

EGR1010 Lab MATLAB

Supplemental Instruction Assignment #3

INSTRUCTIONS:

Complete the following assignment and submit with your laboratory notebook. Use the template provided on the course content page on Pilot. *Please follow the instructions for submission located at the end of this document.*

PROBLEM STATEMENT:

In this supplemental assignment we will investigate the definition of the derivative using MATLAB. Recall that the derivative of a function $y(t)$ at a point $t = p$ is equal to the slope of a tangent line to $y(t)$ at p . Thus, it represents the change in the variable y with respect to a change in the variable t , which are denoted as Δy and Δt , respectively. This is demonstrated graphically in Figure 1.

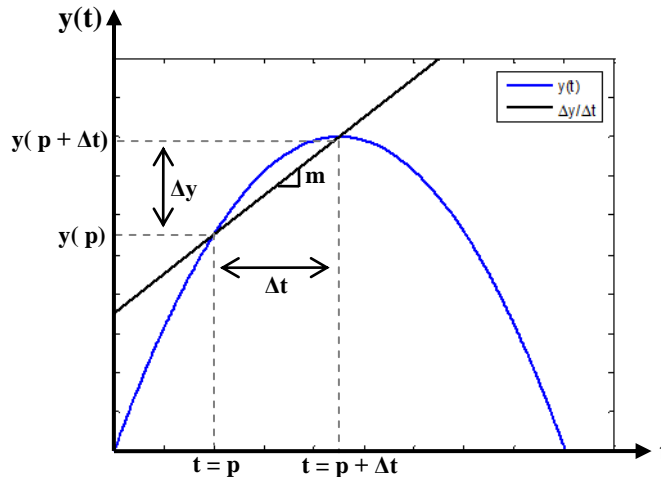


Figure 1. Approximation of the derivative as the slope of a tangent line.

We note from Figure 1, we can compute the slope of the line (and approximate the slope of the *tangent line* or *derivative*) as:

$$m = \frac{\Delta y}{\Delta t} = \frac{y(p + \Delta t) - y(p)}{\Delta t} \approx \frac{dy}{dt}$$

We also know that by making Δt smaller and smaller, we get a better and better approximation for the slope of the tangent line. This process of reducing the increment Δt is simply represented by the limit in the definition the derivative for any point t given by:

$$\frac{dy}{dt} \equiv \lim_{\Delta t \rightarrow 0} \frac{y(t + \Delta t) - y(t)}{\Delta t}$$

For the remainder of this assignment, we will use MATLAB to explore of reducing the increment according to the definition above to converge to the exact value of the derivative.

Let us consider the vertical height $h(t)$, of a model rocket that has been launched from the ground with an initial velocity of $v_o = 200 \text{ m/s}$. The vertical position of the rocket with respect to the ground is given by:

$$h(t) = 200t - 4.905t^2 \text{ m}$$

We wish to approximate the derivative of $h(t)$ at 7.5 seconds.

MATLAB PROCEDURES:

Complete the following procedures in the supplemental template file available in the course content on Pilot. DO NOT alter the formatting of the existing template, but rather simply add in your code where appropriate. Also download the additional file 'approxTangentLine.txt' and save it as a MATLAB function.

1. Create a vector of time ranging from 0 to 42 seconds with an increment of 0.01 seconds between points. Suppress output to the command window.
2. Evaluate the function $h(t)$ at the points in the time vector using element-by-element math. Suppress output to the command window.
3. Create a new figure window and plot the height versus time of the rocket. Be sure to include appropriate labels and units for the x and y axes.
4. Since we are concerned with approximating the derivative of the height at the critical time of $t_c = 7.5$ seconds, define this time value as a scalar variable and evaluate the height at this point. DO NOT suppress output to the command window.
5. In order to approximate the derivative at the critical time, we should move forward in time some increment Δt away from the critical point as previously demonstrated in Figure 1. We would like to explore the effect of moving forward by different amounts. Create a vector that contains increments corresponding to 20, 15, 10, 5, and 1 second forward in time. DO NOT suppress command window output.
6. According to the equation for the approximation of the derivative, we need to determine the incremental change in height that results from an incremental change in time. Thus, compute the values of the height corresponding to the incremental change in time. That is, use element-by-element math to find:

$$h(t_c + \Delta t) = 200(t_c + \Delta t) - 4.905(t_c + \Delta t)^2$$

DO NOT suppress output to the command window.

7. Now determine the incremental change in height by using element-by-element math to subtract the height at the critical point from the heights evaluated at the critical point plus the increments.

$$\Delta h = h(t_c + \Delta t) - h(t_c)$$

DO NOT suppress output to the command window.

8. We can now approximate the slope of the tangent line according to the equation given for the approximate derivative earlier. Do this using element-by-element math and DO NOT suppress command window output. The results in your output correspond to approximations of the derivative at the critical time for decreasing values of the increment in time.
9. We now use the supplied function `approxTangentLine()` plot lines corresponding to the approximations of the tangent lines we have just computed. Type 'help `approxTangentLine`' in the command window for information for how to use function and complete parts (a) through (e) below.
 - a. Create a tangent line for the **first** approximate value of the derivative. Make this plot a **black line** ('k') and run the code after you have done so. Observe the plot.
 - b. Create a tangent line for the **second** approximate value of the derivative. Make this plot a **red line** ('r') and rerun the code after you have done so. Observe the plot.
 - c. Create a tangent line for the **third** approximate value of the derivative. Make this plot a **blue line** ('b') and rerun the code after you have done so. Observe the plot.
 - d. Create a tangent line for the **fourth** approximate value of the derivative. Make this plot a **cyan line** ('c') and rerun the code after you have done so. Observe the plot.
 - e. Create a tangent line for the **fifth** approximate value of the derivative. Make this plot a **green line** ('g') and rerun the code after you have done so. Observe the plot.
10. What did you observe about the tangent lines in the previous part? What are they converging towards? Answer this as a comment in the template file.
11. Take the derivative with respect to t of $h(t)$ analytically and evaluate it at the critical point in time. Is this value close to the approximations you made? DO NOT suppress output to the command window.
12. Add another element to the vector in Part (5) with a value smaller than 1.0 and rerun the code. Note the new approximation for the derivative in the output to Part (8) corresponding to this point. Begin to reduce this increment, each time rerunning the code and noting the output. How small should this value be such that the approximation is accurate to within two decimals of the exact value of the derivative? Leave this value in your final code.

SUBMISSION:

To submit your code, in the editor window select *File > Publish filename.m*. A new window will open up with an HTML document. **Print this document to turn in along with your script file.**