Math/Comp 128 NLA
Take-Home Exam 2
Fall 2007

Instructions: You may use only your book for this class, your notes/handouts from my lectures to complete this exam. You may use calculators or Matlab if you think either would be helpful. You may NOT talk to anyone else about the exam except the instructor. **Do all 5 problems. The last problem is worth 20pts.**

1. Sometimes, the ill-conditioning of a matrix is due simply to poor scaling of the columns. For example if we define

$$A_1 = \begin{bmatrix} 1 & 2 \\ 1 & -2 \\ 3 & 1 \end{bmatrix}, \text{ and } A_2 = \begin{bmatrix} 1 & 2\times 10^{-12} \\ 1 & -2\times 10^{-12} \\ 3 & 1\times 10^{-12} \end{bmatrix}$$

then $\kappa(A_1) \approx 1.38$ but $\kappa(A_2) \approx 1.16 \times 10^{12}$. Note A_1 and A_2 *only differ by a column scaling* (postmultiplication by a diagonal matrix). This may be due simply to the physics of the problem you are trying to solve. We know that even backward stable algorithms can give inaccurate results on ill-conditioned problems, but that backward stable algorithms should produce accurate results on well-conditioned problems. Fortunately, in this case, we don't have to run the algorithm on an ill-conditioned problem, if we use the relation between A_1 and A_2. Give a backward stable procedure to solve

$$\min_x \|A_2 x - b\|_2$$

that you expect should still produce accurate results and explain your reasoning. Remember: to get a backward stable algorithm, *each step* of your algorithm must be a backward stable procedure.

2. Let

$$A = \begin{bmatrix} 1 & 3 & 3 \\ 3 & 4 & 4 \\ -1 & 4 & 5 \end{bmatrix}$$

   (a) Find a 3x3 matrix,G, (using Givens/plane rotations) such that the matrix GA will have a zero in the (2,2) position. Show your work, compute the product GA.

   (b) Find a 3x3 matrix J such that AJ will have a zero in the (1,3) position. (Note, it's all where you put the c and the s, and what length-2 subvector you use to define them.)

3. Let B and C be two $m \times m$, symmetric, positive definite matrices. Show that the block 2x2 matrix

$$A = \begin{bmatrix} B & 0 \\ 0 & C \end{bmatrix}$$

is also symmetric and positive definite.

4. Compute the Cholesky factorization, by hand, for

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 5 & 5 \\ 1 & 5 & 14 \end{bmatrix}$$

and use it to solve $Ax = \begin{bmatrix} 3 \\ 11 \\ 20 \end{bmatrix}$. (Don't convert to decimal, and leave in square roots - all the numbers should come out nicely.)

5. In control theory, one must compute the response matrix

$$G(\omega) = C(i\omega I - A)^{-1}B$$

where $i = \sqrt{-1}$, ω is a real scalar denoting frequency and the $n \times n$ matrix A, the $n \times m$ matrix B, and the $r \times n$ matrix C are called the control matrices with $m \leq n$. In practice, one usually wants to compute these matrices for *several values of* ω.

(a) What must be true about ω in order for $(i\omega I - A)$ to be invertible?

(b) Recall that to compute $(i\omega I - A)^{-1}B$, one can solve instead the system $(i\omega I - A)X = B$ for X. On a previous homework, we discussed using a matrix factorization to do that, solving for each column of B. It costs $O(n^3)$ flops to factor $(i\omega I - A)$, then $O(n^2 m)$ to solve for B.

However, this is **expensive** if we want to do it for many ω values, because we have to keep refactoring before anything else can happen! So if n_ω is the number of values of ω we want to try, the total cost for computing $G(\omega)$ for all the values is $O((n^3+n^2m+rnm)n_\omega)$ flops.

Assume instead you first reduce A to upper Hessenberg form: i.e. you compute unitary V so that $V^*AV = H$ (equivalently, $A = VHV^*$). Show $G(\omega) = CV(i\omega I - H)^{-1}V^*B$.

(c) Give the number of flops for computing $G(\omega)$ using the Hessenberg reduction (use the fact that solving a linear system involving an $n \times n$ upper Hessenberg matrix costs only $O(n^2)$ flops), and discuss why this is an advantage over the factor-the-matrix-for-each-ω approach.

Math 250 Instructions: Do problems 1,2,4 and 5 above, then 24.2 part a