Section 1

MANE 3351

Subsection 1

Lecture 12, September 29

Classroom Management

Agenda

- Lab today in ENGR 2.268 at 3:30 pm
- GitHub Accounts
    - Sign up for free student GitHub account at https://education.github.com/students
    - Register your student GitHub account with MANE 3351 GitHub classroom email
- Lecture

Subsection 2

Resources

Handouts

- Lecture 12 Slides
- Lecture 12 Marked Slides

Calendar

Lecture	Date	Content
1	August 23	Welcome to Class, Syllabus
2	August 25	Error Analysis
3	August 30	Introduction to Jupyter Notebook
4	September 1	Markdown
5	September 6	Labor Day Holiday
6	September 8	Taylor Polynomials, Homework 1
7	September 13	Roots of Equations, Bisection Method
8	September 15	Bisection Error Analysis
9	September 20	False Position Algorithm
10	September 22	Newton Raphson Algorithm
11	September 27	Secant Method
12	September 29	Numerical Integration
13	October 4	Simpson's Rule

Assignments

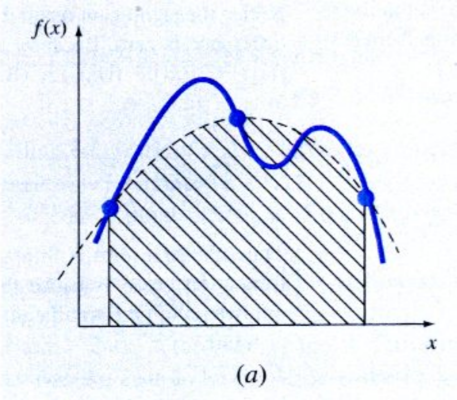
Looking Ahead

Lecture	Date	Content
14	October 6	Romberg Integration
15	October 11	Gaussian Quadrature (end of material for Test 1)
16	October 13	Vectors
17	October 18	Vector Operations
18	October 20	Test 1

Lecture 13, October 4

Today's topic is Simpson's Rule.

- Simpson's (1/3) rule is Newton-Cotes with $n = 2$
- Chapra and Canale (2015)[1] illustrate Simpson's rule in the figure shown below



[1] Chapra, S., and Canale, R., (2015), *Numerical Methods for Engineers, 7th edition*

Definition of Simpson's Rule

- Chapra and Canale(2015)[2] provide the following definition of Simpson's rule

$$I = \frac{h}{3}\left[f(x_0) + 4f(x_1) + f(x_2)\right]$$

where $h = (b - a)/2$

- Note that Simpson's rule is of the form

$$I \approx (b - a)\,\frac{f(x_0) + 4f(x_1) + f(x_2)}{6}$$
$$I \approx \text{ width} \times \text{ average height}$$

[2] Chapra, S., and Canale, R., (2015), *Numerical Methods for Engineers, 7th edition*

Simpson's Rule Error Analysis

- Chapra and Canale (2015)[3] provides the following definition:

$$I = \frac{h}{3}\left[f(x_0) + 4f(x_1) + f(x_2)\right] - \frac{1}{90} f^{(4)}(\xi) h^5$$

$$I = \text{Simpson's } 1/3 \text{ approximation} - \text{ Truncation error}$$

[3] Chapra, S., and Canale, R., (2015), *Numerical Methods for Engineers, 7th edition*

Multiple Applications of Simpson's Rule

- The number of segments must be even
- The formula is

$$I \approx (b-a)\,\frac{f(x_0) + 4\sum_{i=1,3,5}^{n-1} f(x_i) + 2\sum_{j=2,4,6}^{n-2} f(x_j) + f(x_n)}{3n}$$

where $h = \frac{b-a}{n}$

Pseudo-code: Simpson's 1/3 Rule

- CodeSansar[4] provides the following pseudo-code Simpson's 1/3 pseudo-code

[4] https://www.codesansar.com/numerical-methods/integration-simpson-1-3-method-algorithm.htm

Jupyter Notebook Demonstration

Simpson's 3/8 Rule

- Simpson's 3/8 rule is Newton-Cotes with $n = 3$
- Chapra and Canale (2015)[5] illustrate Simpson's rule in the figure shown below Simpson's3/8 Rule

[5]Chapra, S., and Canale, R., (2015), *Numerical Methods for Engineers, 7th edition*

Definition of Simpson's 3/8 Rule

- Chapra and Canale(2015)[6] provide the following definition of Simpson's rule

$$I = \frac{3h}{8}\left[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)\right]$$

where $h = (b-a)/3$

- Note that Simpson's 3/8 rule is of the form

$$I \approx (b-a)\,\frac{f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)}{8}$$
$$I \approx \text{ width} \times \text{ average height}$$

[6] Chapra, S., and Canale, R., (2015), *Numerical Methods for Engineers, 7th edition*

Truncation Error of Simpson's 3/8 Rule

$$
\begin{aligned}
E_t &= -\frac{3}{80} h^5 f^{(4)}(\xi) \\
&= -\frac{(b-a)^5}{6480} f^{(4)}(\xi)
\end{aligned}
$$

Pseudo-code: Simpson's 3/8 Rule

CodeSansar[7] provides the following pseudo-code for Simpson's 3/8 Rule

Simpson's 3/8 Rule Pseudocode

```
1. Start

2. Define Function f(x)

3. Input lower_limt, upper_limit, sub_interval

4. Calculate: step_size = (upper_limit - lower_limit)/sub_interval

5. Calculate: integration = f(lower_limit) + f(upper_limit)

6. Set: i=1

7. Loop
        k= lower_limit + i * step_size

        If i mod 3 = 0
                integration =  integration + 2 * f(k)
        Else
                integration =  integration + 3 * f(k)
        End If

        i = i+1

   While i< sub_interval
```