CAAM 353: Computational Numerical Analysis

Homework 10, March 27, 2008

Due: April 10, 2008

Note: All MATLAB *functions mentioned in this homework assignment can be found on the CAAM353 homepage, or come with* MATLAB *. Turn in all* MATLAB *code that you have written and turn in all output generated by your* MATLAB *functions/scripts.* MATLAB *functions/scripts must be commented and output must be formatted nicely.*

This homework is pledged! You may use your notes and books for this homework. However, you are not allowed to discuss this homework assignment with another person, except your instructor. **Write out and sign the pledge.**

Problem 1 (35 points)

Given data

x_1	x_2	$\cdots$	x_n
f_1	f_2	$\cdots$	f_n
f_1'	f_2'	$\cdots$	f_n'

.

(think of $f_i = f(x_i)$ and $f_i' = \frac{d}{dx} f(x_i)$) we want to compute a polynomial p_{2n-1} of degree at most $2n-1$ such that

$$p_{2n-1}(x_i) = f_i, \qquad i = 1, \dots, n, \tag{1a}$$

$$\frac{d}{dx} p_{2n-1}(x_i) = f_i', \qquad i = 1, \dots, n. \tag{1b}$$

This is called *Hermite interpolation*.

a. (10 points) Write the polynomial $p_{2n-1}(x) = a_1 + a_2 x + \ldots + a_{2n} x^{2n-1}$ in the monomial basis and describe how to solve the Hermite interpolation problem (1).

Write a MATLAB program that applies your procedure to compute and plot the Hermite interpolation polynomial for the data

x_i	0	2	3
f_i	1	0	2
f_i'	0	1	-1

.

What is the Hermite interpolation polynomial in monomial basis?

b. (10 points) Consider the polynomials

$$h_{2i-1}(x) = \prod_{j \neq i} \left(\frac{x - x_j}{x_i - x_j} \right)^2, \quad h_{2i}(x) = (x - x_i) \prod_{j \neq i} \left(\frac{x - x_j}{x_i - x_j} \right)^2, \quad i = 1, \dots, n,$$

and define the so-called *Hermite polynomials*

$$H_{2i-1}(x) = h_{2i-1}(x) - h_{2i-1}'(x_i) h_{2i}(x), \quad H_{2i}(x) = h_{2i}(x), \quad i = 1, \dots, n.$$

The Hermite polynomials are of degree at most $2n-1$. Show that the Hermite polynomials satisfy

$$H_{2i-1}(x_k) = \begin{cases} 1 & \text{if } k = i \\ 0 & \text{if } k \neq i \end{cases}$$
$$H_{2i}(x_k) = 0 \quad \forall k$$
$$\frac{d}{dx}H_{2i-1}(x_k) = 0 \quad \forall k$$
$$\frac{d}{dx}H_{2i}(x_k) = \begin{cases} 1 & \text{if } k = i \\ 0 & \text{if } k \neq i \end{cases}$$

for $i = 1, \ldots, n$. (Hint: Observe that $h_{2i}(x) = (x - x_i)h_{2i-1}(x)$.)

Express the Hermite interpolation polynomial for the data given in part a. using Hermite basis polynomials $H_1, \ldots, H_{2n}$.

c. (15 points) Consider the polynomials

$$\begin{aligned}
N_1(x) &= 1, \\
N_2(x) &= (x - x_1), \\
N_3(x) &= (x - x_1)^2, \\
N_4(x) &= (x - x_1)^2(x - x_2), \\
&\vdots \\
N_{2i-1}(x) &= (x - x_1)^2 \ldots (x - x_{i-1})^2, & i &= 3, \ldots, n, \\
N_{2i}(x) &= (x - x_1)^2 \ldots (x - x_{i-1})^2(x - x_i), & i &= 3, \ldots, n.
\end{aligned}$$

These polynomials are of degree at most $2n-1$. Show that they satisfy

$$\begin{aligned}
N_{2i-1}(x_j) &= 0 & &\text{for } j < i, \\
\frac{d}{dx}N_{2i-1}(x_j) &= 0 & &\text{for } j < i, \\
N_{2i}(x_j) &= 0 & &\text{for } j \leq i, \\
\frac{d}{dx}N_{2i}(x_j) &= 0 & &\text{for } j < i,
\end{aligned}$$

for $i = 1, \ldots, n$.

Describe how to solve the Hermite interpolation problem (1) using the basis polynomials $N_1, \ldots, N_{2n}$.

Write a MATLAB program that applies your procedure to compute the Hermite interpolation polynomial for the data given in part a. (Your program should use the properties of the polynomials $N_1, \ldots, N_{2n}$ as much as possible). Plot the Hermite interpolation polynomial.

What is the Hermite interpolation polynomial in the basis $N_1, \ldots, N_{2n}$?

Problem 2 (25 points) The cubic polynomials

$$H_0(x) = (1-x)^2(1+2x),$$
$$H_1(x) = x^2(3-2x),$$
$$h_0(x) = x(1-x)^2,$$
$$h_1(x) = x^2(x-1)$$

satisfy (here $'$ denotes differentiation with respect to x)

$$\begin{array}{llll} H_0(0)=1 & H_0(1)=0 & H_0'(0)=0 & H_0'(1)=0, \\ H_1(0)=0 & H_1(1)=1 & H_1'(0)=0 & H_1'(1)=0, \\ h_0(0)=0 & h_0(1)=0 & h_0'(0)=1 & h_0'(1)=0, \\ h_1(0)=0 & h_1(1)=0 & h_1'(0)=0 & h_1'(1)=1. \end{array}$$

Given a subdivision

$$a = x_0 < x_1 < \ldots < x_n = b$$

of the interval $[a,b]$ and data

x_0	x_1	$\cdots$	x_n
f_0	f_1	$\cdots$	f_n
f_0'	f_1'	$\cdots$	f_n'

.

we want to find a function $P : [a,b] \to \mathbb{R}$ such that

- On each $[x_i, x_{i+1}]$, $i = 0, \ldots, n-1$, P is a polynomial of degree at most 3.
- $P(x_i) = f_i$ and $P'(x_i) = f_i'$, $i = 0, \ldots, n$.

a. (10 points) Describe how the function P can be constructed using the polynomials H_0, H_1, h_0, h_1.

b. (15 points) Write a MATLAB program that implements your approach and apply it to approximate Runge's function $f(x) = 1/(1+x^2)$ on $[-5,5]$ using equally spaced points.
Plot the computed function for $n = 5$ and for $n = 10$.

Problem 3 (15 points) We are given values (x_i, y_i), $i = 0, \ldots, n$, and we want to fit a piecewise smooth curve through these point. Specifically, we want to find piecewise cubic functions $x(t), y(t)$ such that $x(t_i) = x_i$, $y(t_i) = y_i$, $i = 0, \ldots, n$. The t_i's are computed recursively using

$$\begin{aligned} t_0 &= 0, \\ t_i &= t_{i-1} + \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}, \quad i = 1, \ldots, n. \end{aligned}$$

Consider the points

i	0	1	2	3	4	5	6	7	8	9	10	11
x_i	25.0	19.0	13	9.0	5.0	2.2	1.0	3.0	8.0	13.0	18.0	25.0
y_i	5.0	7.5	9.1	9.4	9.0	7.5	5.0	2.1	2.0	3.5	4.5	5.0

Use (natural) cubic splines to compute smooth piecewise cubic polynomials $x(t), y(t)$ that fit the data above.

Return a plot that shows the data (x_i, y_i), $i = 0, \ldots, n$, as well as the computed $(x(t), y(t))$ (use a fine partition of $[t_0, t_n]$ to obtain a smooth plot).