



## Numerische Mathematik I

### 8. Übung

**Die Aufgaben 1+2 werden besprochen am 7.12.2016**

#### 1. Penrose-Axiome

Beweisen Sie Satz III.12 aus der Vorlesung, also:

**Satz:** Die Pseudoinverse  $A^+ \in \mathbb{R}^{n \times m}$  ist eindeutig charakterisiert durch folgende vier Eigenschaften:

- (i)  $(A^+A)^T = A^+A$
- (ii)  $(AA^+)^T = AA^+$
- (iii)  $A^+AA^+ = A^+$
- (iv)  $AA^+A = A$ .

#### 2. Programmieraufgabe: Nichtlineare Ausgleichsrechnung

(Abgabe bis Mi, 7.12.2016, 10:00 Uhr)

Gegeben sei das Modell einer gedämpften Schwingung

$$y(t; x_1, x_2, x_3, x_4) = x_1 e^{-x_2 t} \sin(x_3 t + x_4),$$

mit den Parametern  $x_1, \dots, x_4$ . Es seien die Daten in der folgenden Tabelle für  $b_i \approx y(t; x_1, x_2, x_3, x_4)$ ,  $i = 1, \dots, 10$ , gegeben.

|       |       |       |       |        |        |       |       |        |        |       |
|-------|-------|-------|-------|--------|--------|-------|-------|--------|--------|-------|
| $t_i$ | 0.1   | 0.3   | 0.7   | 1.2    | 1.6    | 2.2   | 2.7   | 3.1    | 3.5    | 3.9   |
| $b_i$ | 0.558 | 0.569 | 0.176 | -0.207 | -0.133 | 0.132 | 0.055 | -0.090 | -0.069 | 0.027 |

- (a) Formulieren Sie das nichtlineare Ausgleichsproblem.
- (b) Schreiben Sie ein Skript `schwingung_ZZ.m` zur Lösung des nichtlinearen Ausgleichsproblems unter Verwendung der Matlab-Funktion `lsqnonlin` mit den Optionen

```
options=optimset('Display','iter','Jacobian','on',  
'Algorithm','levenberg-marquardt','TolX',1e-8,'TolFun',1e-6);
```

D.h. Programmieren Sie also eine Funktion `myfun_ZZ`, die sie an `lsqnonlin` übergeben, wie folgt:

```
function [F,J] = myfun_ZZ(x)
F = ...           % Zielfunktionswert in x, F=y(t_i)-b_i
if nargin > 1    % Zwei Rueckgabewerte
    J = ...      % Jacobi-Matrix von F in x
end
```

Plotten Sie anschließend die Daten mit ihrer gefitteten Lösungskurve. Speichern Sie diesen als `plot_ZZ.jpg`.

Abgabe der Programmieraufgabe:

- Abgabe per email an `armin.rund@uni-graz.at`
- Betreff: Programmieraufgabe Schwingung
- In email bitte die Namen und Matrikelnr aller beteiligten Studierenden des Teams aufführen
- Abzugeben sind die Funktion, Ihr Test-Skript und der Plot mit vorgegebenen Namen unter Verwendung der Teamnummer

3. **Programmieraufgabe:  $QR$ -Zerlegung (Householder)**  
(Abgabe bis Mo, 12.12.2016, 23:59 Uhr)

- (a) Programmieren Sie die  $QR$ -Zerlegung mit Householder für eine allgemeine Matrix  $A \in \mathbb{R}^{m \times n}$  und rechte Seite  $b \in \mathbb{R}^m$  in MATLAB mit dem Syntax

```
function [...] = qr_zerlegung_ZZ(A,b)
```

wobei  $ZZ$  Ihre Teamnummer ist.

- (b) Entwerfen und programmieren Sie eine zweite function

```
function [x] = qr_solve_ZZ(...)
```

welche anschließend die Pseudo-Inverse  $x = A^+b$  gemäß dem Algorithmus aus der Vorlesung (siehe auch Deuffhard, Algorithmus 3.20) berechnet. Benutzen Sie dazu entweder Ihre Cholesky-Zerlegung aus Übung 5, oder die MATLAB-Variante.

- (c) Passen Sie das gegebene Test-Skript `qr_ZZ.m` an und benennen Sie es um.

Abgabe der Programmieraufgabe:

- Abgabe per email an `armin.rund@uni-graz.at`
- Betreff: Programmieraufgabe QR
- In email bitte die Namen und Matrikelnr aller beteiligten Studierenden des Teams aufführen
- Abzugeben sind alle Funktionen und Ihr Test-Skript mit vorgegebenen Namen unter Verwendung der Teamnummer
- Der schnellste Code erhält einen Sonderpunkt.