

Problem 1.

(25 points)

- (a) Use appropriate Lagrange interpolating polynomials of degrees one, two, and three to approximate $f(0.43)$ if

$$f(0) = 1, f(0.25) = 1.64872, f(0.5) = 2.71828, f(0.75) = 4.48169$$

- (b) Use Neville's method to obtain the approximations for (a). (You may use Matlab or other softwares to solve this part. Copies of codes and results must be submitted)
- (c) The data in (a) is generated using $f(x) = e^{2x}$. Use the error formula to find a bound for the error, and compare the bound to the actual error for the cases $n=1$ and $n=2$.
- (d) Repeat (a) using Newton divided-difference formula. (You may use Matlab or other softwares to solve this part. Copies of codes and results must be submitted.)

Solution**Part (a)**

For the first degree polynomial we use the points 0.25 and 0.5 as we would like to approximate 0.43 within their bounds.

$$\begin{aligned} L_1(x) &= 1.64872 \frac{(x - 0.5)}{0.25 - 0.5} + 2.71828 \frac{(x - 0.25)}{0.5 - 0.25} \\ &= 4.27824x + 0.57916 \end{aligned}$$

$$L_1(0.43) = 2.4188032$$

For the second degree polynomial I could either the first or last point, I found that adding the last point produced a more accurate solution, so I'll use that here.

$$\begin{aligned} L_2(x) &= 1 \times \frac{(x - 0.5)(x - 0.75)}{(0.25 - 0.5)(0.25 - 0.75)} + 1.64872 \times \frac{(x - 0.25)(x - 0.75)}{(0.5 - 0.25)(0.5 - 0.75)} \\ &\quad + 2.71828 \times \frac{(x - 0.25)(x - 0.5)}{(0.75 - 0.25)(0.75 - 0.5)} \\ &= 5.5508^2 + 0.11514x + 1.27301 \end{aligned}$$

$$L_2(0.43) = 2.34886312$$

Then for the third degree polynomial, simply use all of the points available.

$$\begin{aligned} L_3(x) &= \frac{(x - 0.25)(x - 0.5)(x - 0.75)}{(-0.25)(-0.5)(-0.75)} + 1.64872 \frac{(x - 0)(x - 0.5)(x - 0.75)}{(0.25)(0.25 - 0.5)(0.25 - 0.75)} \\ &\quad + 2.71828 \frac{(x - 0)(x - 0.25)(x - 0.75)}{(0.5)(0.5 - 0.25)(0.5 - 0.75)} + 0.75 \frac{(x - 0)(x - 0.25)(x - 0.5)}{(0.75)(0.75 - 0.25)(0.75 - 0.5)} \\ &= 2.91211x^3 + 1.18264x^2 + 2.11721x + 1 \end{aligned}$$

$$L_3(0.43) = 2.36060356577$$

Part (b)

$$N_1(0.43) = 2.41880, N_2(0.43) = 2.34886, N_3(0.43) = 2.36060$$

Here we can see that the results by Neville's method are equivalent to those produced by Lagrange's method, although the Matlab results produced a slightly lesser degree of accuracy. The code to produce these results can be found at the bottom of this problem.

Part (c)

Firsts, we compute the actual error for each degree of our polynomial. Where $e^{2(0.43)} = 2.36316$ is the actual result.

$$\text{degree one: } |2.4188032 - 2.36316| = 0.0556425$$

$$\text{degree two: } |2.34886312 - 2.36316| = 0.01429688$$

$$\text{degree three: } |2.36060356577 - 2.36316| = 0.00255643423$$

The Lagrange polynomial error term is given by

$$\left| \frac{f^{(n+1)}(\xi(x))}{(n+1)!} (x-x_0)(x-x_1) \cdots (x-x_n) \right|$$

To find maximum error, we would like to maximize the term: $|f^{(n+1)}(\xi(x))|$.

For degree one we want the maximum on the bounds $(0.25, 0.5)$:

$$\max(|f''(\xi(x))|) = \max(|4e^{2(\xi(x))}|) = |4e^{2*0.5}| = 10.8731.$$

With just a little more work we have the maximum error as

$$\left| \frac{10.8731}{2!} (0.43 - 0.25)(0.43 - 0.5) \right| = 0.0685. \text{ We can see that our actual error of } 0.0556425 \text{ is just squeezing underneath this maximal value.}$$

Next for degree two, we want the maximum on the bounds $(0.25, 0.75)$:

$$\max(|f^{(3)}(\xi(x))|) = \max(|8e^{2(\xi(x))}|) = |8e^{2*0.75}| = 35.8535.$$

Then to compute maximum error, $\left| \frac{35.8535}{3!} (0.43 - 0.25)(0.43 - 0.5)(0.43 - 0.75) \right| = 0.0225877$. This error is obviously improved accuracy over degree one, and again we see that our actual error of 0.01429688 is well within its bounds.

Finally for degree three, we want the maximum on the full bounds $(0, 0.75)$:

$$\max(|f^{(4)}(\xi(x))|) = \max(|16e^{2(\xi(x))}|) = |16e^{2*0.75}| = 71.7070.$$

Then to compute maximum error, $\left| \frac{71.7070}{4!} (0.43 - 0)(0.43 - 0.25)(0.43 - 0.5)(0.43 - 0.75) \right| = 0.00485636$. Again, this behaves as we'd expect, less error than degree two, and our actual error of 0.00255643423 is again well within its bounds.

Part (d)

$$\text{ndd}_1(0.43) = 2.4188$$

$$\text{ndd}_2(0.43) = 2.3489$$

$$\text{ndd}_3(0.43) = 2.3764$$

```
function p2
```

```
x = 0.43;  
xi = [0 0.25 0.5 0.75];  
fi = [1 1.64872 2.71828 4.48169];
```

```
%% Neville's Method
```

```
% degree one  
neville(x, xi(:,2:3), fi(:,2:3))  
% degree two  
neville(x, xi(:,2:4), fi(:,2:4))  
% degree three  
neville(x, xi(:,1:4), fi(:,1:4))
```

```
%% Newton's Divided-Difference
```

```
% degree one  
ndd(x, xi(:,2:3), fi(:,2:3))  
% degree two  
ndd(x, xi(:,2:4), fi(:,2:4))  
% degree three  
ndd(x, xi(:,1:3), fi(:,1:3))
```

```
end
```

```
function res = ndd(x, xi, fi)

M = divideddifference(xi, fi);
n = length(M);
res = 0;

X = 1;
for i=1:n
    C = M(i,i);
    res = res + C * X;
    X = X * (x - xi(i));
end

end
```

```
function F = divideddifference(x,f)

n = length(x) - 1;
F = zeros(n+1,n+1);

F(:,1) = f(:);

for i=1:n
    for j=1:i
        F(i+1,j+1) = (F(i+1,j)-F(i,j))/(x(i+1)-x(i-j+1));
    end
end

end
```

```
function Q = neville(x, xi, fi)

n = length(xi) - 1;
Q = zeros(n+1,n+1);
Q(:,1) = fi(:);

for i=1:n
    for j=1:i
        Q(i+1,j+1) = ((x-xi(i-j+1))*Q(i+1,j)-(x-xi(i+1))*Q(i,j))...
            /(xi(i+1)-xi(i-j+1));
    end
end

end
```

Problem 2.

(25 points)

The Bernstein polynomial of degree n for $f \in C[0, 1]$ is given by

$$B_n(x) = \sum_{k=0}^n \binom{n}{k} f(k/n) x^k (1-x)^{n-k},$$

where $\binom{n}{k} = n! / (k!(n-k)!)$. These polynomials can be used in a constructive proof of Weierstrass Approximation Theorem since $\lim_{n \rightarrow \infty} B_n(x) = f(x)$, for each $x \in [0, 1]$.

(a) Find $B_3(x)$ for the functions: (i) $f(x) = x$ and (ii) $f(x) = 1$.

(b) Show that for each $k \leq n$,

$$\binom{n-1}{k-1} = \frac{k}{n} \binom{n}{k}$$

(c) Use part (b) and the fact, from (ii) in part (a), that

$$1 = \sum_{k=0}^n \binom{n}{k} x^k (1-x)^{n-k}$$

Solution

TODO

Problem 3.

(10 points)

- (i) Show that the cubic polynomials

$$P(x) = 3 - 2(x + 1) + 0(x + 1)(x) + (x + 1)(x)(x - 1)$$

and

$$Q(x) = -1 + 4(x + 2) - 3(x + 2)(x + 1) + (x + 2)(x + 1)(x)$$

both interpolate the data $f(-2) = -1, f(-1) = 3, f(0) = 1, f(1) = -1, f(2) = 3$

- (ii) Why does part (a) not violate the uniqueness property of interpolating polynomials?

Solution

TODO

Problem 4.

(20 points)

(You may use Matlab or other softwares to solve this problem. Copies of codes and results must be submitted.) Let $f(x) = 3xe^x - e^{2x}$.

- (i) Approximate $f(1.03)$ by the Hermite interpolating polynomial of degree at most three using $x_0 = 1$ and $x_1 = 1.05$. Compare the actual error to the error bound.
- (ii) Repeat (a) with the Hermite interpolating polynomial of degree at most five, using $x_0 = 1, x_1 = 1.05$ and $x_2 = 1.07$.

Solution

TODO

Problem 5.

(20 points)

- (i) Determine the free cubic spline S that interpolates the data $f(0) = 0$, $f(1) = 1$ and $f(2) = 2$.
- (ii) Determine the clamped cubic spline s that interpolates the data $f(0) = 0$, $f(1) = 1$, $f(2) = 2$ and satisfies $s'(0) = s'(2) = 1$.

Solution

TODO