Assignment 1, due February 13

Corrections: Formula (4) has 2^p replaced by the correct 2^{-p}. The previous version of (7) was massively garbled. There is much more explanation of continuous and discrete symbols in part (2). There is new notation to clarify the distinction between differential operators and finite difference operators on the space of continuous functions.

Download the files related to Python and work through them. You should be able to modify the posted codes to do the computational parts of the assignments below. I encourage you to do this. You may also copy code used as examples (particularly in graphics) in the web. You may not copy code other students have written. You may not copy code from the web that answers questions here. For example, there probably is Python code on the web that computes symbols of difference operators. You cannot use this. You may have to use some judgement when making these distinctions.

1. (*High order difference operators*) This set of exercises walks you through some basics about high order finite difference approximations. Many of you will have seen much of this in earlier classes. Suppose $u(x)$ is a smooth function of the single variable x. A *finite difference* approximation to the (second) derivative takes the form

$$\partial_x^2 u(x) = \frac{1}{h^2}\sum_{j=-r}^{j=s} a_j u(x-jh) + O(h^p) \ . \tag{1}$$

The *stencil* is the set of evaluation points $\{x-rh,\ldots,x+sh\}$. A *centered* approximation is one that has $r=s$ and $a_{-j}=a_j$. The *order of accuracy* is p (more precisely, the largest p that satisfies (1) as $h\to 0$). The generic finite difference operator is

$$D_h u(x) = \frac{1}{h^2}\sum_{j=-r}^{j=s} a_j u(x-jh) \ .$$

There are specific notations for certain common difference operators.

(a) Do the Taylor series calculation to find the centered difference operator with a three point stencil that is second order accurate (has $p=2$).

(b) A difference approximation has an *asymptotic error expansion* if there are coefficients b_k and powers p_k for $k=1,2,\ldots$ so that

$$D_h u(x) - \partial_x^2 u(x) = b_1 h^{p_1} + b_2 h^{p_2} + \cdots + b_m h^{p_m} + O(h^{p_{m+1}}) \ . \tag{2}$$

The powers satisfy $p_1 = p$, and $p_{k+1} > p_k$. The coefficients b_k depend on derivatives of u at x. Use the Taylor series of u to show that the centered three point approximation from part (1a) has an asymptotic error expansion with powers $p_1 = 2$, $p_2 = 4$, etc.

(c) (*Richardson extrapolation*) Show that if D_h has order p and an asymptotic expansion, then you can choose c_1 and c_2 so that the Richardson extrapolated operator

$$\widetilde{D}_h = c_1 D_h + c_2 D_{2h} \tag{3}$$

has order p_2. The coefficients c_1 and c_2 depends only on the power p_1. Show that $\widetilde{D}_h$ also has an asymptotic error expansion.

(d) Use the formula (3) with your values of c_1 and c_2 to derive a fourth order ($p = 4$) centered difference approximation with a five point stencil from the second order three point centered approximation.

(e) (*You don't have to do this, but you should understand that it takes longer to derive the formula directly than to derive it using Richardson extrapolation.*) Verify the fourth order accuracy directly using Taylor series.

(f) Show that if D_h has an asymptotic error expansion and order of accuracy p, then

$$\lim_{h \to 0} \frac{D_h u(x) - \partial_x^2 u(x)}{D_{2h} u(x) - \partial_x^2 u(x)} = 2^{-p} \; . \tag{4}$$

(g) (*Convergence study*) Write a Python script to verify the behavior (4) for the second and fourth order operators above. Use the function $u(x) = e^x$ and the point $x = 0$. You will need to use `numpy` to evaluate the exponential, but there is nothing to plot. Your program should print a sequence of ratios (4) for the second and fourth order operators. The output should be clear (give h and say which operator, etc.). Hand in a printout of the program and its output. The program should have a clear programming style and follow other standards of good programming.

2. (*Symbols of difference operators*) We use the standard notation for periodic grid functions. There are $|B| = N$ grid points of the form $x_i = ih$, with grid spacing $h = L/N$. The box B is a sequence of consecutive grid points, such as $B = \{0, h, 2h, \ldots, (N-1)h\}$. The finite difference operator (1) defines a linear transformation of the space V of periodic grid functions:

$$Du_i = \frac{1}{h^2} \sum_{j=-r}^{j=s} a_j u_{i+j} \; . \tag{5}$$

The *symbol* of D satisfies the equation

$$m_D(n) v_n(x_i) = D v_n(x_i) \; ,$$

where $v_n(x) = e^{2\pi inx/L}$ is the Fourier mode and $x_i = ih$ with $h = L/N$ is the grid spacing.

In addition to grid functions, we will consider functions $u(x)$ defined for all $x \in \mathbb{R}$. The precise definition of "all functions" is not important here and will stay vague. The functions may need to be differentiable up to some order. A *linear operator*, A, is a mapping $u \overset{A}{\rightsquigarrow} w = Au$ that is linear. For example, $u \rightsquigarrow w$ with $w(x) = \partial_x^2 u(x)$ is the linear operator $A = \partial_x^2$. The right shift by a is the linear transformation $T_a u(x) = u(x-a)$. A linear operator is translation invariant if it commutes with any shift. These definitions are almost the same as the corresponding definitions for grid functions. A differential operator, such as $u(x) \rightsquigarrow c_0(x)u(x) + c_1(x)\partial_x + \cdots$, is translation invariant if and only if it has *constant coefficients*, which means that the functions $c_j(x)$ are independent of x.

(a) Find a relationship between the mode number n and the wave number ξ so that the symbol is proportional to the *scaled symbol*, which is

$$r_D(\xi) = \sum_{j=-r}^{j=s} a_j e^{ij\xi} \ . \tag{6}$$

(b) The symbol of a translation invariant differential operator A is defined by $r_A(\xi)e^{i\xi x} = Ae^{i\xi x}$. The symbol of the operator $A = \partial_x^2$ is $r_{\partial_x^2}(\xi) = -\xi^2$. Show that a finite difference approximation, D, to ∂_x^2 has order p if and only if

$$r_D(\xi) = -\xi^2 + O\big(\xi^{2+p}\big) \ . \tag{7}$$

(c) Calculate the scaled symbols of the three point second order, and the five point fourth order, difference approximations from part (1). Show that you can determine their orders of accuracy agree with the criterion (7).

(d) Write a Python script to plot the two scaled symbols and $-\xi^2$. You can do this by modifying the script `aliasing.py`. Hand in the program and the plot. The program should be well commented. The parameters used in the plot should appear in the title or legend. Comment on the range of ξ where the scaled symbols are close to $-\xi^2$. Do the scalings so that the order of agreement is clearly visible on the plot.

3. (*Condition number and eigenvalue distribution*) Consider the operator D_h on V, the space of grid functions with periodic boundary conditions. The *condition number* of D_h is (there are multiple notations)

$$\text{cond}(D_h) = \kappa(D_h) = \frac{\max_j \sigma_j(D_h)}{\min_j \sigma_j(D_h)} \ ,$$

where σ_j are the singular values of the matrix D_h.

(a) Show that for any translation invariant operator, the condition number is also given by

$$\kappa(D_h) = \frac{\max_j |\lambda_j(D_h)|}{\min_j |\lambda_j(D_h)|} = \frac{\max_{n \in B'} |m_{D_h}(n)|}{\min_{n \in B'} |m_{D_h}(n)|} , \tag{8}$$

where $\lambda_j(D_h)$ are the eigenvalues of D_h and $m_{D_h}(n)$ is the symbol as defined in class. There are two things to show. The second part of (8) is *von Neumann* analysis. Numerical Methods II will do a lot of it.

(b) Use von Neumann analysis to compute the condition number of the second order three point approximation to the differential operator $\partial_x^2 - 1$, which is defined by $u \to \partial_x^2 u - u$. Describe the grid functions that achieve the max and min in (8). Show that the condition number grown (asymptotically) as a power of N (the number of grid points). Give the power.