Assignment No 3

R Sumedhan

March 3, 2021

1 Overview

We are supposed to plot the data which are read from files and study the effect of noise in the fitting process.
By doing so, we wrote a python code for plotting the parsed data, separate the noise present in the data and use various engineering methods of computing errors based on quantities like σ (standard deviation).

2 Plots

2.1 Function for computation

The function along with noise is parsed from the data and is plotted against time, keeping the actual function as a reference.
We basically plot to find how much the plot varies from the actual plot so as to precisely find the *coefficients* of the equation.

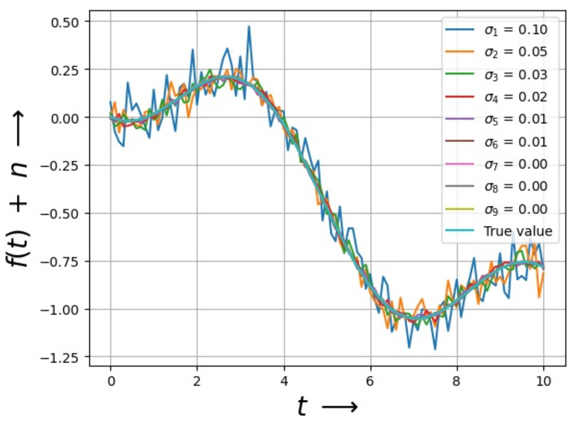

Figure 1: Function with noise

The equation is given as:

$$f(t) = 1.05J_2(t) - 0.105t + n(t) \tag{1}$$

and the standard deviation σ is given by the equation

$$Pr(n(t)|\sigma) = \frac{1}{\sigma\sqrt{2\pi}} exp(\frac{-n(t)^2}{2\sigma^2}) \tag{2}$$

2.2 Errorbar along with function

Here, we use a function **Errorbar** to plot every 5th data point along with the function to get a visualization of the extent of the error.

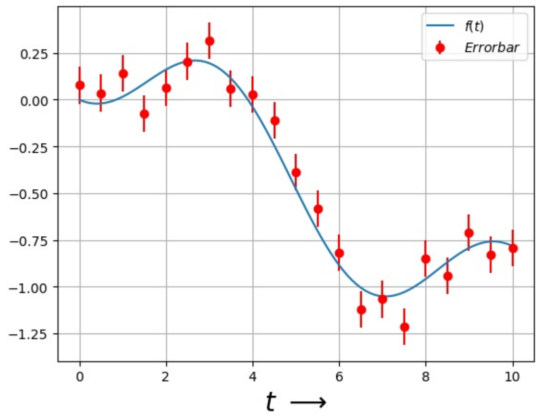

Figure 2: Errorbar plotted with the actual function

2.3 Finding the coeffecients by matrix method

We write down the given function f(t) as a matrix to find the best estimates for the coefficients in the parsed data which contains the noise.

$$g(t, A, B) = \begin{pmatrix} J_2(t_1) & t_1 \\ \cdots & \cdots \\ J_2(t_m) & t_m \end{pmatrix} \begin{pmatrix} A \\ B \end{pmatrix} \tag{3}$$

We prove that the above dot product of two matrices is equal to the function *g(t,A,B)*.

2.4 Plotting the epsilon contour

To find the best estimates of A and B, we calculate what is called the **mean squared error**, which is a form of computation of error in the values of A and B parsed from the data.

The equation is given by:

$$\varepsilon_{ij} = \frac{1}{101} \sum (f_k - g(t_k, A_i, B_j)^2 \tag{4}$$

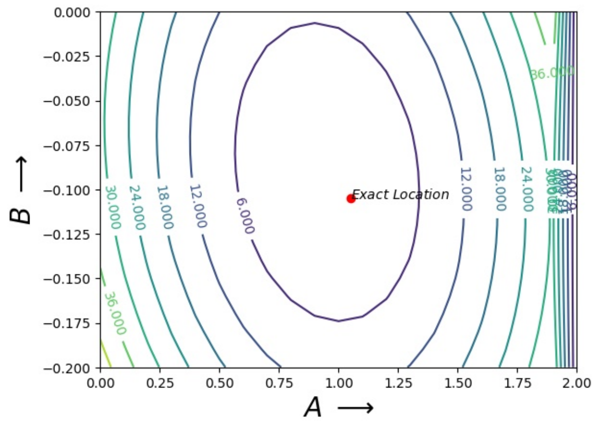

Figure 3: Mean squared error contour for A and B

2.5 Creating points of error of A and B based on *lstsq*

We now use what is called the *least squares program* for finding the error values in A and B and plot it against the standard deviation to find its relation.

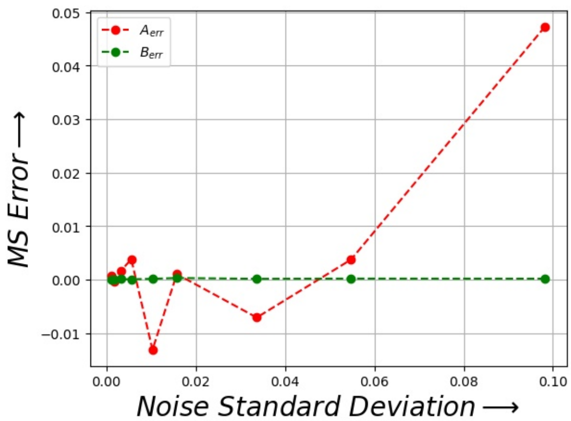

Figure 4: *Lstsq* based error against σ

2.6 Plotting *Lstsq* on log-log scale

The same *least squares program* is used but now is plotted against σ on a `log-log` scale to see if there is a linear relationship between the error and the standard deviation.

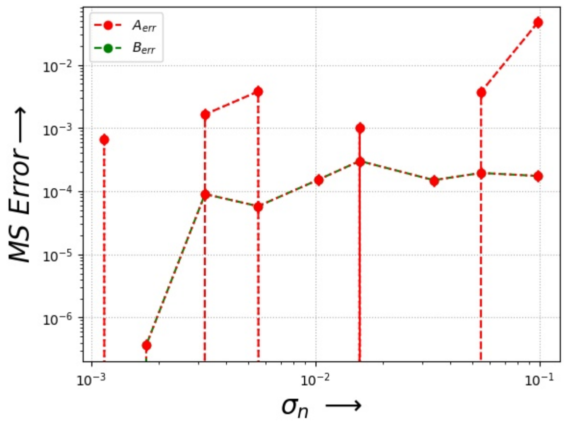

Figure 5: *Lstsq* based error against σ on `log-log`

3 Conclusion

The python code is too large for it to be present here and could be referred to in the folder.