CS 621: Matrix Computations
Fall 2001
Problem Set 5

Handed out: Fri., Nov. 2.

Due: Fri., Nov. 16 in lecture.

1. (a) Show that if T is an upper triangular matrix, then its eigenvectors can be found using backsubstitution. For simplicity, assume the diagonal entries of T are all distinct. Note that the eigenvector will have a block of zeros in it.

   (b) Use the result in (a) to draw the following conclusion: if $A = QTQ^H$ (Schur form), then the eigenvector of A corresponding to eigenvalue $T(i,i)$ lies in the range of $Q(:,1:i)$ for each $i = 1:n$.

2. Let Q and U be two unitary matrices such that for each $i = 1:n$, $R(Q(:,1:i)) = R(U(:,1:i))$ where $R(\cdot)$ means "range." Show that Q and U are identical, except for possible unit-length rescalings of the columns.

3. (TB25.1) Let T be a hermitian tridiagonal matrix all of whose off-diagonal entries are nonzero. Show that all of T's eigenvalues are distinct. [Hints: (1) If a hermitian matrix A has a multiple eigenvalue μ, what do we know about the rank of $A - \mu I$? (2) What can be said about the rank of $T - \mu I$, where μ is a real number, when T has the special form as in the question?]

4. In lecture we have not discussed termination tests for eigenvalue and eigenvector algorithms. Termination tests can be quite complicated. Implement the simple power method in the form of a function that takes as input a square matrix A and a user tolerance `tol`. The routine should generate a random $\mathbf{x}^{(0)}$ and continue to iterate until the computed eigenvector $\hat{\mathbf{x}}$ is sufficiently accurate such that if $\mathbf{x}$ is the true eigenvector, then $\|\hat{\mathbf{x}} - \mathbf{x}\|/\|\mathbf{x}\| \leq$ `tol`. Your m-file should not use computations too much more complicated than what is in the power method (so for example, it should not call `eig`). This means that the termination test will have to be based on some heuristic reasoning, since $\mathbf{x}$ is not known to your routine.

   In addition, the routine must have a second termination test for failure to converge. As mentioned in lecture, the power method is not guaranteed to converge.

   Explain the heuristic reasoning behind your termination tests. Also, see if you can concoct a test case (that is, a specification of A and `tol`) in which your program misjudges the accuracy of its computed $\hat{\mathbf{x}}$ and either runs for far too many or far too few iterations. (For this part, it is acceptable to use `eig` to get the "exact" eigenvectors.) If you can find such a test case, then explain the reasoning behind that test case.