Math 391 | **Assignment 1** | **F2019**

Polynomial Interpolation

Due: Thursday, October 10

01. Let $P_n(x)$ be the degree n polynomial that interpolates the function $f(x)$ at the pairwise distinct nodes $\left\{ x_i \right\}_{1\le i\le n+1}$, written in Newton's form

$$P_n(x) = a_1 + a_2(x-x_1) + a_3(x-x_1)(x-x_2) + \cdots + a_{n+1}(x-x_1)(x-x_2)\cdots(x-x_n)$$

where the coefficients are given by the divided differences

$$a_1 = f[x_1],\ a_2 = f[x_1,x_2], a_3 = f[x_1,x_2,x_3], \cdots, a_{n+1} = f[x_1,x_2,\cdots,x_{n+1}]$$

a. Write a MATLAB function newton_dd() that takes as arguments the nodes $X = [x_1\,;\cdots;\,x_{n+1}]$, the function values $F = [f(x_1)\,;\cdots;\,f(x_{n+1})]$, the degree n of the polynomial $P_n(x)$, and returns the coefficients $a = [a_1\,;\,a_2\,;\cdots;\,a_{n+1}]$.

b. Write a MATLAB function plot_newton_poly() that sketches the graph of $P_n(x)$ and the interpolation points $\left\{(x_i, f(x_i))\right\}_{1\le i\le n+1}$, and displays both on the same plot. The function should take as arguments the number of points, the nodes $X = [x_1\,;\cdots;\,x_{n+1}]$, and the function values $F = [f(x_1)\,;\cdots;\,f(x_{n+1})]$. The plot should be based on at least 100 evaluations of $P_n(x)$. You may use the MATLAB function Horner_Newton() that appears at the end of this assignment, to perform the 100 evaluations of $P_n(x)$.

To test your functions, use the data points

$$(-1, f(-1))\ ,\ (-3/4, f(-3/4))\ ,\ (-1/4, f(-1/4))\ ,\ (0, f(0))\ ,\ (1/2, f(1/2)\ ,\ (1, f(1))$$

where $f(x) = \dfrac{4}{1+4x^2}$.

02. The degree n Lagrange polynomials associated with the points $x_0 < x_1 < \cdots < x_n$, are defined by

$$L_{n,j}(x) = \prod_{\substack{i=0\\ i\ne j}}^{n} \frac{x-x_i}{x_j-x_i}, \quad j = 0,\, 1,\, \cdots,\, n$$

Assuming the points to be equidistant, with $x_0 = 0$ and $x_n = 1$, use MATLAB to sketch in the same plot the graph of $L_{3,0}(x)$, $L_{3,1}(x)$, $L_{3,2}(x)$, and $L_{3,3}(x)$

03. Let n be a natural number, and define

$$T_n(x) = \cos\big(n \cos^{-1}(x)\big), \qquad -1 \le x \le 1$$

Recall that $\cos^{-1}(x)$ is the inverse function of $\cos(x)$. Its domain is $[-1\,,\,1]$, and its range is $[0\,,\,\pi]$

a. Display in the same plot the graphs of $T_i(x)$, $i = 1, 2, 3, 4, 5$

b. Show that $T_n(x)$ is a polynomial of degree n. $T_n(x)$ is known as the degree n Chebyshev polynomial.
Hint: Make use of the identity $\cos\big((n+1)\,\theta\big) + \cos\big((n-1)\,\theta\big) = 2\,\cos(\theta)\,\cos(n\,\theta)$ to show that $T_{n+1}(x) = 2\,x\,T_n(x) - T_{n-1}(x)$

c. Find the roots $\Big\{\,\xi_i\,\Big\}_{1\le i\le n}$ of $T_n(x)$

04. Recall that if $P_n(x)$ is the degree n polynomial that interpolates $f(x)$ at the set of points $\big\{x_0, x_1, \cdots, x_n\big\}$, and if $f(x)$ has $(n+1)$ continuous derivatives in an interval $[a, b]$ that contains the interpolating points, then

$$f(x) - P_n(x) = \frac{f^{(n+1)}(\theta)}{(n+1)!}\,(x - x_0)\,(x - x_1)\,\cdots\,(x - x_n) \qquad (*)$$

That means, the error $f(x) - P_n(x)$ is proportional to the size of $f^{(n+1)}(x)$ and to the size of the polynomial $\omega_n\big(x,\ x_0,\ x_1,\ \cdots,\ x_n\big) = \big(x - x_0\big)\,\big(x - x_1\big)\,\cdots\,\big(x - x_n\big)$.

Let $f(x) = \dfrac{1}{1 + 25\,x^2}$.

a. Assuming the points equidistant, i.e., $x_i = -1 + 2\,\dfrac{i}{n}$, $i = 0, 1, \cdots, n$, sketch the graphs of $f(x), P_5(x),$, $P_{10}(x)$, $P_{20}(x)$, and display them in the same plot. Comment on the closeness of the interpolating polynomials to the function $f(x)$ around the middle and the end points of $[-1\,,\,1]$.

b. Repeat question **(a)** with x_i, $i = 0, 1, \cdots, n$, being the roots of the degree $(n+1)$-Chebyshev polynomial $T_{n+1}(x)$.

05. Let $P_n(t)$ be the degre n polynomial that interpolates $f(t)$ at the distinct points $\big\{x_0, x_1, \cdots, x_n\big\}$. Show that if $t \ne x_i$, for $i = 0, 1, \cdots, n$, then

$$f(t) = P_n(t) + f\big[x_0, x_1, \cdots, x_n, t\big]\ \big(t - x_0\big)\big(t - x_1\big)\cdots\big(t - x_n\big)$$

Hint: do you recognize what the right side is?

```
function   y = Horner_Newton(c, a, x, n)
    %
    % Evaluates  a(1)+a(2)(x-x(1))+a(3)(x-x(1))(x-x(2))+\cdots
    %            a(n+1)(x-x(1))(x-x(2))\cdots(x-x(n))
    %
    % Input    c   the value at which the polynomial is to be evaluated
    %          n   the degree of the polynomial
    %          a   (n+1) by 1  array that contains the coefficients of the
    %              polynomial
    %          x   the x-components of the data points.
    %
    % Output   y   the value of the polynomial at x=c
    %
    y = a(n+1);
    for i=1:n
        y = y*(c-x(n+1-i))+a(n+1-i);
    end
```