Assignment 1

1. **Exact on monomials.** Let $D(h)f$ be a finite difference approximation to $f'(x)$ of the form

$$D(h)f(x) = \frac{1}{h}\left(a_{-k}f(x-kh) + a_{-k+1}f(x-(k-1)h) + \cdots + a_k f(x+kh) \right) .$$

The *stencil* is the set of point values used to estimate f'). It contains $2k+1$ points including the target point x and k points in either side. Typically, k is one or two or maybe three. Any of the coefficients may be zero. For example, the first order one sided approximation

$$f'(x) \approx \frac{1}{h}\left[f(x+h) - f(x) \right]$$

has $k = 1$, $a_{-1} = 0$, $a_0 = -1$, and $a_1 = 1$. The second order centered scheme

$$f'(x) \approx \frac{1}{2h}\left[f(x+h) - f(x-h) \right]$$

has $a_{-1} = -\frac{1}{2}$, $a_0 = 0$, and $a_1 = \frac{1}{2}$. The finite difference approximation is defined by the stencil size parameter k and the coefficients $a_{-k}, \cdots, a_k$.

Show that if such a finite difference approximation is exact on monomials 1, x, $\cdots$, x^p, then its order of accuracy is at least p. *Hints*:

- Without loss of generality, you may take $x = 0$.
- A general function f (if it has enough derivatives) has an asymptotic expansion
$$f(x) \sim f_0 + f_1 x + f_2 x^2 + \cdots .$$
The formulas for the coefficients on the right are not important (so it isn't given) except that $f_1 = f'(0)$.
- This means, using the definition of asymptotic approximation, that
$$f(x) = f_0 + f_1 x + f_2 x^2 + \cdots f_p x^p + O(x^{p+1}) .$$
- If D is exact up to x^p, then[1]
$$D(h)\left[f_0 + f_1 x + f_2 x^2 + \cdots f_p x^p \right] = f_1 = f'(0) .$$
- Thus, the error is determined by what D does to $O(x^{p+1})$.

[1] The operator D is *linear*, which means that $D(h)(f+g) = DF + Dg$, etc.

- You can see what's happening with powers of h by doing the one sided first order scheme. For this, you would write $f(x) = f_0 + f_1 x + O(x^2)$. Dividing by h in D turns x^2 into $h^1 = h$, which indicates first order.

2. **Three point one sided approximation.** The approximation is

$$f'(x) \approx \frac{1}{h}\left[\, af(x+2h) + bf(x+h) + cf(x) \,\right] \ .$$

Write linear equations that express the goal that the approximation is exact for 1, x, and x^2. Solve this system of three linear equations for the three unknowns a, b, and c.

3. **Looking for power laws in data.**

*Analysis*A *power law* function is a function of the form $f(x) = Cx^p$. The exponent p does not have to be an integer. Show that f is a power law if and only its on a log/log plot is a straight line. If f is a power law, show that p is the slope of that line. In particular, show that the slope does not depend on the *prefactor* C. We say f is an asymptotic power law as $x \to 0$ if

$$f(x) = Cx^p + O(x^{p+1}) \quad \text{as } x \to 0 \ .$$

Show that if f is an asymptotic power law, then the log/log graph of f has a straight line as an asymptote as $x \to 0$ (so $\log(x) \to -\infty$).

Coding: Do experiments to verify this for the function $f(x) = 3x^2 - x^3$. Details are important. You need to take x values uniformly spaced in "log space" (i.e., $\log(x)$ values are uniformly spaced) ranging from some rather small positive x (think 10^{-20} or smaller, less than ϵ_{mach}) and going up to x values where the asymptotic approximation $f \sim 3x^2$ is not valid. Draw a line on the plot with the expected asymptotic slope 2, so you can see whether the actual graph tends to this slope.

4. **Accuracy of difference approximations**

A finite difference approximation to f' is first order (or order p) if its error is asymptotic to the power law as $h \to 0$

$$D(h)f - f' \approx Ch \quad (\text{or } Ch^p \ )$$

Adapt your code from Exercise 3 to make the log/log of error as a function of h as h becomes small and look for the asymptotic power law behavior the theory predicts: first order for the simple one sided approximations, second order for the centered approximation and the three point one sided approximation (that's three graphs, but simple adaptations of the code for each, if the code is modular). It will be important to choose a range of h from not so small down to the order of ϵ_{math}. Use the function $f(x) = e^x$ (then experiment with others if you want). You should see that the error decreases like a power law until h gets too small and the error starts growing. Please comment on:

(a) Why does the error start growing?

(b) Make a model of the computation (for the simplest method) that takes into account errors in floating point arithmetic:

$$\frac{f(x+h)(1+r_1) - f(x)(1+r_2)}{h} \ .$$

Here r_1 and r_2 are different numbers on the order of ϵ_{mach}. Does this model predict the behavior you see in the plots?

(c) Use part (b) to derive a rough (approximate) bound for the error of the form (for the first order method)

$$\text{error} <\approx Ch + \frac{\epsilon_{\text{mach}}}{h} \ .$$

There is a formula for C involving f'', but this is not so important.

(d) Minimize the right side of the error bound from part (c) over h to find the h_* that gives the best possible accuracy in floating point arithmetic. You should find that h_* is a power of ϵ_{mach} so the best achievable accuracy is a different power of ϵ_{mach}. Show that the best possible accuracy for second order methods is better than for first order methods.