Journal of Econometrics 12 (1980) 3–22. © North-Holland Publishing Company

TEST PROCEDURES AND TEST PROBLEMS FOR LEAST SQUARES ALGORITHMS

Roy H. WAMPLER

National Bureau of Standards, Washington, DC 20234, USA

Numerous test problems have been introduced in the past twenty years for the purpose of studying and comparing least squares algorithms and computer programs. This paper discusses and classifies some of the useful test problems which have appeared in the literature. A recent large scale test procedure is briefly summarized. Several neat, mathematical examples are displayed. One of these, first introduced by Läuchli, is modified so that it can be solved by the method of inverting a matrix of correlation coefficients. Comparative results from running two types of problems on several different algorithms are given which illustrate some of the factors affecting computational accuracy: choice of algorithm, scaling of the data, tolerance parameters, and iterative refinement.

1. Introduction

In the past twenty years, numerous test problems have been introduced for the purpose of testing and comparing linear least squares algorithms and computer programs. Such test problems have proved to be useful in comparing two or more algorithms with respect to numerical accuracy and computational efficiency. They have sometimes been used in comparative evaluations of least squares regression programs. They can be used to study the properties of a particular algorithm or to compare the performance of a particular program on two or more types of computers.

In this paper, many of the least squares test problems which have appeared in the literature are classified in several ways. Three problems of the neat, mathematical type are displayed, and a modification of one of these is given. Comparative results from running two types of problems on several different algorithms serve to illustrate some of the factors which affect the accuracy of computed solutions.

For recent accounts of the state of the art in solving least squares problems, the reader is referred to *Linear Regression Analysis* by Seber (1977), especially Chapter 11 on computational techniques, to *Computational Methods for Data Analysis* by Chambers (1977), especially Chapter 5 on linear models and the appendix which lists some available algorithms, and to *Solving Least Squares Problems* by Lawson and Hanson (1974).

Gregory and Karney (1969) gave a collection of numerical examples related to matrix inversion, systems of linear equations, eigenvalues and eigenvectors.

2. Formulation of the least squares problem

The simplest form of the linear least squares problem can be stated as follows: Given a matrix $X(n\times p)$ and a vector of observations $y(n\times 1)$, find a vector of coefficients $\hat{\beta}(p\times 1)$ which minimizes the sum of squares of the residual vector $\delta=y-X\beta$. When X is of rank p, the unique solution can be expressed as

$$\hat{\beta}=(X'X)^{-1}X'y.$$

Other quantities of interest are $\hat{y}=X\hat{\beta}$, the vector of predicted values, and $(X'X)^{-1}$, the unscaled covariance matrix of the coefficients.

A method frequently used in econometric work for obtaining the solution vector and related quantities requires the forming and inverting of a matrix of correlation coefficients. In this method it is assumed that one column of X, say the first, has each element equal to one. Let

$$\bar{x}_j=\sum_i x_{ij}\Big/n, \qquad \bar{y}=\sum_i y_i\Big/n,$$

$$z_{ij}=x_{ij}-\bar{x}_j, \qquad u_i=y_i-\bar{y},$$

$$s_j=\Big(\sum_i z_{ij}^2\Big)^{\frac{1}{2}}, \qquad s_y=\Big(\sum_i u_i^2\Big)^{\frac{1}{2}},$$

$$r_{jk}=\sum_i z_{ij}z_{ik}\Big/s_js_k, \qquad r_{jy}=\sum_i z_{ij}u_i\Big/s_js_y,$$

$$R_{xx}=(r_{jk}) \text{ for } j=2,3,\ldots,p, \quad k=2,3,\ldots,p,$$

$$r'_{xy}=(r_{2y},r_{3y},\ldots,r_{py}).$$

The $(p-1)\times(p-1)$ matrix R_{xx} is known as a correlation matrix. If we let

$$a=R_{xx}^{-1}r_{xy}=(a_2,a_3,\ldots,a_p)',$$

and

$$\hat{\beta}'=(b_1,b_2,\ldots,b_p),$$

the desired coefficients can be obtained from the formulas

$$b_j=a_js_y/s_j \quad \text{for} \quad j=2,3,\ldots,p,$$

and

$$b_1=\bar{y}-b_2\bar{x}_2-b_3\bar{x}_3-\ldots-b_p\bar{x}_p.$$

Formulas for expressing the diagonals of $(X'X)^{-1}$ in terms of R_{xx}^{-1}, $\bar{x}_j$ and s_j are given in Seber (1977) and Horiba (1971).

Maindonald (1977), in his article discussing least squares computations based on the correlation matrix, observes that this method is well adapted to updating, as in stepwise regression.

3. Classification of test problems

Least squares test problems can be classified in a variety of ways. Below are given several categories of test problems with examples of these types which have appeared in the literature. In most cases, either the data for the test problems were published or an algorithm was given for generating the data. In a few instances, however, the data were not published.

(1) (a) Problems arising from real data. Examples: Freund (1963), Zellner and Thornber (1966), Longley (1967), Krane (1970, 1971), Chambers (1971), Mullet and Murray (1971), Chambers (1973), Gentleman (1975), Beaton, Rubin and Barone (1976), Deegan (1976), Longley (1976), Chambers (1977), and Dent and Cavander (1977).

(b) Problems constructed from contrived data. Examples: Läuchli (1961), Bauer (1965), Golub and Reinsch (1970), Stewart (1973), Lawson and Hanson (1974), Kenkel (1976), Nash and Lefkovitch (1976), etc.

(2) (a) Polynomial regression problems. Examples: Cameron (1957), Ascher and Forsythe (1958), Macdonald (1964), Bright and Dawkins (1965), Jordan (1968), Muhonen (1968), Wampler (1969, 1970), Jennings and Osborne (1974), Fletcher (1975), Shampine (1975), Boehm, Menkhaus and Penn (1976), Longley (1976), and Björck (1978).

(b) Multiple regression problems. Examples: Bauer (1965), Golub (1965), Zellner and Thornber (1966), Longley (1967), Berk (1977), Frane (1977), etc.

(3) Problems having known condition number, or known to be ill-conditioned. Segments of the Hilbert matrix or of its inverse have been used by Golub (1965), Businger and Golub (1965), Golub and Wilkinson (1966), Björck and Golub (1967), Björck (1968), Peters and Wilkinson (1970), Abdelmalek (1971), Plemmons (1974), Tsao (1975), Daniel et al. (1976), and Björck (1978). Björck and Golub (1973) used Vandermonde matrices.

(4) Problems which are mathematically rank-deficient. Examples: Lawson and Hanson (1974, p. 277 and p. 311), using a Fortran function to generate data; Velleman, Seaman and Allen (1977).

(5) Problems constructed from a set of orthogonal vectors. Example: Hastings (1972).

(6) Problems constructed from the singular value decomposition. Examples: Velleman, Seaman and Allen (1977), and Seaman (1977).

(7) Problems having special features, such as:
(a) Solution vector is subject to linear constraints. Examples: Björck and Golub (1967), Björck (1968), Stoer (1971), Hastings (1972), and Gerig and Gallant (1975).

(b) Observations have unequal variances. Examples: Shampine (1975), Björck (1978), and Wampler (to appear).

(c) Data matrix X is sparse. Example: Gentleman (1975).

(d) Solutions are known for several regression methods, as least squares, least absolute residuals, and other robust methods. Example: Holland (1976).

4. A recent large-scale study

Velleman, Seaman and Allen (1977) recently reported the results of a large-scale evaluation of least squares regression routines available in widely distributed statistical packages. Velleman and co-workers (1975, 1977) devised a test procedure which considered five factors that can lead to computational difficulties:

(1) The columns of X can become highly collinear.
(2) A column of X can have a coefficient of variation which approaches zero.
(3) R^2, the coefficient of determination, can approach zero, indicating failure of the statistical model.
(4) The absolute magnitude of the numbers in X can become large, defeating absolute checks for rank-deficiency.
(5) The ratio of the magnitude of y to that of X can vary widely.

The procedure which was developed by Velleman et al. for probing these five factors generated sets of data through use of the singular value decomposition of the matrix X. Some of the generated data sets were mathematically rank-deficient. The seven statistical package regression routines which were evaluated displayed considerable variation in the accuracy obtained as well as in the protection given to users against computational disasters.

5. Neat, mathematical test problems

Läuchli (1961) introduced the following test problem:

$$X=\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \varepsilon & 0 & 0 & 0 & 0 \\ 0 & \varepsilon & 0 & 0 & 0 \\ 0 & 0 & \varepsilon & 0 & 0 \\ 0 & 0 & 0 & \varepsilon & 0 \\ 0 & 0 & 0 & 0 & \varepsilon \end{bmatrix}, \qquad y=\begin{bmatrix} \varepsilon \\ 0 \\ -5 \\ 5 \\ -5 \\ 0 \end{bmatrix}.$$

In the matrix $X'X$ all diagonal terms equal $1+\varepsilon^2$ and all off-diagonal terms equal 1. For $\varepsilon \neq 0$ the rank of $X'X$ is five since the eigenvalues are $5+\varepsilon^2$, ε^2, ε^2, ε^2, ε^2. Golub (1965) noted that if γ is the largest number on the computer such that $1.0+\gamma=1$ in floating point arithmetic, then whenever $\varepsilon<\sqrt{\gamma}/2$ the rank of the computed matrix $X'X$ will be one. No matter how accurate the linear equation solver, it is impossible to solve the normal equations $X'X\beta=X'y$.

This problem (sometimes with variations) has been widely used by later authors – e.g., Björck (1967), Dahlquist and Björck (1974).

Stewart (1973, p. 228) gave the matrix

$$X=\begin{bmatrix} 1 & 1+\varepsilon \\ 1 & 1-\varepsilon \\ 1 & 1 \end{bmatrix},$$

as an exercise in computing an upper bound on the condition number of X as a function of ε.

Lawson and Hanson (1974, p. 127) used the matrix

$$X=\begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1-\varepsilon \end{bmatrix},$$

in comparing the accuracy obtainable from Cholesky decomposition with that obtainable from Householder transformations. They observed that when one works with a fixed precision, a larger class of problems can be solved using Householder transformations than can be solved by forming the normal equations and using the Cholesky decomposition.

The three examples displayed above are quite useful pedagogically, but they are not suitable for comparing the correlation matrix method with other methods for obtaining least squares solutions. This is because Läuchli's example lacks a constant column vector and the other two examples lead to a correlation matrix having but one element, $R_{xx}=[1]$.

Läuchli's problem can be modified so that the matrix X contains a column vector equal to unity. The modified problem can then be solved by the correlation matrix method as well as by other methods.

Let the $n\times p$ matrix $X(n\geqq 3, p=n-1)$ be defined by

$$\begin{aligned}
x_{1j}&=1, && j=1,2,\dots,p,\\
x_{i1}&=1, && i=1,2,\dots,n,\\
x_{jj}&=\varepsilon, && j=2,3,\dots,p,\\
x_{ij}&=0, && \text{otherwise.}
\end{aligned}$$

Let

$$\begin{aligned}
y_1&=n-1+\varepsilon,\\
y_n&=n-1-\varepsilon,\\
y_i&=\varepsilon, \qquad i=2,3,\dots,n-1.
\end{aligned}$$

Then

$$\underset{p\times p}{X'X}=\begin{bmatrix} n & 1+\varepsilon & 1+\varepsilon & \dots & 1+\varepsilon\\ 1+\varepsilon & 1+\varepsilon^2 & 1 & \dots & 1\\ 1+\varepsilon & 1 & 1+\varepsilon^2 & \dots & 1\\ \vdots & \vdots & \vdots & & \vdots\\ 1+\varepsilon & 1 & 1 & \dots & 1+\varepsilon^2 \end{bmatrix},$$

$$\underset{p\times 1}{X'y}=\begin{bmatrix} 2(n-1)+(n-2)\varepsilon\\ n-1+\varepsilon+\varepsilon^2\\ n-1+\varepsilon+\varepsilon^2\\ \vdots\\ n-1+\varepsilon+\varepsilon^2 \end{bmatrix},$$

$$\underset{(p-1)\times(p-1)}{R_{xx}}=\begin{bmatrix} 1 & a & a & \dots & a\\ a & 1 & a & \dots & a\\ a & a & 1 & \dots & a\\ \vdots & \vdots & \vdots & & \vdots\\ a & a & a & \dots & 1 \end{bmatrix},$$

where

$$a=(n-1-2\varepsilon-\varepsilon^2)/[n-1-2\varepsilon+(n-1)\varepsilon^2].$$

We have

$$(X'X)^{-1}=\Delta_1\begin{bmatrix} b & c & c & \dots & c \\ c & d & e & \dots & e \\ c & e & d & \dots & e \\ \vdots & \vdots & \vdots & & \vdots \\ c & e & e & \dots & d \end{bmatrix},$$

where

$$\Delta_1=1/\{\varepsilon^2[(n-1)(n-2)-2(n-2)\varepsilon+2\varepsilon^2]\},$$

$$b=\varepsilon^2(n-2+\varepsilon^2),$$

$$c=-\varepsilon^2(1+\varepsilon),$$

$$d=(n-1)(n-3)-2(n-3)\varepsilon+3\varepsilon^2,$$

$$e=-(n-1)+2\varepsilon+\varepsilon^2,$$

$$R_{xx}^{-1}=\Delta_2\begin{bmatrix} f & g & g & \dots & g \\ g & f & g & \dots & g \\ g & g & f & \dots & g \\ \vdots & \vdots & \vdots & & \vdots \\ g & g & g & \dots & f \end{bmatrix},$$

where

$$\Delta_2=\frac{(n-1)-2\varepsilon+(n-1)\varepsilon^2}{n\varepsilon^2[(n-1)(n-2)-2(n-2)\varepsilon+2\varepsilon^2]},$$

$$f=(n-1)(n-3)-2(n-3)\varepsilon+3\varepsilon^2,$$

$$g=-(n-1)+2\varepsilon+\varepsilon^2,$$

$$\hat{B}'=(1,1,\dots,1),$$

$$\delta'=(\varepsilon,-1,-1,\dots,-1,n-2-\varepsilon).$$

Since ε^2 appears in the denominator of Δ_1 and Δ_2, as ε becomes smaller, matrices $X'X$ and R_{xx} become more ill-conditioned.

Two eigenvalues of $X'X$ are

$$\tfrac{1}{2}\{2(n-1)+\varepsilon^2\pm[4(n-1)+8(n-2)\varepsilon+4(n-3)\varepsilon^2+\varepsilon^4]^{\frac{1}{2}}\},$$

the remaining $n-3$ eigenvalues (for $n>3$) are all equal to ε^2. One eigenvalue of R_{xx} (for $n>3$) is

$$[(n-1)(n-2)-2(n-2)\varepsilon+2\varepsilon^2]/[n-1-2\varepsilon+(n-1)\varepsilon^2],$$

the remaining $n-3$ eigenvalues are all equal to

$$n\varepsilon^2/[n-1-2\varepsilon+(n-1)\varepsilon^2].$$

Example: $n=4$, $p=n-1=3$.

$$X=\begin{bmatrix} 1 & 1 & 1 \\ 1 & \varepsilon & 0 \\ 1 & 0 & \varepsilon \\ 1 & 0 & 0 \end{bmatrix}, \qquad \begin{aligned} y' &= (3+\varepsilon, \varepsilon, 3-\varepsilon), \\ \beta' &= (1,1,1), \\ \delta' &= (\varepsilon, -1, -1, 2-\varepsilon). \end{aligned}$$

Letting d_j denote the jth diagonal term of $(X'X)^{-1}$, we have

$$d_1=(2+\varepsilon^2)/(6-4\varepsilon+2\varepsilon^2),$$
$$d_2=d_3=(3-2\varepsilon+3\varepsilon^2)/[\varepsilon^2(6-4\varepsilon+2\varepsilon^2)].$$

Some results of running this modification of Läuchli's problem on several different algorithms are given in section 6.

6. Comparative results from two examples

The accuracy one obtains in solutions of least squares problems depends on several factors, including arithmetic precision, computational algorithm, and scaling of the data. The severity of ill-conditioning in some problems can be reduced by a judicious scaling of the data.

One quantity of special interest is the rank of the design matrix X. Lawson and Hanson (1974, pp. 77–78) have noted that the computed rank of a given matrix is not a unique property of the matrix, 'but also depends on other factors, such as the details of the computational algorithm, the value of tolerance parameters used in the computation, and the effects of machine round-off errors'.

We shall explore the effects of these various factors by running two types of problems on programs using different algorithms, different scalings, and different types of tolerances.

6.1. Longley's problem, using economic data

The data introduced by Longley (1967) to compare the performance of various least squares programs is the first example considered here. Six columns of the 16×7 matrix X were displayed in Longley's Table 1; the remaining column is a vector of ones. The vector y for this problem is 'Total

Derived Employment' of Longley's Table 2. The full-rank solution (regression coefficients), correct to at least 7 significant digits, appeared in his Table 10. A more precise 18-digit solution obtained with Newman's (1967) exact equation solver was given in Table II of Deegan (1976).

This problem has been discussed extensively by other authors, such as Chambers (1973, 1977), Farebrother (1974), Beaton et al. (1976), Deegan (1976), Dent and Cavander (1977), and Espasa (1977).

The computing described below was done on a Univac 1108 at the National Bureau of Standards using the 1100 Operating System, Version 33.R2 (Exec 8). Single precision arithmetic works with approximately 8 decimal digits and double precision with approximately 18 decimal digits. All calculations were done in single precision except for the double precision accumulation of inner products. Programs implementing five algorithms were used: Cholesky decomposition, Givens transformations, Householder transformations, modified Gram–Schmidt orthogonalization, and singular value decomposition. In the case of the Cholesky decomposition algorithm, solutions were sought in two different ways: (1) forming $X'X$ and $X'y$, inverting $X'X$, and multiplying $(X'X)^{-1}$ by $X'y$; (2) forming and inverting the matrix of correlation coefficients R_{xx} as described in section 2 above.

Sources of programs. The sources of the five programs are briefly described.

(1) *Cholesky decomposition.* Subroutines SYMINV and CHOL of Healy (1968) with modifications given by Farebrother and Berry (1974). The author's main program performed all calculations except matrix inversion, where SYMINV and CHOL were called. To obtain double precision accumulation of inner products, variables X in subroutine SYMINV and W in subroutine CHOL were declared to be double precision, and values assigned to these variables were computed in double precision.

(2) *Givens transformations.* The author's Fortran translation of Gentleman's (1974) Algol procedures were used, together with the author's main program. To obtain double precision inner products, variable DPRI ('d prime i' in the Algol procedure 'include') was declared to be double precision, and values assigned to it were computed in double precision.

(3) *Modified Gram-Schmidt orthogonalization.* Subroutine L2A of Wampler (to appear) and the author's main program were used. This algorithm includes as one of its features iterative refinement of the solution. In ill-conditioned problems, accuracy of solutions can often be improved in this way.

(4) *Householder transformations.* Lawson and Hanson's (1974) subroutine HFTI and main program PROG2. The author modified PROG2 to read data rather than calling a function to generate data.

(5) *Singular value decomposition.* Lawson and Hanson's (1974) subroutine SVDRS and main program PROG3. The author modified PROG3 to read data rather than calling a function to generate data.

Scaling of the data. The computational difficulties which arise in solving least squares problems can often be mitigated by scaling the data. A scaling known to be near-optimal [cf. Chapter 25 of Lawson and Hanson (1974), and Björck (1978)] is now described.

Let $Z=XD$, where D is a diagonal matrix whose jth diagonal element is the reciprocal of the Euclidean norm of the jth column of X. That is, let

$$d_{jj}=\left(\sum_{i=1}^{n} x_{ij}^2\right)^{-\frac{1}{2}}, \quad j=1,\ldots,p.$$

All columns of Z have unit Euclidean norm. If one replaces X by Z in the original least squares problem and computes

$$\hat{\alpha}=(Z'Z)^{-1}Z'y,$$

one can easily obtain $\hat{\beta}$ from $\hat{\beta}=D\hat{\alpha}$.

Dent and Cavander (1977), after computing the seven singular values for Longley's 16 × 7 matrix X, found the condition number of this matrix (the ratio of the largest singular value to the smallest) to be 0.485926×10^{10}. After converting Lawson and Hanson's singular value subroutine SVDRS to double precision, the author computed the singular values and the condition number for the matrix Z associated with the Longley data. The largest and smallest singular values of Z were 2.6194260 and 0.000060529715, so that the condition number is 43275.0. This is considerably less than the condition number of X.

Longley's problem was run on five programs with the original (unscaled) data and with the data scaled in the manner just described.

Tolerances. All five programs require the user to set a tolerance which is used in determining the rank of the matrix X and in detecting rank deficiencies. Several types of tolerances were used in these programs, and a description of them is now given. Some of these tolerances make use of a machine-dependent parameter η, the smallest positive real number such that $1.0+\eta>1.0$ in floating-point arithmetic. On the Univac 1108, we set $\eta=2^{-26}$. The results summarized in table 1 indicate the type of tolerance used as A, B, C, D or E.

Table 1

Longley's problem. Accuracy of coefficients, $\hat{\beta}$.[a]

		X not scaled tolerance = 0		X not scaled tolerance > 0		X scaled tolerance > 0	
Algorithm	Type of tolerance	Computed rank	Accurate digits	Computed rank	Accurate digits	Computed rank	Accurate digits
Cholesky, using $X'X$	A	7	1.9	6		6	
Cholesky, using R_{xx}	B	7	5.2	7	5.2	7	4.7
Givens transformations	A	7	4.7	6		6	
Modified Gram–Schmidt	C	7	4.5	6		7	4.9
Modified Gram–Schmidt, iterative refinement	C	7	7.8	6		7	7.8
Householder transformations	D(1)	7	5.2	5		7	4.8
	D(2)			6		7	4.8
Singular value decomposition	E	7	0.1	6		7	5.2

[a] Accuracy is defined according to the formula in the *Results* discussion of section 6.1.

(A) In the Cholesky decomposition algorithm one is working with a square positive semi-definite matrix $A = (a_{ij})$, from which the Cholesky factor $U = (u_{ij})$ is obtained such that U is upper triangular and $U'U = A$. The tolerance test examines if $|u_{ii}^2| > t_1$, where $t_1 = \eta|a_{ii}|$, for $i = 1, \ldots, p$. Whenever $|u_{ii}^2| \leq t_1$, u_{ii} is set equal to zero. The nullity of A is the number of u_{ii} that have been set equal to zero. Matrix A here is $X'X$.

The Givens transformation algorithm also computes the Cholesky factor U, although it works with X rather than $X'X$ in obtaining it. The tolerance test described in the above paragraph was also used in the Givens program. This was one of several tests suggested by Gentleman (1974) in connection with this algorithm.

(B) This tolerance test is like that of tolerance A, except that matrix A is now R_{xx}, the matrix of correlation coefficients, rather than $X'X$.

(C) In the modified Gram–Schmidt algorithm, a pivoting scheme is used in computing the Cholesky factor U so that the diagonal terms of U are obtained in decreasing order of magnitude. The tolerance test here checks if $u_{ii}^2 > t_2$, where $t_2 = p^2\eta^2 u_{11}^2$, for $i = 1, \ldots, p$.

(D) In the Householder transformation algorithm, the tolerance test checks if $|u_{ii}| > t_3$ for $i = 1, \ldots, p$. Lawson and Hanson (1974, p. 253) in describing the absolute tolerance parameter t_3, stated 'it is intended that the user set the value of this parameter to express information about the accuracy of the data rather than to express the machine's arithmetic precision'. These authors suggested that t_3 be set approximately equal to $\rho\|X\|$, where ρ is the relative uncertainty in the vector of observations y and $\|X\|$ is the spectral norm of X (equal to s_1, the largest singular value of X).

In the results summarized in table 1, tolerance D(1) indicates that the absolute tolerance parameter t_3 was set in the manner just described. The value of ρ here was obtained as $\rho = 0.5/(\bar{y}) = 0.5/65317 = 0.000007655$, where $\bar{y}$ denotes the mean of the 16 observations y_i. (It was assumed that the errors in the y_i were 0.5 in the least significant digit.) With $s_1 = 1663668.2$ in the case of unscaled data, one gets $t_3 = \rho s_1 = 12.74$. With $s_1 = 2.6194260$ in the case of scaled data, one gets $t_3 = \rho s_1 = 0.00002005$.

In table 1, tolerance D(2) indicates that the absolute tolerance parameter t_3 was based on tolerance C used in the modified Gram–Schmidt program. In C, we have $t_2 = p^2\eta^2 u_{11}^2 = 7^2(2^{-26})^2(0.2553 \times 10^{13}) = 0.02778$ in the case of unscaled data, and $t_2 = 7^2(2^{-26})^2(1.0) = 0.1088 \times 10^{-13}$ in the case of scaled data. The corresponding values of t_3 for tolerance D(2) are the square roots of the t_2-values for C, namely $t_3 = 0.167$ for unscaled data and $t_3 = 0.104 \times 10^{-6}$ for scaled data.

(E) In the singular value decomposition, the singular values of X are denoted by s_i, with $s_1 \geqq s_2 \geqq \ldots \geqq s_p$. The tolerance test here checks if $s_i > t_4$, where $t_4 = \eta s_1$, for $i = 1, \ldots, p$.

Results. The results of running Longley's problem on five different programs, with data both unscaled and scaled, are summarized in table 1. This table gives the computed rank of the system of equations and, in cases where the problem was found to be of full rank, the average accuracy of computed regression coefficients. The accuracy of each coefficient $\hat{\beta}_j$ was calculated as

$$a_j = -\log_{10}|(\beta_j - \hat{\beta}_j)/\beta_j|,$$

where β_j denotes the 'true' value of the coefficient (computed with no rounding error) and $\hat{\beta}_j$ denotes the value calculated by the algorithm using the NBS 1108 computer. Entries in the table, averages of seven a_j's, summarize the accuracy for each set of computed coefficients.

In the case of the unscaled data, the tolerance tests described earlier were modified so that the tolerance parameters (t_1, t_2, t_3, t_4) were set equal to zero. This was done merely to facilitate comparison of the five algorithms which use different types of tolerances. Setting tolerance parameters equal to zero is *not* recommended by any of the program authors, and the present author does not ordinarily recommend the use of such a zero tolerance.

Since the modified Gram–Schmidt algorithm was the only one which includes iterative refinement of the solution, the iterative feature was omitted in some runs in order to put the underlying algorithm on an equal footing with the others. Table 1 shows the accuracy obtained from this algorithm both with and without iterative refinement. In the problems covered in table 1, this algorithm converged to a solution after only one iteration beyond the initial solution.

We note that with tolerance parameters set equal to zero all algorithms obtained solutions, but accuracy in the case of the singular value decomposition was very poor – only 0.1 digits. The Cholesky algorithm using $X'X$ gave only 1.9 accurate digits – noticeably less than the 5.2 digits it gave with the use of R_{xx}.

With unscaled data and tolerances greater than zero, we note that all algorithms found the problem to be rank-deficient except the Cholesky algorithm using R_{xx}. This is not surprising since the computation of R_{xx} entails operations very similar to those described earlier in computing $Z = XD$. With the tolerance parameters of type A, C, D(2) and E, the computed rank was 6; with the larger tolerance of D(1), the computed rank was 5.

With scaled data, the computed rank was found to be 7 with all tolerances except type A. This tolerance is such that in order to obtain a computed rank of 7 one would need a slightly larger word-length than the 8 decimal digits available on the Univac 1108. Those algorithms which found the matrix to be of full rank and computed solutions all had similar accuracy – from 4.7 digits (Cholesky, R_{xx}) to 5.2 digits (singular value decomposition). The modified Gram–Schmidt algorithm with the iterative refinement feature included obtained almost full accuracy – 7.8 digits.

This example illustrates that some problems which are reported to be rank-deficient may have the rank deficiency removed by a suitable scaling.

6.2. *Modified Läuchli problem*

The modification of Läuchli's problem considered here was described at the end of section 5. Two parameters can be varied in this example: n and ε. The values of ε were chosen to be 10^{-k}, with $k=1,2,\ldots,8$; all values of n from 4 to 10 were used. Selected results are summarized in tables 2, 3 and 4 which show the accuracy obtained for regression coefficients, residuals, and diagonals of $(X'X)^{-1}$. The tabulated entries, covering $n=4$ and a subset of $n=10$, are typical of all the results obtained.

The algorithms used in this example are a subset of those used in the Longley problem: Cholesky, Householder, and modified Gram–Schmidt. The implementations of the algorithms were much the same as for the Longley problem, except that for the Householder transformation algorithm the author's Fortran translation of the Björck–Golub (1967) procedure was used, with iterative refinement omitted. All computing was done on a Univac 1108 using single precision arithmetic except in the case of modified Gram–Schmidt with iterative refinement, where double precision accumulation of inner products is essential. In the problems covered in tables 2 and 3, whenever solutions were obtained from the modified Gram–Schmidt algorithm with iterative refinement, convergence occurred after only one iteration.

The accuracy of computed quantities was measured in the same manner as for the Longley problem. There are some instances in tables 2, 3 and 4 where the number of accurate digits, a_j, is negative. This occurs, for example, in table 2, $n=4$, $k=5$, b_2, Householder algorithm, where the true coefficient is 1.0 and the computed value was 78.964866. This leads to $a_2=-\log_{10}|(1.0-78.964866)/1.0|=-1.9$.

The results of tables 2, 3 and 4 show that for this example the orthogonalization algorithms (i.e., Householder and modified Gram–Schmidt) obtained satisfactory solutions for a larger class of problems than did the Cholesky algorithm. Where solutions were obtained by all three algorithms, the least accurate results were usually given by Cholesky. Tables

Table 2
Modified Läuchli problem, $\varepsilon = 10^{-k}$. Accuracy of coefficients, $\hat{\beta}$.[a]

$n=4$		k							
Algorithm		1	2	3	4	5	6	7	8
Cholesky, using $X'X$	b_1	7.1	5.6	2.9	[b]	[b]	0.6	0.6	[b]
	b_2	5.7	7.8	7.8			0.0	0.0	
	b_3	7.8	7.8	1.8			0.0	0.0	
Cholesky, using R_{xx}	b_1	6.7	4.7	3.2	1.2	[b]	[c]	[b]	1.2
	b_2	6.4	4.4	2.9	0.9				0.9
	b_3	6.4	4.4	2.9	0.9				0.9
Householder transformations	b_1	7.3	7.8	7.5	7.5	5.9	5.5	3.0	[b]
	b_2	6.2	4.3	2.3	0.6	−1.9	−2.5	−5.2	
	b_3	6.1	4.3	2.3	0.6	−1.9	−2.5	−5.2	
Modified Gram–Schmidt	b_1	7.8	7.8	7.8	7.8	7.8	7.8	7.8	[b]
	b_2	6.9	5.3	3.2	3.8	3.9	2.3	1.4	
	b_3	6.9	5.3	3.2	3.8	3.9	2.3	1.4	
Modified Gram-Schmidt, iterative refinement	b_1	7.8	7.8	7.8	7.8	7.8	7.8	[d]	[b]
	b_2	7.8	7.8	7.8	4.8	6.5	4.6		
	b_3	7.8	7.8	7.8	4.8	6.6	4.6		
$n=10$		k							
Algorithm		1	2	3	4	5	6	7	8
Cholesky, using $X'X$	b_1	6.1	3.7	1.7	[b]	[b]	[b]	[b]	[b]
	b_3	5.0	2.7	1.0					
	b_6	5.2	3.1	1.3					
	b_9	5.1	7.8	0.7					
Cholesky, using R_{xx}	b_1	5.5	4.0	1.7	[b]	0.2	[b]	[b]	[b]
	b_3	5.0	3.5	0.9		−0.5			
	b_6	5.0	3.4	1.2		−0.4			
	b_9	6.2	3.5	1.2		−0.7			
Householder transformations	b_1	7.5	6.8	7.5	6.7	6.6	3.9	[b]	[b]
	b_3	6.3	4.8	3.3	0.6	−1.1	−3.5		
	b_6	6.6	4.6	3.4	0.6	−1.1	−3.5		
	b_9	5.9	4.6	2.0	0.6	−1.1	−3.5		
Modified Gram–Schmidt	b_1	7.2	7.3	7.2	7.1	7.2	7.2	[b]	[b]
	b_3	6.7	6.3	5.4	4.2	3.3	2.0		
	b_6	7.1	6.5	4.5	4.4	3.0	2.4		
	b_9	6.8	6.1	4.5	3.5	2.9	1.7		
Modified Gram–Schmidt, iterative refinement	b_1	7.8	7.8	7.8	7.8	7.8	7.8	[b]	[b]
	b_3	7.8	7.8	7.8	7.5	6.0	3.8		
	b_6	7.8	7.8	7.8	7.5	6.0	3.5		
	b_9	7.8	7.8	7.8	7.8	6.0	4.0		

[a] Accuracy is defined according to the formula in the *Results* discussion in section 6.1.
[b] Rank deficiency was reported.
[c] Matrix was reported to be not positive semi-definite.
[d] Algorithm failed to converge to a solution.

Table 3

Modified Läuchli problem, $\varepsilon = 10^{-k}$. Accuracy of residuals, δ.[a]

$n=4$		k							
Algorithm		1	2	3	4	5	6	7	8
Cholesky, using R_{xx}	δ_1	5.2	2.2	−0.3	−3.3	[b]	[c]	[b]	−7.3
	δ_2	6.8	4.7	3.2	1.2				1.2
	δ_3	6.8	4.7	3.2	1.2				1.2
	δ_4	7.0	5.0	3.5	1.5				1.5
Householder transformations	δ_1	7.3	6.0	5.7	4.8	2.8	2.8	1.0	[b]
	δ_2	7.0	6.2	5.3	4.6	3.1	3.5	1.8	
	δ_3	7.1	6.2	5.3	4.6	3.1	3.5	1.8	
	δ_4	7.8	7.8	8.1	8.1	6.7	7.5	4.1	
Modified Gram–Schmidt	δ_1	7.2	5.8	4.7	3.4	2.8	1.8	0.7	[b]
	δ_2	7.8	7.3	6.2	7.8	7.8	7.8	7.5	
	δ_3	7.5	7.3	6.2	7.8	7.8	7.8	7.8	
	δ_4	7.8	7.8	8.1	8.1	8.1	8.1	7.8	
Modified Gram–Schmidt iterative refinement	δ_1	7.8	7.9	7.8	5.0	7.9	7.8	[d]	[b]
	δ_2	7.8	7.8	7.8	7.8	7.8	7.8		
	δ_3	7.8	7.8	7.8	7.8	7.8	7.8		
	δ_4	7.8	8.1	8.1	7.8	7.8	7.8		

$n=10$		k							
Algorithm		1	2	3	4	5	6	7	8
Cholesky, using R_{xx}	δ_1	3.6	1.1	−2.2	[b]	−5.7	[b]	[b]	[b]
	δ_4	5.6	4.0	1.7		0.2			
	δ_7	5.6	4.0	1.7		0.2			
	δ_{10}	6.4	4.9	2.6		1.1			
Householder transformations	δ_1	6.5	6.5	4.4	3.2	3.0	1.7	[b]	[b]
	δ_4	7.2	6.7	6.3	4.6	3.9	2.5		
	δ_7	7.3	6.7	6.4	4.6	3.9	2.5		
	δ_{10}	7.6	8.1	7.8	7.5	7.5	5.2		
Modified Gram–Schmidt	δ_1	6.1	4.4	3.9	2.8	1.8	0.6	[b]	[b]
	δ_4	7.3	7.0	7.8	7.5	7.2	7.5		
	δ_7	7.5	7.0	7.8	7.2	7.1	7.5		
	δ_{10}	7.8	7.3	8.1	7.8	7.8	7.8		
Modified Gram–Schmidt, iterative refinement	δ_1	8.0	7.9	7.8	7.6	7.9	7.8	[b]	[b]
	δ_4	7.8	7.8	7.8	7.8	7.8	7.8		
	δ_7	7.8	7.8	7.8	7.8	7.8	7.8		
	δ_{10}	8.1	8.1	8.1	7.8	7.8	8.1		

[a] Accuracy is defined according to the formula in the *Results* discussion of section 6.1.

[b] Rank deficiency was reported.

[c] Matrix was reported to be not positive semi-definite.

[d] Algorithm failed to converge to a solution.

Table 4

Modified Läuchli problem, $\varepsilon = 10^{-k}$. Accuracy of diagonals of $(X'X)^{-1}$, d_j.[a]

$n=4$		k							
Algorithm		1	2	3	4	5	6	7	8
Cholesky, R_{xx}	d_1	6.6	5.1	2.8	1.2	[b]	[c]	[b]	1.2
	d_2	6.6	4.4	2.2	1.0				0.0
	d_3	6.6	4.4	2.2	1.0				0.0
Householder	d_1	7.2	5.9	2.9	1.5	−1.1	−2.9	0.6	[b]
	d_2	6.8	5.8	4.7	3.8	2.7	2.4	1.5	
	d_3	6.6	5.8	4.7	3.8	2.7	2.4	1.5	
Modified Gram–Schmidt	d_1	6.8	5.0	3.2	0.8	−1.1	0.6	0.6	[b]
	d_2	7.3	6.5	5.7	4.4	3.6	2.5	1.2	
	d_3	8.0	6.5	5.7	4.4	3.6	2.5	1.2	

$n=10$		k							
Algorithm		1	2	3	4	5	6	7	8
Cholesky, R_{xx}	d_1	6.0	6.3	2.8	[b]	0.6	[b]	[b]	[b]
	d_3	6.4	4.9	2.1		0.0			
	d_6	6.3	4.9	2.1		0.0			
	d_9	6.3	4.9	2.1		0.0			
Householder	d_1	7.8	4.5	3.5	0.3	[d]	[d]	[b]	[b]
	d_3	7.2	6.3	5.5	4.3	3.5	2.4		
	d_6	8.0	6.4	5.5	4.3	3.5	2.4		
	d_9	7.4	6.4	5.2	4.3	3.5	2.4		
Modified Gram–Schmidt	d_1	6.4	4.5	2.5	1.2	−1.0	−2.9	[b]	[b]
	d_3	7.3	7.0	5.8	5.5	4.1	2.9		
	d_6	7.7	7.2	5.8	5.5	4.1	2.9		
	d_9	7.2	7.2	5.8	5.5	4.1	2.9		

[a] Accuracy is defined according to the formula in the *Results* discussion of section 6.1.
[b] Rank deficiency was reported.
[c] Matrix was reported to be not positive semi-definite.
[d] Diagonal term was computed to be negative.

2 and 3 illustrate the effectiveness of iterative refinement in improving the accuracy of coefficients and residuals in ill-conditioned problems.

References

Abdelmalek, Nabih N., 1971, Round off error analysis for Gram–Schmidt method and solution of linear least squares problems, BIT 11, 345–367.

Ascher, Marcia and George E. Forsythe, 1958, SWAC experiments on the use of orthogonal polynomials for data fitting, Journal of the Association for Computing Machinery 5, 9–21.

Bauer, F.L., 1965, Elimination with weighted row combinations for solving linear equations and least squares problems, Numerische Mathematik 7, 338–352.

Beaton, Albert E., Donald B. Rubin and John L. Barone, 1976, The acceptability of regression solutions: Another look at computational accuracy, Journal of the American Statistical Association 71, 158–168.

Berk, Kenneth, 1977, Tolerance and condition in regression computations, Journal of the American Statistical Association 72, 863–866.

Björck, Ake, 1967, Solving linear least squares problems by Gram–Schmidt orthogonalization, BIT 7, 1–21.

Björck, Ake, 1968, Iterative refinement of linear least squares solutions, II, BIT 8, 8–30.

Björck, Ake, 1978, Comment on the iterative refinement of least squares solutions, Journal of the American Statistical Association 73, 161–166.

Björck, Ake and Gene Golub, 1967, ALGOL Programming, Contribution No. 22: Iterative refinement of linear least square solutions by Householder transformation, BIT 7, 322–337.

Björck, Ake and Gene H. Golub, 1973, Numerical methods for computing angles between linear subspaces, Mathematics of Computation 27, 579–594.

Boehm, William T., D.J. Menkhaus and J.B. Penn, 1976, Accuracy of least squares computer programs: Another reminder, American Journal of Agricultural Economics 58, no. 4, I, 757–760.

Bright, J.W. and G.S. Dawkins, 1965, Some aspects of curve fitting using orthogonal polynomials, Industrial and Engineering Chemistry Fundamentals 4, 93–97.

Businger, Peter and Gene H. Golub, 1965, Linear least squares by Householder transformations, Numerische Mathematik 7, 269–276.

Cameron, J.M., 1957, Some examples of the use of high speed computers in statistics, Proceedings of the First Conference on the Design of Experiments in Army Research, Development and Testing, Office of Ordnance Research Report no. 57-1, 129–135.

Chambers, John M., 1971, Regression updating, Journal of the American Statistical Association 66, 744–748.

Chambers, John M., 1973, Linear regression computations: Some numerical and statistical aspects, Bulletin of the International Statistical Institute (Proceedings of the 39th Session) 45, no. 4, 245–254.

Chambers, John M., 1977, Computational methods for data analysis (Wiley, New York).

Dahlquist, Germund and Ake Björck, 1974, Numerical methods (Prentice-Hall, Englewood Cliffs, NJ).

Daniel, J.W., W.B. Gragg, L. Kaufman and G.W. Stewart, 1976, Reorthogonalization and stable algorithms for updating the Gram–Schmidt QR factorization, Mathematics of Computation 30, 772–795.

Deegan, Jr., John, 1976, A test of the numerical accuracy of some matrix inversion algorithms commonly used in least squares programs, Journal of Statistical Computation and Simulation 4, 269–278.

Dent, Warren T. and David C. Cavander, 1977, More on computational accuracy in regression, Journal of the American Statistical Association 72, 598–600.

Espasa, Antoni, 1977, A note on the acceptability of regression solutions: Another look at computational accuracy, Journal of the American Statistical Association 72, 603.

Farebrother, R.W., 1974, Algorithm AS 79: Gram–Schmidt regression, Applied Statistics 23, 470–476.

Farebrother, R.W. and G. Berry, 1974, Remark AS R12: A remark on Algorithm AS 6: Triangular decomposition of a symmetric matrix, Applied Statistics 23, 477.

Fletcher, R.H., 1975, On the iterative refinement of least squares solutions, Journal of the American Statistical Association 70, 109–112.

Frane, James W., 1977, A note on checking tolerance in matrix inversion and regression, Technometrics 19, 513–514.

Freund, R.J., 1963, A warning of roundoff errors in regression, The American Statistician 17, no. 5, 13–15.

Gentleman, W. Morven, 1974, Algorithm AS 75: Basic procedures for large, sparse or weighted linear least squares problems, Applied Statistics 23, 448–454.

Gentleman, W. Morven, 1975, Row elimination for solving sparse linear systems and least squares problems, in: G.A. Watson, ed., Numerical analysis, Proceedings of the Dundee Conference on Numerical Analysis (Springer-Verlag, Berlin/Heidelberg/New York) 122–133.

Gerig, Thomas M. and A. Ronald Gallant, 1975, Computing methods for linear models subject to linear parametric constraints, Journal of Statistical Computation and Simulation 3, 283–296.

Golub, G., 1965, Numerical methods for solving linear least squares problems, Numerische Mathematik 7, 206–216.

Golub, G.H. and C. Reinsch, 1970, Singular value decomposition and least squares solutions, Numerische Mathematik 14, 403–420.

Golub, G.H. and J.H. Wilkinson, 1966, Note on the iterative refinement of least squares solution, Numerische Mathematik 9, 139–148.

Gregory, Robert T. and David L. Karney, 1969, A collection of matrices for testing computational algorithms (Wiley–Interscience, New York).

Hastings, W. Keith, 1972, Test data for statistical algorithms: Least squares and ANOVA, Journal of the American Statistical Association 67, 874–879.

Healy, M.J.R., 1968, Algorithm AS 6: Triangular decomposition of a symmetric matrix: Algorithm AS 7: Inversion of a positive semi-definite symmetric matrix, Applied Statistics 18, 195–199; 1969, Corrigenda, Applied Statistics 18, 118.

Hocking, Ralph T., 1970, Analysis of factors using multiple regression, Journal of Systems Management 21, 29–34.

Holland, Paul W., 1976, Robust test problems for robust regression programs, Proceedings of the Ninth Interface Symposium on Computer Science and Statistics (Prindle, Weber and Schmidt, Boston, MA) 99–110.

Horiba, Y., 1971, A note on an efficient computation of the standard error of the constant term in multiple regression analysis (Letter to the Editor), The American Statistician 25, no. 4, 51–52.

Jennings, L.S. and M.R. Osborne, 1974, A direct error analysis for least squares, Numerische Mathematik 22, 325–332.

Jordan, T.L., 1968, Experiments on error growth associated with some linear least squares procedures, Mathematics of Computation 22, 579–588.

Kenkel, J.L., 1976, Some methods for examining rounding error in a least squares regression computer program, Journal of Economics and Business 28, 104–110.

Krane, Scott A., 1970, Letter to the Editor, The American Statistician 24, no. 4, 46.

Krane, Scott A., 1971, Letter to the Editor, The American Statistician 25, no. 3, 52 (an article by Hocking, 1970, is discussed.)

Läuchli, Peter, 1961, Jordan-Elimination und Ausgleichung nach kleinsten Quadraten, Numerische Mathematik 3, 226–240.

Lawson, Charles L. and Richard J. Hanson, 1974, Solving least squares problems (Prentice-Hall, Englewood Cliffs, NJ).

Longley, James W., 1967, An appraisal of least squares programs for the electronic computer from the point of view of the user, Journal of the American Statistical Association 62, 819–841.

Longley, James W., 1976, A subroutine for singular value decomposition by normalized modified Gram–Schmidt algorithm for the solution of linear least squares equations, Proceedings of the Statistical Computing Section (American Statistical Association, Washington, DC) 207–212.

Maindonald, J.H., 1977, Least squares computations based on the Cholesky decomposition of the correlation matrix, Journal of Statistical Computation and Simulation 5, 247–258.

Macdonald, J. Ross, 1964, Accelerated convergence, divergence, iteration, extrapolation, and curve fitting, Journal of Applied Physics 35, 3034–3041.

Muhonen, Daniel P., 1968, An experimental comparison of several approaches to the linear least squares problem, Report no. TR-01-1004 (Bendix Field Engineering Corporation, prepared under Contract NAS5-10750 for National Aeronautics and Space Administration, Goddard Space Flight Center, Greenbelt, MD).

Mullet, Gary M. and Tracy W. Murray, 1971, A new method for examining rounding error in least-squares regression computer programs, Journal of the American Statistical Association 66, 496–498.

Nash, J.C. and L.P. Lefkovitch, 1976, Principal components and regression by singular value decomposition on a small computer, Applied Statistics 25, 210–216.

Newman, Morris, 1967, Solving equations exactly, Journal of Research of the National Bureau of Standards 71B, 171–179.

Peters, G. and J.H. Wilkinson, 1970, The least squares problem and pseudo-inverses, The Computer Journal 13, 309–316.

Plemmons, Robert J., 1974, Linear least squares by elimination and MGS, Journal of the Association for Computing Machinery 21, 581–585.

Seaman, Jeff, 1977, Use of computer resources by statistical package regression programs, Proceedings of the Statistical Computing Section (American Statistical Association, Washington, DC).

Seber, G.A.F., 1977, Linear regression analysis (Wiley, New York).

Shampine, L.F., 1975, Discrete least squares polynomial fits, Communications of the ACM 18, 179–180.

Stewart, G.W., 1973, Introduction to matrix computations (Academic Press, New York).

Stoer, Josef, 1971, On the numerical solution of constrained least-squares problems, SIAM Journal on Numerical Analysis 8, 382–411.

Tsao, Nai-Kuan, 1975, A note on implementing the Householder transformation, SIAM Journal on Numerical Analysis 12, 53–58.

Velleman, Paul F. and Ivor Francis, 1975, Measuring statistical accuracy of regression programs, Proceedings of the Computer Science and Statistics Eighth Annual Symposium on the Interface (University of California, Los Angeles, CA) 122–127.

Velleman, Paul F., Jeff Seaman and I. Elaine Allen, 1977, Evaluating package regression routines, Proceedings of the Statistical Computing Section (American Statistical Association, Washington, DC) 82–92.

Wampler, Roy H., 1969, An evaluation of linear least squares computer programs, Journal of Research of the National Bureau of Standards 73B, 59–90.

Wampler, Roy H., 1970, A report on the accuracy of some widely used least squares computer programs, Journal of the American Statistical Association 65, 549–565.

Wampler, Roy H., 1979, Solutions to weighted least squares problems by modified Gram–Schmidt with iterative refinement, to appear in ACM Transactions on Mathematical Software.

Zellner, A. and H. Thornber, 1966, Computational accuracy and estimation of simultaneous equation econometric models, Econometrica 34, 727–729.