Homework10

Sunday, November 5, 2023 5:15 PM

Tony Samow
APPM 4600

1.) 1. For the function $f(x) = \sin(x)$. Determine the Padé approximations of degree 6 with

(a) Both the numerator and denominator are cubic

(b) The numerator is quadratic and the denominator is a fourth degree polynomial.

(c) The numerator is a fourth degree polynomial and the denominator is quadratic.

Compare the accuracy of these approximations with the sixth order Maclaurin polynomial by ploting the error over the interval $[0, 5]$.

Padé Approximation:

$$P_m^n(x) = \frac{a_0 + a_1 x + \cdots + a_m x^m}{1 + b_1 x + \cdots + a_n x^n}$$, which we will match to $T_6(x) = \sum_{n=0}^{6} \frac{f^{(n)}(0)}{n!} x^n$

$$T_6[\sin](x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!}$$

(a) $$P_3^3(x) = \frac{a_0 + a_1 x + a_2 x^2 + a_3 x^3}{1 + b_1 x + b_2 x^2 + b_3 x^3} = x - \frac{x^3}{3!} + \frac{x^5}{5!}$$

$$\therefore \; a_0 + a_1 x + a_2 x^2 + a_3 x^3 = \left(x - \frac{x^3}{3!} + \frac{x^5}{5!}\right)\left(1 + b_1 x + b_2 x^2 + b_3 x^3\right)$$

Term (x^n)	Coefficients
const	$a_0 = 0$
x	$a_1 = 1$
x^2	$a_2 = b_1$
x^3	$a_3 = b_2 - \frac{1}{3!}$
x^4	$0 = b_3 - \frac{b_1}{3!}$
x^5	$0 = -\frac{b_2}{3!} + \frac{1}{5!}$
x^6	$0 = \frac{b_1}{5!} - \frac{b_3}{3!}$

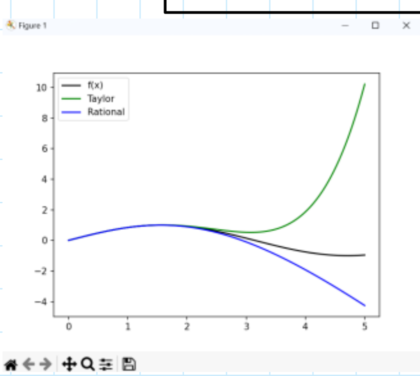
$$\Rightarrow \; b_2 = \frac{3!}{5!} = \frac{1}{20}$$

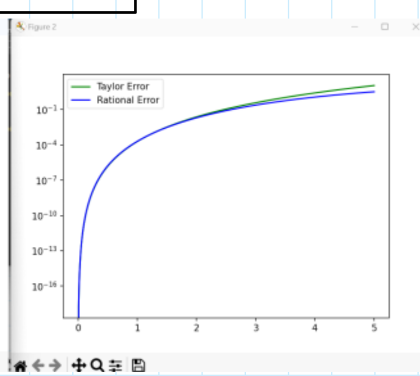
$$\begin{bmatrix} -\frac{1}{6} & 1 \\ \frac{1}{120} & -\frac{1}{6} \end{bmatrix} \begin{bmatrix} b_1 \\ b_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow b_1 = b_3 = 0$$

$b_1 = 0$
$b_2 = 0$
$b_3 = 0$

$a_0 = 0$
$a_1 = 1$
$a_2 = 0$
$a_3 = -\frac{7}{60}$

$$\therefore \; \boxed{P_3^3(x) = \frac{x - \frac{7}{60} x^3}{1 + \frac{1}{20} x^2}}$$





These approximations are extremely close until the end of the interval, where the Taylor error increases much quicker.

(b) $$P_2^4(x) = \frac{a_0 + a_1 x + a_2 x^2}{1 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4} = x - \frac{x^3}{3!} + \frac{x^5}{5!}$$

$$\therefore \; a_0 + a_1 x + a_2 x^2 = \left(x - \frac{x^3}{3!} + \frac{x^5}{5!}\right)\left(1 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4\right)$$

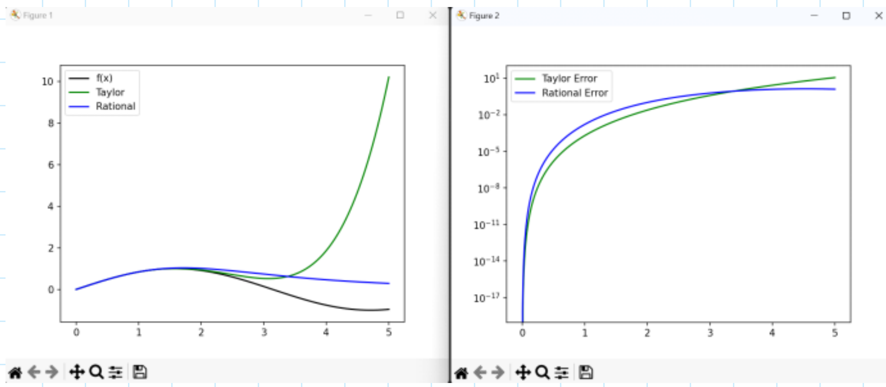
Term (x^n)	Coefficients
const	$a_0 = 0$
x	$a_1 = 1$
x^2	$a_2 = b_1$
x^3	$0 = b_2 - \frac{1}{3!} \Rightarrow b_2 = \frac{1}{6}$
x^4	$0 = b_3 - \frac{b_1}{3!}$
x^5	$0 = \frac{1}{5!} - \frac{b_2}{3!} + b_4 \Rightarrow b_4 = \frac{1}{36} - \frac{1}{120} = \frac{7}{360}$
x^6	$0 = \frac{b_1}{5!} - \frac{b_3}{3!}$

From (a), we find $b_1 = b_3 = 0$

$\therefore$ $b_1 = 0$, $b_2 = \frac{1}{6}$, $b_3 = 0$, $b_4 = \frac{7}{360}$

$a_0 = 0$, $a_1 = 1$, $a_2 = 0$

$$\therefore \quad \boxed{P_2^4(x) = \frac{x}{1 + \frac{1}{6}x^2 + \frac{7}{360}x^4}}$$



In general, this approximation was less exact than in part (a). I found it interesting that the rational error was larger than Taylor at the beginning, but once again the Taylor error shoots up at the end.

(c) $$P_4^2(x) = \frac{a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4}{1 + b_1x + b_2x^2} = x - \frac{x^3}{3!} + \frac{x^5}{5!}$$

$$\therefore \quad a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 = \left(x - \frac{x^3}{3!} + \frac{x^5}{5!}\right)\left(1 + b_1x + b_2x^2\right)$$

Term (x^n)	Coefficients
const	$a_0 = 0$
x	$a_1 = 1$
x^2	$a_2 = b_1$
x^3	$a_3 = b_2 - \frac{1}{3!} \Rightarrow a_3 = -\frac{7}{60}$
x^4	$a_4 = -\frac{b_1}{3!}$
x^5	$0 = \frac{1}{5!} - \frac{b_2}{3!} \Rightarrow b_2 = \frac{3!}{5!} = \frac{1}{20}$
x^6	$0 = \frac{b_1}{5!} \Rightarrow b_1 = 0$

$\therefore$ $b_1 = 0$, $b_2 = \frac{1}{20}$

$a_0 = 0$, $a_1 = 1$, $a_2 = 0$, $a_3 = -\frac{7}{60}$, $a_4 = 0$

$$\therefore \quad \boxed{P_4^2(x) = \frac{x - \frac{7}{60}x^3}{1 + \frac{1}{20}x^2}}$$

Since this approximation matches $P_3^3(x)$, the plots are the same as part (a).

2.) 2. Find the constants x_0, x_1 and c_1 so that the quadrature formula

$$\int_0^1 f(x)dx = \frac{1}{2}f(x_0) + c_1 f(x_1)$$

has the highest possible degree of precision.

Since there are three unknowns, we will need 3 orders of functions in order to approximate. For simplicity, I will choose $f_1(x) = x^0 = 1$

$f_2(x) = x^1 = x$

$f_3(x) = x^2$

$$\therefore \int_0^1 1\,dx = 1 \Rightarrow 1 = \frac{1}{2}(1) + c_1(1)$$

$$\Rightarrow c_1 = \frac{1}{2}$$

$$\int_0^1 x\,dx = \frac{1}{2} \Rightarrow \frac{1}{2} = \frac{1}{2}(x_0) + \frac{1}{2}(x_1)$$

$$\Rightarrow x_0 + x_1 = 1 \Rightarrow x_0 = 1 - x_1$$

$$\int_0^1 x^2 dx = \frac{1}{3} \Rightarrow \frac{1}{3} = \frac{1}{2}(x_0)^2 + \frac{1}{2}(x_1)^2$$

$$\Rightarrow \frac{3}{2}x_0^2 + \frac{3}{2}x_1^2 = 1$$

Substitute:
$$\frac{3}{2}(1-x_1)^2 + \frac{3}{2}(x_1)^2 = 1$$
$$\frac{3}{2}(1 - 2x_1 + x_1^2) + \frac{3}{2}x_1^2 = 1$$
$$\frac{3}{2} - 3x_1 + 3x_1^2 = 1$$
$$3x_1^2 - 3x_1 + \frac{1}{2} = 0$$

$$\frac{3 \pm \sqrt{(-3)^2 - 4(3)(\frac{1}{2})}}{2(3)}$$

$$= \frac{3 \pm \sqrt{3}}{6}$$, choose $x_1 = \frac{3+\sqrt{3}}{6}$

Choose $x_1 = \frac{3+\sqrt{3}}{6}$

$$x_0 + x_1 = 1 \Rightarrow x_0 = 1 - x_1$$
$$x_0 = 1 - \left(\frac{3+\sqrt{3}}{6}\right)$$
$$x_0 = \frac{3-\sqrt{3}}{6}$$

$$\therefore \boxed{c_1 = \frac{1}{2},\quad x_0 = \frac{3-\sqrt{3}}{6},\quad x_1 = \frac{3+\sqrt{3}}{6}}$$

3.) 3. (a) Write a code to approximate $\int_{-5}^{5} \frac{1}{1+s^2} ds$ using a composite Trapezoidal rule. To do this, partition the interval $[-5, 5]$ into equally spaced points $t_0, t_1, \ldots, t_n$.

Write another code to approximate $\int_{-5}^{5} \frac{1}{1+s^2} ds$ using a composite Simpson's rule. To do this, partition the interval $[-5, 5]$ into equally spaced points $t_0, t_1, \ldots, t_n$ where $n = 2k$ is even. The even indexed points should be the endpoints of your subintervals.

You may combine the two into one code that selects the desired method if you wish.

Turn in a listing of your code(s).

For Composite Trapezoidal Rule:

$$\int_a^b f(x)\,dx = \frac{h}{2}\left(f(a) + 2\cdot\sum_{j=1}^{n-1}(x_j) + f(b)\right) - \underbrace{\frac{(b-a)}{2}h^2 f''(\mu)}_{\text{error}}$$

For Composite Simpson's Rule:

For Composite Simpson's Rule:

$$\int_a^b f(x)\,dx = \frac{h}{3}\left(f(a) + 2\cdot\sum_{j=1}^{\frac{n}{2}-1} f(x_{2j}) + 4\cdot\sum_{j=1}^{\frac{n}{2}} f(x_{2j-1}) + f(b)\right) - \underbrace{\frac{b-a}{180}h^4 \cdot f^{(4)}(\mu)}_{\text{error}}$$

My Code:

```python
def driver():

    # Function to integrate
    f = lambda x: 1/(1+x**2)

    # Integration bounds
    a = -5
    b = 5

    # N equispaced intervals
    n = 12

    x = np.linspace(a,b,n+1)

    I_CompTrap = compTrapezoidal(a, b, x, f, n)
    I_CompSimp = compSimpson(a, b, x, f, n)

    print('Composite Trapezoidal Approximation: ', I_CompTrap)
    print('Composite Simpson Approximation: ', I_CompSimp)

def compTrapezoidal(a, b, x, f, n):
    h = (b-a)/(n)
    fSum = 0

    for j in range(1, n-1):
        fSum = fSum + f(x[j])

    I = (h/2)*(f(a) + 2*fSum + f(b))
    return I

def compSimpson(a, b, x, f, n):
    h = (b-a)/n
    fSumEven = 0
    fSumOdd = 0

    for j in range(1, n-1):
        if (j%2) == 0:
            fSumEven = fSumEven + f(x[j])
        elif (j%2) != 0:
            fSumOdd = fSumOdd + f(x[j])

    I = (h/3)*(f(a) + 2*fSumEven + 4*fSumOdd + f(b))
    return I
driver()
```

Output:

```
$ python3 Problem3.py
Composite Trapezoidal Approximation:  2.7030531588427884
Composite Simpson Approximation:  2.6412870352090954
```

(b) b) Use the error estimates derived in class to choose n so that

$$\left|\int_{-5}^{5} \frac{1}{1+s^2}ds - T_n\right| < 10^{-4} \quad \text{and} \quad \left|\int_{-5}^{5} \frac{1}{1+s^2}ds - S_n\right| < 10^{-4},$$

where T_n is the result of the composite Trapezoidal rule and where S_n is the result of the composite Simpson's rule. Be sure to explain your reasoning for choosing n in both cases (these n values will be different in the two cases).

Trapezoidal Error Term: $\frac{b-a}{2}\cdot h^2 f''(\mu) < 10^{-4}$

$f(x) = \frac{1}{1+x^2}$ $\quad f'(x) = -\frac{2x}{(1+x^2)^2}$ $\quad f''(x) = \frac{2(3x^2-1)}{(x^2+1)^3}$

$\max\limits_{x\in[-5,5]} |f''(x)|$ occurs at $x=0$, $|f''(0)| = 2$

$\therefore \quad \frac{5-(-5)}{\cancel{2}}\cdot\left(\frac{5-(-5)}{n}\right)^2\cdot\cancel{(2)} < 10^{-4}$

$$\frac{10^3}{n^2} < 10^{-4}$$

$$\frac{10^3}{n^2} < 10^{-4}$$

$$\sqrt{10^7} < \sqrt{n^2}$$

$$n > 10^{7/2} \approx \boxed{n > 3200}$$

Simpson's Error Term : $\frac{b-a}{180} h^4 \cdot f^{(4)}(\mu) < 10^{-4}$

Using an online calculator: $\frac{d^4}{dx^4}\left(\frac{1}{1+x^2}\right) = \frac{24\cdot(5x^4 - 10x^2+1)}{(1+x^2)^5}$

$\max\limits_{x\in[-5,5]} f^{(4)}(x)$ occurs at $x = 0$, $|f^{(4)}(0)| = 24$

$$\therefore \quad \frac{5-(-5)}{180}\cdot\left(\frac{5-(-5)}{(n)}\right)^4 \cdot 24 < 10^{-4}$$

Note: Chose $n = 50$ because n must be even

$$\frac{10^5}{180\, n^4} < 10^{-4}$$

$$\frac{10^9}{180} < n^4 \Rightarrow n \approx 48.5 \Rightarrow n > 49 \Rightarrow \boxed{n = 50}$$

(c) c) Run your code with the predicted values of n and compare your computed values S_n and T_n with that of SCIPY's quad routine on the same problem. Run the built in quadrature twice, once with the default tolerance of 10^{-6} and another time with the set tolerance of 10^{-4}. Report the number of function evaluations required in both cases and compare these to the number of function values your codes (both S_n and T_n) required to meet the tolerance

Turn in your codes and the results of this test.

Want to verify $\left|\int_{-5}^{5} f(x)\,dx - T_n\right| < 10^{-4}$ and $\left|\int_{-5}^{5} f(x)\,dx - S_n\right| < 10^{-4}$

Code:

```python
def driver():

    # Function to integrate
    f = lambda x: 1/(1+x**2)

    # Integration bounds
    a = -5
    b = 5

    # N equispaced intervals
    # For Simpson's Rule, n =2k
    n = 3200

    x = np.linspace(a,b,n+1)

    I_CompTrap, trapEvalCount = compTrapezoidal(a, b, x, f, n)
    I_CompSimp, simpEvalCount = compSimpson(a, b, x, f, n)

    Ieval = quad(f, a, b)

    errTrap = abs(Ieval - I_CompTrap)
    errSimp = abs(Ieval - I_CompSimp)

    print('Composite Trapezoidal Approximation: ', I_CompTrap, 'Num Evaluations: ', trapEvalCount)
    print('Trapezoidal Error: ', errTrap)
    print('Composite Simpsons Approximation: ', I_CompSimp, 'Num Evaluations: ', simpEvalCount)
    print('Simpsons Error: ', errSimp)
    print('Built-In Quadrature: ', Ieval)

def compTrapezoidal(a, b, x, f, n):
    h = (b-a)/(n)
    fSum = 0
    evalCount = 2

    for j in range(1, n-1):
        fSum = fSum + f(x[j])
        evalCount += 1

    I = (h/2)*(f(a) + 2*fSum + f(b))
    return I, evalCount
```

```python
def compSimpson(a, b, x, f, n):
    h = (b-a)/n
    fSumEven = 0
    fSumOdd = 0
    evalCount = 2

    for j in range(1, n-1):
        if (j%2) == 0:
            fSumEven = fSumEven + f(x[j])
            evalCount += 1

    for jj in range(1, n):
        if (jj%2) != 0:
            fSumOdd = fSumOdd + f(x[jj])
            evalCount += 1

    I = (h/3)*(f(a) + 2*fSumEven + 4*fSumOdd + f(b))
    return I, evalCount
driver()
```

Output for $n = 3200$:

```
$ python3 Problem3.py
Composite Trapezoidal Approximation:  2.746681172914857 Num Evaluations:  3200
Trapezoidal Error:  [1.20360975e-04 2.74668116e+00]
Composite Simpsons Approximation:  2.746801533890024 Num Evaluations:  3201
Simpsons Error:  [8.88178420e-15 2.74680152e+00]
Built-In Quadrature:  (2.7468015338900327, 1.4334139675000002e-08)
```

The error term is of order 10^{-4} for trapezoidal ✓

Output for $n = 100$

```
$ python3 Problem3.py
Composite Trapezoidal Approximation:  2.7383834749902114 Num Evaluations:  50
Trapezoidal Error:  [0.00841806 2.73838346]
Composite Simpsons Approximation:  2.746801738009728 Num Evaluations:  51
Simpsons Error:  [2.04119695e-07 2.74680172e+00]
Built-In Quadrature:  (2.7468015338900327, 1.4334139675000002e-08)
```

Much better for $n = 50$

```
tonys@TonyStudio MINGW64 ~/Documents/APPM4600/Samour_APPM4600/Homework/Homework
0 (main)
$ python3 Problem3.py
Composite Trapezoidal Approximation:  2.738195772308147 Num Evaluations:  49
Trapezoidal Error:  [0.00860576 2.73819576]
Composite Simpsons Approximation:  2.7384137098658767 Num Evaluations:  49
Simpsons Error:  [0.00838782 2.7384137 ]
Built-In Quadrature:  (2.7468015338900327, 1.4334139675000002e-08)
```

Error $> 10^{-4}$ for $n = 49$