SIAM J. SCI. STAT. COMPUT.
Vol. 5, No. 2, June 1984

006

CONDITION ESTIMATES*

WILLIAM W. HAGER†

Abstract. A new technique for estimating the l_1 condition number of a matrix is developed and compared to an earlier scheme.

Key word. condition number

1. Introduction. Given an $n \times n$ matrix A and a vector $\mathbf{b} \in R^n$, the condition number measures the sensitivity of $\mathbf{x} = A^{-1}\mathbf{b}$ to changes in A or $\mathbf{b}$. If $\mathbf{x}+\delta\mathbf{x}$ satisfies

$$A(\mathbf{x}+\delta\mathbf{x}) = \mathbf{b}+\delta\mathbf{b},$$

then it is well known [6, p. 285] that

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leqq \|A\|\,\|A^{-1}\|\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}$$

where $\|\cdot\|$ denotes both a vector norm and the corresponding matrix norm defined by

$$\|A\| = \max\{\|A\mathbf{z}\|: \|\mathbf{z}\| = 1\}. \tag{1}$$

The parameter $\kappa = \|A\|\|A^{-1}\|$ is called the *condition number.* Similarly, if $\mathbf{x}+\delta\mathbf{x}$ satisfies

$$(A+\delta A)(\mathbf{x}+\delta\mathbf{x}) = \mathbf{b},$$

we have [6, p. 285]:

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}+\delta\mathbf{x}\|} \leqq \kappa\frac{\|\delta A\|}{\|A\|}.$$

In practice, the most common norms are the l_1, l_2, and l_∞ norms given by

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n}|x_i|, \quad \|\mathbf{x}\|_2 = \left(\sum_{i=1}^{n} x_i^2\right)^{1/2}, \quad \|\mathbf{x}\|_\infty = \max\{|x_1|, |x_2|, \cdots, |x_n|\}.$$

It is well known [5, p. 21–22] that the corresponding matrix norms (1) can be expressed as follows:

$$\|A\|_1 = \max_j \sum_{i=1}^{n}|a_{ij}|, \quad \|A\|_2 = \rho(A^TA), \quad \|A\|_\infty = \max_i \sum_{j=1}^{n}|a_{ij}|,$$

where a_{ij} is the element in row i and column j or A, ^{T} denotes transpose, and ρ is the spectral radius. Both $\|A\|_2$ and $\|A^{-1}\|_2$ can be estimated by the power method [7, Chapter 9] while $\|A\|_1$ and $\|A\|_\infty$ can be evaluated explicitly. We focus on the problem of determining $\|A^{-1}\|_1$ and $\|A^{-1}\|_\infty$. Of course, this problem is trivial when A^{-1} is known. But since A^{-1} is rarely needed in scientific computations and the cost of inverting a matrix is often 3 or more times the cost of factoring a matrix, it is important to estimate $\|A^{-1}\|_1$ from A's factors, not from the inverse. Also note that any scheme for computing the l_1 norm of A^{-1} can be used to evaluate the l_∞ norm since $\|A^{-1}\|_\infty = \|A^{-T}\|_1$.

* Received by the editors March 16, 1982, and in revised form January 12, 1983. This research was partly supported by the National Science Foundation under grant MCS 8101892.

† Department of Mathematics, Pennsylvania State University, University Park, Pennsylvania 16802.

Cline, Moler, Stewart and Wilkinson [1] give a strategy for estimating $\|A^{-1}\|$ that involves solving two systems:

$$A^T\mathbf{x}=\mathbf{b}, \qquad A\mathbf{y}=\mathbf{x}$$

where $\mathbf{b}$ is chosen during the substitution process to "enhance" the growth of $\mathbf{x}$. Their estimate is

$$\|A^{-1}\|_1 \sim \|\mathbf{y}\|_1/\|\mathbf{x}\|_1.$$

This scheme is incorporated in LINPACK [2], a collection of programs for solving linear systems. To study reliability, O'Leary [4] computed the average ratio

$$r=\frac{\text{estimated } \|A^{-1}\|_1}{\text{actual } \|A^{-1}\|_1}$$

for 100 matrices of dimensions ranging from 5 to 50 where the a_{ij} were taken from a uniform distribution on $[-1, 1]$. Obviously, $r \leqq 1$ and $r=1$ if and only if the estimate is perfect. Column 2 of Table 1 is extracted from [4, Table 1]. O'Leary points out that for negligible cost, the strategy [1] can be improved slightly.

TABLE 1

n	Average r	Average s
5	.69	.61
10	.60	.55
20	.52	.42
40	.43	.40

On the surface, the reliability seems good. If the condition number is "big", then its estimate is big, on the average. However, these results are disappointing in the following respect: Setting

$$\mathbf{x}=\begin{bmatrix}1\\0\\0\\\vdots\\0\end{bmatrix},$$

let us solve $A\mathbf{y}=\mathbf{x}$ and consider the estimate $\|A^{-1}\|_1 \sim \|\mathbf{y}\|_1$. That is, $\|A^{-1}\|_1$ is approximated by the absolute sum of elements from column 1 of A^{-1}. Column 3 of Table 1 lists the average ratio

$$s=\frac{\sum_{i=1}^{n}|a_{i1}^{-1}|}{\|A^{-1}\|_1}$$

where a_{ij}^{-1} is the (i, j) entry of A^{-1}. Observe that this simple strategy is almost as good as the sophisticated approach! The next section presents a new scheme for estimating $\|A^{-1}\|_1$.

2. A new idea. Before developing our algorithm, let us note that for certain matrices with special structure, $\|A^{-1}\|_1$ can be computed very quickly. For example, if every element of A^{-1} is nonnegative, we can evaluate $\|A^{-1}\|_1$ by solving $A^T\mathbf{x}=\mathbf{1}$ where $\mathbf{1}$ is the vector whose components are all 1. Since the elements of A^{-1} are

nonnegative, the components of $\mathbf{x}$ are the column sums of A^{-1}, and $\|A^{-1}\|_1$ is the biggest component of $\mathbf{x}$. Our goal, however, is to develop an algorithm that is suitable for matrices whose elements are generated randomly.

Given an $n \times n$ matrix B, define $f: R^n \to R$ by

$$f(\mathbf{x}) = \|B\mathbf{x}\|_1 = \sum_{i=1}^{n} \left| \sum_{j=1}^{n} b_{ij} x_j \right|.$$

Thus we have

$$\|B\|_1 = \max \{f(\mathbf{x}): \|\mathbf{x}\|_1 \leqq 1\}.$$

Abstractly, $\|B\|_1$ is the maximum of the convex function f over the convex set

$$S = \{\mathbf{x} \in R^n: \|\mathbf{x}\|_1 \leqq 1\}.$$

It is well known that a convex function defined on a convex, compact set attains its maximum at an extreme point. The $2n$ extreme points of S are simply

$$\{\pm \mathbf{e}^j: j = 1, \cdots, n\}$$

where $\mathbf{e}^j$ is the unit vector whose components are all 0 except for the jth component which is 1. Since f is convex, it satisfies the inequality

$$f(\mathbf{y}) \geqq f(\mathbf{x}) + \partial f(\mathbf{x})(\mathbf{y} - \mathbf{x}) \tag{2}$$

for all $\mathbf{x}, \mathbf{y} \in R^n$ where $\partial f(\mathbf{x})$ denotes a subgradient of f at $\mathbf{x}$. If

$$\sum_{j=1}^{n} b_{ij} x_j \neq 0$$

for each i, then $\partial f(\mathbf{x})$ is the usual gradient vector. Defining for $i = 1$ to n,

$$\xi_i = \begin{cases} 1 & \text{if } \sum_{j=1}^{n} b_{ij} x_j \geqq 0, \\ -1 & \text{otherwise,} \end{cases} \tag{3}$$

the chain rule gives us

$$\partial f(\mathbf{x}) = \boldsymbol{\xi}^T B. \tag{4}$$

Note that if one or more components of $B\mathbf{x}$ are zero at some point $\mathbf{x}$, then the function $f(\cdot)$ has a corner at $\mathbf{x}$, and the set of subgradients has many elements at this point. That is, if $(B\mathbf{x})_i = 0$, then equation (4) gives us a different element of this set for each value of ξ_i between -1 and 1. Thus equations (3) and (4) specify a particular element of the subgradient set at the corners of $f(\cdot)$. In the special case $B = A^{-1}$, computing $\partial f(\mathbf{x})$ by equations (3) and (4) is equivalent to solving two systems:

$$A\mathbf{y} = \mathbf{x}, \qquad A^T \mathbf{z} = \boldsymbol{\xi} \tag{5}$$

where

$$\xi_i = \begin{cases} 1 & \text{if } y_i \geqq 0, \\ -1 & \text{otherwise,} \end{cases}$$

and $\partial f(\mathbf{x}) = \mathbf{z}^T$.

Our algorithm for estimating $\|B\|_1$ starts at a point $\mathbf{x}$ on the boundary of S. We then find a j for which

$$|\partial f(\mathbf{x})_j| = \max_i |\partial f(\mathbf{x})_i|. \tag{6}$$

If $|\partial f(\mathbf{x})_j| \leqq \partial f(\mathbf{x})\mathbf{x}$, then stop. (Below we show that this $\mathbf{x}$ is a "local maximum" of f over the polytope S). Conversely, suppose that $|\partial f(\mathbf{x})_j| > \partial f(\mathbf{x})\mathbf{x}$. By the convexity inequality (2) and the fact that $f(\mathbf{e}^j) = f(-\mathbf{e}^j)$, we conclude that $f(\mathbf{e}^j) > f(\mathbf{x})$. Replacing $\mathbf{x}$ by $\mathbf{e}^j$, this process repeats. Since f is strictly increasing, vertices of S are visited only once, and the iterations terminate in a finite number of steps. A Fortran code for our algorithm is included in [3].

To prove that the final point $\mathbf{x}$ generated by this algorithm is a local maximum, we assume that every component of $B\mathbf{x}$ is nonzero. In the case that some component of $B\mathbf{x}$ is zero, we should modify (6) by letting the index j correspond to the maximum absolute component over the entire set of subgradient vectors. The algorithm still makes sense without this modification, but $\mathbf{x}$ may not be a local maximum of f. When the components of $B\mathbf{x}$ are nonzero, $f(\cdot)$ is linear near $\mathbf{x}$. Hence $\mathbf{x}$ is a local maximum of f over S if and only if

$$\partial f(\mathbf{x})(\mathbf{y}-\mathbf{x}) \leqq 0$$

for every $\mathbf{y} \in S$. If $\mathbf{y}$ is a vertex of S, then $\partial f(\mathbf{x})\mathbf{y} = \pm \partial f(\mathbf{x})_i$ for some i since all but one component of $\mathbf{y}$ is zero. If $|\partial f(\mathbf{x})_i| \leqq \partial f(\mathbf{x})\mathbf{x}$ for each i, it follows that $\partial f(\mathbf{x})(\mathbf{y}-\mathbf{x}) \leqq 0$ whenever $\mathbf{y}$ is a vertex of S. Since S is the convex hull of its vertices, $\partial f(\mathbf{x})(\mathbf{y}-\mathbf{x}) \leqq 0$ for every $\mathbf{y} \in S$, and $\mathbf{x}$ is a local maximum of f over S.

To test this scheme, we computed the ratio

$$t_1 = \frac{\text{estimated } \|A^{-1}\|_1}{\text{actual } \|A^{-1}\|_1}$$

for 200 matrices of the same dimension where the a_{ij} are taken from a uniform distribution on $[-1, 1]$. Our initial guess is $\mathbf{x} = n^{-1}\mathbf{1}$. Column 3 of Table 2 gives the

TABLE 2

n	Average t_1	Average steps	Probability $t_1 \geqq .99$
5	.96	2.1	.82
10	.97	2.1	.83
20	.98	2.1	.88
40	.97	2.1	.85
80	.98	2.1	.86

average termination step, counting the initial guess $\mathbf{x} = n^{-1}\mathbf{1}$ as step 1. Column 4 is the proportion of the cases where $t_1 \geqq .99$. With few exceptions, $t_1 \geqq .99$ if and only if the algorithm actually found the vertex $\mathbf{e}^j$ for which $\|A^{-1}\mathbf{e}^j\|_1 = \|A^{-1}\|_1$. It appears that the reliability is independent of n. Since the average termination step is 2.1, the scheme starts from $\mathbf{x} = n^{-1}\mathbf{1}$ and almost always moves straight to a locally maximizing vertex of S. Of course, each step involves solving the two systems (5). In column 4 of Table 2, we see that the local maximum computed by the algorithm is a global maximum with high probability.

To estimate $\|A^{-1}\|_1$ more precisely, our scheme is applied repeatedly to suitable subspaces. During the first cycle described above, we visit vertices $\{\mathbf{v}^1, \cdots, \mathbf{v}^m\}$ and stop at a local maximum. Let $\{\mathbf{v}^{m+1}, \cdots, \mathbf{v}^n\}$ be the remaining vertices; that is,

$$\{\mathbf{v}^{m+1}, \cdots, \mathbf{v}^n\} = \{\mathbf{e}^1, \cdots, \mathbf{e}^n\} - \{\mathbf{v}^1, \cdots, \mathbf{v}^m\}.$$

Then starting at the point

$$\mathbf{x} = \frac{1}{n-m} \sum_{i=m+1}^{n} \mathbf{v}^i,$$

we apply the same scheme to the polytope S_2 with vertices

$$\{\pm \mathbf{v}^i : i = m+1, \cdots, n\}.$$

This leads us to a local maximum on S_2. Our estimate for $\|A^{-1}\|_1$ is the bigger local maximum. Letting t_2 be the ratio between the estimated $\|A^{-1}\|_1$ and the actual $\|A^{-1}\|_1$, our results for the two cycle process are summarized in Table 3.

TABLE 3

n	Average t_2	Average steps	Probability $t_2 \geqq .99$
5	.993	4.2	.94
10	.991	4.2	.94
20	.993	4.2	.95
40	.987	4.2	.90
80	.995	4.3	.95

Finally, the three cycle process yields Table 4.

TABLE 4

n	Average t_3	Average steps	Probability $t_3 \geqq .99$
5	.997	6.2	.98
10	.995	6.4	.97
20	.997	6.5	.96
40	.996	6.4	.97
80	.997	6.6	.97

The worst condition estimate that we detected for the 200 random matrices is shown in Table 5. If the hyperplanes $\{\mathbf{x} \in R^n : \sum_{j=1}^{n} b_{ij} x_j = 0\}$ do not intersect some face of S and $\mathbf{v}$ is any vertex of S, then one step of our algorithm starting from $\mathbf{v}$ takes us to a global maximum of f over S. This situation corresponds to f being linear on a face of S. On the other hand, when the hyperplanes intersect all the faces of S, then f has corners on each face, and it is possible to hide the global maximum behind a corner.

TABLE 5

n	t_1	t_2	t_3
5	.32	.67	.70
10	.39	.67	.76
20	.46	.62	.74
40	.43	.44	.78
80	.46	.71	.71

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), pp. 368–375.
[2] J. J. DONGARRA, J. R. BUNCH, C. B. MOLER AND G. W. STEWART, LINPACK *Users' Guide*, Society for Industrial and Applied Mathematics, Philadelphia, 1979.
[3] W. W. HAGER, *Computing*, book in preparation.
[4] D. P. O'LEARY, *Estimating matrix condition numbers*, this Journal, 1 (1980), 205–209.
[5] J. M. ORTEGA, *Numerical Analysis, A Second Course*, Academic Press, New York, 1973.
[6] G. STRANG, *Linear Algebra and Its Applications*, Academic Press, New York, 1980.
[7] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford Univ. Press, London, 1965.