Problem Set 4

18.330 Intro to Numerical Analysis (MIT, Spring 2019)

Due: March 14. To be submitted *online* on Stellar.

Problem 1. Constructing Finite Difference Stencils. (15 points)

As we saw in class, a *finite difference stencil* is a rule for approximating a n-th derivative by weighted samples of a function:

$$f^{(n)}(x) \approx \sum_k w_k f(x + k\,h).$$

For a function of two variables this generalizes to

$$f^{(n)}(x, y) \approx \sum_{n,m} w_{km} f(x + k\,h_x, y + m\,h_y).$$

For example, the centered difference to the first derivative has weights

$$w_{-1} = -\frac{1}{2h} \qquad w_1 = \frac{1}{2h},$$

and the centered difference to the second derivative has weights

$$w_{-1} = \frac{1}{h^2} \qquad w_0 = -\frac{2}{h^2} \qquad w_1 = \frac{1}{h^2}.$$

Mimick the procedure shown in class (Taylor expanding f and then canceling out terms) to construct the following finite difference stencils:

(a) A backward difference stencil for the first derivative that achieves second-order convergence.

(b) A stencil for the mixed partial derivative $\frac{\partial^2 f}{\partial x \partial y}$. How does convergence depend on stepsize h? You can use a single stepsize for both variables.

Problem 2. Convergence of Finite-Difference approximations. (25 points)

Consider the forward difference stencil,

$$f'(x, h) \approx \frac{f(x+h) - f(x)}{h},$$

and the functions

$$f(x) = e^x \qquad g(x) = x^2.$$

Write a program using the forward difference stencil to compute

$$\left.\frac{df}{dx}\right|_{x=1000} \qquad \left.\frac{dg}{dx}\right|_{x=1}.$$

Plot the relative error as a function of the stepsize h and explain what you see.

Problem 3. Boundary value problems. (25 points)

Consider a periodically forced oscillator which is described by the equation of motion,

$$\frac{d^2x}{dt^2} + \omega_0^2 x = \cos(t^2),$$

where we take the natural frequency as $\omega_0 = 4.3$. Initially, the position of the pendulum is $x(0) = 0.3$, and after a time $\Delta t = 10$, the position is measured as $x(10) = -2.9$.

Here, we want to reconstruct the trajectory in between.

(a) Write a brief description (in words) of how you would use ODE solving techniques (such as RK4) to solve this problem.

(b) Write a program that uses finite-difference techniques to solve the problem. Plot the trajectory $x(t)$ and the forcing function $f(t) = \cos(t^2)$ in the same figure.

(c) Re-run your program using the modified forcing function $f(x) = \cos(4.3\,t)$ and plot the results again. What is the name of the phenomenon you observe? (*Hint:* remember 18.03).

Problem 4. Planar Electrostatics. (35 points)

A common problem in physics is to obtain the electric potential $\phi(x, y)$ when the spatial distribution of electric charges $\rho(x, y)$ in two dimensions is known. These two fields satisfy the *Poisson equation*

$$\Delta\phi(x, y) = \rho(x, y),$$

where the *Laplacian* is $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$.

Using the potential, the electric field vector can then be computed as

$$\mathbf{E}(x, y) = -\nabla\phi(x, y).$$

Here, you will numerically solve the Poisson equation in a 2D square domain $0 \le x \le 1$ and $0 \le y \le 1$. We will assume that the boundaries are perfect electric conductors, which implies the boundary conditions $\phi(0, y) = \phi(x, 0) = \phi(1, y) = \phi(x, 1) = 0$.

(a) Discretize the square into $(N+1) \times (N+1)$ points $\mathbf{x}_{nm} = \frac{1}{N+1}(n, m)$. This means that the vector of function samples for a function $f(x, y)$ on this grid will look like $(f_{11}, f_{12}, \ldots, f_{1N}, f_{21}, f_{22}, \ldots f_{2N}, \ldots f_{NN})$. (Note that while the function is defined in 2D, the vector of samples is a *flattened* vector with N^2 entries).

Find the entries of the matrices D_x, D_y of first derivative central differences in x, y and L_x, L_y of second derivative central differences. Don't forget to take the boundary conditions into account.

(b) Write a program that constructs the above matrices for arbitrary N. Construct the Laplacian matrix

$$L = L_x + L_y$$

for $N = 20$.

Use this matrix to solve the electrostatics problem with the charge densities

$$\rho_1(x, y) = e^{-50(x-0.5)^2 - 50(y-0.5)^2}$$

and

$$\rho_2(x, y) = e^{-100(x-0.25)^2 - 100(y-0.25)^2} - e^{-100(x-0.75)^2 - 100(y-0.75)^2}.$$

In each case, plot the electrostatic potential and the electric field. For the electric field, use the central difference matrix you found above.

(c) *Extra credit (10 points):* Can you come up with a scheme to solve the same problem on a circular domain? An arbitrary domain?

Problem 5. *Extra credit:* Reverse-mode automatic differentiation. (25 points)

Implement reverse mode automatic differentiation in Julia. Your implementation does not need to be very optimized, it is enough if it works for very basic functions like those seen in class.

A basic strategy may be to use the same operator overloading techniques shown in class. However, instead of computing the derivatives using dual numbers, construct the computational tree. You will need to define a new data type for the nodes of this tree which will hold information of the parent and child nodes, about how to take the derivative of the operation at that node, and all other information necessary.

Once you have the computational tree, you will need to write a function for the backward pass.

Test your implementation on a few functions. Check that it's accurate to machine precision.