

Bemerkung 3.20 *Zum Bestapproximationsfehler.* Aus der Analysis-Grundvorlesung ist bekannt, dass sich jede stetige Funktion beliebig genau durch Polynome approximieren lässt, Satz von Weierstraß. Das bedeutet, es gilt

$$\lim_{n \rightarrow \infty} E_n(f) = 0. \quad (3.21)$$

□

Bemerkung 3.21 *Zur Lebesgue-Konstanten.* Die Lebesgue-Konstante verhält sich leider nicht besonders erfreulich. So bewies Faber⁷ im Jahre 1914:

- Zu jeder Folge von Stützstellen gibt es eine Funktion $f \in C([a, b])$, so dass gilt

$$\liminf_{n \rightarrow \infty} \|f - p_n\|_{\infty} > 0.$$

Der Interpolationsfehler geht also nicht gegen Null. Gemeinsam mit (3.21) bedeutet dies, dass $\Lambda_n \rightarrow \infty$ für $n \rightarrow \infty$.

Im Jahre 1978 zeigten Brutman, de Boor und Pinkus eine untere Schranke für die Lebesgue-Konstante.

- Für jede Folge von Stützstellen gilt

$$\frac{2}{\pi} \ln(n+1) + \frac{2}{\pi} \left(\gamma + \ln \left(\frac{4}{\pi} \right) \right) \leq \Lambda_n.$$

Hierbei ist

$$\gamma = \lim_{n \rightarrow \infty} \left(\sum_{i=1}^n \frac{1}{i} - \ln(n) \right) \approx 0.57721$$

die Euler⁸-Mascheroni⁹-Konstante.

Das heißt, selbst für die bestmögliche Wahl von Stützstellen wächst Λ_n mindestens logarithmisch an.

Das Wachstumsverhalten kann aber noch viel schlimmer sein. So bewies Turetskii¹⁰ (1940)

- Für die Folge äquidistanter Stützstellen $x_j = a + jh$, $j = 0, \dots, n$, mit der Schrittweite $h = (b - a)/n$ gilt

$$\Lambda_n \approx e^{-1} \frac{2^{n+1}}{n \ln(n)}.$$

Die Lebesgue-Konstante wächst exponentiell! Das bedeutet, äquidistante Stützstellen sollten vermieden werden. □

Bemerkung 3.22 *Geschickte Wahl der Stützstellen.* Für eine geschickte Wahl der Stützstellen kann man sich an der Fehlerabschätzung (3.10) orientieren. Vernachlässigt man den Einfluss von $f(x)$, dann erhält man einen kleinen Fehler, wenn

$$\|\omega_{n+1}\|_{\infty} = \left\| \prod_{j=0}^n (x - x_j) \right\|_{\infty} \quad (3.22)$$

möglichst klein ist. Das Polynom $\omega_{n+1}(x)$ ist vom Grad $n+1$ und der Koeffizient vor dem Term höchsten Grades ist gleich Eins. Die Koeffizienten vor den anderen Termen kann man noch wählen. Damit lautet eine äquivalente Formulierung zur Minimierung von (3.22), ein Polynom n -ten Grades zu finden, welches x^{n+1} in der Norm $\|\cdot\|_{\infty}$ am besten approximiert.

Die Lösung dieser Aufgabe im Fall $[a, b] = [-1, 1]$ ist aus Beispiel 1.37 bekannt. Man erhält die Tschebyscheff-Polynome 1. Art

$$\omega_{n+1}(x) = \frac{1}{2^n} T_{n+1}(x).$$

⁷Georg Faber (1877 – 1966)

⁸Leonhard Euler (1707 – 1783)

⁹Lorenzo Mascheroni (1750 – 1800)

¹⁰A.H. Turetskii

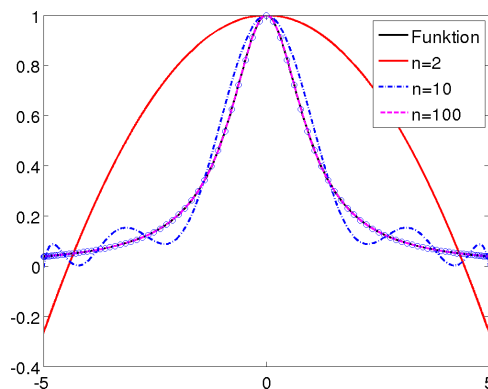


Abbildung 3.2: Beispiel 3.23, Interpolationspolynome für $n \in \{2, 10, 100\}$, Tschebyscheff-Stützstellen.

Die resultierenden Stützstellen sind gerade die Nullstellen von $T_{n+1}(x)$

$$x_j = \cos \left(\frac{2(n-j)+1}{2(n+1)} \pi \right).$$

Im Fall $[a, b] \neq [-1, 1]$ kann man die Tschebyscheff-Stützstellen durch Transformation bestimmen.

Im Jahre 1974 zeigte Rivlin¹¹

- Für die Tschebyscheff-Stützstellen gilt

$$\frac{2}{\pi} \ln(n+1) + \frac{2}{\pi} \left(\gamma + \ln \left(\frac{8}{\pi} \right) \right) \leq \Lambda_n \leq \frac{2}{\pi} \ln(n+1) + 1.$$

In diesem Fall hat man wirklich nur das logarithmische Wachstum der Lebesgue-Konstanten. Der konstante Term auf der linken Seite hat ungefähr den Wert 0.9625. $\square$

Beispiel 3.23 *Beispiel von Runge.* Es wird die gleiche Funktion wie im Beispiel 3.9 betrachtet. Nimmt man als Stützstellen die Tschebyscheff-Stützstellen, so erhält man die in Abbildung 3.2 dargestellten Interpolationen. Man erhält selbst für $n = 100$ eine gute Interpolation. $\square$

3.3 Spline-Interpolation

Bemerkung 3.24 *Motivation, Aufgabenstellung.* Die Polynominterpolation besitzt zwei sehr negative Eigenschaften:

- für große n , das heißt viele Stützstellen, kann sie instabil werden, insbesondere bei äquidistanten Stützstellen,
- für nichtglatte Funktionen kann der Interpolationsfehler zwischen den Stützstellen beliebig groß werden.

Diese beiden Situationen (viele äquidistante Stützstellen oder nichtglatte Funktionen) treten in den Anwendung jedoch häufig auf. In diesen Fällen muss man eine andere Interpolation als die Polynom-Interpolation verwenden. Eine oft genutzte Herangehensweise ist die Spline-Interpolation:

spline (engl.) – längliches, dünnes Stück Holz oder Metall.

$\square$

Definition 3.25 **Spline.** Seien $x_0, \dots, x_n \in [a, b]$ paarweise verschiedene Punkte mit $a = x_0 < x_1 < \dots < x_n = b$. Die Funktion $s_k(x) : [a, b] \rightarrow \mathbb{R}$ wird Spline vom Grad k bezüglich der Stützstellen $\{x_j\}$ genannt, falls

$$s_k(x)|_{[x_j, x_{j+1}]} \in P_k, \quad j = 0, \dots, n-1,$$

¹¹Theodore J. Rivlin

$$s_k(x) \in C^{k-1}([a, b]). \quad (3.23)$$

□

Bemerkung 3.26 *Anzahl der gegebenen Bedingungen.* Definition 3.25 bedeutet: Ein Spline vom Grad k ist

- eine stückweise polynomiale Funktion, in jedem Teilintervall ein Polynom vom Grad k ,
- im Gesamtintervall noch $(k - 1)$ -mal differenzierbar.

In jedem Teilintervall kann man den Spline als

$$s_{k,j}(x) = \sum_{i=0}^k s_{ij}(x - x_j)^i, \quad x \in [x_j, x_{j+1}], \quad j = 0, \dots, n-1,$$

darstellen. Man hat also $n(k + 1)$ unbekannte Koeffizienten s_{ij} .

Aus (3.23) folgt

$$s_{k,j-1}^{(m)}(x_j) = s_{k,j}^{(m)}(x_j), \quad j = 1, \dots, n-1, \quad m = 0, \dots, k-1.$$

Das sind $k(n - 1)$ Bedingungen an die Koeffizienten. Insbesondere hat man, für $m = 0$, die Stetigkeit in den inneren Stützstellen, aber noch nicht den Funktionswert. Der Spline soll eine Funktion $f(x)$ interpolieren, deren Werte in den Stützstellen berechnet werden können. Damit hat man weitere $(n + 1)$ Bedingungen: $s_k(x_j) = f(x_j)$, $j = 0, \dots, n$. Es fehlen noch $(k - 1)$ Bedingungen. □

Beispiel 3.27 *Linearer Spline.* Ein linearer Spline, das heißt $k = 1$, ist nichts weiter als ein Polygonzug. Ein solcher Polygonzug gehört zum Raum S_n , siehe Definition 1.8. Für lineare Splines fehlen keine Bedingungen.

Die Voraussetzungen der Interpolationsaufgabe seien wie in Bemerkung 3.1. Dann lautet die Interpolationsaufgabe für lineare Splines: Finde $u_n \in S_n$, so dass

$$u_n(x_i) = f(x_i), \quad i = 0, \dots, n.$$

Die Lösung dieser Aufgabe ist

$$u_n(x) = \sum_{i=0}^n f(x_i) \varphi_i(x),$$

wobei $\{\varphi_i(x)\}_{i=0}^n$ die in (1.18) definierte Knotenbasis ist. Man verbindet also die gegebenen Punkte $(x_i, f(x_i))$ durch einen Polygonzug. □

Satz 3.28 Fehlerabschätzung für lineare Spline-Interpolation. Seien $a = x_0$, $b = x_n$ und $f \in C^2([a, b])$. Dann gilt für die lineare Spline-Interpolierte $u_n(x)$ die Fehlerabschätzung

$$\|f - u_n\|_\infty \leq h^2 \frac{\|f''\|_\infty}{8}$$

mit

$$h = \max_{j=1, \dots, n} h_j, \quad h_j = x_j - x_{j-1}.$$

Beweis: Man wendet die Fehlerdarstellung der Polynominterpolation (3.9) auf jedes Teilintervall an, da man dort mit einem linearen Polynom interpoliert. Damit ergibt sich für $j = 1, \dots, n$, $\xi \in (x_{j-1}, x_j)$,

$$\begin{aligned} \max_{x \in [x_{j-1}, x_j]} |f(x) - u_n(x)| &= \max_{x \in [x_{j-1}, x_j]} \left| \frac{f''(\xi)}{2!} \omega_2(x) \right| \\ &= \max_{x \in [x_{j-1}, x_j]} \left| \frac{f''(\xi)}{2} (x - x_{j-1})(x - x_j) \right| \\ &\leq \max_{x \in [x_{j-1}, x_j]} \left| \frac{f''(x)}{2} \right| \max_{x \in [x_{j-1}, x_j]} |(x - x_{j-1})(x - x_j)| \\ &= \frac{h_j^2}{8} \max_{x \in [x_{j-1}, x_j]} |f''(x)|, \end{aligned}$$

da der größte Wert des zweiten Faktors (quadratische Funktion mit den Nullstellen x_{j-1} und x_j) für den Mittelpunkt $x = (x_{j-1} + x_j)/2$ angenommen wird. Damit folgt

$$\|f - u_n\|_\infty = \max_{j=1, \dots, n} \left(\max_{x \in [x_{j-1}, x_j]} |f(x) - u_n(x)| \right) \leq \frac{h^2}{8} \max_{x \in [a, b]} |f''(x)|.$$

■

Bemerkung 3.29 *Kubische Splines.* Lineare Splines sind einfach zu handhaben, jedoch relativ ungenau und die Interpolation besitzt Ecken, siehe später in Abbildung 3.3.

In der Praxis möchte man glattere Übergänge und dafür nutzt man oft kubische Splines, das heißt $k = 3$. Ein kubischer Spline ist zweimal stetig bis zum Rand differenzierbar, insbesondere ist seine Krümmung wohldefiniert. Für die zwei fehlenden Bedingungen setzt man zum Beispiel die Randwerte

$$s_3''(a) = s_3''(b) = 0. \quad (3.24)$$

Diese Randbedingung nennt man natürliche Randbedingung. Die Randbedingung

$$s_3'(a) = f'(a), \quad s_3'(b) = f'(b) \quad (3.25)$$

heißt vollständige Randbedingung.

Betrachten wird nun eine Funktion $f(x)$, die durch einen kubischen Spline interpoliert werden soll. Wir verwenden die Bezeichnungen

$$f_i = s_3(x_i), \quad M_i = s_3''(x_i), \quad i = 0, \dots, n.$$

Die zweite Ableitung $s_3''(x)$ ist eine stetige, stückweise lineare Funktion (Polygonzug). Mit den obigen Bezeichnungen gilt

$$s_3''(x) = M_{i-1} \frac{x_i - x}{x_i - x_{i-1}} + M_i \frac{x - x_{i-1}}{x_i - x_{i-1}}, \quad x \in [x_{i-1}, x_i].$$

Die Stetigkeit von $s_3''(x)$ überprüft man durch Einsetzen der Intervallgrenzen. Zweimaliges integrieren liefert eine Darstellung des Splines in $[x_{i-1}, x_i]$

$$s_3(x) = \frac{M_{i-1}}{6} \frac{(x_i - x)^3}{x_i - x_{i-1}} + \frac{M_i}{6} \frac{(x - x_{i-1})^3}{x_i - x_{i-1}} + c_{i-1}(x - x_{i-1}) + d_{i-1}. \quad (3.26)$$

Einsetzen der Endpunkte des Teilintervalls ergibt die Konstanten:

$$\begin{aligned} f_{i-1} &= s_3(x_{i-1}) = \frac{M_{i-1}}{6}(x_i - x_{i-1})^2 + d_{i-1} \implies \\ d_{i-1} &= f_{i-1} - \frac{M_{i-1}}{6}(x_i - x_{i-1})^2, \end{aligned} \quad (3.27)$$

und

$$\begin{aligned} f_i &= s_3(x_i) = \frac{M_i}{6}(x_i - x_{i-1})^2 + c_{i-1}(x_i - x_{i-1}) + d_{i-1} \implies \\ c_{i-1} &= \frac{f_i - f_{i-1}}{x_i - x_{i-1}} - \frac{x_i - x_{i-1}}{6}(M_i - M_{i-1}). \end{aligned} \quad (3.28)$$

Man rechnet direkt nach, dass mit diesen Konstanten die Stetigkeit von $s_3(x)$ in den Stützstellen gewährleistet ist.

Aus der Stetigkeitsbedingung für die erste Ableitung erhält man weitere Gleichungen für $M_0, \dots, M_n$. Es gilt für $x \in [x_{i-1}, x_i]$

$$s_3'(x) = -\frac{M_{i-1}}{2} \frac{(x_i - x)^2}{x_i - x_{i-1}} + \frac{M_i}{2} \frac{(x - x_{i-1})^2}{x_i - x_{i-1}} + c_{i-1}.$$

Für den rechten Randpunkt erhält man

$$s_3'(x_{i-0}) = \frac{M_i}{2}(x_i - x_{i-1}) + \frac{f_i - f_{i-1}}{x_i - x_{i-1}} - \frac{x_i - x_{i-1}}{6}(M_i - M_{i-1})$$