Curve Fitting

AA 651, April 2021

Submitted by: Kishlay Singh

Roll Number: 2003121005

- **Fitting models to the given datasets and checking the goodness of fit:**

  In the following program, two different fitting methods have been used:

  1. The first one is scipy.optimize.curve_fit() with the syntax:

     scipy.optimize.curve_fit(f, xdata, ydata, p0=None, sigma=None, absolute_sigma=False, check_finite=True, bounds=- inf, inf, method=None, jac=None)

     It returns two vectors, popt and pcov. popt is the best fit values of the parameters of the fitting function and pcov is the covarient matrix.

  2. The second one is nimpy.polyfit() with the syntax:

     numpy.polyfit(x, y, deg, rcond=None, full=False, w=None, cov=False)

     It returns a vector containing the values of the parameters of the fitting function.

- **Testing goodness of fit:**

  To test the goodness of fit, the R^2 value for every fit performed. R-squared is a statistical measure of how close the data are to the fitted regression line. It is also known as the coefficient of determination, or the coefficient of multiple determination for multiple regression. A value of $R2$=0implies that the model does not fit the given data at all. R^2 values closer to 1.0 indicate better and better fit. It can be calculated by using the following formula:

$$R^2 = 1 - \frac{RSS}{TSS}$$

where,

R^2= coefficient of determination

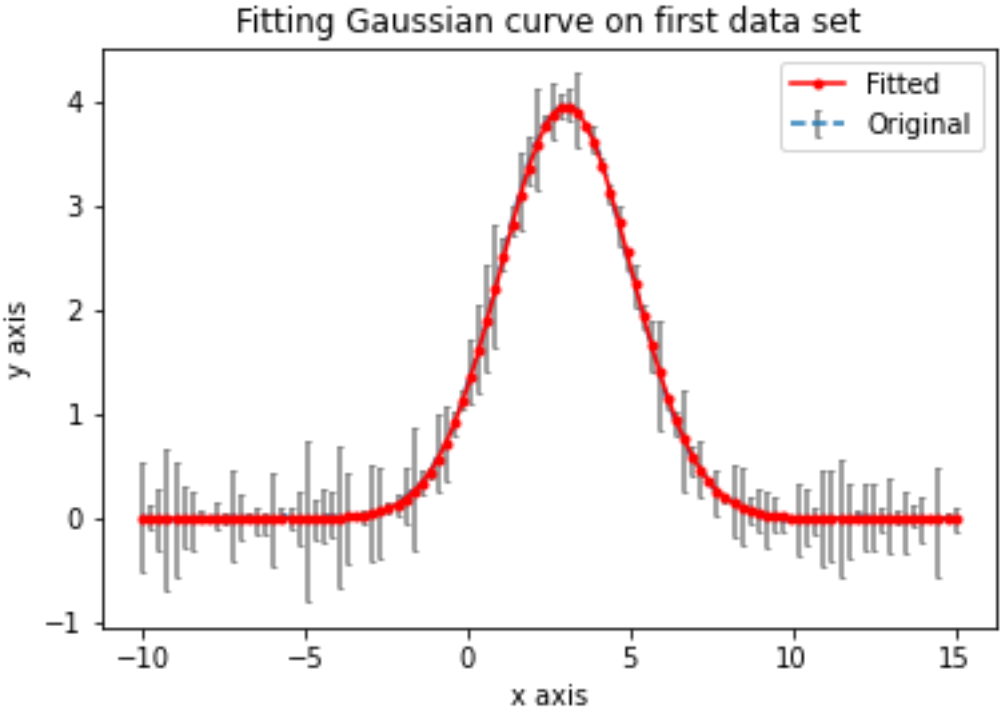
RSS= sum of squares of residuals

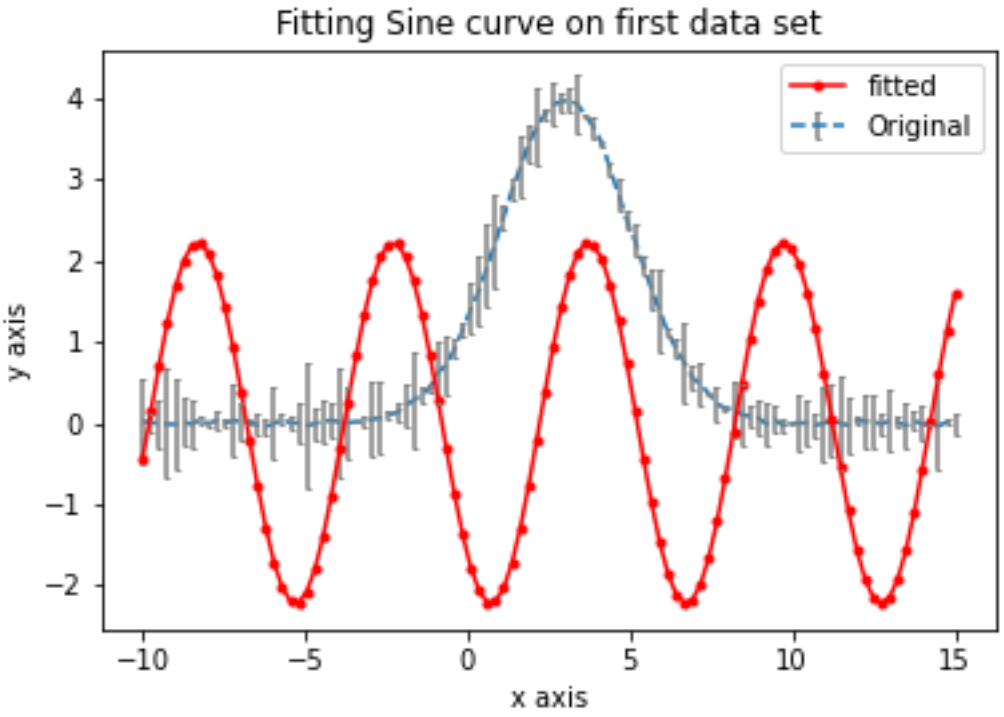
TSS= total sum of squares

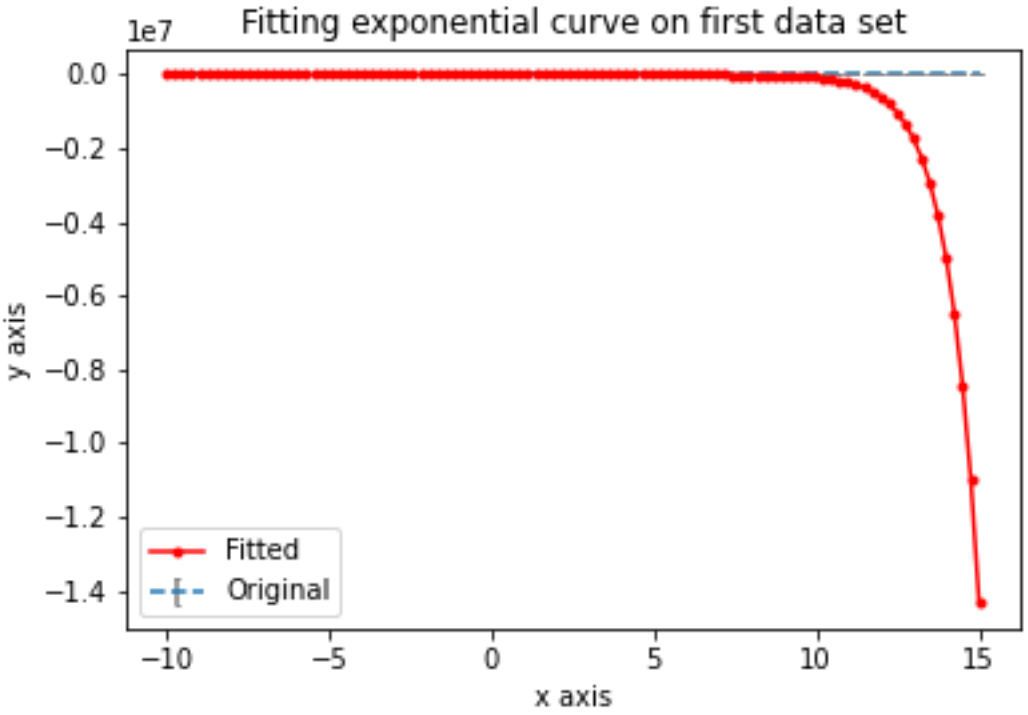
In this program, we indicate RSS by the variable *ssres* and TSS by the variable *sstot*.

- **Results:**

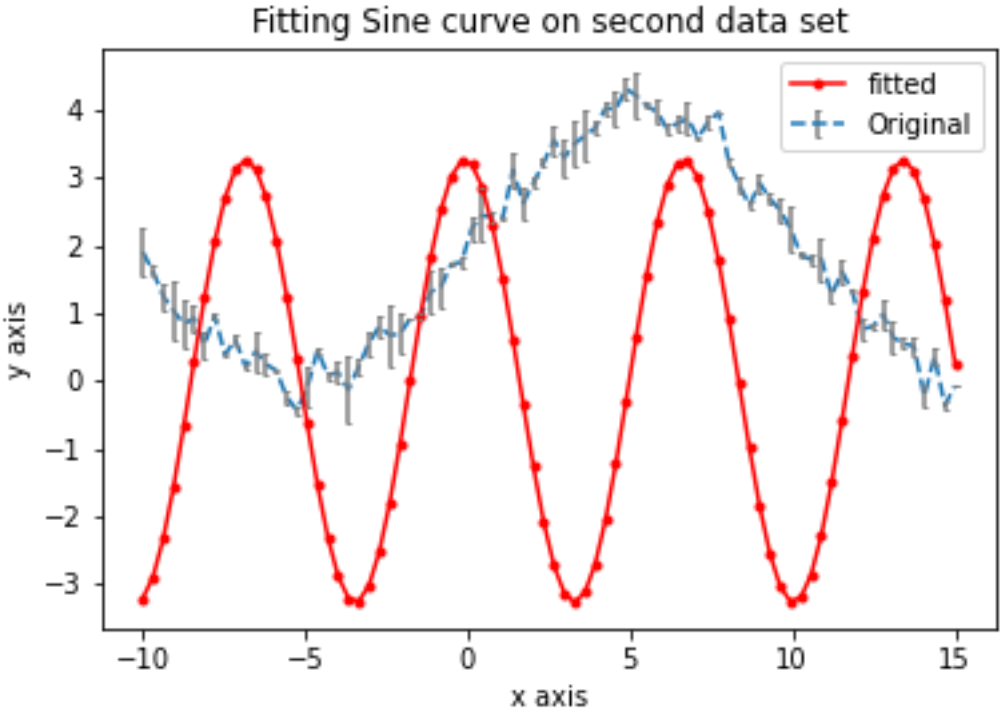
Goodness of fit for Gaussian fit on the first data set is: 0.9995711159812358
Goodness of fit for the Sine fit on the first data set is: -1.6978181951965765
Goodness of fit for the exponential fit on the first data set is: -3111589245948.62

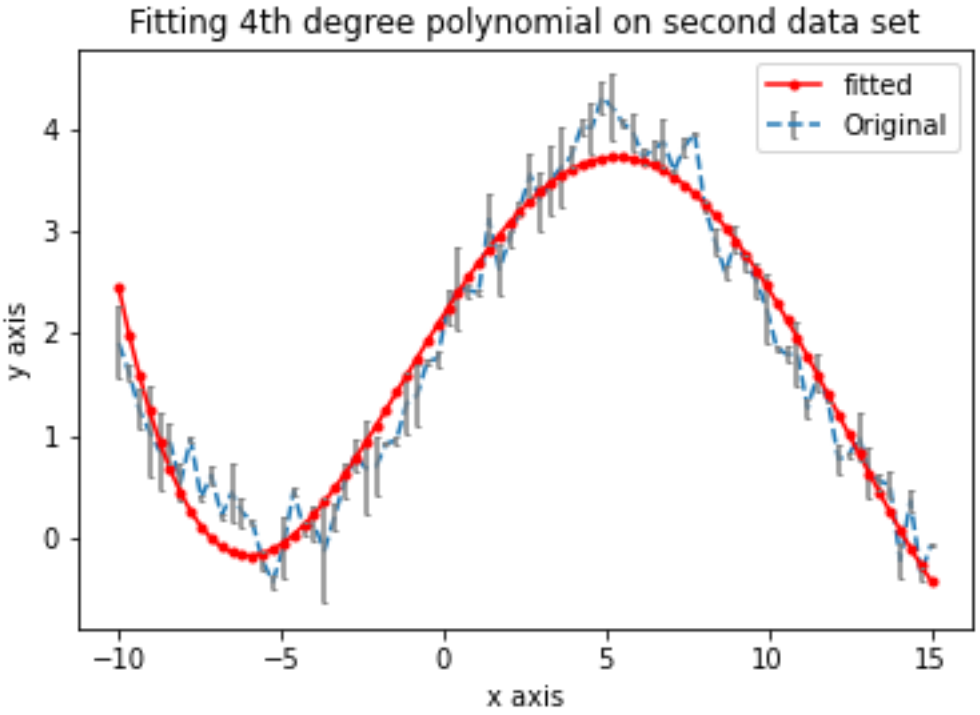


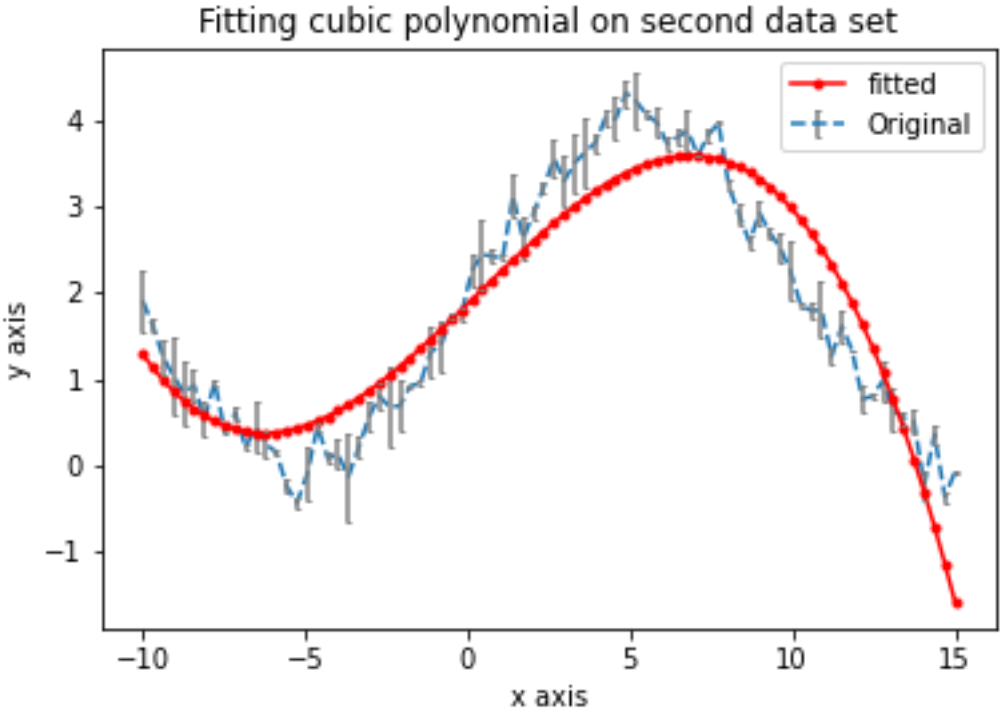




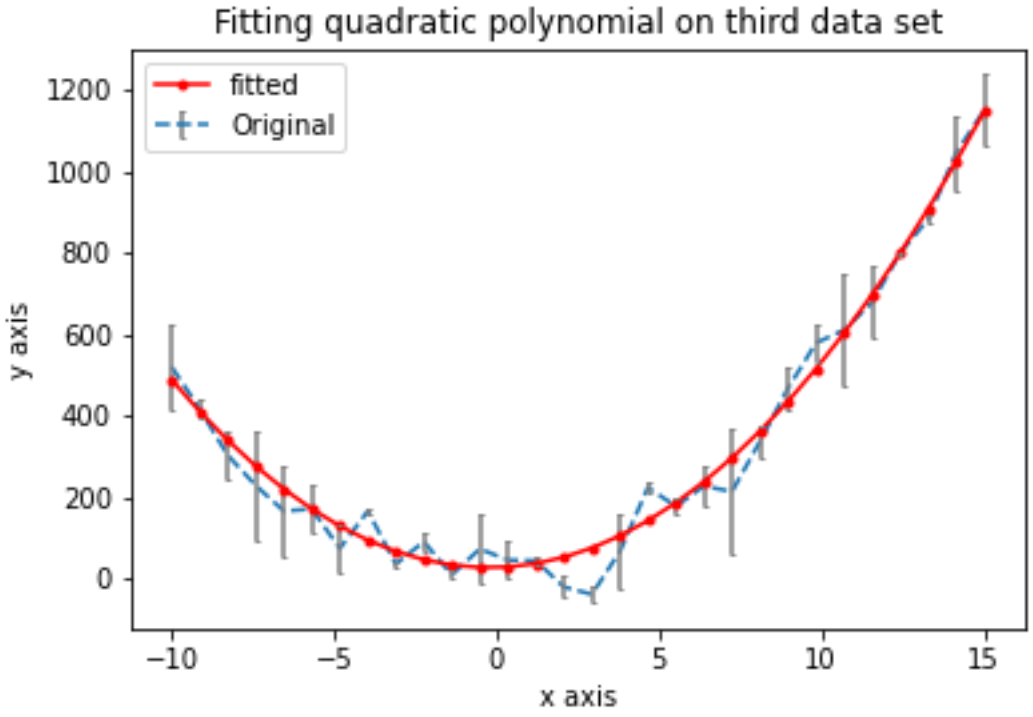
Goodness of fit for sine curve on second data set is: -4.289588471512167
Goodness of fit for 4th degree polynomial on the second data set is: 0.9516496960457752
Goodness of fit for cubic polynomial on the second data set is: 0.8550853502114691

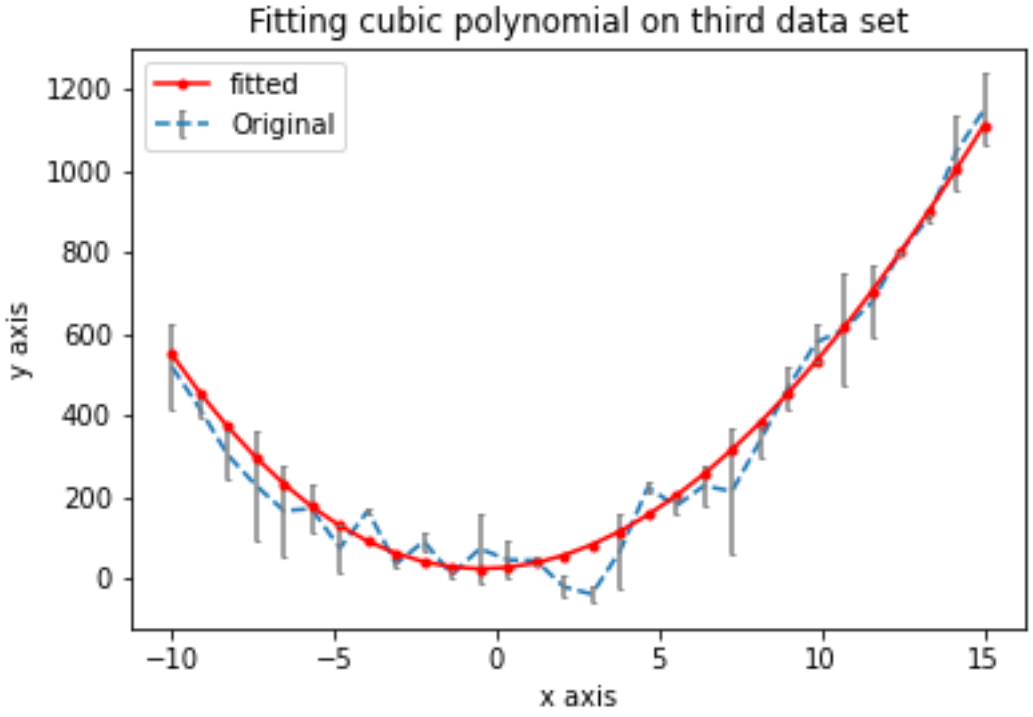


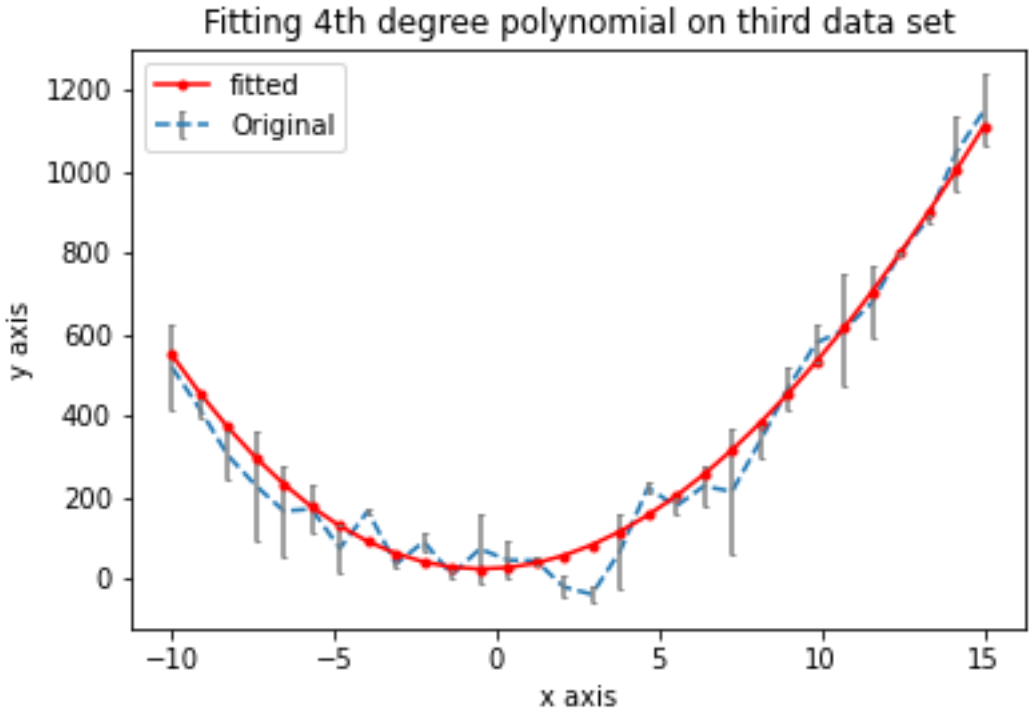




Goodness of fit for a quadratic polynomial on the third data set is: 0.9796039793555448
Goodness of fit for a cubic polynomial on the third set of data is: 0.9739840995444579
Goodness of fit for a 4th degree polynomial on the third set of data is: 0.9651015706151577


Fitting quadratic polynomial on third data set
fitted
Original
y axis
x axis
1200
1000
800
600
400
200
0
−10
−5
0
5
10
15


Fitting cubic polynomial on third data set
fitted
Original
y axis
x axis
1200
1000
800
600
400
200
0
−10
−5
0
5
10
15



- **Acknowledgements:**
  1. [numpy.polyfit — NumPy v1.20 Manual]
  2. [scipy.optimize.curve_fit — SciPy v1.6.2 Reference Guide]