Assignment 1:

1. Root-finding Using Bisection Method

Begin by considering the polynomial:



which as we discussed in class, has three real roots at x=-2.5,0.5, and 1.5

1. Develop a Jupyter Notebook which calculates the roots of this polynomial using the bisection algorithm. The program should fill an array of 10,000 data points between x=-4.0 and x=4.0. Then, the program should do a sparse search of this array, i.e. once every n\_search elements, to look for sign changes, and subsequently using the bisection algorithm to find the root. The program should also keep track of the time taken to find the roots.
2. Create a plot of the time taken to find the roots vs. the sparsification size (n\_search), for n\_search values between 1 and 10,000.
3. Deduce the source of the various patterns that you observe in part b). Hint: You might consider adjusting the number of data points in the original array, as well as the coefficients of the polynomial, slightly.
4. Create a single Jupyter Notebook that both finds the roots of the polynomial, as well as creates the plot from part b)
5. Modify the Notebook created in part d) to allow for the possibility to find the roots of other polynomials or functions.
6. Fitting Data
7. Go to the following site and retrieve data on the density of air as a function of altitude:

<http://www.engineeringtoolbox.com/standard-atmosphere-d_604.html>

1. Make sure to use the second table, in SI units.
2. Modify the Jupyter Notebook that we went through in class for Linear Regression to fit the density data as a function of altitude using a function of the form:
3. Determine the parameters A, B, and C, as well as the uncertainties on these parameters.
4. Plot the fit along with the data. What is your evaluation of the quality of the fit? Can you think of a different approach that might yield a better fit?