CS 421: Numerical Analysis
Fall 2000
Problem Set 2

Handed out: Wed., Sep. 20.

Due: Fri., Sep. 29 in lecture.

1. Exercise proposed by T. Coleman: (a) Show that for any $A \in \mathbf{R}^{n\times n}$, $\|A\|_2 \le \|A\|_F$. (b) Find a 2×2 diagonal matrix A such that $\|A\|_2 = \|A\|_F$.

2. Let A be a symmetric positive *semidefinite* matrix.

   (a) Show that $A(1,1)$ must be nonnegative.

   (b) Show that if $A(1,1) = 0$, then the whole first row and column of A must be all zeros.

   These two facts play a role in an efficient algorithm for testing whether a matrix is positive semidefinite.

3. Let U be an $n \times n$ nonsingular upper triangular matrix. (a) Show that $\|U^{-1}\|_\infty \ge 1/\min_i |U(i,i)|$. This fact leads to a simple but not very reliable condition-number estimator (namely, $\|U^{-1}\|_\infty \approx 1/\min_i |U(i,i)|$) for upper triangular matrices. (b) In fact, show that this estimator is not reliable by constructing a 2×2 upper triangular matrix U in which $\|U^{-1}\|_\infty \ge 10^8/\min_i |U(i,i)|$.

4. This question requires Matlab programming. Consider two different ways to generate an $n \times n$ unit lower triangular matrix L all of whose entries are at most 1 in magnitude. Method 1 is to generate the matrix directly by putting random numbers chosen from the interval $[-1, 1]$ below the diagonal (in Matlab, you need the `rand` function, the `triu` function, and the `eye` function). Method 2 is to generate a square matrix A at random, compute its $P^T LU$ factorization (in Matlab, use the `lu` function), and then save L (ignore P and U).

   For each of these two methods, generate matrices of varying sizes up to $n = 200$. For each L, compute the ∞-norm of L^{-1}. Make two plots: one showing $\|L^{-1}\|_\infty$ versus n for Method 1, and the other for Method 2. The two plots should behave quite differently, and the reason for this difference is not completely understood.

   Hand in: listings of m-files, sample runs, two plots, and a paragraph of conclusions.