A SET OF COUNTER-EXAMPLES TO THE LINPACK CONDITION NUMBER ESTIMATOR

A. K. Cline

Department of Computer Sciences
The University of Texas at Austin

TR-176 March 1981

Summary:

The estimation of the condition number of a matrix relative to inversion is important for bounding the errors in computer solutions of linear systems of equations. A commonly used method of estimation is shown to underestimate the true condition number of a particular class of matrices to an arbitrarily large degree.

1. Introduction

In Cline, Moler, Stewart, and Wilkinson [1], an algorithm was proposed for estimating an ℓ_1 norm condition number of a matrix. Although the algorithm was not proved to produce estimates with small errors, matrices for which previous estimation algorithms produced large underestimates were handled well by the new estimator. Furthermore, in thousands of tests with random matrices of various distributions and dimensions, rarely has the estimate been less than 1/10 of true condition number. On the basis of this evidence, the estimation algorithm was incorporated into LINPACK [2], a popular package of software for solving systems of linear equations.

It will be shown here that for a certain 4 x 4 matrix whose elements depend on a parameter k, the condition number estimate can underestimate the true condition number by a factor arbitrarily large. It will be further shown that the given matrix is not an isolated example, but in fact concurrent purturbations in every element still results in arbitrarily large factors of underestimation.

Finally it will be noted that the matrix also yields a counter-example to the first estimation strategy described in [1] (the one actually employed in LINPACK is the second) but not to the third. The third estimation strategy is further considered for its computational characteristics, and some concluding remarks are made on the larger question of selection of condition number estimators.

2. The LINPACK Estimator

The value of the condition number of a matrix with respect to inversion is described in a large selection of numerical analysis texts. The quantity, which may be defined as

$$\kappa(A) = ||A|| \cdot ||A^{-1}||$$

(for a fixed matrix norm $||\cdot||$) is employed in several inequalities relating the error in the solution to a linear system to perturbations in the matrix or in the right hand side. Two such inequalities are:

1. If $Ax = b$ and $A(x + \Delta x) = b + \Delta b$

   then

$$\frac{||\Delta x||}{||x||} \leq \kappa(A) \frac{||\Delta b||}{||b||},$$

2. If $Ax = b$ and $(A + \Delta A)(x + \Delta x) = b$,

   then

$$\frac{||\Delta x||}{||x||} \leq \frac{\kappa(A) \; ||\Delta A||/||A||}{1 - \kappa(A) \cdot ||\Delta A||/||A||},$$

   provided $||\Delta A|| < 1/||A^{-1}||$.

The vector norm in these inequalities may be any vector norm consistant with the matrix norm.

Given a matrix norm such as the ℓ_1 norm (which equals the maximum of the sums of absolute values of components in matrix columns) or the ℓ_∞ norm (which equals the maximum of the sums of

absolute values of components in matrix rows), then $||A||$ may be computed with a moderate number of arithmetic operations (in either case about n^2 additions and n^2 absolute value evaluations for an n x n matrix A) compared to a factorization of the matrix (about $n^3/3$ multiplications and $n^3/3$ additions). Unfortunately, if $||A^{-1}||$ is to be obtained by first computing A^{-1} then taking its norm, about n^3 multiplications and n^3 additions are required. Since the solution to a linear system $Ax = b$ can be obtained in about n^2 multiplications and n^2 additions after a factorization has been determined, we see that perhaps three times as much computational effort is required to determine $||A^{-1}||$ (and hence $\kappa(A)$) as is required to solve a linear system.

The intention for simply estimating $||A^{-1}||$ (and hence estimating $\kappa(A)$) was to decrease this computational effort, perhaps requiring $0(n^2)$ operations. Some approaches have used the fact that

$$||A^{-1}|| = \max_{x \neq 0} \frac{||A^{-1}x||}{||x||}$$

(where the matrix norm is now assumed to be subordinate to the vector norm) and attempted to find vectors x in which the ratio $||A^{-1}x||/||x||$ was close to maximal.

The estimator of $||A^{-1}||$ in [1] is such an approach. The algorithm for selecting the x can be stated as

step 0. Factorize A. (Either of the forms PA = LU or LPA = U is adequate if P is a permutation matrix, L is unit,

lower triangular, and U is upper triangular)

step 1. Let $b_1 = 1$ and $z_1 = 1$

for $s = 2,\ldots,n$

step s. Let $p_j^{(s-1)} = \sum_{i=1}^{s-1} u_{i,j}\, z_i$ $\quad j = s,\ldots,n,$

$$z_s^+ = (-p_s^{(s-1)} + 1)/u_{ss}, \qquad z_s^- = (-p_s^{(s-1)} - 1)/u_{ss},$$

$$p_j^{(s)+} = p_j^{(s-1)} + u_{s,j}\, z_s^+, \qquad p_j^{(s)-} = p_j^{(s-1)} + u_{s,j}\, z_s^-$$

$j = s+1,\ldots,n.$

If $|-p_s^{(s-1)} + 1| + \sum_{j=s+1}^{n} |p_j^{(s)+}| \geq |-p_s^{(s-1)} - 1| + \sum_{j=s+1}^{n} |p_j^{(s)-}|,$

let $b_s = 1$ and $z_s = z_s^+$, otherwise let $b_s = -1$ and $z_s = z_s^-$.

Having obtained b, solve $A^T x = b$ for x (this actually only requires the use of z with the matrixes P and L) and finally solve $Ay = x$ for y. The estimate of $||A^{-1}||$ is then $||y||/||x||$ and $\kappa(A)$ is estimated by multiplying the estimated $||A^{-1}||$ by the computed $||A||$. The vector norm used in these calculations is the ℓ_1 norm and the matrix norm is that subordinate to the ℓ_1 vector norm. (Henceforth, these are denoted by $||\cdot||_1$).

As implemented in LINPACK, (using a scratch array of length n), this estimation requires about $2\frac{1}{2}\ n^2$ multiplications, $4\frac{1}{2}\ n^2$ additions, and $2\ n^2$ evaluations of absolute value. There may be

an additional $\frac{1}{2}\, n^2$ multiplications and $\frac{1}{2}\, n^2$ additions required depending upon the $b_s = \pm 1$ decisions. Furthermore, in order to minimize the liklihood of computations overflowing, rescaling is done involving perhaps $4\ n^2$ additional multiplications. These figures represent the work in excess of the approximately $n^3/3$ multiplications and $n^3/3$ additions necessary to obtain the factorization. As demanded, this work is $0(n^2)$ but may be as large as $14\ n^2$ total multiplications, additions, and absolute value evaluations and hence comparable to the factorization effort for small matrices (i.e. $n \leq 21$).

3. A Counter-example

The construction of a counter-example began with the determination of an upper triangular matrix U so that U^{-1} had large elements, yet the b and z selected by the algorithm (which satisfy $U^T z = b$) would have z of moderate size. Using a U, so that

$$U^{-T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ k & -k & 1 & 0 \\ 2 & 0 & k^{-1} & k^{-1} \end{bmatrix}$$

with large values of k would have this property if $b_2 = 1$. (The large values k and -k would not affect z since both b_1 and $b_2 = 1$.) The rest of the construction of a counter-example simply required

choosing a unit lower trangular matrix L so that little increase would be made in the solution of Ay = x. For this purpose L was chosen as

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Since

$$U = \begin{bmatrix} 1 & -1 & -2k & 0 \\ 0 & 1 & k & -k \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & k \end{bmatrix},$$

the resultant

$$A = LU = \begin{bmatrix} 1 & -1 & -2k & 0 \\ 0 & 1 & k & -k \\ 0 & 1 & k+1 & -(k+1) \\ 0 & 0 & 0 & k \end{bmatrix},$$

and this is our matrix whose condition number is seriously underestimated.

Theorem:

For $k \geq 2$, the ℓ_1 condition number of A is $8k^2 + 6k + 1$; the estimated ℓ_1 condition number (using the algorithm of section 2) is $(28k^3 + 39k^2 + 32k + 4)/(5k^2 + 2k) = 5.6k + 5.56 + 0(1/k)$. The

ratio of estimated condition number to true condition number is
$(7\ k^2 + 8k + 4)/(10\ k^3 + 9\ k^2 + 2k) = .7/k + .17/k^2 + 0(k^{-3})$

Proof:

The computation of $||A||_1 = 4\ k + 1$ and $||A^{-1}||_1 = 2\ k + 1$ for $k \geq 2$. The estimation algorithm proceeds as follows:

step 1. $b_1 = 1,\ z_1 = 1.$

step 2. $p_2^{(1)} = -1 \quad p_3^{(1)} = -2k \quad p_4^{(1)} = 0$

$z_2^+ = 2 \quad -p_2^{(1)}+1 = 2 \quad p_3^{(2)} = 0 \quad p_4^{(2)+} = -2k$

$z_2^- = 0 \quad -p_2^{(1)}-1 = 0 \quad p_3^{(2)-} = -2k \quad p_4^{(2)-} = 0$

thus since $2+0+2k = 2k+2 > 2k = 0+2k+0$, $b_2 = 1$ and $z_2 = z_2^+ = 2$.

step 3. $p_3^{(2)} = 0 \quad p_4^{(2)} = -2k$

$z_3^+ = 1 \quad -p_3^{(2)}+1 = 1 \quad p_4^{(3)+} = -2k + 1$

$z_3^- = 1 \quad -p_3^{(2)}-1 = -1 \quad p_4^{(3)-} = -2k+1$

thus since $1+2k+1 = 2k+2 > 2k = 1+2k-1$, $b_3 = 1$ and $z_3 = z_3^+ = 1$.

step 4. $p_4^{(3)} = -2k-1$

$z_4^+ = 2+2k^{-1} \quad -p_4^{(3)}+1 = 2k+2$

$z_4^- = 2 \quad -p_4^{(3)}-1 = 2k$

thus since $2k+2 > 2k$, $b_4 = 1$ and $z_4 = z_4^+ = 2k + 2k^{-1}$.

It is easy to confirm that $x = A^{-T} b = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 2+2k^{-1} \end{bmatrix}$

and $y = A^{-1} x = \begin{bmatrix} 6 + 4\ k^{-1} \\ 1 \\ 2k^{-1} + 2k^{-2} \\ 2k^{-1} + 2k^{-2} \end{bmatrix}$. Thus

the estimated norm of A^{-1} is

$$||y||/||x|| = (7 + 8k^{-1} + 4k^{-2})/(5 + 2k^{-1})$$

$$= (7k^2 + 8k + 4)/(5k^2 + 2k)$$

and the estimated condition number multiplys this by $||A||_1 = 4k + 1$ to obtain $(28\ k^3 + 39\ k^2 + 32\ k + 4)/(5k^2 + 2k)$, whereas the true condition number is $(2k+1)(4k+1) = 8k^2 + 6k + 1$.

Experiments with the LINPACK subroutine SGECO using values of $k = 2,4,8,\ldots,1024$ performed exactly as predicted by the theorem.

4. Perturbations of the Counter-example

An examination of the proof of the theorem shows that the three choices (at steps 2, 3, and 4) which resulted in b_2, b_3, and b_4 all being set equal to +1, were forced because

$$|p_s^{(s-1)}+1| + \sum_{j=s+1}^{n} |p_j^{(s)+}| = 2k+2>2k = |p_s^{(s-1)}-1| + \sum_{j=s+1}^{n} |p_j^{(s)-}|.$$

Thus, if the matrix U were subjected to perturbations from that given, then by continuity, for small enough perturbations we would still have $|p_s^{(s-1)}+1| + \sum_{j=s+1}^{n} |p_j^{(s)-}| > |p_s^{(s-1)}-1| + \sum_{j=s+1}^{n} p_j^{(s)-}$. This would imply b_2, b_3, and b_4 being set to +1 as before.

We must be careful about claiming that small perturbations in A lead to small perturbations in the factor U, since this may not be so. If row interchanges are being performed in a partial pivoting strategy then a large positive perturbation in $a_{3,2}$ could lead to an interchange of rows 2 and 3 in the second step of the factorization, and the resulting U would not be a perturbation of the given U. However, if we let e_{ij} be the perturbation in a_{ij} for i, j = 1,...,4 and assume that $(e_{21} - e_{31})(1 - e_{12}) > (e_{32} - e_{22})(1 + e_{11})$ then the perturbed upper triangular factor will be a small perturbation on the given U. As is explained in the previous paragraph, for sufficiently small perturbations, the algorithm will produce $b_1 = b_2 = b_3 = b_4 = 1$. Thus the vector $\tilde{x}$ satisfying $\tilde{x} = (A+E)^{-T} b$ and the vector $\tilde{y}$ satisfying $\tilde{y} = (A+E)^{-1} \tilde{x}$ will be small perturbations of the previous x and y, respectively. We may conclude that since the estimated condition number $||A+E|| \cdot ||\tilde{y}|| / ||\tilde{x}||$ as well as the true condition number $||A+E|| \cdot ||(A+E)^{-1}||$ differ only slightly from $||A|| \cdot ||y|| / ||x||$ and $||A|| \cdot ||A^{-1}||$, respectively; severe underestimates of the condition number are afforded by such matrices A+E as well as A.

An analysis shows that if E is sufficiently small so that the decisions on the components of b lead to $b_1 = b_2 = b_3 = b_4 = 1$, then if $|e_{ij}| \leq \varepsilon$, the ratio of the estimated to true condition number is less than about $.7k^{-1} + (15+5k)\varepsilon$. An example with $\varepsilon = 10^{-5}$ and $k = 100$ showed the ratio smaller than .00806 in 10,000 tests where the perturbations were selected randomly but satisfying $(e_{21} - e_{31})(1 - e_{12}) > (e_{32} - e_{22})(1 + e_{11})$. (The bound would guarantee ratios smaller than .01215.) We may conclude that for every $k \geq 2$, the matrix A sits within an open set of matrices whose condition numbers are severely underestimated.

5. Alternative Estimation Algorithms

It is easy to verify that the matrix A of the previous section will yield the same estimated condition number (and hence the same underestimate) if the first algorithm of [1] is employed. This is also the algorithm described in Forsythe, Malcolm, and Moler [3] (although the code implementing it has a typographical error both in the book and in the distribution source). However, a third algorithm is presented in [1], in which the criterion for selecting $b_s = +1$ (rather than -1) is replaced by

$$\frac{|-p_s^{(s-1)}+1|}{|u_{ss}|} + \sum_{j=s+1}^{n} \frac{|p_j^{(s)+}|}{|u_{jj}|} \geq \frac{|-p_s^{(s-1)}-1|}{|u_{ss}|} + \sum_{j=s+1}^{n} \frac{|p_j^{(s)-}|}{|u_{jj}|}.$$

For this algorithm, the matrix of the previous section has an estimated condition number of $8k^2 + 5 + 0(1/k)$ compared with a true condition number of $8k^2 + 6k + 1$. The ratio is about $1 - 3/4\ k + 0(k^{-2})$ and thus the estimate is quite satisfactory.

The description of this algorithm in [1] is followed by the statement "However, this modification increases the volume of computation appreciably." This statement is misleading. As previously mentioned, the current LINPACK condition number estimator requires between $2\frac{1}{2}\ n^2$ and $7\ n^2$ multiplications, between $4\frac{1}{2}\ n^2$ and $5\ n^2$ additions, and $2\ n^2$ evaluations of absolute value. To implement the third algorithm of [1] (even allowing for rescaling) would require only an additional $\frac{1}{2}\ n^2$ evaluations of absolute value and $\frac{1}{2}\ n^2$ divisions. Assuming all operations require the same computational time, the third algorithm requires between 7% and 11% additional time.

The preceding is not, however, a convincing argument for replacing the current LINPACK condition number estimator with this alternative one. Despite the moderate additional computation and the fact that no counter-example has been presented for the alternative algorithm, a philosophical question should be answered before it (or any other highly complex algorithm) is adopted: Given that many condition number estimators perform well on random tests and in practice, yet fail for particular matrices, is the design of more and more complex algorithms yielding an improvement in estimation or is it simply erecting a higher barrier which makes

the determination of counter-examples more difficult but also makes any theoretical analysis close to impossible? If more and more complex algorithms are constructed simply to escape from counter-examples, the process may terminate only when counter-examples are not to be found only because of the limitations of human minds (not because of their non-existence).

Perhaps a complex estimator should be justified only if it is provably accurate. Unfortunately, very little has been proven about any of the estimators.

References:

1. A. K. Cline, C. B. Moler, G. W. Stewart, and J. H. Wilkinson, "An Estimate for the Condition Number of a Matrix", SIAM J. Numer. Anal. 16 (1979) 368-375.

2. J. J. Dongarra, C. B. Moler, J. R. Bunch, and G. W. Stewart, LINPACK User's Guide, SIAM, Philadelphia (1979).

3. G. E. Forsythe, M. A. Malcolm, and C. B. Moler, Computer Methods for Mathematical Computations, Prentice-Hall, Englewood Clifts (1979).