Die Mathematik ist eine Mausefalle. Wer einmal in dieser Falle gefangen sitzt, findet selten den Ausgang, der zurück in seinen vormathematischen Seelenzustand leitet.

Egmont Colerus

(1888-1939, österreichischer Schriftsteller)

UNIVERSITÄT DES SAARLANDES

FR 6.1 Mathematik

Prof. Dr. S. Rjasanow

L. Huwig, M.Sc.

3. Übung zur Vorlesung
Praktische Mathematik
im Sommersemester 2016

Abgabe: Donnerstag, den 12.05.2016 vor der Vorlesung.

Wegen des Feiertags fällt die Mo-Übungsgruppe in der Pfingstwoche aus. Die davon betroffenen Studierenden werden daher gebeten an den Di- oder Mi-Gruppen teilzunehmen. Die Abgabe der Blätter erfolgt unverändert.

Aufgabe 3.1. (2 + 2.5 = 4.5 Punkte)

(a) Sei $L \in \mathbb{R}^{n\times n}$ eine reguläre, linke, untere Dreiecksmatrix. Zeigen Sie, dass $A = LL^T$ symmetrisch und positiv definit ist.

(b) Sei $b \in \mathbb{R} \setminus \{1\}$ und

$$A = \begin{pmatrix} 4 & 2 & 6 \\ 2 & b & 4 \\ 6 & 4 & \frac{8b}{b-1} \end{pmatrix}$$

Berechnen Sie die Cholesky-Zerlegung von A. Für welche b ist A positiv definit?

Aufgabe 3.2. (2 + 3 = 5 Punkte)

Betrachtet wird die Auswertung der Funktionen

$$\begin{aligned} f_1 &: \quad \mathbb{R}_+^2 \to \mathbb{R}_+, \quad f_1(x_1, x_2) = x_1^2 - x_2^2 \\ f_2 &: \quad \mathbb{R}_+^2 \to \mathbb{R}_+, \quad f_2(x_1, x_2) = x_1^2 + x_2^2 \\ f_3 &: \quad \mathbb{R}_+^2 \to \mathbb{R}_+, \quad f_3(x_1, x_2) = (x_1 + x_2) \cdot (x_1 - x_2) \end{aligned}$$

(a) Berechnen Sie die relativen Konditionszahlen κ_{rel} der Funktionen f_1 und f_2 bzgl. der euklidischen Norm für Vektoren bzw. der Frobeniusnorm für Matrizen.
Für welche x_1 bzw. x_2 ist die Auswertung der jeweiligen Funktion schlecht konditioniert?

(b) Die Auswertung der Funktion erfolgt über die folgenden Algorithmen

$$\text{für } f_1: \quad \begin{aligned} a_1 &= x_1 \cdot x_1 \\ b_1 &= x_2 \cdot x_2 \\ c_1 &= a_1 - b_1 \end{aligned} \qquad (1) \qquad\qquad \text{für } f_3: \quad \begin{aligned} a_2 &= x_1 + x_2 \\ b_2 &= x_1 - x_2 \\ c_2 &= a_2 \cdot b_2 \end{aligned} \qquad (2)$$

Nehmen Sie an, dass in jedem Schritt ein Fehler der Form $\tilde{y} = y(1+\epsilon)$ entsteht und untersuchen Sie die Algorithmen (1) und (2) auf ihre Stabilität.
Hinweise:

$$\epsilon_1\epsilon_2 \approx 0, \qquad \frac{1+\epsilon_1}{1+\epsilon_2} \approx 1 + \epsilon_1 - \epsilon_2, \qquad (1+\epsilon_1)(1+\epsilon_2) \approx 1 + \epsilon_1 + \epsilon_2$$

Aufgabe 3.3. (2.5 + 2.5 = 5 Punkte)

(a) Sei $A \in \mathbb{R}^{n\times n}$ symmetrisch positiv definit. Zeigen Sie, dass für die Spektralkonditionszahl gilt

$$\kappa_2(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)},$$

wobei $\lambda_{\max}(A)$ der größte und $\lambda_{\min}(A)$ der kleinste Eigenwert von A ist.

(b) Sei $A \in \mathbb{R}^{n\times n}$ regulär und $U \in \mathbb{R}^{n\times n}$ unitär. Zeigen Sie, dass $\kappa_2(AU) = \kappa_2(A)$ gilt.

Aufgabe 3.4. (Programmieraufgabe, 2 + 2 + 2 + 2 = 8 Punkte)
In dieser Aufgabe soll die Cholesky-Zerlegung für symmetrisch, positiv definite Matrizen implementiert und Anschließend ein Gleichungssystem $Ax = b$ mit $A \in \mathbb{R}^{n\times n}$ und $x, b \in \mathbb{R}^n$ gelöst werden. Auf der Internetseite der Veranstaltung finden Sie ein Grundgerüst für Ihr Programm. Auf der Kommandozeile kompilieren Sie Ihr Programm mit

```
gcc -Wall main.c CC.c basic_LinAlg.c -lm
```

(a) Um Speicherplatz zu sparen, speichert man eine symmetrische Matrix nicht komplett ab. Wegen der Symmetrie genügt es lediglich eine untere Dreiecksmatrix mit den Einträgen auf und unterhalb der Diagonalen abzuspeichern. Unsere kleine Bibliothek `basic_LinAlg` wurde um diese Funktionalität erweitert und in ihrer Dokumentation finden Sie eine genauere Erläuterung. Ergänzen Sie den fehlenden Quelltext in den Funktionen

```
long symmat_FrobeniusNorm(...);
void symmat_ausgeben(...);
void symmat_ausgeben_dreieck(...);
void symmat_vektor_mult(...);
```

in der Datei `basic_LinAlg.c`.

(b) Betrachten Sie nun die Datei `CC.c` mit der zugehörigen Header-Datei `CC.h`. Schreiben Sie eine Funktion, die die Cholesky-Zerlegung einer Matrix berechnet. Die ursprüngliche Matrix soll dabei durch C überschrieben werden. Schreiben Sie weiter Funktionen für die Vorwärts- und Rückwärtssubstitution.

(c) Schreiben Sie nun in `main.c` ein Testprogramm, das diese Funktionen verwendet, um ein Gleichungssystem zu löst. Testen Sie es anhand des folgenden linearen Gleichungssystems

$$Ax = b \quad \text{mit} \quad A = \begin{pmatrix} 1 & 5 & 2 & 0 \\ 5 & 29 & 16 & 2 \\ 2 & 16 & 22 & 15 \\ 0 & 2 & 15 & 33 \end{pmatrix} \quad \text{und} \quad b = \begin{pmatrix} -4 \\ -30 \\ -5 \\ 67 \end{pmatrix}$$

und geben Sie auf dem Bildschirm jeden Rechenschritt aus, also die Dreiecksmatrix C, den Vektor Cx sowie die Lösung x.
Hinweis: Die Lösung ist

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 5 & 2 & 0 & 0 \\ 2 & 3 & 3 & 0 \\ 0 & 1 & 4 & 4 \end{pmatrix} \quad , \quad x = \begin{pmatrix} 5 \\ -1 \\ -2 \\ 3 \end{pmatrix}.$$

(d) Ergänzen Sie Ihr Programm so, dass die Frobeniusnorm von A und zur Probe Ax berechnet wird. Auf diese Weise kann Ax mit der gegebenen rechten Seite b des Gleichungssystems verglichen werden kann. *Hinweis: Da die Matrix A mit C überschrieben wird, müssen Sie eine Kopie von A anlegen oder die Matrix nochmals einlesen.*