Zuse Institute Berlin (ZIB)
Freie Universität Berlin
S. Röblitz, P. Koltai, S. Winkelmann, L. Fischer

Aufgabenblatt 6
Numerik I, SS 2018

Abgabe: 28. Mai 2018, 16:00 Uhr (Tutorenfach)

Allgemeine Hinweise: Die Punkte unterteilen sich in Theoriepunkte (TP) und Programmierpunkte (PP). Alle Programmieraufgaben und Protokolle müssen pünktlich per E-Mail als Anhang an den jeweiligen Tutor geschickt werden. Aus dem Text der E-Mail muss hervorgehen, wer an der Bearbeitung der Aufgaben mitgewirkt hat. Die Theorieaufgaben sind abzugeben in das Fach des Tutors (Arnimallee 3, 1. OG). Außerdem können zusätzlich und freiwillig Ausdrucke der Dateien abgegeben werden, falls eine detaillierte Korrektur des Codes erwünscht ist. Beachten Sie ferner die Hinweise zur Bearbeitung der Programmieraufgaben auf der Homepage zur Vorlesung:

`http://numerik.mi.fu-berlin.de/wiki/SS_2018/NumerikI.php`

Sie haben außerdem die Möglichkeit, beim Matlab-Tutorium am Freitag (10-12 Uhr, Raum 030, A6) Hilfestellung für die Programmieraufgaben zu bekommen.

Aufgabe 1 (Kondition der Polynominterpolation, 4 TP)
Zeigen Sie, dass der Interpolationsoperator

$$C[a,b] \ni f \mapsto \phi(f) = p_n \in \mathcal{P}_n$$

eine Projektion ist und die Norm $\|\phi\|_\infty = \Lambda_n$ (Operatornorm) hat, wobei

$$\Lambda_n = \max_{x\in[a,b]} \sum_{k=0}^{n} |L_k(x)|$$

die Lebesque-Konstante ist und L_k die Lagrange-Polynome bezeichnet.

Aufgabe 2 (Fehler der linearen Spline-Interpolation, 1+2+1 TP)
Es sei S^1_Δ die Menge der stetigen und stückweise linearen Funktionen bezüglich des Gitters $\Delta = \{x_0, ..., x_n\}$ mit $a = x_0 < x_1 < \cdots < x_n = b$. Für $g \in C[a,b]$ sei weiter $\|g\|_\infty = \max_{a\leq x\leq b} |g(x)|$, und wir bezeichnen die Interpolierte von g bezüglich S^1_Δ mit $s_1(g;\cdot)$.

a) Zeigen Sie, dass $\|s_1(g;\cdot)\|_\infty \leq \|g\|_\infty$ für alle $g \in C[a,b]$ gilt.

b) Zeigen Sie, dass $\|g - s_1(g;\cdot)\|_\infty \leq 2\|g - s\|_\infty$ für alle $g \in C[a,b]$, $s \in S^1_\Delta$ gilt.
(Hinweis: Verwenden Sie die Additivität von $s_1(g;\cdot)$ bezüglich g.)
Wie gut ist $s_1(g;\cdot)$ nun im Vergleich zur bestmöglichen Approximation bezüglich der $\|\cdot\|_\infty$-Norm?
(Hinweis: Schätzen Sie $\|g - s_1(g;\cdot)\|_\infty$ nach oben und unten bezüglich $\|g - s\|_\infty$ ab.)

c) Vergleichen Sie das Ergebnis mit der Fehlerabschätzung aus der Vorlesung für die Polynominterpolation, siehe Satz 3.10. in den Mitschriften bzw. Satz 3.8. im pdf-Skript.

Aufgabe 3 (Interpolationsfehler, 2+2 PP)
Für einen linearen Spline $s_1 \in \mathcal{S}^1_\Delta$, der die Funktion $f \in \mathcal{C}([a,b])$ auf dem Gitter $\Delta = \{x_0, \dots, x_n\}$ mit den Knoten $a = x_0 < x_1 < \cdots < x_n = b$ interpoliert, gilt die Fehlerabschätzung 3.15. aus den Mitschriften. In dieser Aufgabe sollen Sie diese theoretische Aussage numerisch untersuchen. Gehen Sie dabei wie folgt vor.

a) Schreiben Sie eine Matlab-Funktion

`function [err] = plotspline(x,func)`,

welche die Funktion $f(x)$ und den interpolierenden linearen Spline s_1 auf dem Gitter `x` plottet und den Fehler `err` in der Maximumsnorm zurückgibt. Das Argument `func` bezeichnet hierbei eine Matlab-Funktion, die $f(x)$ auswertet.
Erstellen Sie hierfür eine Schleife über die Teilintervalle $[x(i), x(i+1)]$ und plotten Sie jeweils die Werte des Splines (`svec`) und der Funktion (`fvec`) an den Punkten `y = linspace(x(i),x(i+1),20)`. Berechnen Sie den Fehler in jedem Teilintervall als `max(abs(svec - fvec))` und geben Sie den maximalen Wert dieser Fehler über die Teilintervalle in `err` zurück.
Testen Sie Ihr Programm mit der Funktion

$$f(x) = 1/(1 + 25x^2)$$

und dem Gitter `x = linspace(-1,1,5)`.

b) Schreiben Sie ein Skriptfile, das für die Funktion f und $n = 2^k$, $k = 1, \dots, 10$, zu den äquidistanten Gittern `x = linspace(-1,1,n+1)` die Fehler mit der in a) implementierten Matlab-Funktion berechnet. Plotten Sie diese logarithmisch mit x-Achse: n, y-Achse: Fehler zusammen mit einer Gerade $C/n^2 = O(h^2)$, an die sich asymptotisch die Fehlerkurve annähert.