Prof. Dr. Stefan Funken
M.Sc. Andreas Bantle
Dipl.-Math. oec. Klaus Stolle

Universität Ulm
Institut für Numerische Mathematik
Sommersemester 2015

Numerische Analysis - Matlab-Blatt 6

(Besprechung in den MATLAB-Tutorien in KW 27/28)

Hinweise:
Siehe MATLAB-Blatt 1/2.

Aufgabe 10 *(Addition, Multiplikation und Differentiation von Polynomen)* (9 Punkte)

Seien $p, q : \mathbb{R} \to \mathbb{R}$ Polynome mit $p(x) = \sum_{k=0}^{m} a_k x^k$ und $q(x) = \sum_{k=0}^{n} b_k x^k$, sowie $a_m \neq 0$ und $b_n \neq 0$.

a) Schreiben Sie drei Funktionen

(i) `c = add(a,b)`
(ii) `c = mult(a,b)`
(iii) `c = derivative(a)`

die zu den Spaltenvektoren $a = (a_0, \dots, a_m)^T$ und $b = (b_0, \dots, b_n)^T$ jeweils die Koeffizienten des Polynoms $p + q$, $p \cdot q$ und der Ableitung p' als Spaltenvektor `c` ausgeben sollen. Dabei soll der führende Koeffizient, also die letzte Koordinate des Koeffizientenvektors, ungleich Null sein. Das Nullpolynom soll durch den „leeren Vektor“ `[]` dargestellt werden.

b) Schreiben Sie ein Testskript `testPoly.m`, welches die Funktionen an den Beispielen

(i) $p(x) = 0\,,\quad q(x) = 4x^2 - 3$
(ii) $p(x) = 2x^2 - 3x\,,\quad q(x) = -2x^2 + 3x$

testet.

Aufgabe 11 *(Momente und Skalarprodukt)* (9 Punkte)

Sei $\omega : [r, s] \to \mathbb{R}$ eine positive, integrierbare Funktion. Ein Skalarprodukt $(\cdot,\cdot)_\omega$ über den Raum der Polynome $\mathbb{P}$ mit

$$(p, q)_\omega := \int_r^s p(x)q(x)\omega(x)\,dx$$

ist eindeutig durch seine Momente

$$\mu_n = \int_r^s x^n \omega(x)\,dx \qquad (n \in \mathbb{N}_0)$$

bestimmt.

a) Schreiben Sie eine Funktion

`mu = myMoment(n,name)`

welche für die Eingabeparameter $n \in \mathbb{N}_0$ und `name` als String die entsprechende Momente $\mu_0, ..., \mu_n$ berechnet. Dabei gilt für die Momente folgendes

Name	r	s	$\omega(x)$	μ_n
`'legendre'`	-1	1	1	n gerade: $\frac{2}{n+1}$, n ungerade: 0
`'log'`	0	1	$-\log x$	$\frac{1}{(n+1)^2}$
`'tscheb'`	-1	1	$\frac{1}{\sqrt{1-x^2}}$	n gerade: $\pi \cdot \frac{1\cdot 3\cdot 5\cdots n-1}{2\cdot 4\cdot 6\cdots n}$, n ungerade: 0
`'laguerre'`	0	∞	e^{-x}	$n!$
`'hermite'`	$-\infty$	∞	e^{-x^2}	n gerade: $\Gamma\left(\frac{n+1}{2}\right)$, n ungerade: 0

b) Schreiben Sie eine Funktion

```
s = scalar(a,b,name)
```

welche für die Eingabeparameter

- a: der Koeffizientenvektor eines Polynoms $p \in \mathbb{P}$
- b: der Koeffizientenvektor eines Polynoms $q \in \mathbb{P}$
- name: String, der die Gewichtsfunktion festlegt

das Skalarprodukt $(p, q)_\omega$ berechnet. *Hinweis: Verwenden Sie* `myMoment` *und* `mult`.

c) Schreiben Sie ein Testskript `testMomentScalar.m`, welches beide Funktionen ausreichend und sinnvoll für die von Ihnen gewählten Beispiele testet. Beachten Sie hierbei, dass Sie alle Varianten testen sollen. Aufrufe könnten beispielsweise wie folgt aussehen

```
myMoment(5,'tscheb') = [3.1416; 0; 1.5708; 0; 1.1781; 0]
scalar([1;0;-8;0;8],[0;5;0;-20;0;16],'tscheb') = 0
```

Aufgabe 12 *(Determinant Representation of Orthogonal Polynomials)* (12 Punkte)

A positiv weight function ω on an interval determines a system of orthogonal polynomials P_n uniquely, apart from a constant factor. Using the moments of the weight function, i.e.

$$\mu_n = \int_r^s x^n \omega(x)\, dx,$$

we find a determinant representation formula for a point wise evaluation of such orthogonal polynomials. The representation is $(P_0(z) = 1)$ and

$$P_n(z) = \frac{\det M}{\det T} \qquad n = 1, ..$$

with

$$M = \begin{pmatrix} \mu_0 & \mu_1 & \mu_2 & \cdots & \mu_n \\ \mu_1 & \mu_2 & \mu_3 & \cdots & \mu_{n+1} \\ \vdots & \vdots & \vdots & & \vdots \\ \mu_{n-1} & \mu_n & \mu_{n+1} & \cdots & \mu_{2n-1} \\ 1 & z & z^2 & \cdots & z^n \end{pmatrix} \in \mathbb{C}^{(n+1)\times(n+1)} \quad \text{and} \quad T = \begin{pmatrix} \mu_0 & \mu_1 & \mu_2 & \cdots & \mu_{n-1} \\ \mu_1 & \mu_2 & \mu_3 & \cdots & \mu_n \\ \vdots & \vdots & \vdots & & \vdots \\ \mu_{n-1} & \mu_n & \mu_{n+1} & \cdots & \mu_{2n-2} \end{pmatrix} \in \mathbb{R}^{n\times n}.$$

a) Write a function

```
Pz = evalOrtho(z,n,name)
```

which calculates $P_n(z)$ for $n \in \mathbb{N}_0$ corresponding to the weight function defined through `name`.

b) Write a test script `plotOrtho.m`, which generates six plots

(i) $P_n(z)$ for $z \in [-1, 1]$ with `name = 'legendre'` and `n=4,5,6`

(ii) $P_n(z)$ for $z \in [0, 1]$ with `name = 'log'` and `n=4,5,6`

(iii) $P_n(z)$ for $z \in [-1, 1]$ with `name = 'tscheb'` and `n=4,5,6`

(iv) $P_n(z)$ for $z \in [0, 5]$ with `name = 'laguerre'` and `n=4,5,6`

(v) $P_n(z)$ for $z \in [-2, 2]$ with `name = 'hermite'` and `n=4,5,6`

(vi) $\log |P_n(z)|$ for $z \in \mathbb{C}$ with $\mathrm{Re}\,(z)$, $\mathrm{Im}\,(z) \in [-4, 4]$, `name = 'legendre'` and `n=5`
