21/20

Cristian B. Jetomo

MATH 174 – B2L

Exercise 1.2 – Error in Polynomial Interpolation

(3pts) 1. We want to estimate an upper bound for the absolute error by constructing a formula for E which involves a higher-order derivative of f and an unknown number $\xi \in [0,0.5]$ which is dependent on x.
By some properties of integrals,

$$E = \left| \int_0^{0.5} f(x)dx - \int_0^{0.5} P_7(x)dx \right|$$
$$= \left| \int_0^{0.5} (f(x) - P_7(x))dx \right|$$

Note that by Theorem 2.4, since P_7 is the algebraic interpolating polynomial for f relative to the eight equally spaced points $x_1, x_2, \dots, x_8 \in [0, 0.5]$, then

$$f(x) = P_7(x) + E_7$$
$$\implies E_7 = f(x) - P_7(x)$$

where

$$E_7 = \frac{f^{(8)}(\xi)}{(8)!} \cdot \prod_{i=1}^{8}(x - x_i)$$

Hence,

$$E = \left| \int_0^{0.5} (E_7)dx \right| \checkmark$$

this is overkill but an acceptable upper bound

Note that a property of integral states that $\boxed{\left|\int_a^b f(x)dx\right| \leq \int_a^b |f(x)|dx.}$
Thus,

$$E \leq \int_0^{0.5} |E_7|dx$$

we could've settled for

$$E \leq \max_{\xi\in[0,0.5]} \left|\frac{f^{(8)}(\xi)}{8!}\right| \cdot \left|\int_0^{0.5} w(x)\,dx\right|$$

Now, note that

$$|E_7| \leq \max_{x\in[0,0.5]} \left| \frac{f^{(8)}(\xi)}{8!} \cdot \prod_{i=1}^{8}(x - x_i) \right|$$

Therefore, an upper bound for E is given by

$$E \leq \int_0^{0.5} \max_{x\in[0,0.5]} \left| \frac{f^{(8)}(\xi)}{8!} \cdot \prod_{i=1}^{8}(x - x_i) \right| dx$$

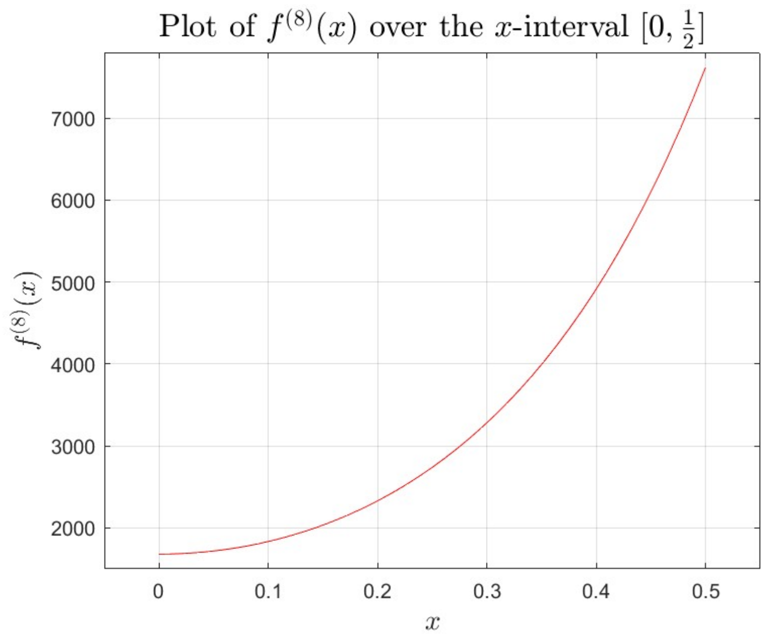
2. Using the ~~obtained formula~~ inequality for E, we can obtain an estimate for its maximum possible value. Since $\frac{1}{8!}$ is a positive constant, we can reconstruct the formula as follows:

$$E \leq \int_0^{0.5} \max_{x\in[0,0.5]} \left| \frac{f^{(8)}(\xi)}{8!} \cdot \prod_{i=1}^{8} (x - x_i) \right| dx$$
$$= \frac{1}{8!} \int_0^{0.5} \max_{x\in[0,0.5]} \left| f^{(8)}(\xi) \cdot \prod_{i=1}^{8} (x - x_i) \right| dx$$

Now, consider the 8th order derivative $f^{(8)}(x)$, which using MATLAB, we obtain to be as follows:

```
1680.0*exp(x^2) + 13440.0*x^2*exp(x^2) + 13440.0*x^4*exp(x^2) + 3584.0*x^6*exp(x^2) + 256.0*x^8*exp(x^2)
```

~~It's~~ Its plot is also illustrated below as



(5pts)

Figure 1.2.1: Plot of $f^{(8)}(x)$ over the x-interval $[0, 0.5]$

As observed, $f^{(8)}(x)$ is increasing over the specified interval. This implies that the maximum absolute value of the function occurs whenever it is evaluated at the largest interpolatory abscissa $x = 0.5$. Additionally from the graph, we see that $f^{(8)}(0.5) > 0$. Hence, the inequality for E can be expressed as

$$E \leq \frac{f^{(8)}(0.5)}{8!} \int_0^{0.5} \left| \prod_{i=1}^{8} (x - x_i) \right| dx$$

Now, we calculate for the definite integral of the absolute value of the polynomial $\omega(x) = \prod_{i=1}^{8}(x - x_i)$. Using MATLAB, we obtain that $\omega(x)$ is equal to

```
x^8 - 2.0*x^7 + 1.643*x^6 - 0.7143*x^5 + 0.1762*x^4 - 0.02442*x^3 + 0.001736*x^2 - 4.781e-5*x
```

Consequently, we get an estimate for an upper bound of E which is given by

(5pts)

$$\begin{aligned} E &\leq \frac{f^{(8)}(0.5)}{8!} \int_0^{0.5} \left| \prod_{i=1}^{8} (x - x_i) \right| dx \\ &= 9.3000 \times 10^{-9} \end{aligned}$$

3. We then compute for the approximate relative error given by

$$\begin{aligned} E_{rel} &\approx \frac{\left| \int_0^{0.5} f(x)dx - \int_0^{0.5} P_7(x)dx \right|}{\left| \int_0^{0.5} P_7(x)dx \right|} \\ &= \frac{E}{\left| \int_0^{0.5} P_7(x)dx \right|} \end{aligned}$$

From our obtained estimated upper bound for E, we can then estimate an upper bound for the relative error E_{rel}. Using MATLAB, the estimated upper bound for the relative error E_{rel} is

(2pts)

```
1.706e-8
```

This obtained value is comparably larger than the obtained relative error in Example 2.6 which is 2.94×10^{-11}. One possible reason for this higher value is that the interpolating polynomial P_7 is obtained using only seven interpolatory abscissas compared to the eight abscissas used in constructing P_8 in the example. Another is that the upper bound for E_{rel} is constructed as an estimate of the maximum possible value for the relative error.

4. Now, we want to find the upper bound for the relative error of the algebraic interpolating polynomial Q_7 relative to the following abscissas:

0.00480367989919189, 0.0421325969243644, 0.495196320100808 0.457867403075636, 0.111107441745099, 0.388892558254901 0.201227419495968 and 0.298772580504032

That is, similarly with item 3, we want to find the following:

$$E_{rel} \approx \frac{\left| \int_0^{0.5} f(x)dx - \int_0^{0.5} Q_7(x)dx \right|}{\left| \int_0^{0.5} Q_7(x)dx \right|}$$

Using the same process as in items 2 and 3, we obtain by the use of MATLAB the estimated upper bound for the relative error which is

(4pts)

```
1.313e-8
```

Compared with the upper bound for the relative error for the integral when using P_7, this new error bound is relatively smaller. From this, it can then be implied that the algebraic interpolating polynomial Q_7 is a better fit for the integral $\int_0^{0.5} e^{x^2} dx$ than P_7 due to it having a smaller error bound obtained. We can also infer from this that we can generate better fitting algebraic interpolating polynomials which would result with lower relative errors by choosing proper interpolatory abscissas.

(1pt)

(1pt)

Appendix A

Program Used in MATLAB for Items 1 to 3

```
clear

close all

syms x  % Declares x as a symbolic variable
%%
X = linspace(0, 0.5, 8);  % generate 8 equally-spaced points from 0 to 0.5
F = exp(X.^2);  % obtains f(x) values for x's in X
f = exp(x^2);  % symbolic function f(x) = e^(x^2)

numPts = length(X);
interval = [0, 0.5];

%% Calculation of functions to maximize

% f^(8)
f8Prime = diff(f, 8);  % obtains 8th order derivative of f
f8Prime = simplify(f8Prime);
disp('f^(8) =')
disp(vpa(expand(f8Prime), 4))

% plotting f8Prime
fplot(f8Prime, interval, 'r')
grid
xlim([-0.05 0.55])
ylim([1500 7800])
xl = xlabel('$x$');
yl = ylabel('$f^{(8)}(x)$');
title('Plot of $f^{(8)}(x)$ over the $x$-interval $[0, \frac{1}{2}]$', ...
    'Interpreter', 'Latex', 'Fontsize', 16)
for j = [xl yl]
    set(j, 'Interpreter', 'Latex', ...
        'Fontsize', 14)
end

% omega
omega = 1;
for i = 1:numPts
    omega = omega * (x - X(i));
end
omega = simplify(omega);
disp('omega(x) =')
disp(vpa(expand(omega), 4))

% integrating omega
integral = int(abs(omega), 0, 0.5);
```

alternatively, abs(int(omega,0,0.5));

```
%% Calculation of Error Bound
maxf8Prime = double(subs(f8Prime, 0.5));
Ebound = (maxf8Prime/factorial(numPts)) * integral;
disp('E <=')
```

```
disp(vpa(expand(Ebound), 4))

%% Calculation of AIP P
P = 0;
for k = 1:numPts
    lk = 1;
    for i = 1:numPts
        if i ~= k
            lk = lk * ((x - X(i))/(X(k) - X(i)));
        end
    end
    P = P + F(k) * lk;
end

P = simplify(P);

%% Calculation of Approximate Relative Error
Erel = Ebound/int(P, 0, 0.5);
disp('E_rel is approximately')
disp(vpa(expand(Erel),4 ))
```

Appendix B

Program Used in MATLAB for Item 4

```
clear
close all

syms x  % Declares x as a symbolic variable
%%
X = [0.00480367989919189 0.0421325969243644 0.495196320100808 0.457867403075636
0.111107441745099 0.388892558254901 0.201227419495968 0.298772580504032];
F = exp(X.^2);  % obtains f(x) values for x's in X
f = exp(x^2);  % symbolic function f(x) = e^(x^2)

numPts = length(X);
interval = [0, 0.5];

%% Calculation of functions to maximize

% f^(8)
f8Prime = diff(f, 8);  % obtains 8th order derivative of f
f8Prime = simplify(f8Prime);

% omega
omega = 1;
for i = 1:numPts
    omega = omega * (x - X(i));
end
omega = simplify(omega);

% integrating omega
integral = int(abs(omega), 0, 0.5);

%% Calculation of Error Bound
maxf8Prime = double(subs(f8Prime, 0.5));
Ebound = (maxf8Prime/factorial(numPts)) * integral;

%% Calculation of AIP P
Q = 0;
for k = 1:numPts
    lk = 1;
    for i = 1:numPts
        if i ~= k
            lk = lk * ((x - X(i))/(X(k) - X(i)));
        end
    end
    Q = Q + F(k) * lk;
end

Q = simplify(Q);

%% Calculation of Approximate Relative Error
Erel = Ebound/int(Q, 0, 0.5);
disp('E_rel is approximately')
disp(vpa(expand(Erel),4 ))
```