

Math 5601 Homework 4

Jacob Hauck

September 24, 2023

Question 1.

- (a) We want to derive a finite difference formula for $f'(t_j)$ involving the sample values $f(t_j - 2h)$, $f(t_j - h)$, $f(t_j + h)$, and $f(t_j + 2h)$. Expanding f at these sample values with Taylor's Theorem (centered at t_j) gives

$$f(t_j - 2h) = f(t_j) - 2hf'(t_j) + \frac{4h^2}{2}f''(t_j) - \frac{8h^3}{6}f'''(t_j) + \frac{16h^4}{24}f''''(t_j) + O(h^5), \quad (1)$$

$$f(t_j - h) = f(t_j) - hf'(t_j) + \frac{h^2}{2}f''(t_j) - \frac{h^3}{6}f'''(t_j) + \frac{h^4}{24}f''''(t_j) + O(h^5), \quad (2)$$

$$f(t_j + h) = f(t_j) + hf'(t_j) + \frac{h^2}{2}f''(t_j) + \frac{h^3}{6}f'''(t_j) + \frac{h^4}{24}f''''(t_j) + O(h^5), \quad (3)$$

$$f(t_j + 2h) = f(t_j) + 2hf'(t_j) + \frac{4h^2}{2}f''(t_j) + \frac{8h^3}{6}f'''(t_j) + \frac{16h^4}{24}f''''(t_j) + O(h^5). \quad (4)$$

Therefore,

$$\begin{aligned} f(t_j - 2h) - 8f(t_j - h) + 8f(t_j + h) - f(t_j + 2h) &= (1 - 8 + 8 - 1)f(t_j) \\ &\quad + (-2 + 8 + 8 - 2)hf'(t_j) \\ &\quad + (4 - 8 + 8 - 4)\frac{h^2}{2}f''(t_j) \\ &\quad + (-8 - 8 + 8 + 8)\frac{h^3}{6}f'''(t_j) \\ &\quad + (16 - 8 + 8 - 16)\frac{h^4}{24}f''''(t_j) \\ &\quad + O(h^5) \\ &= 12hf'(t_j) + O(h^5). \end{aligned} \quad (5)$$

Dividing both sides by $12h$ yields the finite difference formula

$$f'(t_j) = \frac{1}{12h} [f(t_j - 2h) - 8f(t_j - h) + 8f(t_j + h) - f(t_j + 2h)] + O(h^4). \quad (6)$$

- (b) Let \tilde{f}_j be the number actually stored in computer memory as an approximation of $f_j = f(t_j)$, and suppose that $f_j = \tilde{f}_j + e(t_j)$, where, for some $\varepsilon > 0$, the round-off error e satisfies the estimate $|e(t)| \leq \varepsilon|f(t)|$ for all t .

Then the error between $f'(t_j)$ the approximation computed by the computer using the finite difference

formula from (a) is given by

$$\begin{aligned}
 E(t_j) &= \left| f'(t_j) - \frac{\tilde{f}_{j-2} - 8\tilde{f}_{j-1} + 8\tilde{f}_{j+1} - \tilde{f}_{j+2}}{12h} \right| \\
 &\leq \left| f'(t_j) - \frac{f_{j-2} - 8f_{j-1} + 8f_{j+1} - f_{j+2}}{12h} \right| \\
 &\quad + \frac{1}{12h} \left| \tilde{f}_{j-2} - f_{j-2} - 8(\tilde{f}_{j-1} - f_{j-1}) + 8(\tilde{f}_{j+1} - f_{j+1}) - (\tilde{f}_{j+2} - f_{j+2}) \right| \\
 &\leq O(h^4) + \frac{\varepsilon}{12h} (f_{j-2} + 8f_{j-1} + 8f_{j+1} + f_{j+2})
 \end{aligned} \tag{7}$$

- (c) The following Python script computes the numerical derivative $\ln'(3) = \frac{1}{3}$ using the formula from (a) for various values of h .

```

1 from math import log
2 fprime = 1/3
3 for n in range(1, 21):
4     h = 10 ** (-n)
5     fprime_approx = (log(3+2*h) - 8 * log(3-h) + 8*log(3+h) - log(3+2*h))/(12*h)
6     print(f'h = 10^{-n}, error = {abs(fprime - fprime_approx)}')
```

Here is what was printed when I ran this script.

```

h = 10^-1, error = 3.3052928644083934e-07
h = 10^-2, error = 3.293260109060725e-11
h = 10^-3, error = 1.8207657603852567e-14
h = 10^-4, error = 1.0734746425100639e-12
h = 10^-5, error = 4.034106382277969e-12
h = 10^-6, error = 2.8088975589923848e-11
h = 10^-7, error = 2.0258236199666158e-09
h = 10^-8, error = 9.427310487808427e-09
h = 10^-9, error = 4.60838407434494e-08
h = 10^-10, error = 2.1261729443722288e-07
h = 10^-11, error = 1.8779518313749577e-06
h = 10^-12, error = 8.51446786782617e-05
h = 10^-13, error = 8.138877501551178e-05
h = 10^-14, error = 0.012686176008173844
h = 10^-15, error = 0.0002664259457863527
h = 10^-16, error = 0.14829616256247388
h = 10^-17, error = 1.517038374375261
h = 10^-18, error = 18.17038374375261
h = 10^-19, error = 184.7038374375261
h = 10^-20, error = 1850.0383743752611
```

Evidently, the error decreases at first as h gets smaller, achieving a minimum for $h = 10^{-3}$, then it increases again as h gets even smaller. This is consistent with the estimate from part (b). This estimate involves two terms; the first decreases rapidly as $h \rightarrow 0$, and the second blows up as $h \rightarrow 0$. If the relative error ε is small enough compared to h , then the total error $E(t_j)$ should be small at first because the $O(h^4)$ term goes to zero so quickly, but as h gets even smaller, the total error could (and in this case does) increase in inverse proportion to h . In fact, each time h decreases by a factor of 10 below 10^{-5} , the total error roughly increases by a factor of 10, consistent with the $\frac{1}{h}$ relationship in the estimate.

Furthermore, we can see that the point where the error changes from decreasing to increasing is different if we use a different level of precision. Here we do the same experiment using 30 decimal

digits of relative precision ($\varepsilon \approx 10^{-30}$) instead of Python's default 64-bit floating point representation ($\varepsilon \approx 10^{-15}$).

```

1 from mpmath import mp, mpf, log
2 mp.dps = 30
3
4 fprime = mpf(1) / 3
5 for n in range(1, 21):
6     h = mpf(10) ** (-n)
7     fprime_approx = (log(3-2*h) - 8*log(3-h) + 8*log(3+h) - log(3+2*h)) / (12*h)
8     print(f'h = 10^{ -n}, error = {abs(fprime - fprime_approx)}')
```

This results in the following output.

```

h = 10^-1, error = 0.000000330529287394316004402906803334
h = 10^-2, error = 3.29231171680679755975451845909e-11
h = 10^-3, error = 3.2921823763803376195465228809e-15
h = 10^-4, error = 3.29218108325886682162187340946e-19
h = 10^-5, error = 3.29215652897465049577434955075e-23
h = 10^-6, error = 8.68735537064968066371557994169e-26
h = 10^-7, error = 1.12928697564002534249710051324e-24
h = 10^-8, error = 4.12978564366037573305902685826e-24
h = 10^-9, error = 1.02737398845590657985838335357e-22
h = 10^-10, error = 3.73866064741871909975852085448e-22
h = 10^-11, error = 1.23711256649616585150871196032e-20
h = 10^-12, error = 3.2002300253720255539322612194e-20
h = 10^-13, error = 8.86601614243133276799836453504e-19
h = 10^-14, error = 9.46719776678264666922910096583e-18
h = 10^-15, error = 2.59017999588870428473712539715e-17
h = 10^-16, error = 3.35659448267410067502208722659e-16
h = 10^-17, error = 1.36337524150213406861996890051e-14
h = 10^-18, error = 9.31911618336573413130731564024e-14
h = 10^-19, error = 1.36893268855804435281782277331e-13
h = 10^-20, error = 1.39419591102235110300363359764e-11
```

As we can see, the value of h where the error is minimal is now $h = 10^{-6}$. Furthermore, for the smallest value of $h = 10^{-20}$ (at which point the $O(h^4)$ error term should be completely irrelevant), the error has decreased by a factor of $\sim 10^{-15}$, that is, in proportion to ε , as the estimate for $E(t_j)$ suggests.

Question 2.

Suppose that

$$f'(t_j) = \frac{1}{h} [\alpha_1 f(t_j) + \alpha_2 f(t_j + h) + \alpha_3 f(t_j + 2h) + \alpha_4 f(t_j + 3h) + \alpha_5 f(t_j + 4h)] + O(h^4) \quad (8)$$

for some constants $\{\alpha_i\}$. Expanding f about t_j using Taylor's Theorem, we have, for $i \in \{0, 1, 2, 3, 4\}$,

$$f(t_j + ih) = f(t_j) + \sum_{k=1}^4 \frac{(ih)^k}{k!} f^{(k)}(t_j) + O(h^5). \quad (9)$$

Therefore, the approximation in (8) can be estimated by

$$\frac{1}{h} \sum_{i=0}^4 \alpha_{i+1} f(t_j + ih) = \sum_{i=0}^4 \alpha_{i+1} \left[f(t_j) + \sum_{k=1}^4 \frac{i^k}{k!} f^{(k)}(t_j) h^{k-1} \right] + O(h^4) \quad (10)$$

$$= \left(\sum_{i=0}^4 i \alpha_{i+1} \right) f'(t_j) + \sum_{i=0}^4 \alpha_{i+1} \left[f(t_j) h^{-1} + \sum_{k=2}^4 \frac{i^k}{k!} f^{(k)}(t_j) h^{k-1} \right] + O(h^4) \quad (11)$$

$$= \left(\sum_{i=1}^5 (i-1) \alpha_i \right) f'(t_j) + \left(\sum_{i=1}^5 \alpha_i \right) f(t_j) h^{-1} \quad (12)$$

$$+ \sum_{k=2}^4 f^{(k)}(t_j) \frac{h^{k-1}}{k!} \left(\sum_{i=1}^5 \alpha_i (i-1)^k \right) + O(h^4).$$

So, to obtain (8), we would need the coefficient of $f'(t_j)$ in (12) to be 1, and we would need the rest of the coefficients of h^k to be zero when $k < 4$, since otherwise we would not have an error of $O(h^4)$. That is, $\{\alpha_i\}$ must satisfy the system of equations

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 \\ 0 & 1 & 4 & 9 & 16 \\ 0 & 1 & 8 & 27 & 64 \\ 0 & 1 & 16 & 81 & 256 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (13)$$

Thus, $\alpha_1 = -\sum_{i=2}^5 \alpha_i$, and the remaining $\{\alpha_i\}_{i=2}^5$ must satisfy

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 4 & 9 & 16 \\ 1 & 8 & 27 & 64 \\ 1 & 16 & 81 & 256 \end{bmatrix} \begin{bmatrix} \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (14)$$

Then $\alpha_2 = 1 - \sum_{i=3}^5 (i-1)\alpha_i$, and the last three rows are equivalent to

$$\begin{bmatrix} 4 & 9 & 16 \\ 8 & 27 & 64 \\ 16 & 81 & 256 \end{bmatrix} \begin{bmatrix} \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix} = -\alpha_2 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \quad (15)$$

Let A be the matrix on LHS(15). Then

$$\det(A) = 4(27 \cdot 256 - 64 \cdot 81) - 9(8 \cdot 256 - 64 \cdot 16) + 16 \cdot (8 \cdot 81 - 27 \cdot 16) = 1152.$$

Using Cramer's Rule, the solution of $Ax = [1 \ 0 \ 0]^T$ is

$$\frac{1}{\det(A)} \left[\begin{vmatrix} 27 & 64 \\ 81 & 256 \end{vmatrix} - \begin{vmatrix} 8 & 64 \\ 16 & 256 \end{vmatrix} \begin{vmatrix} 8 & 27 \\ 16 & 81 \end{vmatrix} \right]^T = \frac{1}{1152} \begin{bmatrix} 1728 \\ -1024 \\ 216 \end{bmatrix}, \quad (16)$$

the solution of $Ax = [0 \ 1 \ 0]^T$ is

$$\frac{1}{\det(A)} \left[-\begin{vmatrix} 9 & 16 \\ 81 & 256 \end{vmatrix} \begin{vmatrix} 4 & 16 \\ 16 & 256 \end{vmatrix} - \begin{vmatrix} 4 & 9 \\ 16 & 81 \end{vmatrix} \right]^T = \frac{1}{1152} \begin{bmatrix} -1008 \\ 768 \\ -180 \end{bmatrix}, \quad (17)$$

and the solution of $Ax = [0 \ 0 \ 1]^T$ is

$$\frac{1}{\det(A)} \left[\begin{vmatrix} 9 & 16 \\ 27 & 64 \end{vmatrix} \quad - \begin{vmatrix} 4 & 16 \\ 8 & 64 \end{vmatrix} \quad \begin{vmatrix} 4 & 9 \\ 8 & 27 \end{vmatrix} \right]^T = \frac{1}{1152} \begin{bmatrix} 144 \\ -128 \\ 36 \end{bmatrix}. \quad (18)$$

Therefore,

$$A^{-1} = \frac{1}{1152} \begin{bmatrix} 1728 & -1008 & 144 \\ -1024 & 768 & -128 \\ 216 & -180 & 36 \end{bmatrix}. \quad (19)$$

This implies that

$$\begin{bmatrix} \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix} = -\alpha_2 A^{-1} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = -\frac{\alpha_2}{1152} \begin{bmatrix} 1728 - 1008 + 144 \\ -1024 + 768 - 128 \\ 216 - 180 + 36 \end{bmatrix} = -\frac{\alpha_2}{1152} \begin{bmatrix} 864 \\ -384 \\ 72 \end{bmatrix} = \frac{\alpha_2}{48} \begin{bmatrix} -36 \\ 16 \\ -3 \end{bmatrix}, \quad (20)$$

or $\alpha_3 = -\frac{3}{4}\alpha_2$, $\alpha_4 = \frac{1}{3}\alpha_2$, and $\alpha_5 = -\frac{1}{16}\alpha_2$. Next,

$$\alpha_2 = 1 - 2\alpha_3 - 3\alpha_4 - 4\alpha_5 = 1 + \frac{3}{2}\alpha_2 - \alpha_2 + \frac{1}{4}\alpha_2 \implies \alpha_2 = 4. \quad (21)$$

Thus, $\alpha_3 = -3$, $\alpha_4 = \frac{4}{3}$, and $\alpha_5 = -\frac{1}{4}$. Lastly, we get

$$\alpha_1 = -\alpha_2 - \alpha_3 - \alpha_4 - \alpha_5 = -4 + 3 - \frac{4}{3} + \frac{1}{4} = -\frac{25}{12}. \quad (22)$$

Therefore, the desired finite difference formula is

$$f'(t_j) = \frac{-25f(t_j) + 48f(t_j + h) - 36f(t_j + 2h) + 16f(t_j + 3h) - 3f(t_j + 4h)}{12h} + O(h^4). \quad (23)$$

Question 3.

Let $f \in C^\infty(-\infty, \infty)$, and let $x \in \mathbf{R}$.

(a) Define

$$S_h = \frac{f(x+h) - f(x-h)}{2h}. \quad (24)$$

Then

$$S_h = f'(x) + \sum_{i=1}^{\infty} c_i h^{2i} \quad (25)$$

for constants c_i independent of h .

Proof. By Taylor's Theorem (assuming f is analytic in a neighborhood of x and h is sufficiently small),

$$f(x+sh) = \sum_{i=0}^{\infty} \frac{f^{(i)}(x)}{i!} s^i h^i, \quad (26)$$

where $s \in \{-1, 1\}$. Then

$$S_h = \frac{1}{2h} \left[\sum_{i=0}^{\infty} \frac{f^{(i)}(x)}{i!} h^i - \sum_{i=0}^{\infty} \frac{f^{(i)}(x)}{i!} (-1)^i h^i \right] = \frac{1}{2h} \sum_{i=0}^{\infty} \frac{f^{(i)}(x)}{i!} (1 - (-1)^i) h^i. \quad (27)$$

Since $1 - (-1)^i = 0$ if i is even and 2 if i is odd, it follows that

$$S_h = \frac{1}{h} \sum_{i=0}^{\infty} \frac{f^{(2i+1)}(x)}{(2i+1)!} h^{2i+1} = f'(x) + \sum_{i=1}^{\infty} \frac{f^{(2i+1)}(x)}{(2i+1)!} h^{2i}. \quad (28)$$

The claim follows if we choose $c_i = \frac{f^{(2i+1)}(x)}{(2i+1)!}$. \square

(b) Suppose that

$$\alpha_1 S_h + \alpha_2 S_{\frac{h}{2}} = f'(x) + O(h^4). \quad (29)$$

Using part (a), we see that

$$\alpha_1 S_h + \alpha_2 S_{\frac{h}{2}} = (\alpha_1 + \alpha_2) f'(x) + \sum_{i=1}^{\infty} c_i \left(\alpha_1 + \frac{\alpha_2}{2^{2i}} \right) h^{2i}. \quad (30)$$

Since $\sum c_i h^{2i}$ converges, the remainder of the infinite series starting from $i = 2$ is $O(h^4)$. The first term of the infinite series, however, is at best $O(h^2)$, so, in order to satisfy the assumption, we must choose α_1 and α_2 so that the coefficient of the h^2 term in the series is zero. We must also ensure that the coefficient of $f'(x)$ is 1. That is, α_1 and α_2 must solve the equations

$$\alpha_1 + \alpha_2 = 1, \quad (31)$$

$$\alpha_1 + \frac{1}{4} \alpha_2 = 0. \quad (32)$$

Then $\alpha_2 = -4\alpha_1$, so $-3\alpha_1 = 1$, and $\alpha_1 = -\frac{1}{3}$, and $\alpha_2 = \frac{4}{3}$. Then

$$\alpha_1 S_h + \alpha_2 S_{\frac{h}{2}} = \frac{4S_{\frac{h}{2}} - S_h}{3} = f'(x) + O(h^4). \quad (33)$$