Dr. Steffen Basting
Dipl.-Technomath. Christopher Basting
Numerische Mathematik für Physiker und Ingenieure

Sommersemester 2017
Übungsblatt 4

Abgabe der **Theorieaufgaben bis Donnerstag, 01.06.2017, 16:00 Uhr vor der Vorlesung** (einzeln oder zu zweit) **in den Briefkasten Ihrer Übungsgruppe**. Bitte vermerken Sie auf Ihrer Abgabe Ihre Übungsgruppe sowie Name und Matrikelnummer. Heften Sie mehrere Blätter zusammen.
Abgabe der **Programmieraufgaben bis Donnerstag, 01.06.2017, 16:00 Uhr digital im Moodle-Arbeitsraum der Veranstaltung**. Bei Abgabe zu zweit bitte nur einmal einreichen und im Kommentar den Namen und E-Mail Adresse des/der Koautors/Koautorin nennen. Die Aufgaben sollen mit MATLAB bzw. OCTAVE gelöst werden.

Aufgabe 4.1 (Ausgleichsrechnung | 4 Punkte)

Unter Einwirkung der Schwerkraft fliegen geworfene Körper auf Parabelbahnen. In dieser Aufgabe wollen wir den *senkrechten Wurf* nach oben betrachten, bei dem ein Körper mit einer Geschwindigkeit v_y nach oben geworfen wird. Die Wurfhöhe $y(t)$ in Abhängigkeit der Zeit t folgt dabei dem Zusammenhang

$$y(t) = v_y t - \frac{1}{2} g t^2,$$

wobei $g \approx 9.81 \frac{m}{s^2}$ die Erdbeschleunigung bezeichnet. In einem Experiment wird zur Bestimmung dieses Werts ein Gegenstand in die Höhe geworfen, und anschließend die Wurfhöhe y zu unterschiedlichen Zeitpunkten gemessen. Es ergeben sich folgende Werte:

i	1	2	3	4	5	6
t_i [s]	0.2	0.5	1	1.5	2	2.5
y_i [m]	2.8	6.3	10	11.5	10.3	6.8

Bestimmen Sie die Abwurfgeschwindigkeit v_y und den (experimentellen) Wert für die Erdbeschleunigung g durch Lösen des Least-Squares-Problems (2.5) aus der Vorlesung.

Hinweis: Zum Lösen der Normalengleichung dürfen Sie einen Taschenrechner (oder MATLAB) verwenden.

Aufgabe 4.2 (Ausgleichsrechnung | 4 Punkte)

Ein periodisches Signal soll durch eine Funktion der Form

$$y(t) = a_1 + a_2 \cdot \sin(t) + a_3 \cdot \sin(2t)$$

approximiert werden. Für das Signal werden folgende Werte gemessen:

i	1	2	3	4
t_i	0	$\frac{\pi}{4}$	$\frac{\pi}{2}$	$\frac{3\pi}{2}$
y_i	1.1	3.9	11.2	-9.2

a) Wie lautet das lineare Ausgleichsproblem zur Bestimmung der Koeffizienten $a_1, \ldots, a_3$?

b) Lösen Sie das Ausgleichsproblem über die Normalengleichung und berechnen Sie anschließend die Norm des Fehlers

$$\|y(\mathbf{t}) - \mathbf{y}\| = \left(\sum_{i=1}^{4} |y(t_i) - y_i|^2 \right)^{1/2} .$$

Hinweis: Zum Lösen der Normalengleichung und zum Berechnen des Fehlers dürfen Sie einen Taschenrechner (oder MATLAB) verwenden.

Programmieraufgabe 4.1 (Stückweise Interpolation | 4 + 6 Punkte)

In dieser Aufgabe wollen wir das Verhalten der stückweisen linearen Interpolation mit der polynomiellen Interpolation (in der Newtonschen Darstellung) vergleichen. **Zur Bearbeitung dieser Aufgabe dürfen Sie den Code aus Programmieraufgabe 3.1 wiederverwenden.**

a) Schreiben Sie eine Funktion `[yEval]=myPiecewiseInterpol(x,f,xEval)`, die zu gegebenen Stützstellen $x_0, \dots, x_n$ und zugehörigen Werten $f_0, \dots, f_n$ die Werte der stückweisen linearen Interpolation an den Knoten des Vektors `xEval` berechnet. Beispielsweise sollte Ihr Code folgende Ausgabe erzeugen:

```
>> x = [0 1 2]; y = exp(x); xEval = 0 : 0.5 : 2;
>> c = myPiecewiseInterpol(x,y,xEval)

c =

    1.0000    1.8591    2.7183    5.0537    7.3891
```

(In diesem Beispiel wurden drei Stützstellen $x_i = 0, 1, 2$ sowie Werte $y_i = \exp(x_i)$ vorgegeben, und die Werte der stückweise linear Interpolierenden an den durch `xEval` definierten x-Werten $0, 0.5, 1, 1.5, 2$ berechnet.)

b) Es sollen nun die beiden Funktionen

$$\exp(x) \quad \text{und} \quad \frac{1}{1+25x^2} \quad \text{auf} \quad I = [-1, 1]$$

durch stückweise lineare Interpolation sowie polynomielle Interpolation approximiert werden.

- Erstellen Sie ein Skript `myPiecewiseInterpolTest()`, welches zu äquidistanten Stützstellen

  $$x_i = -1 + \frac{2i}{n}, i = 0, \dots, n$$

  für $n = 2, 4, 8, 16, \dots, 128$ zur stückweisen Interpolation $S_h(f)$ sowie polynomiellen Interpolation $I_h(f)$ auf einem feinen Gitter $\Delta = \{-1 + 2j/m,\ j = 0, \dots, m\}$ für $m = 100$ die Fehler

  $$E_{S_h} := \max_{\xi \in \Delta} |f(\xi) - S_h(f)(\xi)| \quad \text{und} \quad E_{I_h} := \max_{\xi \in \Delta} |f(\xi) - I_h(f)(\xi)|$$

berechnet. **Wählen Sie in Ihren Experimenten als maximales n für die polynomielle Interpolation $I_h(f)$ den Wert $n_{max} = 16$, da es bei höheren Polynomgraden zu Instabilitäten bei der Polynomauswertung kommt**. Für die stückweise Interpolation $S_h(f)$ können und sollen Sie zur Fehlerberechnung auch Werte $n > n_{max}$ berücksichtigen. Ihr Skript soll beide Interpolierende sowie die exakte Funktion in einem Plot darstellen und abspeichern können (Darstellung soll ebenfalls nur für $n \leq n_{max}$ erfolgen).

- Vergleichen Sie das Fehlerverhalten der beiden Interpolierenden E_{S_h} und E_{I_h} in Bezug auf n für die Funktion $f(x) = \exp(x)$. Was beobachten Sie? Welches Verfahren konvergiert schneller für wachsendes n (solange $n \leq n_{max}$), und warum?
- Wiederholen Sie das Experiment für die Funktion $f(x) = \frac{1}{1+25x^2}$. Was beobachten Sie nun? (Begründung!)