

## MATLAB Assignment 3

Released: 9/19/2016

Due: 9/26/2015

***Please read “MATLAB Assignment Submission Guidelines” in Blackboard before submission. Not following the guidelines will result in loss of credit, even though you may have the correct answer(s).***

Write code of your own and answer the questions in this assignment.

- Submit your code through Cody Coursework.
- Submit a report through Blackboard

You will practice the use of symbolic variables, built-in MATLAB functions for finding derivatives and their substitution values, and write your own function of the divided difference schemes.

### **Problem 1: Derivatives with MATLAB Built-In Functions**

1) Analytical Derivatives with `diff()`:

- Define a **symbolic** variable “ $x$ ” and create a **symbolic** function  $f(x) = \sin(2x)$ . Using “`diff`” command, find and report the first derivative of  $f(x)$ . Your output display should be a symbolic expression with proper explanatory narratives. Show the output display in your report.
- Plot the **symbolic** first derivative in domain  $[-2\pi, 2\pi]$  using the `ezplot` command. Consult the MATLAB Help documents and explain the use of this command.
- Using `subs()` and `double()` commands, find and report  $f'(\pi/3)$ . (The 1<sup>st</sup>-order derivative of  $f(x) = \sin(2x)$  evaluated at  $x = \pi/3$ .) Your output display should show both the symbolic expression and the numerical value with proper explanatory narratives. Show the output display in your report.

*(Hint: use the slide set “Class 12 MATLAB 3 Numerical Differentiation\_Self Study” as a reference.)*

2) Approximate Derivatives with `diff()`:

- Create a domain  $X$  starting from  $-2\pi$  to  $2\pi$  with 0.001 increments. Find the functional values  $Y$  across the domain  $X$  for function  $f(x) = \sin(2x)$ . Do NOT use any loops for this calculation. Demonstrate this portion of your code in your report.
- What is the output of `diff(Y)`? Consult the MATLAB Help documents and explain the use of `diff()` in this particular case.
- Do you see an approach for using `diff()` for approximation of derivatives? Please explain your approach.
- Implement your approach above in MATLAB for function  $f(x) = \sin(2x)$  in domain  $[-2\pi, 2\pi]$ . Plot the numerical derivatives across the domain.

3) Comparison between analytical and numerical derivatives

- Compare the derivatives from 1) and 2) graphically. You may use the same graph for both plots, and after plotting, zoom in to see subtle differences in the lines.
- Compare the derivatives from 1) and 2) numerically. Use `subs()` and `double()` to calculate the values of the analytical derivative in 1) across the domain  $X$ . Compute the

difference in errors numerically. Do NOT use any loops for the calculations in this question. Demonstrate this portion of your code in your report.

**Problem 2: Modified from #3 in Chapter 02.02 Problem Set**

Write a MATLAB **function** to calculate the 1<sup>st</sup>-order derivative of the function  $f(x) = \sin(2x)$  at any given point correct within a specified significant digits using forward or central divided difference schemes.

**Your function should meet the following requirements:**

- It should be named **NumericalDiff**
- It should have **four** input arguments:
  - 1) The first input argument is the point at which the first-order derivative is to be estimated.
  - 2) The second input argument is the number of correct significant digits specified.
  - 3) The third input argument is the starting step size.
  - 4) The fourth input argument is either 1 or 2, where 1 means an FDD scheme should be used and 2 means a CDD scheme should be used.
- It should have **three** output arguments stored as a **row vector**:
  - 1) The first output argument is the number of iterations performed.
  - 2) The second output argument is the final approximate first-order derivative.
  - 3) The third output argument is the final absolute relative approximate error.
- The algorithm implemented in your function should start with the specified step size, keep halving it till you find the answer within a specified significant digits. Use a **while-end** loop for the procedure. You could initialize the relative approximate error to 1 at the beginning of the procedure. Create a flowchart first and demonstrate this portion of your code in your report.
- Your function should display the following:
  - 1) At the beginning of your function, display on screen the divided difference scheme used for the calculation
  - 2) In each iteration, your function should display on screen the current approximate value and the current absolute relative approximate error. Make sure your display has proper explanatory narrative.

Show the screen display in your report.