18.303 Problem Set 2

Due Friday, 23 September 2016.

Problem 1: Inner products, adjoints, and definiteness

Here, we consider inner products $\langle u, v\rangle$ on some vector space V of real-valued functions and the corresponding adjoint $\hat{A}^*$ of real-valued operators $\hat{A}$, where the transpose is defined, as in class, by whatever satisfies $\langle u, \hat{A}v\rangle = \langle \hat{A}^* u, v\rangle$ for all u and v in the vector space (usually, $\hat{A}^*$ is obtained from $\hat{A}$ by some kind of integration by parts).

(a) Suppose V consists of the functions $u(x)$ on $x \in [0, L]$ with quasiperiodic boundary conditions $u(0) = e^{i\phi}u(L)$ and $u'(0) = e^{i\phi}u'(L)$, as in pset 1, and the inner product is $\langle u, v\rangle = \int_0^L \overline{u(x)}v(x)dx$. Show that $\hat{A} = -d^2/dx^2$ is self-adjoint ($\hat{A} = \hat{A}^*$). For what values (if any) of ϕ is it positive-definite? Is this consistent with what you found in problem 3(a) of pset 1?

(b) Suppose that $\hat{A} = -\frac{d^2}{dx^2} + q(x)$ where $q(x)$ is some real-valued function with $q(x) \geq q_0$ for some constant q_0. Show that this is self-adjoint, for the same vector space V and inner product as in the first part. Furthermore, show that all eigenvalues λ of $\hat{A}$ are $\geq q_0$. (Hint: consider whether $\hat{A} - q_0$ is definite.)

Problem 2: Modified inner products for column vectors

Consider the inner product $\langle \mathbf{x}, \mathbf{y}\rangle_B = \mathbf{x}^* B\mathbf{y}$ from pset 1 (problem 1b), where B is a real-symmetric positive-definite matrix.

If A is a real-symmetric matrix, then show that the matrix $C = B^{-1}A$ is self-adjoint with respect to the $\langle \mathbf{x}, \mathbf{y}\rangle_B$ inner product, i.e. that $\langle \mathbf{x}, C\mathbf{y}\rangle_B = \langle C\mathbf{x}, \mathbf{y}\rangle_B$.

[Hence the result from pset 1b (real λ and orthogonal eigenvectors of $B^{-1}A$) follows immediately by the proof in class.]

Problem 3: Finite-difference approximations

Suppose that we want to analyze the operation (from class)

$$\hat{A}u = c\frac{d^2u}{dx^2}$$

where $c(x) > 0$ is a real-valued positive function. 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) Write down a finite-difference approximation, using center differences as in class, that corresponds 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 matrix of the form $A = -CD^TD\ldots$ that is, give the matrix C, where D is the same as the 1st-derivative matrix from lecture.

(b) Explain why you expect the matrix A to have real, negative eigenvalues, even though $A \neq A^T$. (Hint: choose the correct inner product, with help from problem 2!)

(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 (except missing the $1/\Delta x$ factor) via `D = diff1(M)` for any given value of M. Using these two commands, construct the matrix A from part (a) for $M = 100$ and $L = 1$ and $c(x) = e^{3x}$ via

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

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, by sorting them in decreasing order, these become the first four columns of U. You can sort and plot them with:

```
using PyPlot
i = sortperm(λ, rev=true) # i sorts λ in descending order
plot(x, U[:,i[1:4]])
xlabel("x"); ylabel("eigenfunctions")
legend(["first", "second", "third", "fourth"])
```

(ii) Verify that the first two eigenfunctions are indeed orthogonal for the correct inner product with `dot(U[:,i[1]], X*U[:,i[2]])` in Julia, where you replace `X` by an appropriate matrix (hint: see problem 2): the result of `dot(...)` should be zero up to roundoff errors $\lesssim 10^{-15}$.

(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?

(e) How would the matrix A change if the boundary conditions were quasiperiodic $[u(0) = e^{i\phi}u(L)]$, as in problem 1? (Hint: review how we derived the `A` or `D` matrices in class, and look at the first and last rows.)