/14

# Numerische Lineare Algebra - Matlabblatt 3

(Besprechung: Freitag, 6. Dezember 2013)

**Aufgabe 6** *(Berechnung der euklidischen Norm für Matrizen)* (10 Punkte)

Wir wollen uns mit der numerischen Berechnung der euklidischen Norm für Matrizen $A \in \mathbb{R}^{3\times 3}$ befassen. Es gilt

$$||A||_2 = \sup_{||x||_2=1} ||Ax||_2.$$

Um das Supremum zu bestimmen, diskretisieren wir die Einheitssphäre mittels Kugelkoordinaten und bestimmen das Maximum über diese diskrete Menge, also

$$B(0,1) \approx D_n := \left\{ \begin{pmatrix} \sin\frac{k\pi}{n}\cos\frac{\ell\pi}{n} \\ \sin\frac{k\pi}{n}\sin\frac{\ell\pi}{n} \\ \cos\frac{k\pi}{n} \end{pmatrix} \in \mathbb{R}^3 \ : \ k = 1, ..., n-1;\ \ell = 0, ..., 2n-1 \right\} \cup \left\{ \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}$$

und

$$||A||_2 = \sup_{||x||_2=1} ||Ax||_2 \approx \max_{x\in D_n} ||Ax||_2.$$

a) Schreiben Sie eine Funktion

```
y = myNorm2(A,n)
```

welche zu einer Matrix $A \in \mathbb{R}^{3\times 3}$ und $n \in \mathbb{N}$ die euklidische Norm berechnet, also

$$y = \max_{x\in D_n} ||Ax||_2.$$

Verwenden Sie **nicht** die Matlabroutine `norm` um auftretende euklidische Vektornormen zu berechnen.

b) Ergänzen Sie das Skript `test_myNorm2.m`, welches den relativen Fehler

$$\frac{||A||_2 - \texttt{norm2(A,2\^{}k)}}{||A||_2}$$

für $k = 1, ..., 10$ berechnet, um ein doppelt logarithmischen Plotbefehl, der diesen Fehler visualisiert. Veranschaulichen Sie an dieser Grafik, dass die Formel

$$\frac{||A||_2 - \texttt{norm2(A,2\^{}k)}}{||A||_2} \le \frac{\pi^2}{4}\left(\frac{1}{n}\right)^2$$

gilt. Achten Sie auf eine saubere Achsenbeschriftung, Titel und Legenden.

**Aufgabe 7** *(Drei-Term Rekursion der Legendrefunktionen)* (10 Punkte)

Die Legendrefunktionen spielen u.a. bei der Lösung von partiellen Differentialgleichungen eine wichtige Rolle. Es gibt Legendrepolynome $P_n$, auch Legendrefunktionen erster Art genannt, und Legendrefunktionen zweiter Art $Q_n$. Beide erfüllen (die gleiche) Drei-Term Rekursion für $n \geq 1$

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x) \quad \text{und} \quad (n+1)Q_{n+1}(x) = (2n+1)xQ_n(x) - nQ_{n-1}(x),$$

haben allerdings unterschiedliche Startwerte

$$P_0(x) := 1, \quad P_1(x) := x \qquad \text{und} \qquad Q_0(x) := \frac{1}{2}\log\left(\frac{x+1}{x-1}\right), \quad Q_1(x) := \frac{x}{2}\log\left(\frac{x+1}{x-1}\right) - 1.$$

Man kann zeigen, dass

$$\lim_{k\to\infty} \frac{Q_k(x)}{P_k(x)} = 0 \quad \forall\, x \in \mathbb{R} \setminus \{\pm 1\},$$

gilt, was die Begriffe dominante Lösung (hier $P_k$) und Minimallösung (hier $Q_k$) rechtfertigt. Vergleiche dazu [1]: Seite 5ff, Gleichung (2.2), Lemma 2.4.3. Wir wollen uns nun mit der Berechnung der Minimallösung beschäftigen.

a) Schreiben Sie eine Funktion

```
y = myLegendre_foreward(n,x,f0,f1)
```

die für gegebenes $n \in \mathbb{N}$, $(x_1, x_2, ..., x_k) \in \mathbb{R}^k$ und zwei *function handles* `f0, f1`, was die Startfunktionen darstellt, die Werte $f_p$ an den einzelnen Punkten $x_i$ für alle $0 \leq p \leq n$ berechnet. Der Rückgabewert $y \in \mathbb{R}^{n+1\times k}$ ist also eine Matrix, in der in jeder Zeile die Werte $(f_i(x_1), f_i(x_2), ..., f_i(x_k))$ stehen.

b) Testen Sie ihre Funktion mit Hilfe des Skriptes `test_myLegendre_foreward.m` und erweitern Sie das Skript derart, dass jeweils $P_n$ und $Q_n$ für $n \in \{0,1,2,3,4,5\}$ in separaten Figures geplottet werden. *Hinweis:* `subplot`

c) Schreiben Sie ein Skript `showQ25.m`, welches für $x \in [0, 2.5]$ die Funktion $Q_{25}$ plottet (Komplexwertige Resultate ignorieren wir an dieser Stelle). Was fällt auf?

d) Schreiben Sie eine Funktion

```
y = myLegendre(n,x)
```

die für $n \in \mathbb{N}$, $x \in \mathbb{R}^k$ jeweils die Werte der Legendrefunktion zweiter Art $Q_p$ für $0 \leq p \leq n$ berechnet, indem sie innerhalb eines Intervalls $(-r, r)$ die Funktion `myLegendre_foreward(n,x,f0,f1)` verwenden und außerhalb die gegebene Funktion `myLegendre_backward(n,x,f0,f1)`. Wählen Sie

$$r = \sqrt{1 + \max\left\{1, \frac{4.5}{(n+1)^{1.17}}\right\}} \qquad \text{(vgl. [1], Gleichung (3.4))}$$

und ändern Sie das Skript aus c) derart ab, dass Sie die Funktion `myLegendre(n,x)` benutzen und speichern Sie dieses Skript unter `test_myLegendre.m`. Was fällt nun auf?

*Hinweis: Beachten Sie, dass die Routine* `myLegendre_backward` *nur für $x \in \mathbb{R}$ funktioniert.*

# Literatur


[1] Pascal F. Heiter, *Stable Implementation of Three-Term Recurrence Relations*. Bachelorarbeit, Universität Ulm, Institut für Numerische Mathematik, 2010.