Assignment #6

Due Friday 22 March, 2013

Please read Lectures 7, 8, 9, 10, and 11 in the textbook *Numerical Linear Algebra* by Trefethen and Bau. Do these exercises:

Exercise 7.3 in Lecture 7.

Exercise 8.1 in Lecture 8.

Exercise 9.1 in Lecture 9.

Exercise 10.1 in Lecture 10.

P12. Let $A \in \mathbb{C}^{m\times n}$, $m \ge n$, be a matrix with orthogonal, but not orthonormal, columns. Fully-describe the reduced QR factorization assuming $\hat{R}$ has positive diagonal entries.

P13. We have three algorithms for computing $A = QR$, namely Algorithms 7.1, 8.1, and 10.1, and each of these algorithms requires a finite number of steps. If you flip through the text or look at the inside front cover you will see an apparently related factorization for square matrices which could be written "$A = QRQ^*$." Instead it is denoted $A = UTU^*$ in this book, namely the Schur factorization. Here U is unitary and T is upper triangular. This problem explains why algorithms for computing the Schur factorization must be completely different from those for $A = QR$.

(a) Recall that the (monic) *characteristic polynomial* of $A \in \mathbb{C}^{m\times m}$ is $p(\lambda) = \det(\lambda I - A)$. Recall that the eigenvalues of A are the roots of p. Show by using properties of determinants that if $X \in \mathbb{C}^{m\times m}$ is invertible then the eigenvalues of A are the same as the eigenvalues of $B = XAX^{-1}$. Show, therefore, that if $A = UTU^*$ is a Schur factorization then the eigenvalues of T and of A are the same.

(b) Show that the eigenvalues of a triangular matrix are its diagonal entries. (It follows that the Schur factorization, unlike the QR or SVD or LU factorizations, is an "eigenvalue-revealing" factorization.)

(c) Consider the polynomial

$$p(z) = z^m + c_{m-1}z^{m-1} + c_{m-2}z^{m-2} + \cdots + c_1 z + c_0.$$

Show, presumably by expanding a determinant in minors, that p is the characteristic polynomial of this *companion matrix* of p,

$$C(p) = \begin{bmatrix} 0 & 0 & \dots & 0 & -c_0 \\ 1 & 0 & \dots & 0 & -c_1 \\ 0 & 1 & \dots & 0 & -c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -c_{m-1} \end{bmatrix}$$

(*I found it easiest to expand by minors using the first column and then work recursively. You might try the* 2×2 *and* 3×3 *cases first*.) It follows that for any polynomial there is a matrix for which the roots of the polynomial are the eigenvalues of the matrix.

(d) There is a famous theorem in mathematics that says that not all solutions of higher-degree polynomial equations with rational or integer coefficients can be obtained by starting with the polynomial coefficients and doing finitely many steps of sums, differences, products, quotients, and radicals (n-th roots, for some integer n) of previously-obtained numbers.[1] In particular, such an algorithm for computing roots of polynomials does not exist in degree 5 and higher. Using this famous theorem, show by contradiction that there is no algorithm which has finitely-many steps and which computes the Schur factorization of a square matrix A of size $m \geq 5$.

[1] `http://en.wikipedia.org/wiki/Abel-Ruffini_theorem`