Assignment 3
Phys 512 Computational Physics

Hans Hopkins 260919593

October 9, 2021

Q1:
Solving the ODE analytically gives:

$$y = ce^{\arctan x}$$

And plugging in the initial given value:

$$1 = ce^{\arctan(-20)}$$
$$\implies c \approx 4.57605801$$

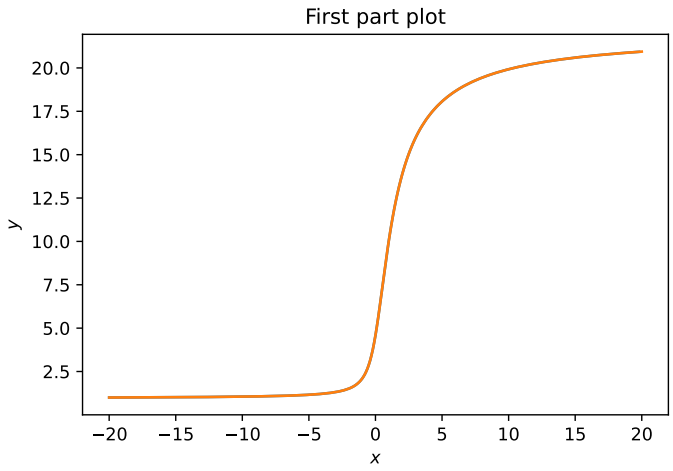

Figure 1: The real function and the Rk4 approximation overlap

For the second part, I'll use the formula (17.2.3) derived in Numerical Recipes:

$$y(x+h) = y_2 + \frac{\Delta}{15} + O(h^6)$$

where y_2 is the approximation of two step sizes of 2, and Δ is $y_2 - y_1$, where y_1 is the original approximation.

This process costs 11 function evaluations since the k_0 for the original approximation can be reused for the first of the two smaller steps.

Now we need to figure out how many steps to take for comparison. We can't use 200 again since 4*200 isn't a multiple of 11. Let's just use the first number over 200 which is a multiple of 11, i.e. 209 steps. That means we'll do 836 function evaluations total, and for the half-step approximation we'll have to use 76 steps.

For the error calculation I'll be using

$$\text{average error} = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1}(y_{true} - y_i)^2}$$

The average error using the first method came out to 0.0013963504159896977, but the stepd method came out to 1.7704318515121642e-05, so the stepd method is a few orders of magnitude better for the same cost.

Q2:

a) Let Y be the amounts of each product, but scaled so that $\text{sum}(Y) = 1$. (Or in other words, 1 is the initial amount of U238)

Then the ODE is

$$\frac{dY}{dt} = \tau Y$$

Where τ is a matrix containing all the decay rates.

The solver I chose to use is "Radau" because it was the first implicit method listed on the page for solve_ivp. I knew I need to use an implicit method because my matrix contains both very large and very small values, so I'm probably going to run into machine rounding errors.

I ran it for a full half-life of Uranium-238, and these are the product fractions by the end:

Name	Fraction
Uranium-238	5.00001080e-01
Thorium-234	7.38387942e-12
Protactinium-234	8.55324899e-14
Uranium-234	2.74747113e-05
Thorium-230	8.43616570e-06
Radium-226	1.79064340e-07
Radon-222	1.17154775e-12
Polonium-218	6.59626509e-16
Plomb-214	5.70257757e-15
Bismuth-214	4.23437663e-15
Polonium-214	5.82670083e-22
Plomb-210	2.49570925e-09
Bismuth-210	5.61255424e-07
Polonium-210	4.23994440e-11
Plomb-206	4.99962267e-01

There's pretty much nothing except the original uranium and the final lead.

b)

Here are the two plots:

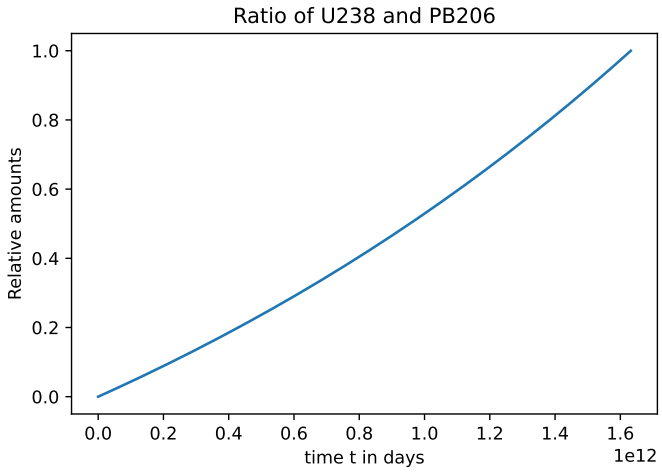

Figure 2: This is exactly what we expect. Over a full half-life, only half of the uranium is left, and it's pretty much entirely converted into lead, so the final ratio will be 1:1.

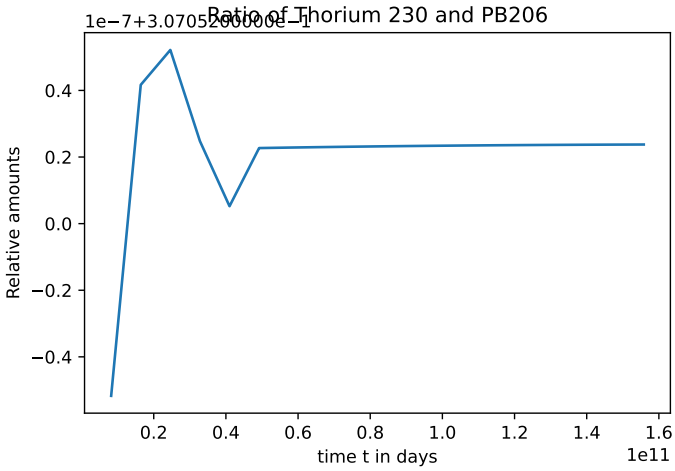

Figure 3: Well this is a little wonky, and changing the number of points didn't seem to help to much. I think the important thing is that it levels out to about a third, which corresponds to how the half life of of thorium-230 is about a third of the half life of Uranium-234.

Q3:

a) Let's expand the equation to find parameters which are linearly associated with functions of x and y.

$$\begin{aligned} z - z_0 &= a((x - x_0)^2 + (y - y_0)^2) \\ z &= a(x^2 - 2xx_0 + x_0^2 + y^2 - 2yy_0 + y_0^2) + z_0 \\ &= a(x^2 + y^2) + ax_0(-2x) + ay_0(-2y) + (ax_0^2 + ay_0^2 + z_0) \end{aligned}$$

So define:

$$\begin{aligned} a &= a \\ b &= ax_0 \\ c &= ay_0 \\ d &= ax_0^2 + ay_0^2 + z_0 \end{aligned}$$

These equations can be inverted to give:

$$\begin{aligned} a &= a \\ x_0 &= \frac{b}{a} \\ y_0 &= \frac{c}{a} \\ z_0 &= d - a\left(\frac{b}{a}\right)^2 - a\left(\frac{c}{a}\right)^2 = d - \left(\frac{b^2 + c^2}{a}\right) \end{aligned}$$

b) Running the fit gives the results:

$$\begin{aligned} a &= 0.00016670445477401358, & x_0 &= -1.360488622197728 \\ y_0 &= 58.22147608157934, & z_0 &= -1512.3118166896197 \end{aligned}$$

c) Printing out the residuals squared shows that they also pretty much follow the same shape as the dish:

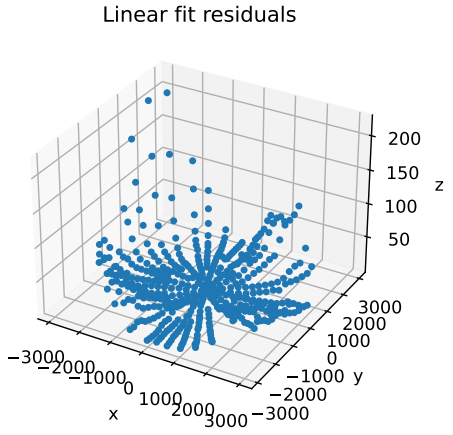

Figure 4: The residuals look like they also pretty much only have radial dependence. And they look like the same shape as the paraboloid.

Notice how it plots out a pretty clear shape. That means there's probably some significance to it. It can mean either a) the fit doesn't work so well as you get farther from the center of the disk (probably) or b) the noise just gets higher as you get farther from the center of the disk. I'm leaning toward A because if it was just caused by noise, the points should still be randomly fluctuating rather than tracing out a smooth surface.

The only nice thing here is that it seems kind of rotationally symmetric which we can use to estimate the error. Since it's rotationally symmetric, every point in a thin ring around the center should be about the same height. Therefore the amount that the points deviate from this average-ring-height should approximate the noise at that radius.

Well I wrote the noise thing and I think it's working. I estimated my error for a to be 5.8826111039569845e-08 in whatever units it is.

We have that

$$\begin{aligned} f &= \frac{1}{4a} \\ \frac{\delta f}{\delta a} &\approx \frac{1}{4a^2} \\ \implies \delta f &= \frac{\delta a}{4a^2} \end{aligned}$$

Plugging in values:

$$f = \frac{1}{4(0.00016670445477401358)} = 1499.66$$

I'm hoping that's in millimeters. And the uncertainty is:

$$\delta f = \frac{5.8826111039569845e-08}{4(0.00016670445477401358)^2} = 0.5292$$

So the final result would be $f = 1.4997 \pm 0.0005$ m. That's within error.

Bonus:
Our new equation for the possibly elliptical dish is:

$$z - z_0 = a(x - x_0)^2 + b(y - y_0)^2$$

Applying the rotation:

$$z - z_0 = a(\cos\theta x' + \sin\theta y' - x_0)^2 + b(-\sin\theta x' + \cos\theta y' - y_0)^2$$

And to simplify the algebra a bit let's work with the primed offsets instead.

$$\begin{aligned}
z - z_0 &= a(\cos\theta(x' - x'_0) + \sin\theta(y' - y'_0))^2 + b(-\sin\theta(x' - x'_0) + \cos\theta(y' - y'_0))^2 \\
z &= a\cos^2\theta(x' - x'_0)^2 + a\sin(2\theta)(x' - x'_0)(y' - y'_0) + a\sin^2\theta(y' - y'_0)^2 \\
&\quad b\sin^2\theta(x' - x'_0)^2 - b\sin(2\theta)(x' - x'_0)(y' - y'_0) + b\cos^2\theta(y' - y'_0)^2 + z_0 \\
&= (a\cos^2\theta + b\sin^2\theta)(x' - x'_0)^2 + (a - b)\sin(2\theta)(x' - x'_0)(y' - y'_0) \\
&\quad + (a\sin^2\theta + b\cos^2\theta)(y' - y'_0)^2 + z_0 \\
&= (a\cos^2\theta + b\sin^2\theta)x'^2 + (a\cos^2\theta + b\sin^2\theta)x'_0(-2x') + (a\cos^2\theta + b\sin^2\theta)x'^2_0 \\
&\quad + (a - b)\sin(2\theta)(x'y') + \tfrac{1}{2}(a - b)\sin(2\theta)x'_0(-2y') + \tfrac{1}{2}(a - b)\sin(2\theta)y'_0(-2x') \\
&\quad + (a - b)\sin(2\theta)x'_0y'_0 + (a\sin^2\theta + b\cos^2\theta)y'^2 + (a\sin^2\theta + b\cos^2\theta)y_0(-2y') \\
&\quad + (a\sin^2\theta + b\cos^2\theta)y_0^2 + z_0
\end{aligned}$$

Well that wasn't fun. They can be grouped together to give our linear parameters. We're going to label them all k_n because why not.

$$\begin{aligned}
z = &\underbrace{(a\cos^2\theta + b\sin^2\theta)}_{k_1}x'^2 \\
&+ \underbrace{((a\cos^2\theta + b\sin^2\theta)x'_0 + \tfrac{1}{2}(a - b)\sin(2\theta)y'_0)}_{k_2}(-2x') \\
&+ \underbrace{(\tfrac{1}{2}(a - b)\sin(2\theta)x'_0 + (a\sin^2\theta + b\cos^2\theta)y'_0)}_{k_3}(-2y') \\
&+ \underbrace{(a\sin^2\theta + b\cos^2\theta)}_{k_4}y'^2 \\
&+ \underbrace{(a\cos^2\theta + b\sin^2\theta)x'^2_0 + (a - b)\sin(2\theta)x'_0y'_0 + (a\sin^2\theta + b\cos^2\theta)y_0^2 + z_0}_{k_5}
\end{aligned}$$

Now we need to invert to find a and b somehow. Let $\gamma = (a - b)\sin(2\theta)$.

$$\begin{aligned}
k_2 &= k_1x'_0 + \tfrac{1}{2}\gamma y'_0 \\
k_3 &= \tfrac{1}{2}\gamma x'_0 + k_4y'_0 \\
k_5 &= k_1x_0^2 + \gamma x'_0y'_0 + k_4y_0^2 + z_0
\end{aligned}$$

This is as far as I got on the bonus before the due date. Hopefully I'll upload the finished version soon.