Numerical Derivative

Rebecca Corley†

[1]University of North Georgia, Department of Physics and Astronomy

1. INTRODUCTION

In this project, we were given noisy data to implement a numerical derivative. Initial findings showed that finding a derivative is not as trivial as numerical integration. We found that when given a set of noisy data to try to take a simple derivative of, we get noisy data back. To avoid this problem, we implemented two newly learned techniques: binning and the five-point stencil method. Binning takes the average of points in some region. The number or width of bins used depends on the data set to find the appropriate number of bins. The five-point stencil method is a technique used for numerical differentiation. This technique takes a central point with four surrounding points to calculate finite difference approximations to derivatives at each point. In one dimension, the stencil is simply a line. In two dimensions the stencil makes a + shape with one point in the middle and four surrounding points. Here h represents the spacing between the points in the grid, then the five-point stencil of of a point x is

$$\{x - 2h, x - h, x + h, x + 2h\}. \tag{1}$$

Using the equation to find a derivative:

$$f^{'}(x) = \frac{f(x-h) - f(x)}{f(a-b)} \tag{2}$$

and equation (1), we can find the first derivative using the five-point stencil given by:

$$f^{'}(x) \approx \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h}. \tag{3}$$

Thus the second derivative is given by:

$$f^{"}(x) \approx \frac{-f(x+2h) + 16f(x+h) - 30f(x) + 16f(x-h) - f(x-2h)}{12h^2}. \tag{4}$$

2. THE CODE

The code is structured as follows and can be seen in Fig. 1 and Fig. 2. First import required libraries and data. Covert the data to a list and sort the x-data. Define any constant integers used throughout the code. Using statistics from scipy, we can find the bin mean, edges, and numbers, which will be used in the derivative and five-point stencil method. The code calculates the first simple derivative, which returns a noisy function, then applies the five-point stencil method to return a smoother curve. Using the same logic, I then attempted to find the second derivative, but was unsuccessful. The last bit of code shows the plot functions used to generate plots.

E-mail: rlcorl0436@ung.edu

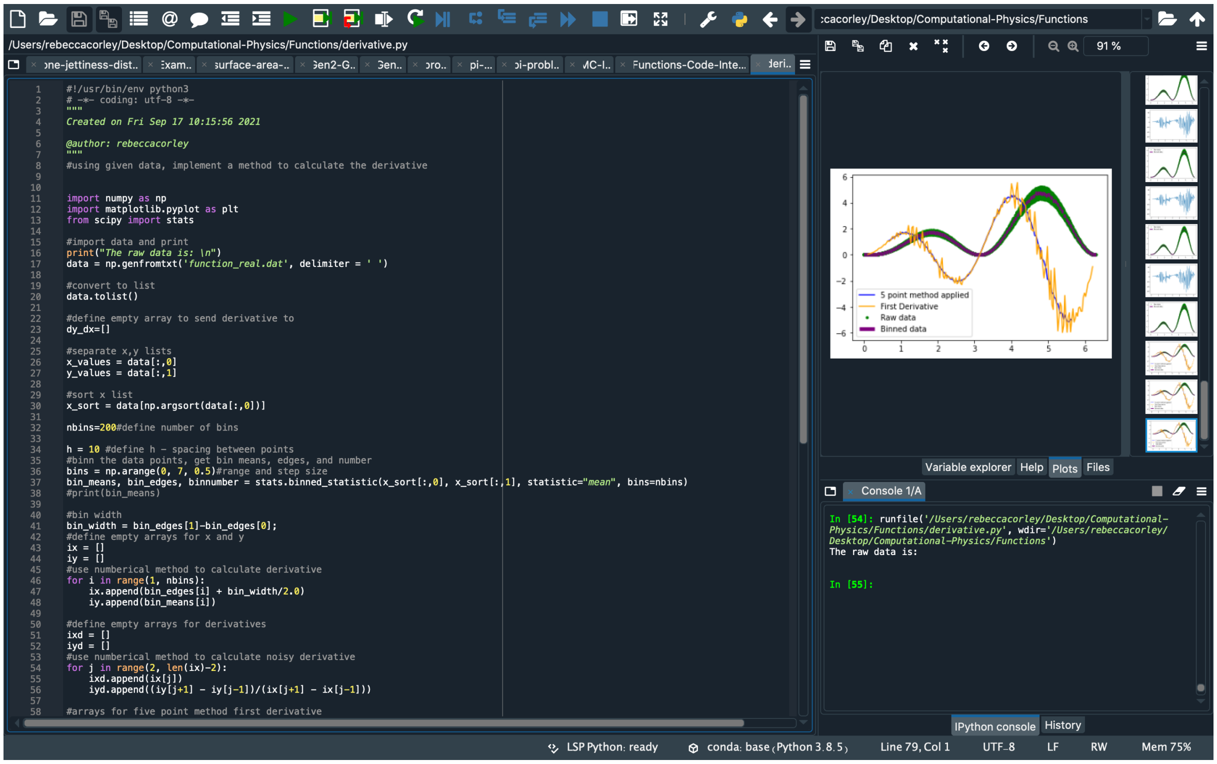

Figure 1.   Figure 1 shows the first half of the code used to find the derivatives.

3. RESULTS AND ANALYSIS

Taking the derivative of noisy data was not as trivial as anticipated. Fig. 3 shows the raw data (green), binned data (purple), simple first derivative (orange), and five-point method applied (blue). Binning the data took averages of points within a specified region, here the appropriate bin-width was 200. Taking a simple first derivative gave the noisy orange curve, so the five-point stencil method was applied, which gave the smooth first derivative. For this data, I used a bin-width of 200 and a step-size of 10. Increasing the number of bins, which decreases the width, makes the curve seem noisier. Decreasing the number of bins, which increases the bin-width, generates a smoother curve, but information is lost. A smaller step-size, around 5, shows some noise at the ends of the curve. Increasing the step-size past 20 caused the curve to disappear entirely. There seemed to be more appropriate range of useful step-sizes than bin-widths. Particularly bad combinations of bin-width and step-size would be extremes of both. Too large of sizes skips too much information, but too small of sizes does not seem to give enough room to make use of averages. The shape of the noisy second derivative in Fig 4 is what I would expect, but applying the five-point stencil for the second derivative still returned noisy data and it seemed to have changed the second derivative when I plotted them on the same plot, as seen in Fig. 5. Due to time constraints, I was not able to debug the code yet!

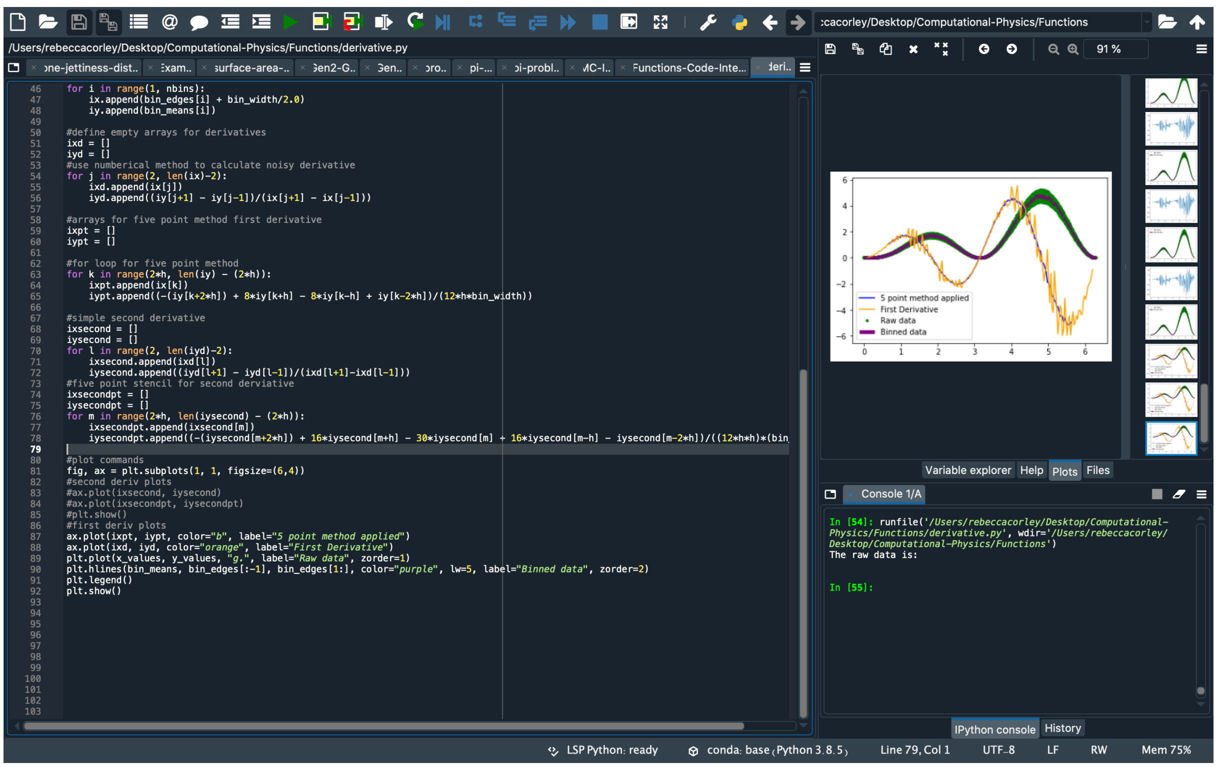

Figure 2. Figure 2 shows the second half of the code used to find the derivatives.

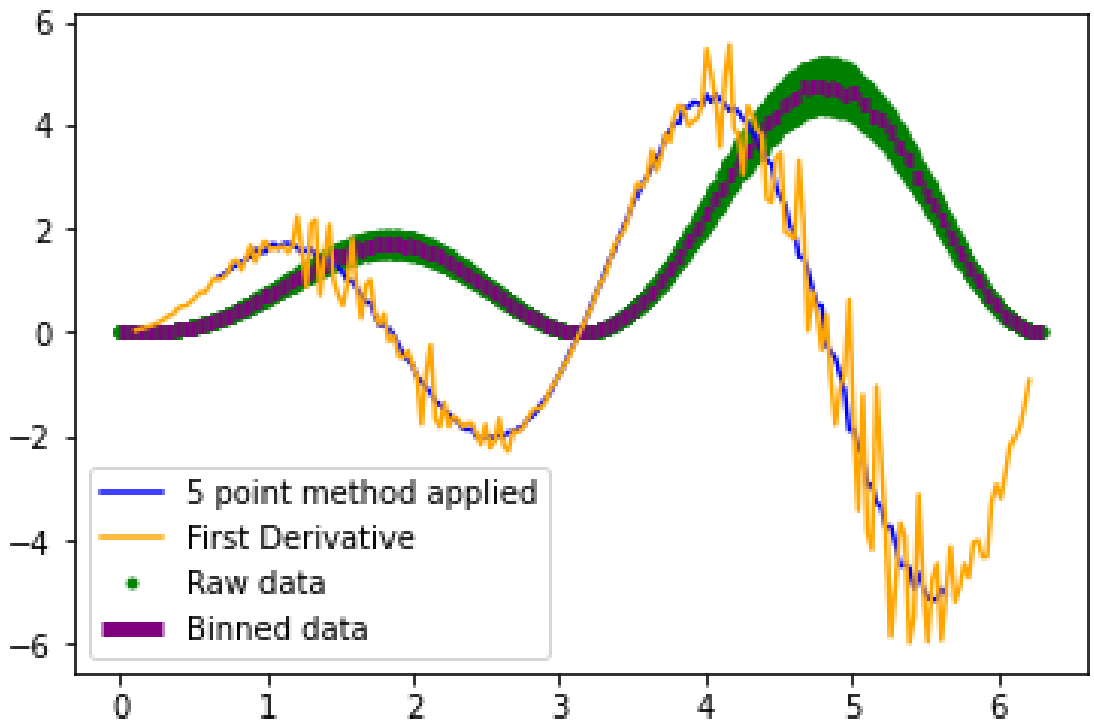

Figure 3. Figure 3 shows the raw data (green), binned data (purple), simple first derivative (orange), and the first derivative implementing the five-point stencil technique (blue). The simple first derivative still appears noisy, so we bin the raw data then use the five-point stencil method, which gives a smooth derivative.

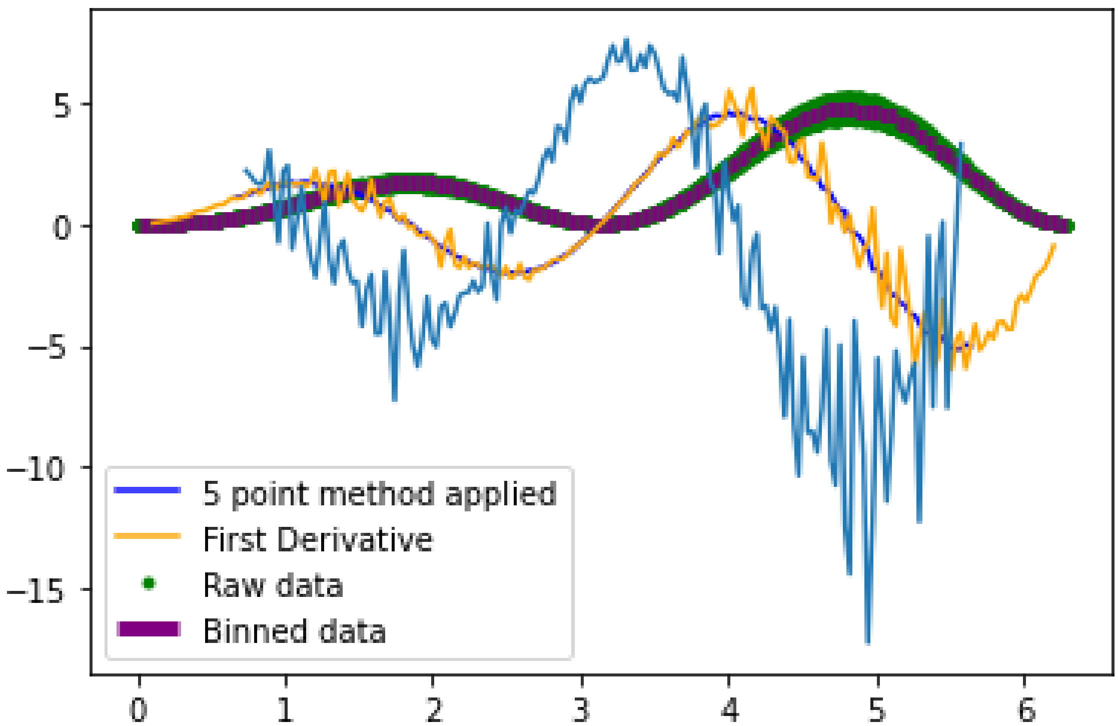

Figure 4. Figure 4 shows the addition of the second derivative (light blue) to the raw data and first derivative data. The second derivative curve is still noisy, but the shape is expected.

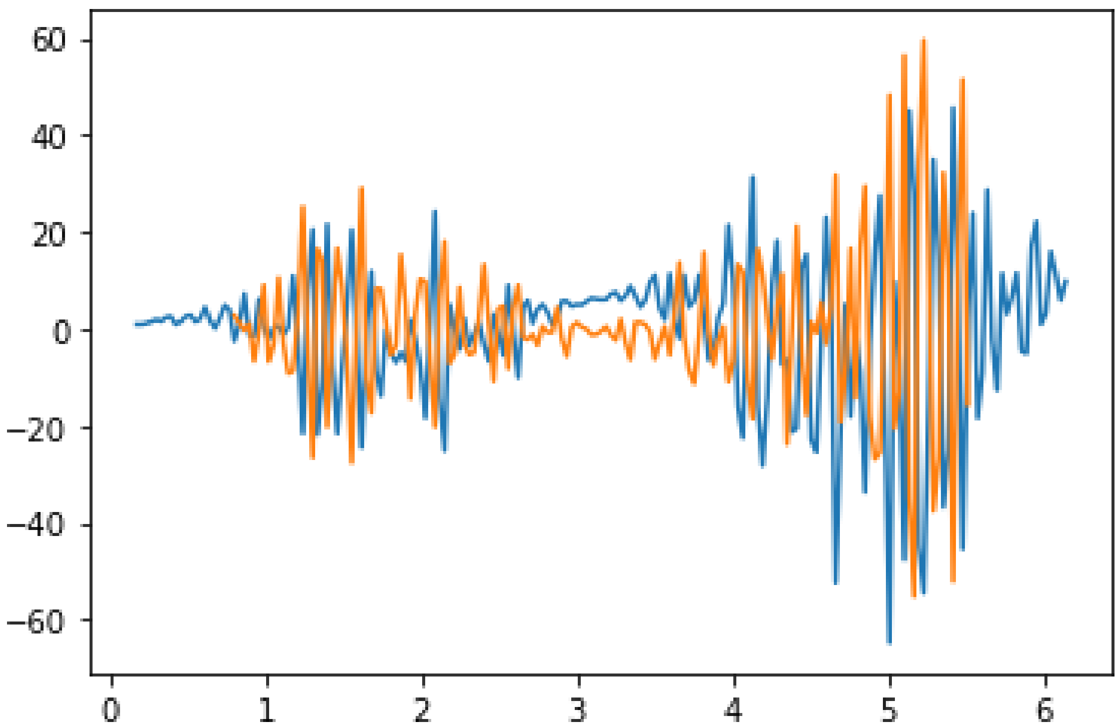

Figure 5. Figure 5 shows what happened when I tried to apply the five-point stencil method to the second derivative.