Assignment 2

Yifei Gu

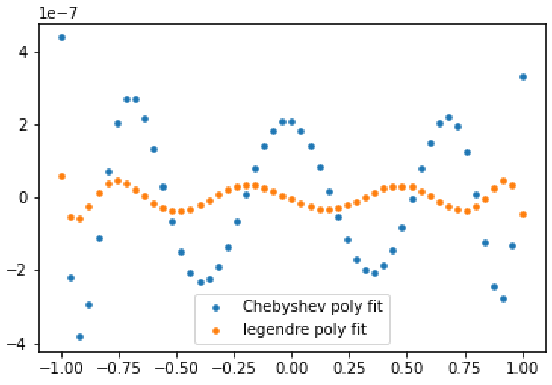
260906596

Q1

For question 1, My mind road is to compare the new x in the new (x1, x2) in every recursion using a non_match_elements function I defined and only calculate y(x) for those new x. The new y(x) will be add in an array storing calculated y(x). In this case, we do not recalculate the same y(x) multiple times. But unfortunately, my coding skill really limit what I can do. q1.py file is what I get so far.

Q2

For this question, I wrote a routine for Chebyshev Polynomial fit using the similar routine presented in the class. Since the range of x given in the question (0.5, 1) is not in the Chebyshev Polynomial range, I rescale x to (-1, 1) by applying a linear transfer.



The terms we need is **9** (including the constant) to reach an accuracy in the region better than **1*10^-6.**

As expected:
My RMS error of Chebyshev poly fit is 1.89454222761301e-07
My RMS error of Legendre poly fit is 2.9057336755082354e-08
The **average** error of **Chebyshev** poly fit is **bigge**r than that of **Legendre** poly fit.

My max error of Chebyshev poly fit is 4.392196097935397e-07
My max error of Legendre poly fit is 5.940760317013627e-08

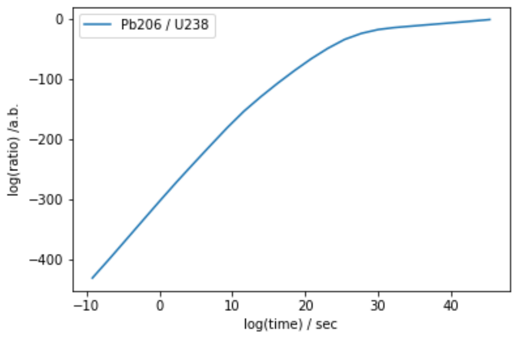
In class, we looked at two Chebyshev polynomials fit. One is similar to Legendre fit, one behaves differently. I am not sure if the reversed trend of max error is due to the way I did Chebyshev or it something else messed up.

Q3

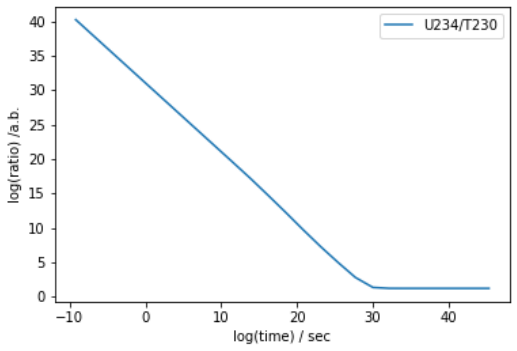
a)

For this question, I chose Scipy ODE driver to solve implicitly, which will save much more time. What I did is to build a system of 15 differential equations. It took 184 evaluations and 0.07795262336730957 seconds to solve. I also tried to solve it with RK4. It couldn't give me the result as quick as the implicit option.

b)



In the beginning U238 decay slowly, so the ratio is small. As Pb206 is grow faster, the ratio increases.



There is a linear relation between the log of the decay rate.