18.303 Problem Set 2

Due Monday, 22 September 2014.

Problem 2: Modified inner products for column vectors

Consider the inner product $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^* B \mathbf{y}$ from class (lecture 5.5 notes), where the vectors are in $\mathbb{C}^N$ and B is an $N \times N$ Hermitian positive-definite matrix.

(a) Show that this inner product satisfies the required properties of inner products from class: $\langle \mathbf{x}, \mathbf{y} \rangle = \overline{\langle \mathbf{y}, \mathbf{x} \rangle}$, $\langle \mathbf{x}, \mathbf{x} \rangle > 0$ except for $\mathbf{x} = 0$. (Linearity $\langle \mathbf{x}, \alpha \mathbf{y} + \mathbf{z} \rangle = \alpha \langle \mathbf{x}, \mathbf{y} \rangle + \langle \mathbf{x}, \mathbf{z} \rangle$ is obvious from linearity the of matrix operations; you need not show it.)

(b) If M is an arbitrary (possibly complex) $N \times N$ matrix, define the adjoint $M^\dagger$ by $\langle \mathbf{x}, M\mathbf{y} \rangle = \langle M^\dagger \mathbf{x}, \mathbf{y} \rangle$ (for all $\mathbf{x}, \mathbf{y}$). (In this problem, we use $\dagger$ instead of $*$ for the adjoint in order to avoid confusion with the conjugate transpose: for this inner product, the adjoint $M^\dagger$ is *not* the conjugate transpose $M^* = \overline{M^T}$.) Give an explicit formula for $M^\dagger$ in terms of M and B .

(c) Using your formula from above, show that $M^\dagger = M$ (i.e., M is self-adjoint/Hermitian for *this* inner product) if $M = B^{-1} A$ for some $A = A^*$.

Problem 2: Finite-difference approximations

For this question you may find it helpful to refer to the notes and readings from lecture 3. Suppose that we want to compute the operation

$$\hat{A} u = \frac{d}{dx} \left[c \frac{du}{dx} \right]$$

for some smooth function $c(x)$ (you can assume c has a convergent Taylor series everywhere). Now, we want to construct a finite-difference approximation for $\hat{A}$ with $u(x)$ on $\Omega = [0, L]$ and Dirichlet boundary conditions $u(0) = u(L) = 0$, similar to class, approximating $u(m\Delta x) \approx u_m$ for M equally spaced points $m = 1, 2, \ldots, M$, $u_0 = u_{M+1} = 0$, and $\Delta x = \frac{L}{M+1}$.

(a) Using center-difference operations, construct a finite-difference approximation for $\hat{A} u$ evaluated at $m\Delta x$. (Hint: use a centered first-derivative evaluated at grid points $m + 0.5$, as in class, followed by multiplication by c, followed by another centered first derivative. Do *not* separate $\hat{A} u$ by the product rule into $c'u' + cu''$ first, as that will make the factorization in part (d) more difficult.)

(b) Show that your finite-difference expressions correspond to approximating $\hat{A} u$ by $A\mathbf{u}$ where $\mathbf{u}$ is the column vector of the M points u_m and A is a real-symmetric matrix of the form $A = -D^T C D$ (give C, and show that D is the same as the 1st-derivative matrix from lecture).

(c) In Julia, the `diagm(c)` command will create a diagonal matrix from a vector `c`. The function
`diff1(M) = [ [1.0 zeros(1,M-1)]; diagm(ones(M-1),1) - eye(M) ]`
will allow you to create the $(M+1) \times M$ matrix D from class via `D = diff1(M)` for any given value of M. Using these two commands, construct the matrix A from part (d) for $M = 100$ and $L = 1$ and $c(x) = e^{3x}$ via

```
L = 1
M = 100
D = diff1(M)
dx = L / (M+1)
x = dx*0.5:dx:L # sequence of x values from 0.5*dx to <= L in steps of dx
C = ....something from c(x)...hint:  use diagm...
```

`A = -D' * C * D` / dx^2

You can now get the eigenvalues and eigenvectors by `λ, U = eig(A)`, where λ is an array of eigenvalues and U is a matrix whose columns are the corresponding eigenvectors (notice that all the λ are < 0 since A is negative-definite).

(i) Plot the eigenvectors for the smallest-magnitude four eigenvalues. Since the eigenvalues are negative and are sorted in increasing order, these are the *last* four columns of U. You can plot them with:

```
using PyPlot
plot(dx:dx:L-dx, U[:,end-3:end])
xlabel("x"); ylabel("eigenfunctions")
legend(["fourth", "third", "second", "first"])
```

(ii) Verify that the first two eigenfunctions are indeed orthogonal with `dot(U[:,end], U[:,end-1])` in Julia, which should be zero up to roundoff errors $\lesssim 10^{-15}$.

(iii) Verify that you are getting second-order convergence of the eigenvalues: compute the smallest-magnitude eigenvalue λ_M`[end]` for $M = 100, 200, 400, 800$ and check that the *differences* are decreasing by roughly a factor of 4 (i.e. $|\lambda_{100} - \lambda_{200}|$ should be about 4 times larger than $|\lambda_{200} - \lambda_{400}|$, and so on), since doubling the resolution should multiply errors by 1/4.

(d) For $c(x) = 1$, we saw in class that the eigenfunctions are $\sin(n\pi x/L)$. How do these compare to the eigenvectors you plotted in the previous part? Try changing $c(x)$ to some other function (note: still needs to be real and > 0), and see how different you can make the eigenfunctions from $\sin(n\pi x/L)$. Is there some feature that always remains similar, no matter how much you change c?

Problem 3: Discrete diffusion

In this problem, you will examine thermal conduction in a system of a finite number N of pieces, and then take the $N \to \infty$ limit to recover the heat equation. In particular:

- You have a metal bar of length L and cross-sectional area a (hence a volume La), with a varying temperature T along the rod. We conceptually subdivide the rod into N (touching) pieces of length $\Delta x = L/N$.

- If Δx is small, we can approximate each piece as having a uniform temperature T_n within the piece ($n = 1, 2, \ldots, N$), giving a vector $\mathbf{T}$ of N temperatures.

- Suppose that the rate q (in units of W) at which heat flows across the boundary from piece n to piece $n+1$ is given by $q = \frac{\kappa a}{\Delta x}(T_n - T_{n+1})$, where κ is the metal's *thermal conductivity* (in units of W/m·K). That is, piece n *loses* energy at a rate q, and piece $n+1$ *gains* energy at the same rate, and the heat flows faster across bigger areas, over shorter distances, or for larger temperature differences. Note that $q > 0$ if $T_n > T_{n+1}$ and $q < 0$ if $T_n < T_{n+1}$: heat flows from the hotter piece to the cooler piece.

- If an amount of heat ΔQ (in J) flows into a piece, its temperature changes by $\Delta T = \Delta Q/(c\rho a \Delta x)$, where c is the specific heat capacity (in J/kg·K) and ρ is the density (kg/m^3) of the metal.

- The rod is insulated: no heat flows out the sides or through the ends.

Given these assumptions, you should be able to answer the following:

(a) "Newton's law of cooling" says that that the temperature of an object changes at a rate (K/s) proportional to the temperature difference with its surroundings. Derive the equivalent here: show that our assumptions above imply that $\frac{dT_n}{dt} = \alpha(T_{n+1} - T_n) + \alpha(T_{n-1} - T_n)$ for some constant α, for $1 < n < N$. Also give the (slightly different) equations for $n = 1$ and $n = N$.

(b) Write your equation from the previous part in matrix form: $\frac{d\mathbf{T}}{dt} = A\mathbf{T}$ for some matrix A.

(c) Let $T(x,t)$ be the temperature along the rod, and suppose $T_n(t) = T([n - 0.5]\Delta x, t)$ (the temperature at the *center* of the n-th piece). Take the limit $N \to \infty$ (with L fixed, so that $\Delta x = L/N \to 0$), and derive a partial differential equation $\frac{\partial T}{\partial t} = \hat{A}T$. What is $\hat{A}$? (Don't worry about the $x = 0, L$ ends until the next part.)

(d) What are the boundary conditions on $T(x,t)$ at $x = 0$ and L? Check that if you go backwards, and form a center-difference approximation of $\hat{A}$ with these boundary conditions, that you recover the matrix A from above.

(e) How does your $\hat{A}$ change in the $N \to \infty$ limit if the conductivity is a function $\kappa(x)$ of x?

(f) Suppose that instead of a thin metal bar (1d), you have an $L \times L$ thin metal *plate* (2d), with a temperature $T(x,y,t)$ and a constant conductivity κ. If you go through the steps above dividing it into $N \times N$ little squares of size $\Delta x \times \Delta y$, what PDE do you get for T in the limit $N \to \infty$? (Many of the steps should be similar to above.)