Journal Report 5

9/30/19-10/3/19
Jialin Tso
Computer Systems Research Lab
Period 2, White

Daily Log

Monday September 30

Attempted to resolve ”TypeError: ’numpy.float64’ object cannot be interpreted as an integer” message. Discovered that the list of x coordinates must be converted to a numpy array. Graphed both linreg function’s original data and fitted line.

Tuesday October 1

Wrote polyreg function for all polynomials (equations and graphs generated).

Thursday October 3

Wrote expreg function (almost working) and attempted to solve ”raise RuntimeError(”Optimal parameters not found: ” + errmsg), RuntimeError: Optimal parameters not found: Number of calls to function has reached maxfev = 800.” error.

Timeline

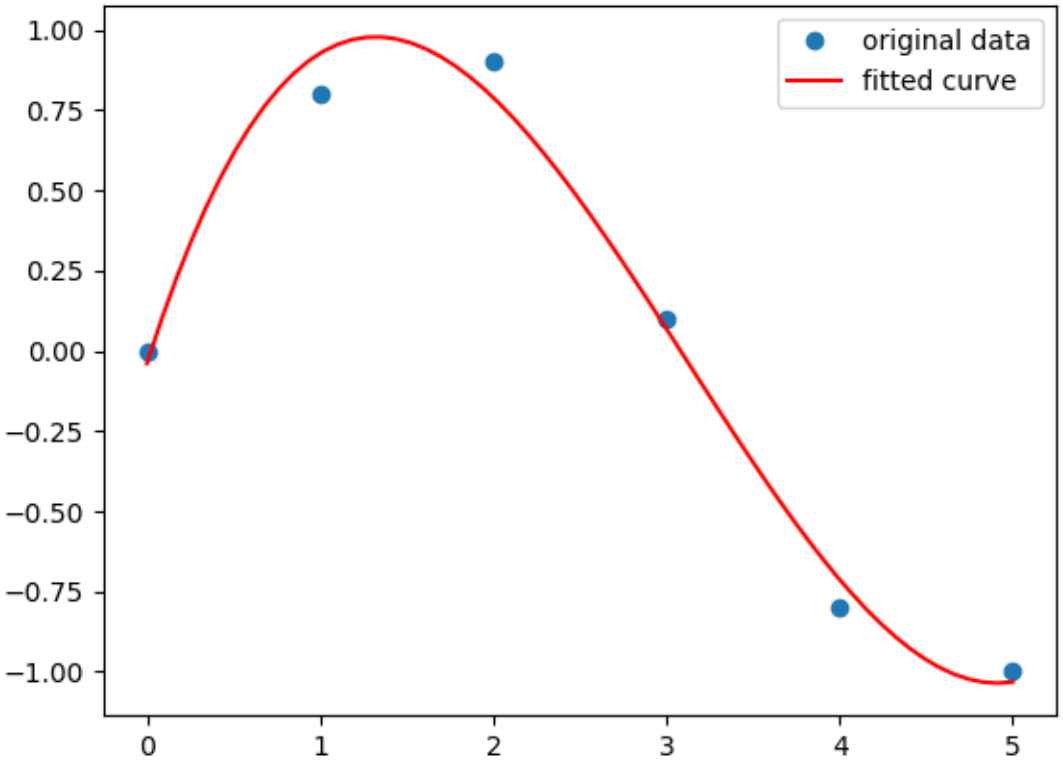
Date	Goal	Met
Sept 9	Install SpeechRecognition and word2number; text to expression for factorial, logarithms, and other useful functions	SpeechRecognition, word2number, and num2words installed. Factorial and logarithms work.
Sept 16	Solve integrals, derivatives, and permutations and combinations;	Indefinite and definite integrals and derivatives (not at specified value) solved. Permutations and combinations work for a certain format.
Sept 23	Solve matrices and find 1-var stats for user-inputted list.	Common matrix functions like addition, reduced row echelon, and determinant work. 1-var stats outputted.
Sept 30	Find regression equations from user-inputted lists	Linreg and Polyreg can generate equations and graphs. Expreg almost works.
Oct 7	Begin experimenting with SpeechRecognition and converting answers from expression to text	

Reflection

The biggest challenge of this week was resolving issues relating to graphing equations. I took longer than expected to graph polynomials because when I tried graphing the fitted curves, I would get an error about plt.plot() not recognizing the new y-list. I did lots of research, tried many methods, and discovered that poly.polyfit() and poly.polyval() worked the best. Another issue I had was with expreg() running out of memory to call the exponential function formula. I resolved this by setting maxfev = 2000 when calling curvefit. Some of my trial lists had numbers that were too large for the program to handle, so I moved the decimal points in the y-list two places to the left. However, I got coefficients that were too small after running my code, so for next week, I shall continue to figure out what went wrong.

Example of new input and output of current program: Input: polynomial left left-bracket zero comma one comma two comma three comma four comma five right-bracket comma left-bracket zero comma point eight comma point nine comma point one comma negative point eight comma negative one right-bracket comma three right

Expression output: polyreg([0,1,2,3,4,5],[0,.8,.9,.1,-.8,-1],3) Answer (coefficients of curve equation): [0.08703704 -0.81349206 1.69312169 -0.03968254] Graph:


original data
fitted curve
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
1
2
3
4
5