CS 421: Numerical Analysis
Fall 2004
Problem Set 2

Handed out: Wed., Sep. 22.

Due: Fri., Oct. 1 in lecture.

1. Consider a program to evaluate

$$F(a,b) = \sqrt{a^2+b^2} - |a|.$$

Implementing this formula directly in Matlab (i.e., as `sqrt(a^2+b^2)-abs(a)`) is prone to overflow (e.g., in the case $a, b \approx 10^{180}$), underflow (e.g., in the case $a, b \approx 10^{-180}$), and severe cancellation (e.g., in the case $|a| \gg |b|$). Write (on paper) a matlab program to evaluate F that should be more robust against overflow, underflow and cancellation than the direct implementation. It is OK if your program needs some **if** statements.

2. Let U be an $n \times n$ nonsingular upper triangular matrix. (a) Show that $\|U^{-1}\|_\infty \geq 1/\min_i |U(i,i)|$. 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 $\|U^{-1}\|_\infty \geq 10^8/\min_i |U(i,i)|$.

3. 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.

4. Write a Matlab function `invlower` that computes L^{-1} given a lower triangular matrix L by applying forward substitution to the columns of the identity matrix. Make sure the inner loop is vectorized, and make sure that unnecessary operations on 0's are omitted.

Then write an m-file called `mycond` that computes the condition number of a lower triangular matrix by multiplying its norm (the matrix 2-norm, which is computed by `norm`) in matlab by the norm of the inverse as computed by `invlower`. Compare this to the builtin `cond` function. They should nearly identical answers for reasonably well-conditioned matrices, e.g., the matrix returned by `tril(randn(10,10))`. Which seems to be more accurate for extremely ill-conditioned lower triangular matrices? You can make a lower triangular matrix ill-conditioned by putting a number very close to 0 (say 1e-40) on the main diagonal, or by putting a very big number in an off-diagonal position, or both. You can get some idea of which routine (`cond` vs `mycond`) is more accurate by checking whether the inequalities of question 1 are satisfied by the results. Hand in listings of all m-files, some sample runs, and a paragraph of conclusions.