Prof. Dr. Angela Kunoth
Sandra Boschert
Jose Licon

Numerische Mathematik
SoSe 2018

Übungsblatt 7

Ausgabe: Mittwoch, 13.6.2018
Abgabe: Mittwoch, 20.6.2018 bis 16 Uhr im Briefkasten vor dem Eingang Gyrhofstr. 8a.

Aufgabe 23: (10 Punkte)

Seien $n-2$ verschiedene Stützstellen $\Delta := \{a = x_1, \dots, b = x_{n-2}\}$ und dazugehörige Daten $f(x_1), \dots, f(x_{n-2})$ bezüglich einer Funktion f gegeben. Außerdem seien noch die Ableitungen $f'(a)$ und $f'(b)$ gegeben. Die dazugehörige Knotenfolge T wird gewählt als $T = \{\theta_i\}_{i=1,\dots,n+4}$ wobei

$$\theta_i = x_{i-3}, \quad i = 4, \dots, n+1$$

und $\theta_1 = \cdots = \theta_4$ sowie $\theta_{n+1} = \cdots = \theta_{n+4}$.
Nun ist ein kubischer Spline $S \in S_{4,\Delta}$ eindeutig durch die Interpolationsbedingungen

$$\begin{aligned} S(x_i) &= f(x_i), \qquad i = 1, \dots, n-2 \\ S'(a) &= f'(a), \quad S'(b) = f'(b) \end{aligned}$$

gegeben. Die Umsetzung dieser vollständigen kubischen Spline-Interpolation lässt sich mit B-Splines wie folgt realisieren. Zunächst gilt:

$$\begin{aligned} N_{j,4}(\theta_4) &= 0, \quad j = 2, 3, \dots \\ N_{j,4}(\theta_{n+1}) &= 0, \quad j = n-1, n-2, \dots, \end{aligned}$$

woraus sich

$$N_{1,4}(\theta_4) = 1, \quad N_{n,4}(\theta_{n+1}) = 1$$

ergibt. Damit sind die Koeffizienten c_1 und c_n bereits bestimmt als

$$\begin{aligned} S(\theta_4) &= c_1 = f(\theta_4) \\ S(\theta_{n+1}) &= c_n = f(\theta_{n+1}). \end{aligned}$$

Ähnlich lassen sich auch direkt die Koeffizienten c_2 und c_{n-1} angeben als

$$\begin{aligned} c_2 &= \frac{f'(a) - f(a)\, N'_{1,4}(a)}{N'_{2,4}(a)} \\ c_{n-1} &= \frac{f'(b) - f(b)\, N'_{n,4}(b)}{N'_{n-1,4}(b)}. \end{aligned}$$

Da die B-Splines auf halboffenen Intervallen definiert sind, aber auch der rechte Randpunkt $x = b$ interpoliert werden soll, setzt man formal

$$N_{j,4}(b) := \lim_{\substack{\varepsilon \to 0 \\ \varepsilon > 0}} N_{j,4}(b - \varepsilon),$$

anstatt den Wert auf Null zu setzen. Analog geht man bei den Ableitungen vor. Die verbleibenden Koeffizienten $\tilde{\mathbf{c}} := (c_3, \dots, c_{n-2})^T$ sind nun als Lösung des linearen Gleichungssystems

$$\tilde{A}\tilde{\mathbf{c}} = \tilde{\mathbf{f}}$$

gegeben, wobei

$$\tilde{A} := \begin{pmatrix} N_{3,4}(\theta_5) & N_{4,4}(\theta_5) & & \\ N_{3,4}(\theta_6) & \ddots & \ddots & \\ & \ddots & \ddots & \\ & \ddots & \ddots & N_{n-2,4}(\theta_{n-1}) \\ & & N_{n-3,4}(\theta_n) & N_{n-2,4}(\theta_n) \end{pmatrix},$$

und

$$\tilde{\mathbf{f}} := (f_3, \dots, f_{n-2})^T,$$

mit

$$\begin{aligned} f_3 &= f(\theta_5) - c_2 N_{2,4}(\theta_5), \\ f_j &= f(\theta_{j+2}), \qquad j = 4, \dots, n-3, \\ f_{n-2} &= f(\theta_n) - c_{n-1} N_{n-1,4}(\theta_n), \end{aligned}$$

wie in der Vorlesung. Die Einträge der Matrix, sowie Funktionsauswertungen der interpolierenden Spline-Funktion, lassen sich allgemein mit Hilfe des *Neville-artigen Schemas* (7.3.15) (siehe Aufgabe 18) bestimmen. Die auftretenden Ableitungen der B-Splines lassen sich nach Bemerkung 7.3.8 als Linearkombination von B-Splines niedrigerer Ordnung ausdrücken.

a) Implementieren Sie die obige vollständige kubische Spline-Interpolation für die äquidistanten Stützstellen

$$x_i = 0.25 + \frac{i-1}{20}, \qquad i = 1, \dots, 11$$

und der zu interpolierenden Funktion

$$f(x) := \sin(2\pi x)$$

auf dem Intervall $[\frac{1}{4}, \frac{3}{4}]$. Plotten Sie die Lösung und erzeugen Sie einen Differenzplot des interpolierenden Splines S und der zu interpolierenden Funktion f. Tragen Sie zudem die Datenpaare $(x_i, f(x_i))$ mit in den Plot der Lösung ein, sowie die Punkte $(x_i, f(x_i) - S(x_i))$ in den Differenzplot. Das Gitter zum Plotten sollte selbstverständlich feiner gewählt werden als die Stützstellen.

b) Führen Sie nun die kubische Spline-Interpolation für die äquidistanten Stützstellen

$$x_i = 0.25 + \frac{i-1}{20}, \qquad i = 1, \dots, 11$$

wie in a) und der zu interpolierenden Funktion

$$f(x) := |\sin(2\pi x)|$$

auf dem Intervall $[\frac{1}{4}, \frac{3}{4}]$ durch. Plotten Sie die Lösung zusammen mit der zu interpolierenden Funktion *f in einem Plot* und erzeugen Sie einen Differenzplot des interpolierenden Splines S und der zu interpolierenden Funktion f. Tragen Sie zudem die Datenpaare $(x_i, f(x_i))$ mit in den Plot der Lösung ein, sowie die Punkte $(x_i, f(x_i) - S(x_i))$ in den Differenzplot. Vergleichen Sie den Interpolationsfehler in b) mit dem Interpolationsfehler in a). In welcher Region ist der Interpolationsfehler am Größten, und was ist der Grund dafür?

c) Wählen Sie nun die *nicht*-äquidistanten Stützstellen

$$\begin{aligned} x_1 &= 0.25, \\ x_i &= 0.45 + \frac{(i-2)}{100}, \qquad i = 2, \dots, 12, \\ x_{13} &= 0.75 \end{aligned}$$

und führen Sie erneut die kubische Spline-Interpolation bezüglich der zu Interpolierenden

$$f(x) := |\sin(2\pi x)|$$

auf dem Intervall $[\frac{1}{4}, \frac{3}{4}]$ durch. Plotten Sie ihre Lösung wie in b) zusammen mit der zu interpolierenden Funktion *f in einem Plot* und erzeugen Sie einen Differenzplot des interpolierenden Splines S und der zu interpolierenden Funktion f. Tragen Sie zudem die Datenpaare $(x_i, f(x_i))$ mit in den Plot der Lösung ein, sowie die Punkte $(x_i, f(x_i) - S(x_i))$ in den Differenzplot. Was fällt nun im Vergleich zu b) auf? Wie könnte man die *kubischen* B-Splines verändern, um die Singularität an der Stelle $x = 0.5$ besser aufzulösen?

d) Berechnen Sie die interpolierenden Splines bezüglich

$$f(x) := sin(2\pi x)$$

wie in a) für die verschiedene Stützstellen

$$x_{i,h} = \frac{1}{4} + (i-1)h, \quad i = 1, \dots, \frac{1}{2h} + 1 \quad \text{mit zugehöriger Gitterweite } h = \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \dots.$$

Eine Möglichkeit, den Interpolationsfehler zu schätzen, ist

$$\|f - S\|_{\infty,[a,b]} \approx \max\{|f(z_i) - S(z_i)| : z_i \in [a,b],\ i = 1, \dots, M\}, \tag{1}$$

wobei man hier M 'genügend groß' im Vergleich zu h^{-1} wählen sollte. Erzeugen Sie eine Fehlertabelle, in der Sie die Gitterweite h, die Maximumsnorm des Interpolationsfehlers (approximiert gemäß (1)), sowie den Quotienten zweier aufeinanderfolgender Interpolationsfehler auftragen. Welche Fehlerordnung lässt sich hieraus erkennen?

Implementierung:
Zur Implementierung können Sie, wie üblich, Matlab verwenden. Hierbei dürfen allerdings die bereits integrierten Spline-Interpolations-Routinen *nicht* benutzt werden! Eine *mögliche* Vorgehensweise ist die Folgende:

- Verwenden Sie das Neville-Schema von Übung 5, Aufgabe 18. Beachten Sie, dass später für die Auswertung von Ableitungen von B-Splines auch B-Splines niedrigerer Ordnung benötigt werden und dass auch nicht äquidistante Stützstellen vorkommen (siehe c)). Daher bietet es sich an, dieses Schema so allgemein zu halten, dass für beliebige Entwicklungskoeffizienten $\mathbf{c} := (c_1, \dots, c_n)^T$ und einer nicht-äquidistanten Knotenfolge $T := \{\theta_i\}_{i=1,\dots,n+k}$ die Spline-Funktion

  $$S(x) := \sum_{i=1}^{n} c_i N_{i,k}(x)$$

  an jeder beliebigen Stelle $x \in [a,b] = [\theta_4, \theta_{n+1}]$ und für allgemeines $k \in \mathbb{N}$ mit $k \geq 1$ ausgewertet werden kann. Achten Sie auf die Besonderheit des rechten Intervallstücks $[\theta_n, \theta_{n+1})$ bzw. den Auswertungen am rechten Randpunkt $x = b$.

- Als nächstes bietet es sich an, eine Funktion zu schreiben, welche anhand der gegebenen Daten $\{f(x_1), \dots, f(x_{n-2}), f'(x_1), f'(x_{n-2})\}$ und Knotenpunkten T die Matrix $\tilde{A}$ sowie die dazugehörige rechte Seite $\tilde{\mathbf{f}}$ erstellt.

- Nun lässt sich mit Hilfe der ersten beiden Funktionen eine weitere Funktion schreiben, welche letztendlich für gegebene Daten und Knotenpunkte die entsprechenden Entwicklungskoeffizienten $\mathbf{c}$ des interpolierenden Splines in der kubischen B-Spline-Basis liefert.

- Schließlich muss dann in der 'main' oder einer zusätzlichen Funktion nur noch der entsprechende interpolierende Spline visualisiert werden, wozu erneut das Neville-Scheme benutzt werden kann. Wie bereits erwähnt, sollte das Gitter, das zum Plotten benutzt wird, feiner gewählt werden als die Knotenfolge: das Gitter zum Plotten ist natürlich vollkommen unabhängig von der Knotenfolge. Denken Sie auch hier wieder an eine aussagefähige Beschriftung.

Aufgabe 24: (5 Punkte)
Sei $I(f) = \int_a^b f(x)\,dx$ und das Intervall $[a,b]$ unterteilt in die n Teilintervalle $[t_{k-1}, t_k]$ für $k = 1, \dots, n$ mit äquidistanten Stützstellen $t_k = a + kh$ und $h = \frac{b-a}{n}$. Ein klassisches Verfahren zur numerischen Integration ist die *Mittelpunktsregel*. Hier approximiert man f auf den Teilintervallen $[t_{k-1}, t_k]$ durch $g_k(x) := f\left(\frac{t_{k-1}+t_k}{2}\right)$ für $x \in [t_{k-1}, t_k]$. Damit ist eine Näherung des Integrals $I(f)$ gegeben durch

$$R(h) = h \sum_{k=1}^{n} f\left(\frac{t_{k-1} + t_k}{2}\right).$$

Zeigen Sie, dass dieses Verfahren ein Verfahren *zweiter Ordnung* ist, d.h. dass der Fehler wie $O(h^2)$ gegen 0 geht für $h \to 0$.

Aufgabe 25: (5 Punkte)
Implementieren Sie die Rechteckregel und die Trapezregel.

a) Berechnen Sie $I(f) := \int_1^2 \sqrt{x^7}\,dx$ approximativ mit der Rechteckregel

$$R(h) := h\sum_{k=1}^{n} f(t_{k-1}),$$

für verschiedene Gitterweiten $h = \frac{b-a}{n}$ mit $n = 2^j$ und $j = 1,2,\ldots,15$, d.h. $t_k = a+kh$ mit $k = 0,\ldots,n$, wobei in diesem Fall $a = 1$ und $b = 2$ ist. Erstellen Sie eine Fehlertabelle, in der Sie die Gitterweite h bzw. j, den Fehler $|I(f)-R(h)|$ sowie den Quotient zweier aufeinanderfolgender Fehler auftragen. Welche Ordnung des Verfahrens lässt sich hier ablesen?

b) Führen Sie Aufgabenteil a) analog mit der Trapezregel

$$R(h) := h\left[\frac{1}{2}(f(1)+f(2)) + \sum_{k=1}^{n-1} f(t_k)\right]$$

durch.

c) Ein bekanntes Beispiel eines aus der Stochastik motivierten Integrals, welches sich nicht durch eine geschlossene Funktion darstellen lässt, ist die *Gaußsche Fehlerfunktion*

$$\operatorname{erf}(z) := \frac{2}{\sqrt{\pi}}\int_0^z e^{-x^2}\,dx, \qquad z\in\mathbb{R}.$$

Wie groß müssen Sie n mindestens wählen, um garantieren zu können, dass die Approximation der Gaußschen Fehlerfunktion mit der Trapezregel für alle Werte $z = 0.1, 0.2, \ldots, 1$ bis auf einen Fehler von höchstens 10^{-4} genau ist? Nutzen Sie hierzu die Fehlerabschätzung für die Trapezregel aus Beispiel 8.1.3. Approximieren Sie die Gaußsche Fehlerfunktion an diesen Stellen $z = 0.1, 0.2, \ldots, 1$ mit Hilfe der Trapezregel und tragen Sie die Werte in eine Tabelle ein.

Hinweis:
Schreiben Sie bitte auf jedes Blatt Ihren Namen. Auf die erste Seite der Übung schreiben Sie bitte zusätzlich Ihre Matrikelnummer und Ihre Übungsgruppennummer. Den gesamten Programmcode schicken Sie bitte bis zum Abgabetermin per E-Mail an Ihren jeweiligen Übungsleiter mit:

- Subject (Betreff): Uebung Name, Vorname Mat.-Nr.
- Body: Erläuternde Hinweise, falls nötig, ansonsten einfach leer lassen.
- Attachement: Sämtliche .m-Dateien, die zur Ausführung Ihres Programmes benötigt werden, archiviert als .zip- oder .rar-Datei. Bitte benennen Sie die Dateien mit Nachname.zip bzw. Nachname.rar .