Printout

Wednesday, September 15, 2021 9:22 AM

Section 1

MANE 3351

Subsection 1

Lecture 8, September 15

Classroom Management

Agenda

- Proctor & Gamble
- Lecture
- Handout Raspberry Pi
- No lab today

Subsection 2

Resources

Handouts

- Lecture 8 Slides
- Lecture 8 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

Assignments

- Homework 1 (assigned 9/8/2021, due 9/16/2021)

Homework 2 9/15/2021, 9/23/2021

Bisection Method: Error Analysis

Bisection Method Theorem

Cheney and Kincaid (2004)[^1] provide a definition of the bisection method

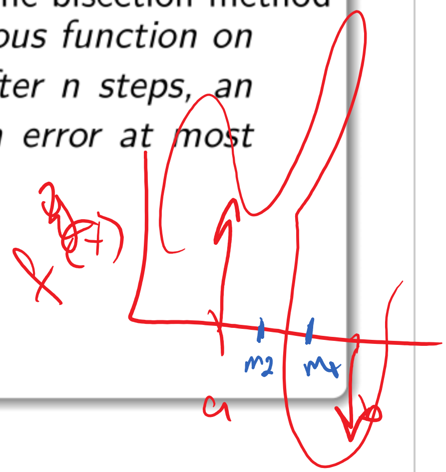
> *If the bisection algorithm is applied to a continuous function on an interval $[a, b]$, where $f(a)f(b) < 0$, then, after n steps, an approximate root will have been computed with error at most $(b-a)/2^{n+1}$.*

This definition provides the following useful results:

- $|e_n| \leq \frac{1}{2^{n+1}}(b-a)$
- $n > \frac{\log(b-a) - \log 2\varepsilon}{\log 2}$

epsilon - tolerance or error bound

pseudo code



f(a)
f(b)

Percent Relative Error

Chapra and Canale (2015) [^2] provide a formula for the approximate percent relative error

- $\varepsilon_a = \left| \frac{x_r^{new} - x_r^{old}}{x_r^{new}} \right| 100\%$ where x_r^{new} is the root for the present iteration and x_r^{old} is the root from the previous iteration

Pseudo Code

- "**Pseudocode** is an informal high-level description of the operating principle of a computer program or other algorithm"[^3]
- Highly encourage textbook recommendation: "Though technically no necessary for coding, when we can, we will preface each method's pseudo-code with mathematical assumptions that guarantee success.", page 39

First example of Pseudo-code for Bisection

Assumptions: f is continuous on $[a, b]$. $f(a)$ and $f(b)$ have opposite signs.

Input: Interval $[a, b]$; function f; desired accuracy tol; maximum number of iterations N.

Step 1: Set $err = |b - a|$; $L = f(a)$;

Step 2: For $j = 1 \ldots N$ do Steps 3-5:

Step 3: Set $m = \frac{a+b}{2}$; $M = f(m)$; $err = err/2$;
Step 4: If $M = 0$ or $err \leq tol$ then return m;
Step 5: If $LM < 0$ then set $b = m$; else set $a = m$ and $L = M$;

Step 6: Print "Method failed. Maximum iterations exceeded."

Output: Approximation m within tol of exact root, or message of failure.

Figure 1: Pseudo-code 1

Second Example of Pseudo-code for Bisection

Step 1: Set $L = f(a)$;
Step 2: Set $m = \frac{a+b}{2}$; $M = f(m)$;
Step 3: If $LM < 0$ then set $b = m$; else set $a = m$ and $L = M$;
Step 4: Go to Step 2.

Figure 2: Pseudo-code 2

Review Jupyter Notebook for Monday

Error for Monday's First Quartile Example

[1]: Cheney, W. and Kincaid, D. (2004), *Numerical Mathematics and Computing, 5th edition*

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