TU Chemnitz
Professur Numerische Mathematik
Prof. Dr. Oliver Ernst
Dipl.-Math. Ingolf Busch
Dipl.-Math. techn. Tommy Etling

5. Februar 2014

Numerische Mathematik

Sommersemester 2013

10. Übungsblatt

Aufgabe 39:

Das US–amerikanische NIST (National Institute of Standards and Technology) stellt Referenzdatenmengen StRD (Statistical Reference Datasets) zum Testen und Zertifizieren statistischer Software bereit. Die entsprechende Webseite des NIST ist unter

`http://www.itl.nist.gov/div898/strd/lls/lls.shtml`

zu finden. Für diese Aufgabe ist die sogenannte „Filip“ Datenmenge zu benutzen.

Diese Datenmenge ist „umstritten“. Eine Websuche nach „filip strd“ liefert dutzende Ergebnisse einschliesslich der Originalseite vom NIST. Einige Mathematik– und Statistikpakete sind in der Lage, die Polynomkoeffizienten zu reproduzieren, welche NIST als „certified values“ ausgewiesen hat. Andere Pakete geben Warnungen oder Fehlermeldungen, dass das Problem zu schlecht konditioniert ist. Wieder andere liefern abweichende Koeffizienten ohne Warnungen. Es sind im Web verschieden Meinungen zu finden, ob dies ein „vernünftiges“ Problem ist oder nicht.

Laden Sie die entsprechende Seite vom NIST, extrahieren Sie die Daten und untersuchen Sie diese mit MATLAB. Die Datenmenge besteht aus 82 Messdaten (x_i, y_i), wobei in der Datenmenge zuerst der y–Wert aufgeführt ist. Diese sind durch ein Polynom zehnten Grades auszugleichen.

Es sei n die Anzahl der Daten und $p = 11$ die Anzahl der Polynomkoeffizienten.

(a) Als erstes Experiment sind die Daten in MATLAB zu laden und mit dem Linientyp `'.'` zu plotten. Verwenden Sie dann `Basic Fitting` aus dem Tools–Pulldownmenü des figure–Fensters und wählen Sie `10th degree polynomial` aus. Sie erhalten eine Warnung, dass das Problem schlecht konditioniert ist, welche Sie aber an dieser Stelle ignorieren können.
Wie sind die berechneten Koeffizienten im Vergleich mit den zertifizierten vom NIST? Wie unterscheidet sich der von MATLAB gezeichnete Graph von dem auf der NIST Webseite? Das `Basic Fitting`–Tool liefert auch die Residualnorm $\|r\|$. Vergleichen Sie diese mit dem NIST–Wert der "Residual Standard Deviation", welcher durch $\|r\|/\sqrt{(n-p)}$ definiert ist.

(b) Im zweiten Teil des Experiments soll diese Datenmenge sorgfältiger analysiert werden, indem 6 verschiedene Methoden zur Berechnung der Polynomkoeffizienten benutzt werden. Erklären Sie dabei alle auftretenden Warnungen.

- Polyfit: Benutzen Sie `polyfit(x,y,10)`.

- Backslash: Verwenden Sie `X\y`, wobei X die auf $n \times p$ reduzierte Vandermondesche Matrix mit den Elementen

$$X_{i,j} = x_i^{p-j}\,, \quad i = 1, \dots, n,\ j = 1, \dots, p$$

  bezeichnet.
- Pseudoinverse: Bestimmen Sie `pinv(X)*y`.
- Normalgleichungen: Ermitteln Sie `inv(X'*X)*X'*y`.
- Balancierung: Es sei $\mu = \texttt{mean}(x),\ \sigma = \texttt{std}(x),\ t = (x - \mu)/\sigma$.
  Berechnen Sie `polyfit(t,y,10)`.
- Zertifizierte Koeffizienten: Beschaffen Sie sich die Koeffizienten von der NIST Webseite.

(c) Wie groß sind jeweils die Residualnormen für die 6 ermittelten Lösungen?

(d) Welche der 6 Methoden liefern einen ziemlich schlechten Datenausgleich? (Vielleicht verwenden die im Web für schlechte Ergebnisse kritisierten Softwarepakete gerade diese Methode.)

(e) Vergleichen Sie die Lösungen (Koeffizienten des Polynoms 10. Grades) mit den von NIST zertifizierten Koeffizienten. (*Hinweis:* Bei Methode 5 (Balancierung) ist vorher eine Rücktransformation notwendig!)

(f) Zeichnen Sie die fünf "guten" Lösungen. Verwenden Sie Punkte (Linientyp `'.'`) in den Daten und zeichnen Sie die Polynomkurven mit mehreren hundert Punkten über den x–Bereich. Es gibt 5 verschiedene Plots, aber nur zwei unterscheiden sich auch visuell. Welche Methode liefern welchen Plot?
Hinweis: Benutzen Sie zur Funktionsauswertung die MATLAB–Funktion `polyval`.

(g) Warum produzieren polyfit und backslash unterschiedliche Resultate?

Aufgabe 40:

(a) Zu $a \le x_0 < \cdots < x_n \le b$ seien $\ell_j(x), j = 0, \dots, n$, die Lagrange-Grundpolynome. Man zeige

$$\sum_{j=0}^{n} \ell_j(x)(x - x_j)^k = \begin{cases} 1 & \text{für } k = 0, \\ 0 & \text{für } k = 1, \dots, n. \end{cases}$$

(b) Sei $f(x) := \cos(x)\sin(x)$. Für jedes $n \in \mathbb{N}$ wählen wir Knoten $0 \le x_0^{(n)} < x_1^{(n)} < \cdots < x_n^{(n)} \le 2\pi$ und bezeichnen das Interpolationspolynom für f an diesen Knoten mit p_n. Zeigen Sie

$$\lim_{n\to\infty} \left(\max_{x\in[0,2\pi]} |f(x) - p(x)| \right) = 0.$$

Aufgabe 41:

Eine Verallgemeinerung der in der Vorlesung behandelten Interpolationsaufgabe für Polynome besteht darin, zu $n+1$ Knoten $a \le x_0 < x_1 < \cdots < x_n \le b$ sowie zwei Sätzen von je $n+1$ Zahlen $\{f_j\}_{j=0}^n$, $\{f'_j\}_{j=0}^n$ ein Polynom $p \in \mathscr{P}_{2n+1}$ zu bestimmen, sodass

$$p(x_j) = f_j, \quad p'(x_j) = f'_j \qquad (j = 0, \ldots, n).$$

Interpolationsaufgaben, in denen Ableitungswerte vorgeschrieben sind, nennt man **Hermite-Interpolation**.

(a) Zeigen Sie: Sind $\{\ell_j\}_{j=0}^n$ die Lagrange-Grundpolynome der Knoten $\{x_j\}_{j=0}^n$, so gelten für die Polynome $\{h_j\}_{j=0}^n$ sowie $\{k_j\}_{j=0}^n$ definiert durch

$$\begin{aligned} h_j(x) &:= [\ell_j(x)]^2\big(1 - 2\ell'_j(x_j)(x - x_j)\big), \\ k_j(x) &:= [\ell_j(x)]^2(x - x_j) \end{aligned}$$

folgende Aussagen:

(i) $h_j, k_j \in \mathscr{P}_{2n+1}$ für alle j.

(ii) $h_i(x_j) = \delta_{i,j}$, $h'_i(x_j) = 0$ für alle $1 \le i, j \le n$.

(iii) $k_i(x_j) = 0$, $k'_i(x_j) = \delta_{i,j}$ für alle $1 \le i, j \le n$.

(b) Konstruieren Sie mit diesen Hilfspolynomen eine Lösung der o.g. Hermiteschen Interpolationsaufgabe.

(c) Zeigen Sie, dass die Lösung eindeutig ist.

(d) Zeigen Sie: Ist $f \in C^{2n+2}([a,b])$ und $p \in \mathscr{P}_{2n+1}$ das oben definierte Hermite-Polynom für die Daten $f_j = f(x_j)$ und $f'_j = f'(x_j)$, so existiert zu jedem $x \in [a,b]$ ein $\xi = \xi(x) \in (a,b)$, sodass

$$f(x) - p(x) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!}[\omega_{n+1}(x)]^2.$$

Hierbei ist ω_{n+1} das Knotenpolynom zu $\{x_j\}_{j=0}^n$.

Aufgabe 42:

Mit $f[x_0, x_1, \ldots, x_n, x]$ sei die dividierte Differenz der Ordnung $n+1$ der $n+2$ paarweise verschiedenen Zahlen $x_0, x_1, \ldots, x_n, x \in \mathbb{R}$ bezeichnet. Man zeige

(a) Bezüglich des Knotenpolynoms $\omega_{n+1}(x) = (x - x_0)(x - x_1)\cdots(x - x_n)$ besitzen die Lagrange-Grundpolynome die Darstellung

$$\ell_j(x) = \frac{\omega_{n+1}(x)}{(x - x_j)\omega'_{n+1}(x_j)}, \qquad j = 0, \ldots, n.$$

Man zeige damit

$$f[x_0, x_1, \ldots, x_n, x] = \sum_{j=0}^{n} \frac{f[x, x_j]}{\prod\limits_{\substack{k=0 \\ k \neq j}}^{n} (x_j - x_k)} .$$

(b) Ist $f \in C[a,b]$ und existiert $f'(x_j)$ für $x_j \in [a,b]$, so ist die Funktion $g(x) := f[x, x_j]$ (x_j fest) mit der Festlegung $f[x_j, x_j] := f'(x_j)$ stetig auf $[a,b]$.

(c) Ist $f' \in C[a,b]$ und ist f'' in einer Umgebung von $x_j \in [a,b]$ stetig, so ist $g'(x) = d/dx f[x, x_j]$ eine auf $[a,b]$ stetige Funktion.
Hinweis: Taylor-Entwicklung um x_j.

(d) Ist $f \in C^2[a,b]$, so ist die Funktion $h(x) = f[x_0, x_1, \ldots, x_n, x]$ in $C^1[a,b]$.

(e) Seien $x_0, \ldots, x_n \in [a,b]$ paarweise verschieden, $f \in C[a,b]$ und es existiere $f'(x_n)$. Aus Teil (b) folgt

$$\lim_{h \to 0} f[x_0, , \ldots, x_n, x_n + h] = f[x_0, \ldots, x_n, x_n].$$

Folgern Sie hieraus die Beziehung

$$\frac{d}{dx_n} f[x_0, \ldots, x_n] = f[x_0, \ldots, x_{n-1}, x_n, x_n].$$

Hinweis: Nutzen Sie die ursprüngliche Definition der dividierten Differenzen und beachten Sie, dass diese symmetrisch bezüglich aller Argumente sind.