Computational Engineering

Übungen zu: Interpolation und Approximation

Aufgabe 1:

Gegeben sei die Funktion

$$f(x) = \frac{1}{1+x^2}.$$

Unterteilen Sie das Intervall $x \in [-3, 3]$ in $n = 6, 8, 10, 12$ gleich große Teilintervalle und bilden Sie daraus eine Wertetabelle der Art

x_i	x_0	x_1	$\cdots$	x_n
y_i	$f(x_0)$	$f(x_1)$	$\cdots$	$f(x_n)$

a) Berechnen Sie jeweils das Interpolationspolynom für die Daten. Benutzen Sie dazu die Matlab-Befehle `polyfit` und `polyval`. Stellen Sie die Ergebnisse graphisch dar. Was beobachten Sie?

b) Stellen Sie zusätzlich noch die Interpolierende mit einem kubischen Spline dar. Benutzen Sie hierzu den Matlab-Befehl `spline`.

c) Verändern Sie die Datentabelle, in dem Sie zum Beispiel eine andere erzeugende Funktion f wählen.

Aufgabe 2:

Gegeben sind die folgenden Datenpaare:

x_i	0	1	2	3	4
y_i	3	1	0.5	0.2	0.05

und die Ansatzfunktionen haben die Form

$$f_a(x, \lambda_1, \lambda_2) = \lambda_1 e^{\lambda_2 x}$$

bzw.

$$f_b(x, \lambda_1, \lambda_2, \lambda_3) = \frac{\lambda_1}{x^{\lambda_2} + \lambda_3}$$

Finden Sie für beide Ansatzfunktionen jeweils Werte für die Parameter λ_k, in dem Sie das quadratische Fehlerfunktional

$$E(\lambda_1, \ldots, \lambda_m) = |\vec{y} - f(\vec{x}, \lambda_1, \ldots, \lambda_m)|$$

minimieren. Benutzen Sie hierfür einen der in Matlab implementierten Optimierungsalgorithmen (z.B. `fminsearch` oder `fminunc`).

Aufgabe 3:

Gegeben sei der folgende Datensatz

x_i	-1	0	1	2	3
y_i	3	2	9	21	49

und die Funktionen

$$f_1(x) = 1, \quad f_2(x) = x, \quad f_3(x) = x^2, \quad f_4(x) = \sqrt{1+x}, \quad f_5(x) = e^x.$$

Approximieren Sie wieder durch Minimierung des Fehlerfunktionals (entweder mit fminsearch, oder fminunc) die Daten mit den gegebenen Funktionen **linear**, also durch den Ansatz

$$f(x) = \sum_k \lambda_k f_k(x)$$

Variieren Sie dabei welche der einzelnen Funktionen in der Summe vorkommen. Nehmen Sie auch eigenständig noch andere Funktionen hinzu. Was lässt sich beobachten?

Aufgabe 4:

Gegeben ist die Funktion $f : \mathbb{R}^d \to \mathbb{R}$

$$f(x_1, x_2, \dots, x_d) = \sum_{k=1}^{d} 100(x_{k+1} - x_k^2)^2 + (x_k - 1)^2$$

Diese Funktion hat ein (globales) Minimum bei $(x_1, x_2, \dots, x_d) = (1, 1, \dots, 1)$.
Untersuchen Sie die Performance der Algorithmen `fminsearch` und `fminunc` in dem Sie für Werte von $d = 2, 3, 4, \dots, 10$ und für verschiedene Startpunkte $\vec{x}_0 \in [-5, 10] \times [-5, 10] \times \cdots \times [-5, 10]$ das Minimum von f mittels der beiden Befehle suchen und die Ergebnisse hinsichtlich der Anzahl der benötigten Iterationsschritte, der Rechenzeiten und der Qualität der Ergebnisse vergleichen.