DERIV2D – 2nd Project Progress Report

Name: Hai-Y Michael Tran Nguyen | Peoplesoft: 0925358

Date: 4/16/18

So far, I have started creating my program in C# .net Core using Visual Studio 2018.

My algorithm currently does the following parts of the algorithm:

* Loads the DERIV2D\_functionA\_XY.csv data into two lists from the supplied dataset
* Loads the DERIV2D\_functionB\_XY.csv data into two lists from the supplied dataset
* Computes the derivative of the X and Y from function A and creates a separate list of the derivative values at each interval. (e.g. Between points 1-2, 2-3, 3-4, etc.)
* Outputs the raw derivative values into a .csv file determined by the user input
* Compares the Derivative of function A with function B through 2 different methods
  + First method is to compare 1 point from function A to 5 points from function B. For example, point [0] from function A will be compared to the average of points [0, 1, 2, 3, 4] from function B. And then point [1] from function A will be compared to the average of points [5, 6, 7, 8, 9] from function B. This will be repeated until it reaches the end of function A. (There will be excess points left over in function B)
  + Second method is to compare 2 points from function A to 5 points from function B. For example, the averages from point [0, 1] from function A will be compared to the average of points [0, 1, 2, 3, 4] from function B. And then the averages from point [1, 2] from function A will be compared to the average of points [5, 6, 7, 8, 9] from function B. This will be repeated until it reaches the end of function A. (There will be excess points left over in function B)
* Outputs the raw Derivative comparison values into a .csv file

One additional comparison technique I plan to implement is to dynamically choose the average from 5-6 points in function B using some sort of calculation technique such that there will be no excess points in function B left over.

These 3 techniques will be compared to see the differences between the different techniques.

With the outputs from above (and future outputs) I plan to load the data onto R, and create the graphs, plots, comparisons, statistical analysis, etc. for the report.

For comparisons on the raw Derivative comparison values, I use a couple of comparison techniques like: comparing x and y values separately, comparing magnitude of x and y combined, comparing magnitude and angle of x and y combined.

Next steps to complete:

Normalizing my data in function A derivative and function B. My planned method is to find the absolute maximum of function A and function B, and then divide all the data in function A derivative and function B by that maximum. This will make all my data range from [-1, 1]. This will potentially make the data easier to analyze? I’m not entirely sure what the point of normalizing is at the moment.

As stated in my first progress report I also wanted to interpolate both of the functions but I’m not sure if I’ll have enough time to do so, so I’ll leave that on the back burner for now and if I do end up having time I will attempt this part.

I’m coding my code in such a way that it supports any dimensions so I believe I should be able to do something with DERIV3D, but my main analysis will be done on DERIV2D.

If you have any comments or tips on how I’m proceeding with my project, please let me know!