Problem 1

a)

Consider expanding $f(x \pm \delta)$ and $f(x \pm 2\delta)$ using its Taylor series:

$$f(x \pm \delta) = f(x) \pm f'(x)\delta + \frac{1}{2}f''(x)\delta^2 \pm \frac{1}{6}f'''(x)\delta^3 + \frac{1}{24}f^{(4)}(x)\delta^4 \pm \frac{1}{120}f^{(5)}(x)\delta^5 + O(\delta^6), \tag{1}$$

$$f(x \pm 2\delta) = f(x) \pm 2f'(x)\delta + 2f''(x)\delta^2 \pm \frac{4}{3}f'''(x)\delta^3 + \frac{2}{3}f^{(4)}(x)\delta^4 \pm \frac{4}{15}f^{(5)}(x)\delta^5 + O(\delta^6). \tag{2}$$

We need to combine the four functions above in such a way that their third order terms cancel one another (the logical step after we've seen cancelling first and second order). Using the definition of the double-sided derivative, we have:

$$f'_\delta(x) \approx \frac{f(x+\delta) - f(x-\delta)}{2\delta}, \tag{3}$$

$$f'_{2\delta}(x) \approx \frac{f(x+2\delta) - f(x-2\delta)}{4\delta}. \tag{4}$$

Expanding,

$$\begin{aligned}
2\delta f'_\delta(x) &\approx f(x+\delta) - f(x-\delta) \\
&\approx f + f'\delta + \frac{1}{2}f''\delta^2 + \frac{1}{6}f'''\delta^3 + \frac{1}{24}f^{(4)}\delta^4 + \frac{1}{120}f^{(5)}\delta^5 \\
&\quad - (f - f'\delta + \frac{1}{2}f''\delta^2 - \frac{1}{6}f'''\delta^3 + \frac{1}{24}f^{(4)}\delta^4 - \frac{1}{120}f^{(5)}\delta^5) \\
&\approx 2f'\delta + \frac{1}{3}f'''\delta^3 + \frac{1}{60}f^{(5)}\delta^5
\end{aligned}$$

$$\begin{aligned}
4\delta f'_{2\delta}(x) &\approx f(x+2\delta) - f(x-2\delta) \\
&\approx f + 2f'\delta + 2f''\delta^2 + \frac{4}{3}f'''\delta^3 + \frac{2}{3}f^{(4)}\delta^4 + \frac{4}{15}f^{(5)}\delta^5 \\
&\quad - (f - 2f'\delta + 2f''\delta^2 - \frac{4}{3}f'''\delta^3 + \frac{2}{3}f^{(4)}\delta^4 - \frac{4}{15}f^{(5)}\delta^5) \\
&\approx 4f'\delta + \frac{8}{3}f'''\delta^3 + \frac{8}{15}f^{(5)}\delta^5
\end{aligned}$$

From here, we must find constants a, b, $c(a,b)$ and $d(a,b)$ such that the third order term cancels out, i.e.:

$$2a\delta f'_\delta + 4b\delta f'_{2\delta} = cf'\delta + df^{(5)}\delta^5 \tag{5}$$

Through observation, we see that this is satisfied for $a = 8, b = -1$. Plugging in, we get

$$8(2\delta f'_{\delta}) - 4\delta f'_{2\delta} = 12f'\delta - \frac{2}{5}f^{(5)}\delta^5. \tag{6}$$

The third order term has been eliminated as desired. Re-arranging and substituting equations (3) and (4) into equation (6), we can now write the derivative of f as:

$$f'(x) \approx \frac{1}{12\delta}[8f(x+\delta) - 8f(x-\delta) - f(x+2\delta) + f(x-2\delta)] + \frac{1}{30}f^{(5)}(x)\delta^4 + O(\delta^5) \tag{7}$$

b)

Notice that the leading order truncation error in equation (7) is now $e_t \sim \delta^4 f^{(5)}$. Assuming round-off error of $e_r \sim \epsilon_f |f(x)/\delta|$, the variance can be expressed as:

$$\text{Var}[e] = e_t^2 + e_r^2 = (\delta^4 f^{(5)})^2 + (\epsilon_f \frac{f}{\delta})^2 = \delta^8 f^{(5)2} + \epsilon_f^2 \frac{f^2}{\delta^2}. \tag{8}$$

Minimizing the variance,

$$\begin{aligned} \frac{\partial \text{Var}}{\partial \delta} &= 8f^{(5)2}\delta^7 - 2\epsilon_f^2 f^2 \delta^{-3} = 0 \\ \delta^{10} &= \frac{2\epsilon_f^2 f^2}{8f^{(5)2}} \end{aligned}$$

Dropping the constants, we find that:

$$\delta \sim \left(\frac{\epsilon_m f}{f^{(5)}}\right)^{1/5} \tag{9}$$

where $\epsilon_f \approx \epsilon_m$. With the choice of equation (9), the fractional accuracy of the computed derivative is

$$\begin{aligned} (e_r + e_t)/|f'| &= \frac{1}{|f'|}(\epsilon_m |f(x)/\delta| + \delta^4 f^{(5)}) \\ &= \frac{1}{f'}\left(\frac{\epsilon_m f f^{(5)1/5}}{\epsilon_m^{1/5} f^{1/5}} + \frac{\epsilon_m^{4/5} f^{4/5} f^{(5)}}{f^{(5)4/5}}\right) \\ &\sim \epsilon_m^{4/5} f^{4/5} f^{(5)1/5}/f' \end{aligned}$$

When applying these results to `exp(x)`, I noticed that the predicted fractional error was consistently underestimating the error compared to the true fractional error by a factor of approximately 10. I noticed this in problem 2 as well.

Problem 2

Consider equations (1) and (2). Here, we derive an analytical expression to get an estimate for $f'''(x)$ to use for determining the optimal h for the double-sided derivative.

Utilizing our results from Problem 1, we can eliminate f' in the expansion of the double-sided derivatives by setting $a = 2$ and $b = -1$, i.e.:

$$2(2\delta f'_\delta) - 4\delta f'_{2\delta} = -2f'''(x)\delta^3 + O(\delta^5) \tag{10}$$

This is implies that

$$f'''(x) \approx \frac{2(2\delta f'_\delta) - 4\delta f'_{2\delta}}{-2\delta^3} \tag{11}$$

is a fifth-order centered difference approximation of the third derivative. Substituting the appropriate expressions yields:

$$\begin{aligned} f'''(x) &\approx \frac{2(f(x+\delta) - f(x-\delta)) - (f(x+2\delta) - f(x-2\delta))}{-2\delta^3} \qquad (12)\\ &\approx \frac{f(x+2\delta) - 2f(x+\delta) + 2f(x-\delta) - f(x-2\delta)}{2\delta^3} \qquad (13)\end{aligned}$$

where for the sake of this analysis, we will set $\delta = \epsilon_m^{1/3}$ to estimate the optimal δ.

Problem 3

Firstly, I wanted to have an idea of what the data looked like, so I plotted the Temperature as a function of Voltage (see Figure 1). In the end, I decided to perform a cubic polynomial

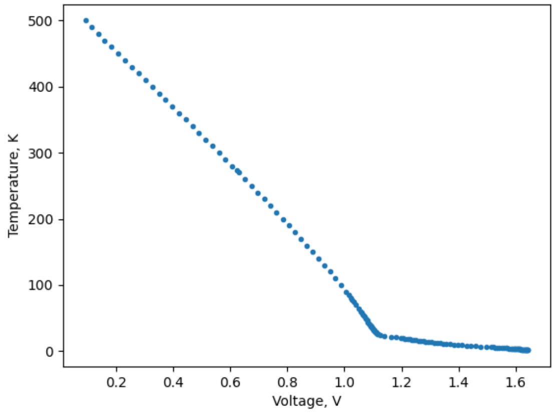

Figure 1: Raw data provided by `lakeshore.txt`.

interpolation primarily because I didn't want to have to deal with having continous derivatives around 1.1V required for splines. In order to carry this out, I utilized a python Class from `sklearn`, namely `NearestNeighbor`, which given an array of numerical values, determines the index of the n nearest neighbors of a specified value (see https://scikit-learn.org/stable/modules/neighbors.html).

3.1 Estimating Error in Interpolated Values

Suppose we are given a function $f(x)$ on $x \in [a, b]$ and a set of distinct points $x_i \in [a, b]$, $i = 0, 1, ..., n$. Let $P_n(x) \in P_n$ s.t.,

$$P_n(x_i) = f(x_i), \quad i = 0, 1, ..., n \tag{14}$$

where we define the error function as:

$$e(x) = f(x) - P_n(x), \quad x \in [a, b]. \tag{15}$$

Theorem: There exists some value $\zeta \in [a, b]$, such that

$$e(x) = \frac{1}{(n+1)!} f^{(n+1)}(\zeta) \prod_{i=0}^{n} (x - x_i), \quad \text{for all } x \in [a, b] \tag{16}$$

Let's see how we can apply this to get an upper bound on the error of our interpolated value.

Example: Let n = 3 (i.e. cubic polynomial), $x_0 = a$, $x_1 = b$, $x_2 = c$ and $x_3 = d$. We would like to find an upper bound on the error. Let

$$M = \max_{a \leq x \leq d} |f^{(4)}(x)| \tag{17}$$

denote the maximum value of the fourth derivative of f evaluated between $a \leq x \leq d$. The maximum error is given by:

$$|e(x)| = \frac{1}{4!} |f^{(4)}(\zeta)| \cdot |(x - a)(x - b)(x - c)(x - d)|. \tag{18}$$

How is this useful? In the context of this problem, suppose we have a voltage V which we'd like to know the temperature of. We choose the 4 nearest neighbours of V

$$V_i, \quad i = 0, 1, 2, 3$$

and obtain a temperature $\mathtt{T}_{\mathtt{interp}}$ by performing a cubic interpolation on the corresponding temperatures $\mathtt{T}_i$, $i = 0, 1, 2, 3$. To get an estimate of the error, we must determine

$$\max_{V_0 \leq V \leq V_3} |(V - V_0)(V - V_1)(V - V_2)(V - V_3)|.$$

To do this, let

$$g(V) \equiv (V - V_0)(V - V_1)(V - V_2)(V - V_3).$$

The max value can be obtained by taking `max(np.abs(g(V)))` for `V = np.linspace(`V_0`, `V_3`, 1001)`. This taken care of, all that is left is getting an estimate of $|f^{(4)}(\zeta)|$. Following a similar approach as in question 1 and 2, we can obtain a central difference approximation of the fourth derivative. Quoting the result from here, we find that:

$$f^{(4)}(x) \approx \frac{f(x+2h) - 4f(x+h) + 6f(x) - 4f(x-h) + f(x-2h)}{h^4}. \tag{19}$$

To get a rough estimate, the equation reduces to

$$f^{(4)}(V) \approx \frac{\mathtt{T_0} - 4\mathtt{T_1} + 6\mathtt{T_{interp}} - 4\mathtt{T_2} + \mathtt{T_3}}{(\mathtt{np.min}(\mathtt{V} - \mathtt{V_i}))^4} \tag{20}$$

Since we want the largest value of $f^{(4)}$, we take the minimum difference on the denominator in order to assure that we get the largest estimate on our fourth derivative.

Combining everything, our rough estimate on our interpolated value is:

$$|e(V)| \approx \frac{1}{24} \times \mathtt{max(np.abs(g(V))} \times \left| \frac{\mathtt{T_0} - 4\mathtt{T_1} + 6\mathtt{T_{interp}} - 4\mathtt{T_2} + \mathtt{T_3}}{(\mathtt{np.min}(\mathtt{V} - \mathtt{V_i}))^4} \right| \tag{21}$$

for `V = np.linspace(`V_0`, `V_3`, 1001)` and $i = 0, 1, 2, 3$.

Problem 4

a)

For $f(x) = \cos(x)$ and allowing ourselves to interpolate through 10 points, we find that the rational fit performs best, followed by the cubic spline, with the cubic polynomial performing the worst.

b) `np.linalg.inv`

For $f(x) = \frac{1}{1+x^2}$ and allowing ourselves to interpolate through 10 points again, we find that the spline performs best, followed by the cubic polynomial, and the rational performing exceptionally poorly. This is a bit worrisome: we would expect the error to be extremely small given that the Lorentzian is by definition a rational function ($p(x) = 1$ and $q(x) = 1 + x^2$).

c) `np.linalg.pinv`

Repeating the above process but changing the way we invert our matrix of coefficients, we find that the rational function does exceptionally well, going from an error of 15 to an approximate error of $\sim 1e^{-16}$! This is much more inline with what we would expect. What explains this massive increase in precision?

d)

As stated in the problem, `np.linalg.pinv` sets sufficiently small eigenvalues in A to be 0 in A^{-1}. To visualize the implications of this, recall that a matrix A has eigenvalue λ if and only if the inverse matrix A^{-1} has eigenvalue λ^{-1}. So by setting small eigenvalues in A to 0 in A^{-1}, we don't allow them to blow up and dominate the inverse matrix.

By comparing the coefficients, we see that higher order coefficients have been reduced considerably. It seems that in some way the new coefficients have been normalized. I don't really understand what's going on to be quite honest.