Algorithmische Mathematik II

Sommersemester 2012
Prof. Dr. Beuchler
Markus Burkow

universitätbonn

Übungsblatt 9. Abgabe am **Mittwoch vor der Vorlesung (bis 10:15 Uhr)**.

Aufgabe 1. (Matrix Zerlegungen)

a) Man zeige, daß

$$A = \begin{bmatrix} 0 & 2 \\ 2 & 3 \end{bmatrix} \in \mathbb{R}^{2\times 2}$$

keine L–R Zerlegung besitzt (d.h. es gibt keine untere Dreiecksmatrix L und keine obere Dreiecksmatrix R mit $A = L \cdot R$).
Man bestimme eine Permutationsmatrix P, eine obere Dreiecksmatrix R und eine untere Dreiecksmatrix L mit $P \cdot A = L \cdot R$.

b) Es sei

$$A = \begin{bmatrix} 1 & 5 & 0 & 0 \\ 5 & 29 & 12 & 0 \\ 0 & 12 & 45 & 21 \\ 0 & 0 & 21 & 65 \end{bmatrix} \in \mathbb{R}^{4\times 4}$$

Man bestimme die Cholesky Zerlegung von A und berechne mit Hilfe dieser Zerlegung die Determinante von A.

(5 Punkte)

Aufgabe 2. (Positive Definitheit)

Für welche $a \in \mathbb{R}$ ist

$$\mathbf{A} = \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -a & 0 \\ 0 & -a & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}$$

positiv definit ?

(5 Punkte)

Aufgabe 3. (Tschebyscheff-Polynome)

Die aus der Vorlesung bekannten Tschebyscheff-Polynome lauten für $t \in [-1, 1]$

$$C_k(t) = \cos(k \cdot \arccos(t)).$$

a) Berechnen Sie $C_0(t)$ und $C_1(t)$ und beweisen Sie die Rekursionsformel

$$C_{k+1}(t) = 2tC_k(t) - C_{k-1}(t) \text{ für } k = 1, 2, \ldots$$

b) Beweisen Sie, dass der Höchstkoeffizient des n-ten Tschebyscheff–Orthogonalpolynoms $2^{n-1}x^n$ ist ($n \geq 1$).

c) Zeigen Sie: die Nullstellen t_i, $1 \le i \le k$, des Tschebyscheff–Polynoms $C_k(t)$ sind für $k \ge 1$ gegeben als

$$t_i = \cos\left(\pi \frac{2i-1}{2k}\right).$$

d) i sei die imaginäre Einheit. Beweisen Sie die Identität

$$\begin{aligned}\frac{1}{2}\left[\left(t+\sqrt{t^2-1}\right)^k + \left(t-\sqrt{t^2-1}\right)^k\right] &= \frac{1}{2}\left[\left(t+i\sqrt{1-t^2}\right)^k + \left(t-i\sqrt{1-t^2}\right)^k\right] \\ &= \cos(k \cdot \arccos(t)).\end{aligned}$$

Bemerkung: Diese Polynomdarstellung der Tschebyscheff-Polynome gilt auch für $t \in \mathbb{R}$.

e) Zeigen Sie

$$C_k(t) = \sum_{j=0}^{\lfloor k/2 \rfloor} (-1)^j \binom{k}{2j} t^{k-2j}(1-t^2)^j.$$

f) Sei $t = \frac{r^2+1}{r^2-1}$ mit $r > 1$. Zeigen Sie, dass

$$C_k(t) = \frac{1}{2}\left[\left(\frac{r+1}{r-1}\right)^k + \left(\frac{r-1}{r+1}\right)^k\right].$$

Insbesondere gilt

$$\frac{1}{C_k\left(\frac{r^2+1}{r^2-1}\right)} < 2\left(\frac{r-1}{r+1}\right)^k.$$

g) Zeigen Sie die Orthogonalitätsrelation

$$\int_{-1}^{1} \frac{1}{\sqrt{1-t^2}}\, C_k(t)\, C_l(t)\, dx = \begin{cases} 0 & \text{für } k \neq l \\ \frac{\pi}{2} & \text{für } k = l \neq 0 \\ \pi & \text{für } k = l = 0. \end{cases}$$

h) Zeigen Sie, dass die C_k auf $[-1, 1]$ der Differentialgleichung

$$(1-t^2)C_k''(t) - tC_k'(t) + k^2 C_k(t) = 0$$

genügen.

(15 Punkte)

Aufgabe 4. (Interpolationsproblem)

Gegeben sei das lineare Gleichungssystem

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \\ 1 & 3 & 9 & 27 \\ 1 & 4 & 16 & 64 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 26 \\ 58 \\ 112 \end{bmatrix}.$$

a) Formulieren Sie ein äquivalentes Polynom–Interpolationsproblem und bestimmen Sie dessen Lösung.
(Beachten Sie, daß die Koeffizientenmatrix eine Vandermonde-Matrix ist).

b) Lösen Sie das lineare Gleichungssystem.

(5 Punkte)

Programmieraufgabe 1. Schreiben Sie folgende Programme:

(a) Den Neville–Algorithmus zur Auswertung eines Interpolationspolynoms an einer beliebigen Stelle x für gegebene (x_i, f_i), $0 \leq i \leq n$.

(b) Den Newton–Algorithmus zur Bestimmung der Polynomkoeffizienten in Newton–Darstellung. Hierbei ist die Lösung des Gleichungssystems S.186 Skript zu berechnen. Bereits erstellte LGS-Löser sollen benutzt werden.

(c) Ein Verfahren zur direkten Auswertung des Interpolationspolynoms in Newton–Darstellung an einer beliebigen Stelle x.

(d) Das dividierte Differenzenverfahren zur Bestimmung der Polynomkoeffizienten in Newton–Darstellung.

(e) Das Horner–Schema zur Auswertung des Interpolationspolynoms in Newton–Darstellung an einer beliebigen Stelle x.

Betrachten Sie als Beispiel die Gauß'sche Hutfunktion

$$f(x) = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{x^2}{2}}$$

im Intervall $[-3, 3]$ mit $n = 1, 3, 5, 9, 17$ und 33 Stützstellen.

(f) Werten Sie zur Überprüfung Ihrer Programme das Interpolationspolynom für $f(x)$ an den Stellen x=0.02, $x = 0.2$ und $x = 2$ einmal mit dem Neville–Algorithmus, einmal mit dem Newton–Algorihmus und dem direkten Verfahren und einmal mit dividierten Differenzen und dem Horner–Schema aus.

(g) Plotten Sie $f(x_k) - P(x_k)$ gegen n (Aufwand zu Genauigkeit) in einen gemeinsamen Graphen, wobei Sie Punkte für gleiches x_k miteinander verbinden. Verwenden Sie für beide Axen eine logarithmische Skala. Bestimmen Sie jeweils die Konvergenzrate.

(h) Vergleichen Sie den Rechenaufwand der Programme (a), (b)+(c) und (d)+(e) für den Fall, daß die Zahl der Stützstellen anwächst und für den Fall, daß die Zahl der Auswertungen anwächst.

(i) Nutzen Sie die erstellten Programme zur Interpolation von

$$f : [-1, 1] \to \mathbb{R} \qquad f(x) = 1/(1 + 25x^2)$$

auf n äquidistanten Stützstellen x_k. Vergleichen Sie an den Stellen $x_{k/2} = (x_k + x_{k+1})/2$ das Verhalten des Fehlers $\max_k |f(x_{k/2}) - P(x_{k/2})|$ mit steigendem n. Was ist zu beobachten? Welche Stellen weisen den höchsten Fehler auf?

Die Abgabe der Programmieraufgaben erfolgt in den CIP-Pools vom 06.06. bis 13.06.2012. Die Listen für die Anmeldung zu den Abgabe-Terminen hängen in der Woche vom 23.05. bis 05.06.2012 aus.