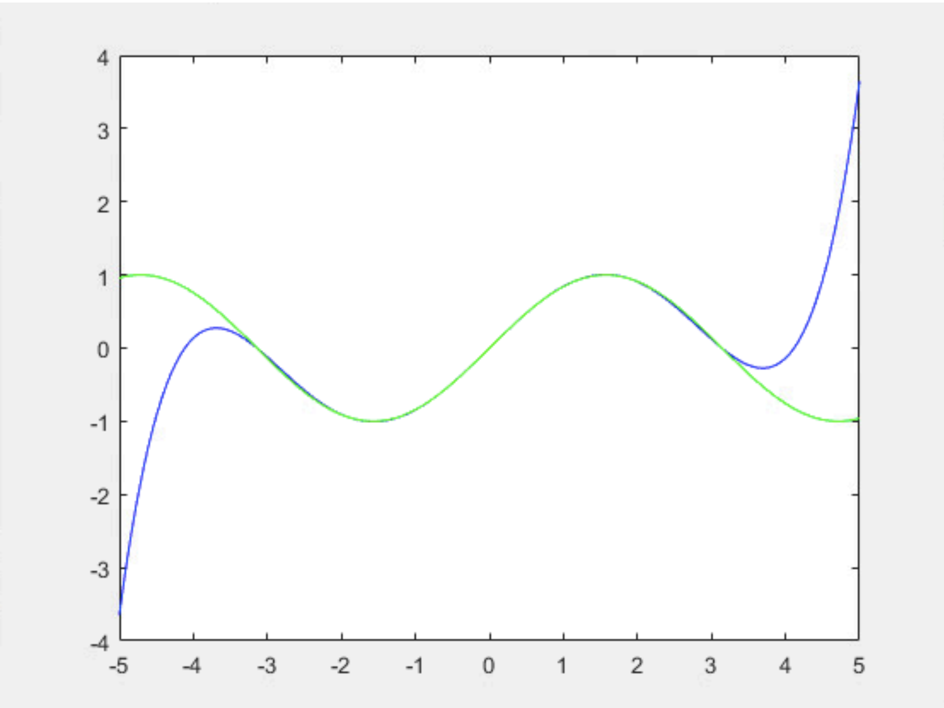
INTERPOLATING POLYNOMIAL AND THEIR OSCILLATORY BEHAVIOR

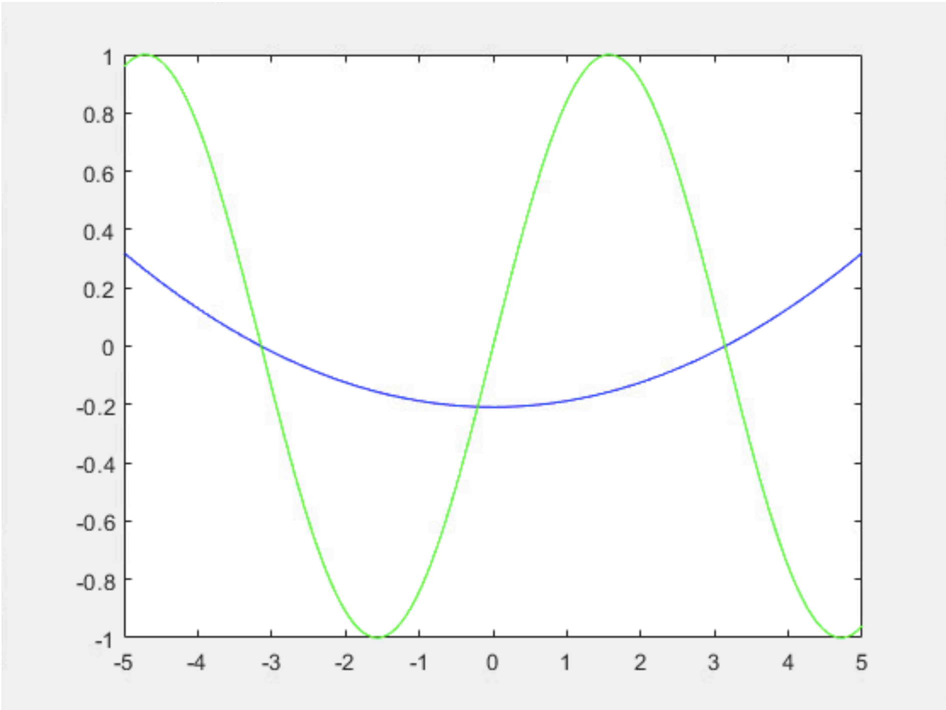
From the experiment we have done in part 1-3 of the project, we have seen that building the Vandermonde matrices and solve the Ax = y equation have produced a fairly close estimation of the function in a given interval. However, using the method to get the interpolating polynomial can be extremely error prone when dealing with large data set.

Researches and experiment have shown that using a sparse large data set, the interpolating polynomial will work very effectively. However, for dense data set, the computational cost is enormous. Hence, interpolation methods are mostly used in theoretical context, when we are trying to prove a theorem. For real life application, we can use interpolating polynomial with other more efficient algorithms, like artificial neural network, to get a better prediction of the data set.

For example, I have conducted an experiment with f(x) = sin(x) function and trying to find the interpolating polynomial of f. Using 6 data points, we get a pretty close match of the real function, demonstrated by the figure below:



In the figure above, the green line represent the real graph of the sin function, and the blue line represent the graph of our interpolating polynomial. However, when we use the data set of 15 points, the result we get is not at all close to the function of f:



Thus, through the experiment above, we have seen how interpolating polynomial is not effective with large, dense data set, due to the error generated when performing matrix operations and rounding real values.

Reference:
Robert Raturi (2018), Large Data Analysis via Interpolation of Functions: Interpolating Polynomials vs Artificial Neural Networks, American Journal of Intelligent Systems