CS 377P Spring 2017
Assignment 1: Computational science methods
Due: February 7th, 2017

January 29, 2017

Some of the problems in this assignment ask you to write code. This code must be written in MATLAB, which is available on the public machines in the CS department, or in Mathematica. You can also use Octave, which is free software and is similar to MATLAB.

1. (Difference equations, 5 points) Use induction to show that the closed-form solution of the difference equation

$$\begin{aligned} u(nh) &= a * u(nh - h) + b \quad (n{=}1{,}2{,}3{,}..) \\ u(0) &= 1 \end{aligned}$$

   is $u(nh) = a^n + b * \frac{(1-a^n)}{(1-a)}$.

2. (Iterative solution of linear systems, 15 points) Consider the linear system

$$\begin{aligned} 3x - 4y &= -1 \\ x + 2y &= 3 \end{aligned}$$

   (a) Write down the recurrence relation that corresponds to solving this system using the Jacobi method, starting with the initial approximation ($x_1 = 0, y_1 = 0$). Use the first equation to refine the approximation for x and the second equation to refine the approximation for y. Express this recurrence as a computation involving matrices and vectors.

   (b) Compute the first 25 approximations (x_i, y_i) and plot a 3D plot (x, y, i) in which the z-axis is the iteration number i. Give an intuitive explanation of this 3D plot.

   (c) Determine experimentally roughly how many iterations you need to obtain an approximate solution in which both x and y are within 1% of the exact solution.

   (d) Repeat these three parts for the Gauss-Seidel method. You can find a description of the Gauss-Seidel method online.

3. (ODE's, 20 points)) Consider the ordinary differential equation
$\frac{dy}{dx} = y * sin(x)$
with initial condition $y(0) = 1$ in the interval $0 < x < 2$. Use finite-differences to find an approximate solution to this differential equation, using a step size $h = 0.1$, as specified below.

   (a) (10 points) Recall that the forward-Euler approximation of the derivative is the following:
   $\frac{dy}{dx}|_{i*h} \approx \frac{y_f((i+1)*h) - y_f(i*h)}{h}$
   What is the recurrence for computing the Forward-Euler approximation to the solution of this ode? Draw a graph of this solution.

   (b) (10 points) Recall that the centered-difference approximation of the derivative is the following:
   $\frac{dy}{dx}|_{i*h} \approx \frac{y_c((i+1)*h) - y_c((i-1)*h)}{2h}$.
   Write down the corresponding recurrence formula and draw a graph of this solution.

   Hint: the differential equation tells you the derivative at 0 is 0. Use this fact to determine $y_c(h)$, and use the recurrence equation to compute the rest of the values.

4. (PDE's, 30 points) In this problem, we will solve the one-dimensional diffusion equation, which models how heat spreads through a material of uniform conductivity and similar problems. The diffusion equation is usually written as follows
$\frac{\delta f}{\delta t} = D\frac{\delta^2 f}{\delta x^2}$
The solution $f(x, t)$ depends on both x and t. Assume that we have a rod of length 10 meters, and that the two ends of the rod are kept at a fixed temperature of 0^oC, so $f(0, t) = 0.0$ and $f(10, t) = 0.0$. Assume that initially the temperature in the interior of the rod is $f(x, 0) = e^{-4(x-5)^2}$.

   The approximate solution $\hat{f}$ can be thought of a two-dimensional array in which $\hat{f}(i, j) \approx f(i\Delta x, j\Delta t)$. Intuitively, $\hat{f}(i, j)$ is the computed solution after j time steps at a spatial position i spatial steps away from the origin.

   (a) Compute the array $\hat{f}$ using the following discretization scheme, which discretizes time using Forward-Euler and discretizes space using centered-differences.
   $\Delta x = 0.25$
   $\Delta t = 0.025$
   $D = 1$
   $0 \leq x \leq 10$
   $0 \leq t \leq 10$
   $\frac{\delta f}{\delta t}|_{(i*\Delta x, j*\Delta t)} \approx \frac{\hat{f}(i,j+1) - \hat{f}(i,j)}{\Delta t}$ (Forward-Euler)
   $\frac{\delta^2 f}{\delta x^2}|_{(i*\Delta x, j*\Delta t)} \approx \frac{\hat{f}(i+1,j) - 2*\hat{f}(i,j) + \hat{f}(i-1,j)}{\Delta x^2}$ (Centered differences)

i. Draw the stencil for this discretization scheme.

ii. Use the recurrence equation to find an approximate solution to this problem. Plot the temperature distribution along the rod for different time steps on the same graph (so the x-axis is the distance from the origin in the rod and the y-axis is the temperature). Does this graph jive with your intuition?

iii. Increase the time step to 0.050 and repeat the previous steps. Explain your observations briefly.

5. (PDE's, 30 points) In the previous problem, we used the forward-Euler method to discretize time and centered differences to discretize space. As you should have observed in that problem, this discretization scheme is stable only if the time step is below a critical threshold (this threshold is given by something called the CFL-condition). For this reason, an implicit scheme called the Crank-Nicolson method is often used to discretize time when solving the heat equation. The Crank-Nicolson scheme is similar to the well-known trapezoidal method for integration that you may have learnt in freshman calculus. A good description of the Crank-Nicolson method can be found online in wikipedia.

   Repeat the previous problem using the Crank-Nicolson method. Experiment with different time-steps. Does the instability you should have observed for forward-Euler discretization show up for this method?

   Hint: As we discussed in class, implicit methods usually require solving linear systems. You do not have to write code for this - use the built-in Matlab functions for solving linear systems.

What to turn in: Create one zip file with your answers, MATLAB/Octave/-Mathematica code and graphs for all problems, and submit to Canvas. No credit will be given for problems in which you are asked to write code unless you include your code and we can run it.