Mathematisches Institut
Prof. Dr. Achim Schädle
Lennart Jansen

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

11. Februar 2015

Computergestützte Mathematik zur Linearen Algebra – Hauptklausur

Wichtige Hinweise zur Klausur

- In Ihrem Home-Verzeichnis finden Sie die Datei `WerBinIch.txt` sowie die Dateien `aufgabe1a.m`, `aufgabe1b.m`, `aufgabe2a.m`, `aufgabe2b.m`, `aufgabe3.m`, `MChol.m` `aufgabe4.m`, `aufgabe5a.m`, und `aufgabe5b.m` in die sie ihre Lösungen schreiben soll.
- **Ergänzen Sie zuerst die Datei `WerBinIch.txt` mit Ihrem Namen, Vornamen, usw.**
- Als Hilfsmittel dürfen Sie ein von Ihnen selbst beschriebenes DIN-A4-Blatt verwenden. In Ihrem Home-Verzeichnis finden Sie darüber hinaus die Dateien der Vorlesungen (pdf und m) von der Website (Matlabdateien und Vorlesungsunterlagen).
- Es werden nur Lösungsvorschläge gewertet, die in Dateien mit dem jeweils in der Aufgabe angegebenen Namen in Ihrem Home-Verzeichnis gespeichert sind. Speichern Sie daher in kurzen Abständen Ihre Lösungen, um ggf. den Verlust von Daten zu vermeiden, falls Matlab einmal abstürzt.
- Anders als bei den Übungsaufgaben werden bei den Klausuren in der Regel keine Testfälle vorgegeben. Diese müssen Sie sich ggf. selbst konstruieren, um Ihre Implementierungen zu überprüfen.
- Nicht lauffähiger Code kann in der Regel höchstens die Hälfte der Punkte erzielen. Kommentieren Sie ihren Code sinnvoll, so dass nachvollzogen werden kann, was Sie tun wollten.
- Es werden nur Lösungsvorschläge gewertet, bei denen der Lösungsweg (mit Matlab) klar zu erkennen ist.
- Die vorgegebene Form einer zu implementierenden Funktion muss ggf. eingehalten werden (das heißt keine zusätzlichen Eingaben oder Rückgaben).
- Fehlerabfragen vom Typ “überprüfen Sie, ob die Eingabematrix symmetrisch ist” oder “geben Sie einen Fehler aus, falls. . . ” werden in der Aufgabenstellung explizit gefordert. Fehlermeldungen sollen das aufgetretene Ereignis sinnvoll charakterisieren.
- Schreiben Sie ihre Matlab-Skripte so, dass sie auch nach einem `clear all` noch lauffähig sind.
- Die Verteilung der Punkte auf die einzelnen Aufgaben ist angegeben. Zum Bestehen der Klausur sind 25 Punkte hinreichend.
- Sie haben 120 Minuten Zeit für die Bearbeitung der Klausur.

Aufgabe 1: (4 + 3 + 3 Punkte)

Für einen Parameter $k \in \mathbb{N}_0$ ist die reellwertige Funktion f_k definiert durch

$$f_k(x) = \frac{\left(x + \sqrt{x^2-1}\right)^k - \left(x - \sqrt{x^2-1}\right)^k}{2\sqrt{x^2-1}}.$$

(a) Schreiben Sie eine MATLAB-Funktion

```
function y = aufgabe1a(x,k)
```

in die Datei `aufgabe1a.m`, die zu einem Parameter $\mathtt{k} \in \mathbb{N}_0$ und $\mathtt{x} \in \mathbb{R}^n$ die Funktion auswertet und einen Vektor $\mathtt{y} = f_{\mathtt{k}}(\mathtt{x}) \in \mathbb{R}^n$ ausgibt. Überprüfen sie, dass erstens die Eingabe `k` eine ganze nicht negative Zahl und zweitens `x` ein Vektor und keine Matrix ist. Geben Sie jeweils eine aussagekräftige Fehlermeldung aus, falls dies nicht der Fall ist.

(b) Schreiben sie ein MATLAB-Skript `aufgabe1b.m`, welches für $\mathtt{k} = 1$, $\mathtt{k} = 2$, $\mathtt{k} = 3$, $\mathtt{k} = 4$ und $\mathtt{k} = 5$ die Funktionen $f_{\mathtt{k}}$ in einem gemeinsamen Fenster auf dem Intervall $(-1, 1)$ graphisch darstellt.

(c) Wählen sie für jedes `k` eine andere Farbe. Ergänzen sie das Skript so, dass eine Legende, die die einzelnen Funktionen beschreibt, in ihrer Graphik angezeigt wird. Platzieren sie die Legende rechts außerhalb der Achsen.

Aufgabe 2: (5+3+2 Punkte)

Gegeben sind zwei Polynome a_ℓ und b_ℓ in x, die für $\ell = 1, \dots$ rekursiv durch

$$\begin{aligned} a_0(x) &= 1, \quad a_1(x) = 2 + x/2, \quad a_{\ell+1}(x) = (2+x)a_\ell(x) + (x/2)^2 a_{\ell-1}(x) \\ b_0(x) &= 0, \quad b_1(x) = 1, \qquad\qquad b_{\ell+1}(x) = (2+x)b_\ell(x) + (x/2)^2 b_{\ell-1}(x). \end{aligned}$$

definiert sind.

(a) Schreiben sie eine Funktion

```
function y = aufgabe2a(x,k)
```

in die Datei `aufgabe2a.m`, die zu gegebenem $\mathtt{x} \in \mathbb{R}^n$ und einer natürlichen Zahl `k` die Werte der rationalen Funktion $p_k(x) := a_k(x)/b_k(x)$ in einen Vektor $\mathtt{y} \in \mathbb{R}^n$ ausgibt.

(b) Schreiben sie ein MATLAB-Skript `aufgabe2b.m`, das die Funktion p_k in einem Fenster auf dem Intervall $[0.1, 20]$ für $\mathtt{k} = 2$ und $\mathtt{k} = 6$ graphisch darstellt. Verwenden sie für `x` $n = 30$ äquidistante Werte zwischen 0.1 und 20 und beschränken sie den y-Achsenabschnitt auf das Intervall $[1, 10]$.

(c) Zeichnen sie die $p_{\mathtt{k}}$ für $\mathtt{k} = 2$ mit einer roten durchgezogenen Linie und für $\mathtt{k} = 6$ mit einer grünen gestrichelten Linie.

Aufgabe 3: (10 Punkte)

Für eine symmetrische, positiv definite $n \times n$ Matrix A kann man eine Zerlegung $A = LL^T$ mit einer linken unteren Dreicksmatrizen L berechnen, die sogenannte Choleskyzerlegung. (Matlabbefehl `L = chol(A,'lower')`) Ein Verfahren dazu ist durch den folgenden Pseudocode gegeben.

Eingabe: A
Ausgabe: L, flag
for j *von* 1 *bis* n **do**
  $A_{jj} = \sqrt{A_{jj}}$ % falls A positiv definit ist, sind die Diagonaleinträge positiv
  for i *von* $j+1$ *bis* n **do**
    $A_{ij} = A_{ij}/A_{jj}$
  endfor
  for k *von* $j+1$ *bis* n **do**
    for i *von* j *bis* n **do**
      $A_{ik} = A_{ik} - A_{ij}A_{kj}$
    endfor
  endfor
endfor

L ist der linke untere Teil des überschriebenen A

In der Datei `MChol.m` wurden die innneren "for" Schleifen vektorisiert. Es haben sich zudem sechs Fehler eingeschlichen. Finden und korrigieren sie diese Fehler. Schreiben sie in die Datei `aufgabe3.m` ein Skript, das ihre korrigierte Choleskyzerlegung an geeignet gewählten Beispielen testet.

Handschriftliche Korrekturen in der unten abgedruckten Datei werden bei der Bewertung nicht berücksichtigt. Eine schreibgeschützte Kopie von `MChol.m` finden sie zudem in der Datei `MCholKopie.m`

```
function [L,flag] = MChol(A)
 %  Eingabe: A  eine symmetrische positiv definite n x n Matrix.
 %  Ausgabe: L  eine untere Dreiecksmatrix, so, dass A = L*L'
 %           flag Statusmeldung  flag = k > 0 Fehler im k-ten Schritt
 %                                    = 0     erfolgreiche Zerlegung

[m,n] = size(A);
flag = 1;

if m %= n
   error('Die Matrix muss quadratisch sein')
end

for j = 1:n,
  %    Abbruch falls A nicht positiv definit ist
  if (A(j,j) <= -1), flag = j; return; end
  % Berechnung der Diagonale und der darunterliegenden Spalte
  A(j,j) = sqrt(A(j,j));
  A(j+1:n,j) = A(j+1+n,j)/A(j,j);
  % Aktualisiere den restlichen linken unteren Block
  for k = j+1:n,
    A(j:n,k) = A(j:n,k) - A(j:n,j)*A(k,j);
  end
end
L = trill(A);
```

Aufgabe 4: (5+2+5 Punkte)

Gegeben sind die folgenden $M = 10$ Datenpunkte (x_m, z_m, y_m) für $m = 1, \ldots, M$

y	2.0	1.5	3.2	-1.8	-0.8	-0.9	2.6	-1.2	2.9	2.2
x	0	1	1.5	2.0	-1.2	1.8	-1.0	-2.0	-1.2	-0.5
z	0.1	0.1	-1.5	2.0	-1.5	1.0	2.0	-1.0	1.3	0.5

Tabelle 1: Daten zur Aufgabe 4, (vgl auch `aufgabe4daten.mat`).

Es wird angenommen, dass diese Daten für drei Parameter a, b und c folgender Gesetzmäßigkeit gehorchen

$$g(x, z) = a + b\cos(x + z) + c(xz)$$

Schreiben sie ein Matlabskript `aufgabe4.m`, das

(a) Parameter a, b, und c bestimmt, so, dass die Funktion g in den Punkten (x_m, z_m) $m = 1, \ldots, M$ im Sinne der kleinsten Fehlerquadrate die Werte y_m bestmöglich approximiert und

(b) den Fehler

$$f = \left(\sum_{m=1}^{M} |y_m - g(x_m, z_m)|^2 \right)^{\frac{1}{2}}$$

berechnet.

(c) Stellen sie die Daten als Punkte im $\mathbb{R}^3$ und für die gefundenen Parameter a, b und c die durch g definierte Fläche für $(x, z) \in [-2, 2] \times [-2, 2]$ graphisch dar.

Aufgabe 5: (6 + 2 Punkte)

Gegeben sei die Singulärwertzerlegung einer regulären Matrix $A \in \mathbb{R}^{n\times n}$, $A = USV^T$. Diese Singulärwertzerlegung soll nun verwendet werden, um ein lineares Gleichungssystem $Ax = b$ zu lösen.

(a) Schreiben sie eine Matlabfunktion

```
function x = aufgabe5a(U,S,V,b)
```

die ohne Verwendung des Backslashoperators zum Lösen linearer Gleichungssysteme (Matlabbefehl \) die Lösung x berechnet. Divisionen mit Skalaren dürfen sie natürlich verwenden.

(b) Verwenden sie die im Matlabskript `aufgabe5b.m` gegebenen U, V, S und b, um ihre Implementierung in `aufgabe5a.m` zu testen.