Dr. rer. nat. Armin Rund
Institut für Mathematik und Wiss. Rechnen
Karl-Franzens-Universität Graz

WS 16/17

Numerische Mathematik I

10. Übung

Die Aufgaben werden besprochen am 11.01.2016

1. **Konvergenzordnung**

   Sei (x_n) eine durch einen Algorithmus erzeugte Folge die quadratisch mit Konstante $C = 0.95$ gegen x^* konvergiert. Angenommen $|x_0 - x^*| <= 1$. Wie viele Iterationen sind nötig um sicherzustellen dass $|x_k - x^*| <= 10^{-10}$?

2. **Gauß-Newton-Verfahren**

   Gegeben seien folgende Stützstellen t_i und Meßwerte y_i

t_i	0	$\frac{\pi}{2}$	π
y_i	0	$\frac{\sqrt{2}}{2}$	1

   Aus theoretischen Überlegungen geht hervor, dass diese Meßdaten einer Funktion

   $$y\left(t; x_1, x_2\right) = \sin\left(x_1 t\right) + x_2 t$$

   genügen. Bestimmen Sie die Parameter x_1 und x_2 optimal im Sinne der kleinsten Fehlerquadrate, indem Sie

   - das nichtlineare Ausgleichsproblem formulieren und
   - ausgehend vom Startwert $\left(x_1^{(0)}, x_2^{(0)}\right)^T = \left(\frac{1}{2}, 2\right)^T$ einen Gauß-Newton-Schritt durchführen.

3. **Polynom-Interpolation**

Sei $\Lambda_n(K, I)$ die Lebesguekonstante bzgl. der Knotenmenge K auf dem Intervall I.

(a) Seien $K = \{t_0, \ldots, t_n\} \subset I = [a, b]$ paarweise verschiedene Knoten. Die Affintransformation

$$\chi : I \to I_0 = [-1, 1], \quad t \mapsto \frac{2t - a - b}{b - a}$$

dieses Intervalls auf das Einheitsintervall I_0 bilde die Knotenmenge K auf die Knotenmenge $K_0 = \chi(K)$ ab. Zeigen Sie, dass die Lebesguekonstante invariant unter dieser Transformation ist, d.h.

$$\Lambda_n(K, I) = \Lambda_n(K_0, I_0).$$

(b) Seien $K = \{t_0, \ldots, t_n\}$ mit $a \leq t_0 < t_1 < \cdots < t_n \leq b$ Knoten im Intervall $I = [a, b]$. Geben Sie die Affintransformation

$$\chi : [t_0, t_n] \to I$$

auf I an, so dass für $\bar{K} = \chi(K) = \{\bar{t}_0, \ldots, \bar{t}_n\}$ gilt

$$a = \bar{t}_0 < \bar{t}_1 < \cdots < \bar{t}_n = b.$$

Zeigen Sie, dass

$$\Lambda_n(\bar{K}, I) \leq \Lambda_n(K, I),$$

d.h. die Einbeziehung der Randknoten verbessert die Lebesguekonstante.

4. **Programmieraufgabe: Gedämpftes Newton-Verfahren (Abgabe bis Mi, 11.1.2017, 10:00 Uhr)**

Algorithm 1: Gedämpftes Newton-Verfahren mittels natürlichem (affin-invariantem) Monotonietest [Deuflhard p. 103f]

Input: stetige-differenzierbare Funktion $f : \mathbb{R}^n \to \mathbb{R}^n$ mit regulärer Jacobimatrix $f'(x)$, Startwert x_0, Toleranzen abstol > 0, reltol > 0, $0 < \lambda_{\min} \ll 1$

Output: Näherung für Nullstelle x_k

$k = 0$; $\lambda = 1$;
while $\|f(x_k)\| > \max(\text{abstol}, \text{reltol}\|f(x_0)\|)$ **do** `// Newton-Iteration`
    Löse $f'(x_k)\Delta_k = -f(x_k)$;
    $\lambda = \min(1, 2\lambda)$;
    Löse $f'(x_k)\tilde{\Delta} = -f(x_k + \lambda\Delta_k)$;
    while $\|\tilde{\Delta}\| > (1 - \frac{\lambda}{2})\|\Delta_k\|$ **do** `// Schrittweitensuche`
        $\lambda = \lambda/2$;
        if $\lambda < \lambda_{\min}$ **then** return; `// Kein weiterer Abstieg`
        Löse $f'(x_k)\tilde{\Delta} = -f(x_k + \lambda\Delta_k)$;
    $x_{k+1} = x_k + \lambda\Delta_k$; $k = k + 1$;

(a) Denken Sie sich ein Nullstellenproblem für $n \geq 2$ aus als Testproblem für ihr folgendes Programm.

(b) Programmieren Sie dieses gedämpfte Newtonverfahren in MATLAB mit dem Syntax

`function [x] = newton_ZZ(@myfun_ZZ,x_0)`

wobei ZZ Ihre Teamnummer ist. Die Funktion `myfun_ZZ` habe den üblichen Syntax (siehe Aufgabe 8.2). Typische Toleranzen sind z.B. abstol $= 10^{-4}$, reltol $= 10^{-5}$.

(c) Geben Sie je Newton-Iterationen die relevanten Daten formatiert $(k, \|f(x_k)\|, \|\Delta_k\|, \lambda)$ aus.

(d) Entwerfen Sie ein Test-Skript `main_ZZ`, welches ihr Testproblem mit ihrem Newton-Verfahren löst. Vergleichen Sie ihre Lösung darin mit einem MATLAB standard Verfahren.

Abgabe der Programmieraufgabe:

- Abgabe per email an `armin.rund@uni-graz.at`
- Betreff: Programmieraufgabe Newton
- In email bitte die Namen und Matrikelnr aller beteiligten Studierenden des Teams aufführen
- Abzugeben sind beide Funktionen und Ihr Test-Skript mit vorgegebenen Namen unter Verwendung der Teamnummer