Numerical Analysis and Computational Mathematics

Fall Semester 2019 - Section CSE

Dr. Rafael Vázquez Hernández

Assistant: Ondine Chanon

Session 5 - 16 October 2019

Approximation of functions and data

Exercise I (MATLAB, tutorial)
For any set of couples $\{(x_i, y_i)\}_{i=0}^n$ such that $x_i \neq x_j$ when $i \neq j$, we can compute and evaluate the *interpolating polynomial* or a *least–squares approximating polynomial* by using the MATLAB commands `polyfit` and `polyval`.

Example 1: Let us consider the following data set composed by couples $\{(x_i, y_i)\}_{i=0}^4$:

i	0	1	2	3	4
x_i	0	0.25	0.5	0.75	1
y_i	3.38	3.86	3.85	3.59	3.49

The coefficients of the interpolating polynomial of degree $n = 4$ are computed using the following commands:

```
x_nodes = [0:0.25:1];
y_nodes = [3.38 3.86 3.85 3.59 3.49];
P = polyfit( x_nodes, y_nodes, 4 )   % Coefficients of the L. polynomial
%  P = 1.8133   -0.1600   -4.5933    3.0500    3.3800
```

where `P` is a vector containing the coefficients of the polynomial $\Pi_4(x) = 1.8133x^4 - 0.16x^3 - 4.5933x^2 + 3.05x + 3.38$. Given any vector of points `x_values`, we correspondingly evaluate the polynomial using the command `P_values = polyval( P, x_values )`, e.g.:

```
% polynomial value in a single point
x_value = 0.4;
P_value = polyval( P, x_value )    % value of the polynomial at point x_value
%  P_value =
%      3.9012

% polynomial values in multiple points
x_values = linspace( 0, 1, 1001);  % 1001 equispaced points between 0 and 1
P_values = polyval( P, x_values ); % values of the polynomial at points x_values
plot( x_values, P_values )         % plot the interpolating polynomial
```

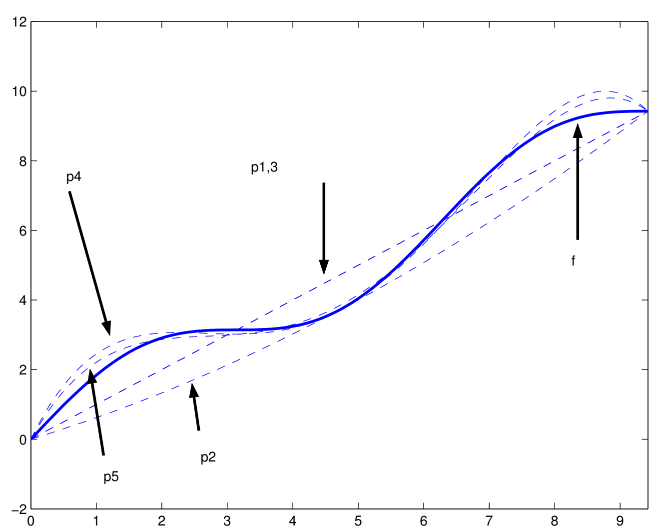

Example 2: To compute the interpolating polynomial of degree n of an arbitrary continuous function $f(x)$ we have to define a set of $n+1$ couples $\{(x_i, f(x_i))\}$, with x_i distinct nodes; e.g. for $f(x) = \cos(x)$, $n = 4$, and $n+1 = 5$ equally distributed nodes in $I = [a, b] = [0, 1]$ we execute the following commands:

```
f = @(x) cos(x); a=0; b=1;
n = 4;
x_nodes = linspace( a, b, n +1 );
y_nodes = f(x_nodes);
P = polyfit( x_nodes, y_nodes, n )
%  P =
%       0.0362    0.0063   -0.5025    0.0003    1.0000
```

Remark: If the length of x_nodes (and y_nodes) is larger than $n+1$ (where n is the degree of desired interpolating polynomial) the command polyfit(x_nodes, y_nodes, n) returns the *least–squares approximating polynomial* of degree n.

Example 3: Let us interpolate the function $f(x) = \sin(x) + x$ at $n + 1 = 2, 3, \ldots, 6$ equally distributed nodes in $I = [0, 3\pi]$ by means of interpolating polynomials $\Pi_n f(x)$ of degree n; we use the following commands:

```
f = @(x) sin(x) + x; a=0; b=3*pi;
x_values = linspace(a,b,1001);
plot( x_values, f(x_values), 'b' );
hold on
for n = 1:5
    x_nodes = linspace( a, b, n+1 );
    P = polyfit( x_nodes, f(x_nodes), n );
    plot( x_values, polyval( P, x_values ), '--b' )
end
```



Exercise II (MATLAB)
Let us consider the function $f(x) = \sin(x)$ on the interval $I = [a, b] = [0, 3\pi]$.

a) By equally distributing the nodes in the interval I, compute the interpolating polynomial $\Pi_n f$ of the function $f(x)$ for $n = 1, \ldots, 7$, where n is the degree of the polynomial. Compare the results obtained with the analytical expression of $f(x)$. (*Hint: plot* $\Pi_n f$ *and* $f(x)$ *on the same figure using at least 1001 nodes.*)

b) Compute the true errors $e_n(f)$:

$$e_n(f) := \max_{x \in I} |E_n f(x)|, \qquad \text{with } E_n f(x) := f(x) - \Pi_n f(x),$$

for $n = 1, \ldots, 7$, plot them in a figure as a function of n, and comment the result.

c) We recall that, for interpolating polynomials of functions $f(x) \in C^{n+1}(I)$ at equally spaced nodes, the true errors $e_n(f)$ can be estimated as:

$$e_n(f) \leq \widetilde{e}_n(f), \qquad \text{where } \widetilde{e}_n(f) := \frac{1}{4(n+1)} \left(\frac{b-a}{n} \right)^{n+1} \max_{x \in I} \left| f^{(n+1)}(x) \right|.$$

Plot in the same figure of point b) the behavior of the error estimators $\widetilde{e}_n(f)$ vs. the degree n and verify the validity of the estimation.

Exercise III (Theoretical)
Let us consider the function $f(x) = -x^3 + 3x^2 - 2$ with $x \in I = [a, b] = [0, 2]$.

a) Compute the Lagrange interpolating polynomial of degree 2, say $\Pi_2 f(x)$, of the function $f(x)$ at the nodes $x_0 = a = 0$, $x_1 = \frac{1}{2}$, and $x_2 = b = 2$. Provide the expressions of the Lagrange characteristic functions.

b) Repeat point a) with $x_1 = 1$ and motivate the result obtained.

c) Which are the coefficients of the Lagrange polynomial of degree $n = 3$, say $\Pi_3 f(x)$, which interpolates $f(x)$ at the nodes $x_0 = 0$, $x_1 = e^{-\sqrt{2}}$, $x_2 = 3e^{-\sqrt{1/2}}$ and $x_3 = 2$?

Exercise IV (MATLAB)
Let us consider the Runge function $f(x) = \dfrac{1}{1+x^2}$ in the interval $I = [a, b] = [-5, 5]$.

a) By uniformly distributing the nodes in I, use MATLAB to compute and plot the interpolating polynomials $\Pi_n f(x)$ of the function $f(x)$ for the degrees $n = 2, 4, 8$, and 12. Compare the results with the plot of the function $f(x)$.

b) Compute the true errors $e_n(f) := \max_{x \in I} |f(x) - \Pi_n f(x)|$ for $n = 2, 4, 8$, and 12, and plot them on a figure vs. the degree n. Is the error decreasing when n increases? Motivate the answer.

c) Repeat point a) by using the Chebyshev–Gauss–Lobatto distribution of the nodes in I. In this case the $n + 1$ nodes x_i are obtained as:

$$x_i = \frac{a+b}{2} + \frac{b-a}{2} \hat{x}_i, \qquad \text{where } \hat{x}_i = -\cos\left(\pi \frac{i}{n} \right), \quad i = 0, \ldots, n.$$

For $n = 8$ compare the interpolating polynomial with the corresponding polynomial obtained at point a) for equally spaced nodes and the function $f(x)$.

d) Repeat point b) by using the Chebyshev–Gauss–Lobatto distribution of the nodes in I obtained at point c) and motivate the results obtained.

Exercise V (Theoretical)

Let us consider the function $f(x) = \sin\left(\frac{x}{3}\right)$ in the interval $I = [a, b] = [0, 1]$.

a) Let $\Pi_n f(x)$ be the interpolating polynomial of $f(x)$ at the $n + 1$ equally spaced nodes $x_0, x_1, \ldots, x_n$, with $x_0 = a$ and $x_n = b$. Provide an estimate, say $\widetilde{e}_n(f)$, of the interpolation error $e_n(f) := \max_{x \in I} |f(x) - \Pi_n f(x)|$ in the interval I as a function of the degree n of the polynomial $\Pi_n f(x)$ and study its limit for $n \to \infty$.

b) Find the minimum number of equally spaced nodes in I which guarantee that $e_n(f) < 10^{-4}$. (*Hint: try for* $n = 1, 2, 3, \ldots$ *until the condition is satisfied*).

c) Let us consider now the polynomial interpolation at the Chebyshev–Gauss–Lobatto nodes. Compute the first four Chebyshev–Gauss–Lobatto nodes x_0, x_1, x_2, x_3 in the interval I.

d) We define $\omega_n(x) := \prod_{i=0}^{n}(x - x_i)$, with x_i, for $i = 0, \ldots, n$, the nodes used for the interpolation. By setting $n = 3$ and by choosing the Chebyshev–Gauss–Lobatto nodes obtained at point c), we obtain the plot of $|\omega_3(x)|$ vs. $x \in I$ in Figure 1. Use the plot to estimate the interpolation error $e_n(f)$ corresponding to the Chebyshev–Gauss–Lobatto nodes in I for $n = 3$.

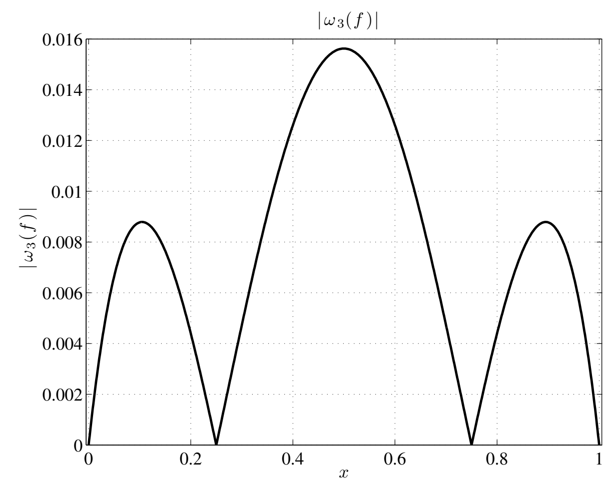


Figure 1: Function $|\omega_3(x)|$ in the interval I.