

Programming Assignment Five – Interpolation and Numerical Differentiation¹

Background

This assignment has various problems in interpolation, regression, and numerical differentiation that involve MATLAB, VBA and the Excel worksheet. Unless stated otherwise all problems are to be attempted on the “standard” data used in classroom examples. This data set is shown below:

x	805	825	845	865	885	905	925	945	965	985
y	0.710	0.763	0.907	1.336	2.169	1.598	0.916	0.672	0.615	0.606

1. Using MATLAB

The two assignments below can be done by following examples in the MATLAB help files.

- a. Review the help file for the MATLAB spline function. Copy the standard data set above into MATLAB and plot a curve showing a comparison of the data and a cubic spline fit. (Obtain plot data for $\Delta x = 1$.) Compare this plot to the one in the course notes. Discuss this comparison and list the option that you used in MATLAB for handling the endpoints.
- b. Review the help file for the MATLAB function polyfit. Obtain a 9th order polynomial to fit the same data used for the spline fit. Use the transformed x variable to reduce error as described in the help file for polyfit. Prepare a plot that compares the interpolated results with the original data. (Obtain plot data for $\Delta x = 1$.) Discuss any difference between your results and those in the course notes.

2. Using EXCEL VBA

Download the workbook pa5Start.xlsm from the course web site and use this workbook for these exercises. Be sure to change the name of the workbook to <yourLastName>_5.xlsm. The workbook contains two tabs: the “Interpolation” tab has the data for part a, and the Regression” tab has the data for part b. Your Excel work should be on these tabs.

- a. You have just been employed by a company which has some staff members who use Excel and would like a simple way to interpolate. They use various spreadsheets that have data tables on two variables, which can be called x and y in your VBA code.

In one or more cells on the worksheet, the users will have a value of x (not part of the data table) for which they want to find an interpolated value of y. (If there is more than one such value, they will be in adjacent cells in the same column.) You have been assigned to write a user-defined function² to accomplish this task. For this task, you should use the Newton polynomial discussed in class.

Your function should allow individuals to select the x and y data they want to use for the interpolation, the cell with an x value that they want to use as the input to the interpolation formula, and the location for the output y value. The input x and y data that the user selects for

¹ Assignment due Wednesday, Monday, April 14, at 11:59 pm. May be submitted by Wednesday, April 16, at 11:59 pm with a 30% penalty. No grade for assignments submitted after this date.

² You can write a macro if you prefer to do so, but it will be much harder to accomplish all the following requirements: (1) the user can choose all inputs with the mouse, (2) the user can set any cell location for the output with the mouse, (3) the interpolated result should update automatically if the input x value is changed.

the interpolation will determine the order of the polynomial. *E.g.*, if the user selects 3 x values and 3 y values a second-order polynomial should be fit. You can assume that the cells for the x data are in adjacent rows in the same column. You can make the same assumption for the y data. Your function should ensure that the user has selected the same number of cells for both the x and the y data.

Apply your VBA code to obtain interpolation results for $x = 853$ and 962 using linear, quadratic and cubic polynomials **that properly select the correct input data to place these x values near the center of the range for the interpolation data**. Discuss the differences in your results from the different polynomials for each interpolated x value.

Some suggestions

You can use code samples provided for exercises one and four to see how to get cell ranges into VBA variables and check the size of two arrays for compatibility. You can use example code provided in class for constructing the divided-difference table and the Newton polynomial. Remember that arrays entered from the worksheet always have an initial subscript of one.

Once you have the divided-difference table and identify the a_k coefficients in the Newton polynomial, you have to write the code to construct the polynomial and solve it for the given user input value. To see how to do this, look at the following polynomial equations from the lecture on Newton polynomials.

$$p_0(x) = a_0$$

$$p_1(x) = a_0 + a_1(x - x_0) = p_0(x) + a_1(x - x_0)$$

$$p_2(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) = p_1(x) + a_2(x - x_0)(x - x_1)$$

$$p_3(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)(x - x_2) = p_2(x) + a_3(x - x_0)(x - x_1)(x - x_2)$$

$$p_4(x) = p_3(x) + a_4(x - x_0)(x - x_1)(x - x_2)(x - x_3)(x - x_4)$$

This sequence should suggest a way for you to construct the polynomial with only one for loop. Use two program variables to create the polynomial, one to represent the running sum of the polynomial terms, and the other to represent the product terms, $(x - x_0)(x - x_1)(x - x_2)(x - x_3)(x - x_4)\dots$. Each time through the loop you will multiply this product-term variable by the new $(x - x_k)$ term to give a new value for the product term. After that you will multiply the new product term by a_k and add it to the running sum term.

- b. Use the LINEST function to find the regression coefficients for the data shown on the "Regression" tab of the workbook for this assignment and use them as described in the steps below.
 - i. Obtain the regression coefficients for y as a function of three variables, x_1 , x_2 , and x_3 ; identify the regression coefficients and their standard errors on the worksheet.
 - ii. Once you have obtained these coefficients, use them to compute the estimated y values for all the data points. Use these estimated values to compute the R^2 coefficient for the regression and compare it to the value given by LINEST.
 - iii. Use the T.INV.2T function (or the compatibility function, TINV) with a confidence level of 95% to find the confidence limits for the regression coefficients.

3. Numerical differentiation

Do the following assignment on an Excel worksheet or in the MATLAB command window, as you prefer. Use each formula below to compute the derivatives of $\cos(x)$ at $x = 1$ for a range of h values from 0.1 to 10^{-10} . Plot the errors (the difference between the numerical result and the correct value of $d\cos(x)/dx = -\sin(x)$ and $d^2\cos(x)/dx^2 = -\cos(x)$) on a log-log plot and verify that the order of the error is correct. Identify the locations on the plot where the roundoff error begins to occur.

$$f'_i = \frac{f_{i+1} - f_{i-1}}{2h} + O(h^2) \quad f'_i = \frac{f_{i-2} - 8f_{i-1} + 8f_{i+1} - f_{i+2}}{12h} + O(h^4)$$
$$f''_i = \frac{f_{i-1} - 2f_i + f_{i+1}}{h^2} + O(h^2) \quad f''_i = \frac{-f_{i-2} + 16f_{i-1} - 30f_i + 16f_{i+1} - f_{i+2}}{12h^2} + O(h^4)$$

Submission Requirements

You should submit only two files for this assignment. The first is a Word file that discusses the questions given above and summarized below. Include all your relevant MATLAB results in the Word file. The second is an Excel file with your Excel results. If you use Excel for part 3, create a separate worksheet tab for this part; if you use MATLAB, include the results from this part in the Word file.

Your discussion for this assignment should contain the following items:

1. Your answers to the discussion questions for each of the two parts of the MATLAB assignment.
2. A set of user instructions for the VBA interpolation program that you wrote in part 2a.
3. A brief discussion of your regression results. Do you think you got a good fit? If so, why? If not, why not?
4. A brief discussion about the third part of the assignment. Describe where you think roundoff error started for each formula and if the results corresponded to the order shown for the finite-difference expressions. Give your reasoning for these answers.