COMPUTATION OF THE LIMITED INFORMATION MAXIMUM LIKELIHOOD ESTIMATOR

BY

WARREN DENT

GENE H. GOLUB

STAN-CS-73-339

FEBRUARY 1973

COMPUTER SCIENCE DEPARTMENT
School of Humanities and Sciences
STANFORD UNIVERSITY

COMPUTATION OF THE LIMITED INFORMATION MAXIMUM LIKELIHOOD ESTIMATOR

by

Warren Dent and Gene H. Golub*

The University of Iowa and Stanford University

*The work of this author was supported in part by the AEC and by NSF.

COMPUTATION OF THE LIMITED INFORMATION MAXIMUM LIKELIHOOD ESTIMATOR

by

Warren Dent and Gene H. Golub

The University of Iowa and Stanford University

0. Abstract

Computation of the Limited Information Maximum Likelihood Estimator (LIMLE) of the set of coefficients in a single equation of a system of interdependent relations is sufficiently complicated to detract from other potentially interesting properties. Although for finite samples the LIMLE has no moments [18], asymptotically it remains normally distributed [2] and retains other properties associated with maximum likelihood. The most extensive application of the estimator has been made in the Brookings studies [7]. We believe that current methods of estimation are clumsy, and present a numerically stable estimation schema based on Householder transformations and the singular value decomposition. The analysis permits a convenient demonstration of equivalence with the Two Stage Least Squares Estimator (TSLSE) in the instance of just identification.

1. Introduction

In a system of interdependent relations, suppose a given structural equation is denoted by

$$Y\gamma + (X_1, X_2)\begin{pmatrix}\beta \\ \underset{\sim}{0}\end{pmatrix} + \underset{\sim}{u} = \underset{\sim}{0}.$$

The n x (L+1) matrix Y = (Y*, y) represents n observations on L endogenous variables Y* and n observations on an endogenous variable y elected as subject of the equation. X_1 is the n x K_1 matrix of n observations on the K_1 included exogenous variables, X_2 is the n x K_2 matrix of n observatic on the K_2 excluded exogenous variables, and u is an n x 1 vector of disturbances. $X = (X_1, X_2)$ and is of order n x K, where $K = K_1 + K_2$. γ and β are unknown coefficient vectors, save for the last element of γ which is -1, whence $\gamma' = (\gamma^{*\prime}, -1)$.

Assuming X_1 and X of full rank, define the two residual operators

$$M_1 = I - X_1(X_1'X_1)^{-1}X_1',$$
$$M = I - X(X'X)^{-1}X'.$$

The LIMLE is then determined by solving the determinantal equation

$$|Y'M_1Y - \mu Y'MY| = 0$$

for the smallest value of μ, say $\hat{\mu}$. The corresponding solution for $\hat{\gamma}$, with last element -1, in

$$(Y'M_1Y - \hat{\mu}Y'MY)\ \hat{\gamma} = 0$$

is the LIMLE of γ, with the LIMLE of β being

$$\hat{\underset{\sim}{\beta}} = -(X_1'X_1)^{-1}X_1'Y\hat{\underset{\sim}{\gamma}}$$

Derivation of this solution may be found, for example, in [3], [14, pp. 166-173], [6, pp. 335-344], [20, pp. 500-503, pp. 679-686], and [9, pp. 38-44]. The LIMLE is sometimes identified with the Least Variance Ratio Estimator (LVRE), and Least Generalized Residual Variance Estimator (LGRVE), or the Smallest Canonical Correlation Estimator (SCCE). Because of the simplicity of its derivation the LVRE is usually presented in texts, for example [16, pp. 384-387], [15, pp. 166-173], [6, p. 346], [10, p. 338], [17, pp. 567-571], [5, pp. 411-424, pp. 663-666], and [9 , pp. 45, 46]. The LGRVE is proposed in [3], and [10, pp. 338-341], while the SCCE is derived in [4] and [1]. The four estimators are not necessarily identical at all times, since under certain conditions (to be discussed below) on the number of available observations, one may exist while another may not. This confusion has led to some meaningless statements in certain texts, especially with respect to the equivalence of the TSLSE and LIMLE in cases of just-identification.

In the following section we present the computational schema for the LIMLE, while in section 3 we discuss the TSLSE and in section 4 present the determination of the asymptotic variance matrix of these estimators.

2. Algorithm for the LIMLE

There exists an orthogonal n x n matrix Q, the product of $K \leq n$ Householder transformations [12] such that, for

$$Q = (Q_1, Q_2, Q_3),$$

where Q_1 is n x K_1; Q_2 n x K_2; Q_3 n x (n-K), Q annihilates X as

$$Q'X = \begin{bmatrix} R_1 & R_3 \\ 0 & R_2 \\ 0 & 0 \end{bmatrix}$$

where R_1 is an upper-triangular non-singular K_1 x K_1 matrix, R_2 is similarly upper-triangular non-singular K_2 x K_2, R_3 is K_1 x K_2, and the zero matrices have appropriate order. Clearly

$$X_1 = Q_1R_1$$

and

$$X = Q_4R$$

where

$$Q_4 = (Q_1, Q_2),$$

$$R = \begin{bmatrix} R_1 & R_3 \\ 0 & R_2 \end{bmatrix}.$$

Further, it is readily verified that

$$M = I - Q_4Q_4'$$

$$= Q_3Q_3'$$

and

$$M_1 = I - Q_1Q_1'$$

$$= Q_2Q_2' + Q_3Q_3'.$$

Suppose the transformation Q applied to Y yields

$$Q'Y = \begin{bmatrix} Q_1'Y \\ Q_2'Y \\ Q_3'Y \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \end{bmatrix} = (Q'Y^*, Q'y) = \begin{bmatrix} Z_1^*, & z_1 \\ Z_2^*, & z_2 \\ Z_3^*, & z_3 \end{bmatrix}$$

such that matrices with subscript 1 have K_1 rows, while subscript 2 indicates K_2 rows, and subscript 3 indicates (n-K) rows. An asterisk denotes a matrix with L columns, and the partition of Z_1, Z_2, Z_3 corresponds to that of $Y = (Y^*, y)$. Alternately,

$$Y = \sum_{i=1}^{3} Q_iZ_i$$

$$Y^* = \sum_{i=1}^{3} Q_i Z_i^*,$$

and

$$Y'MY = Z_3'Z_3,$$

$$Y'M_1Y = Z_2'Z_2 + Z_3'Z_3.$$

The determinantal equation for $\hat{\mu}$ now becomes

$$|Z_2'Z_2+Z_3'Z_3-\mu Z_3'Z_3| = 0$$

or

$$|Z_2'Z_2-(\mu-1)Z_3'Z_3| = 0 .$$

If the particular equation is over-identified, $K_2 > L$ or $K_2 \geq L+1$ so that Z_2, which is $K_2 \times (L+1)$, has full column rank with probability one, and the inverse $(Z_2'Z_2)^{-1}$ exists with probability one. We therefore consider

$$|(Z_2'Z_2)^{-1}Z_3'Z_3-\frac{1}{\mu-1} I| = 0$$

and search for the largest eigenvalue of $(Z_2'Z_2)^{-1}Z_3'Z_3$.

Assuming Z_2 has full rank L+1 and since $K_2 \geq L+1$, there exists an orthogonal $K_2 \times K_2$ matrix H, the product of L+1 Householder transformations, such that

$$H'Z_2 = \begin{pmatrix} G \\ 0 \end{pmatrix}$$

$$Z_2 = H \begin{pmatrix} G \\ 0 \end{pmatrix} = (H_1, H_2)\begin{pmatrix} G \\ 0 \end{pmatrix} = H_1G$$

where G is an upper-triangular, non-singular L+1 by L+1 matrix, and H_1 is $K_2 \times (L+1)$. It follows that

$$Z_2'Z_2 = G'G$$

and

$$(Z_2'Z_2)^{-1} = G^{-1}(G')^{-1}$$

The matrix G^{-1} may be readily computed since G is upper-triangular, see e.g. [19 p. 3 1 , [8 , p. 427]. If, now, $n \geq K+L+1$, consider the singular value decomposition [13,14] of Z_3G^{-1} as

$$Z_3G^{-1} = U \Delta V'$$

where U is $(n-K) \times (L+1)$, Δ and V are $(L+1) \times (L+1)$, Δ is diagonal and

$$U'U = I_{L+1} = V'V = VV'.$$

Obviously

$$(G^{-1})'Z_3'Z_3G^{-1} = V\Delta^2V'$$

or

$$(G^{-1})'Z_3'Z_3G^{-1}V = V\Delta^2$$

$$= (\delta_1^2 \underset{\sim}{v}_1, \delta_2^2 \underset{\sim}{v}_2, \ldots \delta_{L+1}^2 \underset{\sim}{v}_{L+1})$$

where $\delta_1^2 \geq \delta_2^2 \geq \ldots \geq \delta_{L+1}^2$, say, are the diagonal elements of Δ^2, and $\underset{\sim}{v}_i$ is the column vector of V corresponding to δ_i^2. Since

$$G^{-1}(G^{-1})'Z_3'Z_3G^{-1}\underset{\sim}{v}_1 = \delta_1^2 G^{-1}\underset{\sim}{v}_1,$$

δ_1^2 is the eigenvalue sought, and we define

$$\hat{\hat{\gamma}} = G^{-1}\underset{\sim}{v}_1.$$

The LIMLE of γ is given by

$$\underset{\sim}{\hat{\gamma}} = -\underset{\sim}{\hat{\hat{\gamma}}}/\hat{\hat{\gamma}}_{L+1}$$

where $\hat{\hat{\gamma}}_{L+1}$ is the last component of $\hat{\hat{\gamma}}$.

Since

$$(X_1'X_1)^{-1}X_1' = R_1^{-1}Q_1' ,$$

$$\hat{\beta} = -(X_1'X_1)^{-1}X_1'Y\underset{\sim}{\hat{\gamma}}$$

$$= -R_1^{-1}Q_1'(\sum_{i=1}^{3}Q_iZ_i)\underset{\sim}{\hat{\gamma}}$$

$$= -R_1^{-1}Z_1\underset{\sim}{\hat{\gamma}}.$$

When $K \leq n < K+L+1$ the singular value decomposition of $(G^{-1})'Z_3' = U\Delta V'$ is used with

$$\hat{\gamma} = G^{-1}\underset{\sim}{u}_1$$

where $\underset{\sim}{u}_1$ is the eigenvector corresponding to the largest value δ_1^2 of δ_i^2 $i=1, \ldots, L+1$; and $\hat{\gamma}$ is $\hat{\hat{\gamma}}$ normalized appropriately. In either case

$$\hat{\mu} = 1+\delta_1^{-2}.$$

3. The LIMLE and TSLSE

When the given structural equation is just-identified, $K_2 = L$. Consider the difference

$$Y'M_1Y - Y'MY = Z_2'Z_2.$$

Since Z_2 has less rows (K_2) than columns (L+1), $Z_2'Z_2$ is necessarily singular and

$$|Y'M_1Y - Y'MY| = 0$$

implying $\mu = 1$, which provides the case of the TSLSE, [3], [20, p. 504].

In [5, p. 424] the equivalence of the TSLSE and LIMLE is claimed by assuming $Y'M_1Y = Y'MY$ ($W^* = W$), which obviously need not be true. Indeed there seems to be general confusion as to the behavior of these two residual moment matrices. In [6, p. 339] $Y'M_1Y$ and $Y'MY$ (W_{11} and W^*_{11}) are claimed to be positive definite, while in [15, p. 172] $Y'MY$ ($W_{\Delta\Delta}$) is claimed to be positive definite. Since $Y'MY = Z_3'Z_3$, a necessary condition for non-singularity, with probability one, is that Z_3 have at least as many rows as columns, or that $n \geq K+L+1$. In the derivation so far we have only required $n \geq K$. (This includes derivation of the classical determinantal equation).

Clearly, when $K \leq n < K+L+1$ the LGRVE does not exist (since it is derived through the minimization of a determinant involving $(Y'MY)^{-1}$). Hence equivalence with the TSLSE [10, p. 344] which exists for $n \geq K_1+L+1$ may be impossible in certain instances. Even the LVRE is of spurious interpretation in such instances since the denominator of the ratio being minimized is positive semi-definite and hence may assume a zero value.

For a range of n values (L+1 of them) the estimators LVRE, LGRVE, and SCCE may not exist, while the LIMLE will. If the number of included endogenous variables is large for a particular relation in an interdependent system, this will define an equivalent number of observation values over which the LIMLE alone will exist.

4. Asymptotic Covariance

The asymptotic variance-covariance matrix of $(\hat{\gamma}^{*\prime}, \hat{\beta}')$ is given by the two-stage least squares asymptotic covariance, or by

$$W = s^2 \begin{bmatrix} Y^{*\prime}(I-M)Y^* & Y^{*\prime}X_1 \\ X_1'Y^* & X_1'X_1 \end{bmatrix}^{-1}$$

where

$$s^2 = \frac{1}{n} (\underset{\sim}{y} - Y^*\hat{\underset{\sim}{\gamma}}{}^*_o - X_1\hat{\underset{\sim}{\beta}}_o)'(\underset{\sim}{y} - Y^*\hat{\underset{\sim}{\gamma}}{}^*_o - X_1\hat{\underset{\sim}{\beta}}_o)$$

and $\hat{\underset{\sim}{\beta}}_o$ is the two-stage least squares estimate of $\underset{\sim}{\beta}$, while $\hat{\underset{\sim}{\gamma}}{}^*_o$ consists of the first L components of the two-stage least squares estimate of y. The normal equations for the two-stage least squares estimators are

$$\begin{bmatrix} Y^{*\prime}(I-M)Y^* & Y^{*\prime}X_1 \\ X_1'Y^* & X_1'X_1 \end{bmatrix} \begin{bmatrix} \underset{\sim}{\gamma}{}^*_o \\ \underset{\sim}{\beta}_o \end{bmatrix} = \begin{bmatrix} Y^{*\prime}(I-M)\underset{\sim}{y} \\ X_1'\underset{\sim}{y} \end{bmatrix}$$

or

$$\begin{bmatrix} Z_1^{*\prime}Z_1^* + Z_2^{*\prime}Z_2^* & Z_1^{*\prime}R_1 \\ R_1'Z_1^* & R_1'R_1 \end{bmatrix} \begin{bmatrix} \underset{\sim}{\gamma}{}^*_o \\ \underset{\sim}{\beta}_o \end{bmatrix} = \begin{bmatrix} Z_1^{*\prime}\underset{\sim}{z}_1 + Z_2^{*\prime}\underset{\sim}{z}_2 \\ R_1'\underset{\sim}{z}_1 \end{bmatrix}$$

We now apply Householder transformations as

$$T' \begin{bmatrix} Z_1^* & R_1 & \underset{\sim}{z}_1 \\ Z_2^* & 0 & \underset{\sim}{z}_2 \end{bmatrix} = \begin{bmatrix} S & \underset{\sim}{t}_1 \\ 0 & \underset{\sim}{t}_2 \end{bmatrix}$$

reducing the first $L+K_1$ columns to upper triangular form S, whence

$$\begin{bmatrix} \hat{\underset{\sim}{\gamma}}_o{}^* \\ \hat{\underset{\sim}{\beta}}_o \end{bmatrix} = S^{-1}\underset{\sim}{t}_1 \quad .$$

s^2 may now be computed from above and the asymptotic covariance matrix found as

$$s^2\ (S'S)^{-1}.$$

A FORTRAN program listing follows.

Acknowledgement

We wish to thank Mrs. Margaret Wright, at Stanford, for programming the algorithm described in this paper and performing the initial calculations used for checking purposes.

REFERENCES

1. Anderson, T. W. "Estimating Linear Restrictions on Regression Coefficients for Multivariate Normal Distributions," AMS, 1957, Vol. 22, pp. 327-351.

2. Anderson, T. W. "An Asymptotic Expansion of the Distribution of the Maximum Likelihood Estimate of a Coefficient in a Simultaneous Equations System." Presented at the IMS meetings, Montreal, August 1972.

3. Anderson, T. W. and Rubin, H. "Estimation of the Parameters of a Single Equation in a Complete System of Stochastic Equations," AMS, 1949, Vol. 20, pp. 46-63.

4. Bartlett, M. S. "A Note on the Statistical Estimation of Demand and Supply Relationships From Time Series," Econometrica, 1948, Vol. 16, pp. 323-329.

5. Christ, C. F. Econometric Models and Methods, John Wiley & Sons, Inc., New York, 1966.

6. Dhrymes, P. Econometrics, Harper & Row, New York, 1970.

7. Duesenberry, J. S., Fromm, G., Klein, L. R., and Kuh, E. (eds). The Brookings Quarterly Econometric Model of the United States, Rand McNally, Chicago, 1965.

8. Durbin, J. "An Alternative to the Bounds Test for Testing for Serial Correlation in Least-Squares Regression," Econometrica, 1970, Vol. 38, pp. 422-429.

9. Fisk, P. R. Stochastically Dependent Equations, Hafner, New York, 1967.

10. Goldberger, A. S. Econometric Theory, John Wiley & Sons, Inc., New York, 1964.

11. Goldberger, A. S. and Olkin, I. "A Minimum Distance Interpretation of Limited-Information Estimation," Econometrica, 1971, Vol. 39, pp. 635-639.

12. Golub, Gene H. "Numerical Methods for Solving Linear Least-Squares Problems," Numer. Math., 1965, Vol. 7, pp. 206-216.

13. Golub, Gene H. and Kahan, W. "Calculating the Singular Values and Pseudo-inverse of a Matrix," J. Siam Numer. Analys., Ser. B. 1965, Vol. 2. pp. 205-224.

14. Golub, Gene H. and Reinsch, C. "Singular Value Decomposition and Least Squares Solutions." Numer. Math., 1970, Vol. 14, pp. 403-420.

15. Hood, W. C., Koopmans, T. C. (eds). Studies in Econometric Methods, Cowles Foundation Monograph No. 14, John Wiley & Sons, Inc., New York, 1953.

16. Johnston, J. Econometric Methods, McGraw-Hill, New York, 2nd ed., 1972.

17. Kmenta, J. Elements of Econometrics, MacMillan, New York, 1971.

18. Mariano, R. S., and Sawa, T. "Exact Finite-Sample Distribution of Limited-Information Maximum Likelihood Estimator for Two Included Endogenous Variables," JASA, 1972, Vol. 67, pp. 159-163.

19. Plackett, R. L. Principles of Regression Analysis, Oxford University Press, 1960.

20. Theil, H. Principles of Econometrics, John Wiley & Sons, Inc., 1971.

```
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XY(25,25),S(25),U(25,25),V(25,25)
      REAL*4 HEAD(20),FMT(20)
      DIMENSION GINV(25,25)
      DIMENSION GAMMA(25),BETA(25),Z3GINV(25,25)
      DIMENSION XXY(625)
      EQUIVALENCE (XY(1,1),XXY(1))
      READ 10,NPROB
      DO 500 NRUN=1,NPROB
      PRINT 726
  726 FORMAT(1H1)
C
      READ 7776, HEAD
 7776 FORMAT(20A4)
      READ 10,N,K1,K2,L
 10   FORMAT(16I5)
C  N IS NUMBER OF ROWS IN ALL MATRICES
C K1,K2, L ARE THE NUMBERS OF COLUMNS IN X1,X2,Y*, RESPECTIVELY
C
      READ 11, FMT
   11 FORMAT(20A4)
C
      KSUM=K1+K2+L+1
C KSUM IS NUMBER OF COLUMNS IN FULL MATRIX XY
C THE MATRIX HAS THE FORM
C          (  X1    X2    Y)
C
C
      NDIM=25
C NUMBER OF ROWS DECLARED
      K=K1+K2
      LP1=L+1
      KP1=K+1
      NMK1=N-K1
C
C
C
C  READ THE XY MATRIX FROM CARDS, BY ROWS, IN THE ORDER
C
C                            (  X1      X2     Y  )
C
      DO 100 I=1,N
      READ (5,FMT) (XY(I,J),J=1,KSUM)
100   CONTINUE
C
C  HEAD IS AN ALPHANUMERIC TITLE TO DESIGNATE THE GIVEN RUN
      PRINT 7775,HEAD
7775  FORMAT(1H1,20A4)
      PRINT 7773, N, K1, K2, L
7773  FORMAT(1H0, 'N=',I5, 2X, 'K1=', I5,2X,'K2=',I5,2X,'L=',I5)
      PRINT 7772
7772  FORMAT(1H0, 'XY  MATRIX' )
      DO 602 I=1,N
      PRINT 601,(XY(I,J),J=1,KSUM)
602   CONTINUE
```

```
      IF(K1+L.EQ.K) GO TO 1091
      IF(K1+L.GT.K) GO TO 1092
      PRINT 640
C
C HCOMP1 REDUCES THE FIRST K COLUMNS OF THE  RECTANGULAR
C  MATRIX XY TO UPPER TRIANGULAR  FORM  AND APPLIES THE RESULTING
C  HOUSEHOLDER TRANSFORMATIONS TO THE REMAINING COLUMNS.
      CALL HCOMP1(NDIM,N, K, KSUM,XY,S)
601   FORMAT(1H0,8D16.6)
640   FORMAT(1H0,/,/)
      DO 667 I=1,K2
      DO 667 J=1,LP1
  667 XY(I+K1,J+K1)=XY(I+K1,J+K)
C
C
C  NOW COMPUTE HOUSEHOLDER REFLECTIONS TO REDUCE Z2, A K2 BY L+1
C  MATRIX WHOSE FIRST LOCATION IS XY(K1+1,K+1), TO UPPER
C  TRIANGULAR  FORM
      CALL HCOMP1(NDIM,K2,LP1,LP1,XY(K1+1,KP1),S)
C
C  CALL  THE RESULTING  SQUARE  TRIANGULAR  MATRIXG,WHEREGIS
C  (L+1) BY (L+1).
      CALL  TNVRT2( LP1,XY(K1+1,KP1),GINV,NDIM,
     1  &220)
C
C FORM PRODUCT OF Z3* GINVERSE
      NMK=N-K
      DO 200 I =1,NMK
      DO 200 J=1,LP1
      SUM=0.0D0
   INDEX1 IS INDEX IN XY OF LOCATION PRECEDING FIRST ELEMENT OF Z3
      INDEX1= K*NDIM + K
      DO 300 KK=1,LP1
      IND1= INDEX1 + (KK-1)*NDIM +I
      SUM = SUM + XXY(IND1)*GINV(KK,J)
C  Z3(I,KK)*  GINVERSE(KK,J)
300   CONTINUE
      Z3GINV(I,J)=SUM
200   CONTINUE
      NW1=NMK
      NW2=LP1
      IFLAG=0
      IF(N.GE.K+LP1) GO TO  700
      IFLAG=1
      no 701 I=1,LP1
      II=I+1
      DO 701 J=II,LP1
      WTD=Z3GINV(I,J)
      Z3GINV(I,J)=Z3GINV(J,I)
  701 Z3GINV(J,I)=WTD
      IEND=NMK-LP1
      DO 702 I=1,IEND
      NROW=LP1+I
      DO 702 J=1,LP1
  702 Z3GINV(NROW,J)=Z3GINV(J,NROW)
```

```
      NW1=LP1
      NW2=NMK
C
C COMPUTE  SINGULAR  VALUE  DECOMPOSITION  OF  Z3*GINVERSE
  700 CALL DSVD(Z3GINV,NDIM,NDIM,NW1,NW2,0,.TRUE.,.TRUE.,S,U,V)
      PRINT 7767
 7767 FORMAT(1H0,'MUHAT')
      RMU=1.0D0+1.0D0/S(1)**2
      PRINT 601, RMU
      PRINT 640
      RNU=RMU-1.0D0
      RNU2=1.0D0/S(2)**2
      RMU2=RNU2+1.0D0
      IF(IFLAG.EQ.0) GO TO 703
      DO 704 I=1,LP1
  704 V(I,1)=U(I,1)
C
C SOLVE THE LINEAR SYSTEM G*GAMMA=V, WHERE  V IS THE SINGULAR VECTOR
C ASSOCIATED WITH THE LARGEST S INGULARVALUE
C
C NOTE THAT G IS ALREADY UPPER TRIANGULAR
  703 CALL TSOLV2(LP1, XY(K1+1,KP1),NDIM, V(1,1),GAMMA,&220)
C
C  NORMALIZE  GAMMA  TOHAVELASTCOMPONENT-1
      IF (GAMMA(LP1).EQ.0.0D0) GO TO 220
C G IS (L+1) BY (L+1)
      DO 350 I=1,L
      GAMMA(I)=-GAMMA(I)/GAMMA(LP1)
  350 CONTINUE
C
C COMPUTE RESIDUAL VARIANCE
      SQUAR=RMU/(RMU-1.0D0)/GAMMA(LP1)/GAMMA(LP1)/DFLOAT(N)
      GAMMA(LP1)=-1.0D0
      PRINT 7766
 7766 FORMAT(1H0, 'GAMMA')
      PRINT 601,(GAMMA(I),I=1,LP1)
      PRINT 640
C
      INDEX1=K*NDIM
C  INDEX1 IS THE FIRST LOCATION OF Z1, WHICH IS IN THE FIRST
C  ROW, AND THE (K+1)ST COLUMN OF XY
C Z1 IS K1BY (L+1)
C
C FORM PRODUCT OF THE MATRIX Z1 * GAMMA
      DO 360 I=1,K1
      SUM=0.0D0
      DO 370 KK=1,LP1
      IND1=INDEX1+ (KK-1)*NDIM + I
c  INDEX OF Z1(I,KK)
      SUM= SUM + XXY(IND1)*GAMMA(KK)
  370 CONTINUE
      BETA(I)=SUM
  360 CONTINUE
c
C SOLVE R1*BETA = -Z1 * GAMMA
```

```
      CALL TSOLV2(K1,XY,NDIM,BETA,BETA,&220)
      DO 380 I=1,K1
      BETA(I)=-BETA(I)
380   CONTINUE
      PRINT 7765
7765  FORMAT(1H0,'BETA')
      PRINT 601,(BETA(I),I=1,K1)
      PRINT 640
      PRINT 707,SQUAR
 707  FORMAT(///,' RESIDUAL VARIANCE',D16.6)
C
C COMPUTE ASYMPTOTIC VARIANCES
      NJ=K1+L
      NJ1=NJ+1
      DO 668 I=1,K2
      DO 668 J=1,LP1
 668     XY(I+K1,J+K)=XY(I+K1,J+K1)
      DO 677 I=1,LP1
      DO 677 J=1,LP1
      U(I,J)=0.0D0
      DO 677 M=1,NMK1
 677  U(I,J)=U(I,J)+XY(M+K1,I+K)*XY(M+K1,J+K)
      DO 669 I=1,K1
      DO 669 J=I,K1
 669  GINV(I,J)=XY(I,J)
      DO 670 J=1,L
      DO 670 I=1,K
 670  XY(I,J)=XY(I,J+K)
      DO 671 I=1,K1
      DO 671 J=I,K1
 671  XY(I,J+L)=GINV(I,J)
      DO 672 J=1,K1
      II=J+1
      DO 672 I=II,K
 672  XY(I,J+L)=0.0D0
      DO 678 I=1,K
 678  XY(I,NJ1)=XY(I,KSUM)
      CALL HCOMP1(NDIM,K,NJ,NJ1,XY,S)
      CALL TNVRT2(NJ,XY,GINV,NDIM,&220)
      CALL TSOLV2(NJ,XY,NDIM,XY(1,NJ1),S,&220)
      S(LP1)=-1.0D0
      SQUAR=0.0D0
      DO 680 I=1,LP1
      SUM=0.0D0
      DO 679 J=1,LP1
 679  SUM=SUM+U(I,J)*S(J)
 680  SQUAR=SQUAR+S(I)*SUM
      SQUAR=SQUAR/DFLOAT(N)
      DO 675 I=1,L
      S(I)=0.0D0
      DO 674 J=I,NJ
 674  S(I)=S(I)+GINV(I,J)*GINV(I,J)
 675  S(I)=S(I)*SQUAR
      DO 676 I=1,K1
      Z3GINV(I,I)=0.0D0
```

```
      II=I+L
      DO 676 J=II,NJ
  676 Z3GINV(I,I)=Z3GINV(I,I)+GINV(II,J)*GINV(II,J)
      PRINT 713
  713 FORMAT(///' ASYMPTOTIC VARIANCES AND Z VALUES FOR GAMMA'/)
      PRINT 601,(S(I),I=1,L)
      DO 714 I=1,L
  714 GAMMA(I)=GAMMA(I)/DSQRT(S(I))
      PRINT 601,(GAMMA(I),I=1,L)
      DO 715 I=1,K1
  715 Z3GINV(I,I)=Z3GINV(I,I)*SQUAR
      PRINT 716
  716 FORMAT(///' ASYMPTOTIC VARIANCES AND Z VALUES FOR BETA'/)
      PRINT 601, (Z3GINV(I,I),I=1,K1)
      DO 717 I=1,K1
  717 BETA(I)=BETA(I)/DSQRT(Z3GINV(I,I))
      PRINT 601,(BETA(I),I=1,K1)
      KJ=2*(K2-L+1)
      KM=K2-L
      AN=DFLOAT(N)
      TEST1=AN* RNU
      TEST2=AN*DLOG(RMU)
      PRINT 730,TEST1,TEST2,KM
      TEST1=TEST1+AN*RNU2
      TEST2=TEST2+AN*DLOG(RMU2)
      PRINT 731,TEST1,TEST2,KJ
      TEST1=DFLOAT(N-K)*RNU/DFLOAT(K2)
      TEST2=DFLOAT(N-K)*RNU2/DFLOAT(K2)
      PRINT 732,TEST1,TEST2,K2,NMK
  730 FORMAT(////' N*(MUHAT-1) IS',F12.3,//,' N*LOG(MUHAT) IS',F11.3,//,
     1' CHI-SQUARE D.F.',I12)
  731 FORMAT(////' N*(MUHAT(1)+MUHAT(2)-2) IS',F12.3,//,' N*LOG(MUHAT(1)
     1+MUHAT(2)) IS',F11.3,//,' CHI-SQUARE D.F.',12X,I12)
  732 FORMAT(////' (N-K)*(MUHAT(1)-1)/K2 IS',F12.3,//,' (N-K)*(MUHAT(2)-
     21)/K2 IS',F12.3//,' F DISTRIBUTION D.F.',I12,3X,',',3X,I4)
      GO TO 500
  220 PRINT 221
  221 FORMAT(1H0,'SINGULAR UPPER TRIANGULAR MATRIX')
      GO TO 500
 1091 PRINT 1093
 1093 FORMAT(//' THIS EQUATION IS JUST IDENTIFIED AND TWO-STAGE'/' LEAST
     1 SQUARES IS APPROPRIATE')
      GO TO 500
 1092 PRINT 1094
 1094 FORMAT(//' THIS EQUATION IS NOT IDENTIFIABLE')
  500 CONTINUE
      STOP
C
C
C
      END
```

```
      SUBROUTINE DSVD(A,MMAX,NMAX,M,N,P,WITHU,WITHV,S,U,V)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(MMAX,NMAX),U(MMAX,NMAX),V(NMAX,NMAX)
      DIMENSION S(N),B(100),C(100),T(100)
C
C     THIS SUBROUTINE COMPUTES THE SINGULAR VALUE DECOMPOSITION
C     OF A REAL M*N MATRIX A, I. E. IT COMPUTES MATRICES U, S, AND V
C     SUCH THAT
C
C                      A = U * S * VT ,
C     WHERE
C               U IS AN M*N MATRIX AND UT*U = I, (UT=TRANSPOSE
C                                                         OF U),
C               V IS AN N*N MATRIX AND VT*V = I, (VT=TRANSPOSE
C                                                         OF V),
C        AND    S IS AN N*N DIAGONAL MATRIX,
C
C     DESCRIPTION OF PARAMETERS:
C
C     A = REAL*8 ARRAY. A CONTAINS THE MATRIX TO BE DECOMPOSED.
C
C     MMAX = INTEGER*4 VARIABLE. THE NUMBER OF DECLARED ROWS IN THE
C            ARRAYS A AND U.
C
C     NMAX = INTEGER*4 VARIABLE. THE NUMBER OF DECLARED ROWS IN THE
C            ARRAY V.
C
C     M, N = INTEGER*4 VARIABLES. THE NUMBER OF ROWS AND COLUMNS
C           IN THE MATRIX STORED IN A. (N<=M<=100. IF IT IS
C           NECESSARY TO SOLVE A LARGER PROBLEM, THEN THE
C           AMOUNT OF STORAGE ALLOCATED TO THE ARRAYS B, C, AND
C           T MUST BE INCREASED ACCORDINGLY.)
C
      INTEGER P
      LOGICAL WITHU,WITHV
C     WITHU, WITHV = LOGICAL*4 VARIABLES. IF WITHU=.TRUE., THEN
C         THE MATRIX U IS COMPUTED AND STORED IN THE ARRAY U.
C         SIMILARLY FOR V.
C
C     S = REAL*8 ARRAY, S(1), . . . , S(N) CONTAIN THE DIAGONAL
C         ELEMENTS OF THE MATRIX S ORDERED SO THAN S(I)>=S(I+1),
C         I=1, . . . , N-1.
C
C     U,V = REAL*8 ARRAYS. U, V CONTAIN THE MATRICES U AND V.
C         IF WITHU=.TRUE. AND WITHV=.FALSE., THEN THE ACTUAL
C         PARAMETER CORRESPONDING TO A AND U MAY BE THE SAME.
C         SIMILARLY FOR V IF WITHV=.TRUE. AND WITHU=.FALSE..
C
C     P = INTEGER*4 VARIABLE. IF P>0, THEN COLUMNS N+1, . . . ,
C         N+P OF A ARE ASSUMED TO CONTAIN THE COLUMNS OF AN M*P
C         MATRIX B. THIS MATRIX IS MULTIPLIED BY UT, AND UPON
C         EXIT, A CONTAINS IN THESE SAME COLUMNS THE N*P MATRIX
C         UT*B. (P>=0)
C
C     THIS SUBROUTINE IS A TRANSLATION OF AN ALGOL 60 PROCEDURE
C     DESCRIBED IN THE ARTICLE "SINGULAR VALUE DECOMPOSITION AND
```

```
C      LEAST SQUARES SOLUTIONS", NUM. MATH. 14 (1970), PP. 403-420.
C      THE TRANSLATION WAS DONE BY P. BUSINGER AT BELL TELEPHONE
C      LABORATORIES WITH SOME CHANGES AND EDITING DONE BY R.
C      UNDERWOOD AT STANFORD UNIVERSITY.
C
       DATA ETA /Z3410000000000000/
       DATA TOL /Z0010000000000000/
C
C      ETA AND TOL ARE MACHINE DEPENDENT CONSTANTS WHOSE
C      VALUES ARE 16**(-13) AND 16**(-52), RESPECTIVELY,
C      ON IBM SYSTEM/360 COMPUTERS.
C
C
       NP=N + P
       N1=N+1
C
C      HOUSEHOLDER REDUCTION TO BIDIAGONAL FORM
       C(1)=0.0D0
       K=1
   10  K1=K+1
C
C      ELIMINATION OF A(I,K), I=K+1, . . . , M
       Z=0.0D0
       DO 20  I=K,M
   20     Z=Z+A(I,K)**2
       B(K)=0.0D0
       IF (Z.LE.TOL) GOTO 70
       Z=DSQRT(Z)
       B(K)=Z
       W=DABS(A(K,K))
       Q=1.0D0
       IF (W.NE.0.0D0) Q=A(K,K)/W
       A(K,K)=Q*(Z+W)
       IF (K.EQ.NP) GOTO  70
       DO 50  J=K1,NP
          Q=0.0D0
          DO 30  I=K,M
   30        Q=Q+A(I,K)*A(I,J)
          Q=Q/(Z*(Z+W))
          DO 40  I=K,M
   40        A(I,J)=A(I,J)-Q*A(I,K)
   50     CONTINUE
C
C      PHASE TRANSFORMATION
       Q=-A(K,K)/DABS(A(K,K))
       DO 60  J=K1,NP
   60     A(K,J)=Q*A(K,J)
C
C      ELIMINATION OF  A(K,J),J=K+2,...,N
   70  IF (K.EQ.N) GOTO  140
       Z=0.0D0
       DO 80  J=K1,N
   80     Z=Z+A(K,J)**2
       C(K1)=0.0D0
       IF (Z.LE.TOL) GOTO  130
```

```
      Z=DSQRT(Z)
      C(K1)=Z
      W=DABS(A(K,K1))
      Q=1.0D0
      IF (W.NE.0.0D0) Q=A(K,K1)/W
      A(K,K1)=Q*(Z+W)
      DO 110 I=K1,M
         Q=0.0D0
         DO 90 J=K1,N
   90       Q=Q+A(K,J)*A(I,J)
         Q=Q/(Z*(Z+W))
         DO 100 J=K1,N
  100       A(I,J)=A(I,J)-Q*A(K,J)
  110    CONTINUE
C
C     PHASE TRANSFORMATION
      Q=-A(K,K1)/DABS(A(K,K1))
      DO 120 I=K1,M
  120    A(I,K1)=A(I,K1)*Q
C
  130 K=K1
      GOTO 10
C
C     TOLERANCE FOR NEGLIGIBLE ELEMENTS
  140 EPS=0.0D0
      DO 150 K=1,N
         S(K)=B(K)
         T(K)=C(K)
  150    EPS=DMAX1(EPS,S(K)+T(K))
      EPS=EPS*ETA
C
C     INITIALIZATION OF U AND V
      IF (.NOT.WITHU) GOTO 180
      DO 170 J=1,N
         DO 160 I=1,M
  160       U(I,J)=0.0D0
  170    U(J,J)=1.0D0
C
  180 IF (.NOT.WITHV) GOTO 210
      DO 200 J=1,N
         DO 190 I=1,N
  190       V(I,J)=0.0D0
  200    V(J,J)=1.0D0
C
C     QR DIAGONALIZATION
  210 DO 380 KK=1,N
         K=N1-KK
C
C     TEST FOR SPLIT
  220    DO 230 LL=1,K
            L=K+1-LL
            IF (DABS(T(L)).LE.EPS) GOTO 290
            IF (DABS(S(L-1)).LE.EPS) GOTO 240
  230       CONTINUE
C
```

```
C       CANCELLATION
  240      CS=0.0D0
           SN=1.0D0
           L1=L-1
           DO 280 I=L,K
              F=SN*T(I)
              T(I)=CS*T(I)
              IF (DABS(F).LE.EPS) GOTO 290
              H=S(I)
              W=DSQRT(F*F+H*H)
              S(I)=W
              CS=H/W
              SN=- F/W
              IF (.NOT.WITHU) GOTO 260
              DO 250 J=1,N
                 X=U(J,L1)
                 Y=U(J,I)
                 U(J,L1)=X*CS+Y*SN
  250            U(J,I)=Y*CS-X*SN
  260         IF (NP.EQ.N) GOTO 280
              DO 270 J=N1,NP
                 Q=A(L1,J)
                 R=A(I,J)
                 A(L1,J)=Q*CS+R*SN
  270            A(I,J)=R*CS-Q*SN
  280      CONTINUE
C
C       TEST FOR CONVERGENCE
  290      W=S(K)
           IF (L.EQ.K) GOTO 360
C
C       ORIGIN SHIFT
           X=S(L)
           Y=S(K-1)
           G=T(K-1)
           H=T(K)
           F=((Y-W)*(Y+W)+(G-H)*(G+H))/(2.0D0*H*Y)
           G=DSQRT(F*F+1.0D0)
           IF (F.LT.0.0D0) G=-G
           F=((X-W)*(X+W)+(Y/(F+G)-H)*H)/X
C
C       QR STEP
           CS=1.0D0
           SN=1.0D0
           L1=L+1
           DO 350 I=L1,K
              G=T(I)
              Y=S(I)
              H=SN*G
              G=CS*G
              W=DSQRT(H*H+F*F)
              T(I-1)=W
              CS=F/W
              SN=H/W
              F=X*CS+G*SN
```

```
            G=G*CS-X*SN
            H=Y *SN
            Y=Y*CS
            IF (.NOT.WITHV) GOTO 310
            DO 300 J=1,N
               X=V(J,I-1)
               W=V(J,I)
               V(J,I-1)=X*CS+W*SN
  300          V(J,I)=W*CS-X*SN
  310       W=DSQRT(H*H+F*F)
            S(I-1)=W
            CS=F/W
            SN=H/W
            F=CS*G+SN*Y
            X=CS*Y-SN*G
            IF (.NOT.WITHU) GOTO 330
            DO 320 J=1,N
               Y=U(J,I-1)
               W=U (J,I )
               U(J,I-1)=Y*CS+W*SN
  320          U(J,I)=W*CS-Y*SN
  330       IF (N.EQ.NP) GOTO 350
            DO 340 J=N1,NP
               Q=A(I-1,J)
               R=A(I,J)
               A(I-1,J)=Q*CS+R*SN
  340          A(I,J)=R*CS-Q*SN
  350       CONTINUE
C
         T(L)=0.0D0
         T(K)=F
         S(K)=X
         GOTO 220
C
C     CONVERGENCE
  360    IF (W.GE.0.0D0) GOTO 380
         S(K)=-W
         IF (.NOT.WITHV) GOTO 380
         DO 370 J=1,N
  370       V(J,K)=-V(J,K)
  380    CONTINUE
C
C     SORT SINGULAR VALUES
      DO 450 K=1,N
         G=-1.0D0
         J=K
         DO 390 I=K,N
            IF (S(I).LE.G) GOTO 390
            G=S(I)
            J=I
  390       CONTINUE
         IF (J.EQ.K) GOTO 450
         S(J)=S(K)
         S(K)=G
         IF (.NOT.WITHV) GOTO 410
         DO 400 I=1,N
```

```
          Q=V(I,J)
          V(I,J)=V(I,K)
  400     V(I,K)=Q
  410   IF (.NOT.WITHU) GOTO 430
        DO 420 I =1,N
          Q=U(I,J)
          U(I,J)=U(I,K)
  420     U(I,K)=Q
  430   IF (N.EQ.NP) GOTO 450
        DO 440 I=N1,NP
          Q=A(J,I)
          A(J,I)=A(K,I)
  440     A(K,I)=Q
  450   CONTINUE
C
C     BACK TRANSFORMATION
      IF (.NOT.WITHU) GOTO 510
      DO 500 KK=1,N
        K=N1-KK
        IF (B(K).EQ.0.0D0) GOTO 500
        Q=-A(K,K)/DABS(A(K,K))
        DO 460 J=1,N
  460     U(K,J)=Q*U(K,J)
        DO 490 J=1,N
          Q=0.0D0
          DO 470 I=K,M
  470       Q=Q+A(I,K)*U(I,J)
          Q=Q/(DABS(A(K,K))*B(K))
          DO 480 I=K,M
  480       U(I,J)=U(I,J)-Q*A(I,K)
  490     CONTINUE
  500   CONTINUE
C
  510 IF (.NOT.WITHV) GOTO 570
      IF (N.LT.2) GOTO 570
      DO 560 KK=2,N
        K=N1-KK
        K1=K+1
        IF (C(K1).EQ.0.0D0) GOTO 560
        Q=-A(K,K1)/DABS(A(K,K1))
        DO 520 J=1,N
  520     V(K1,J)=Q*V(K1,J)
        DO 550 J=1,N
          Q=0.0D0
          DO 530 I=K1,N
  530       Q=Q+A(K,I)*V(I,J)
          Q=Q/(DABS(A(K,K1))*C(K1))
          DO 540 I=K1,N
  540       V(I,J)=V(I,J)-Q*A(K,I)
  550     CONTINUE
  560   CONTINUE
C
  570 RETURN
      END
```

```
      SUBROUTINE TNVRT2 (N, A, AI, IDIM,  *)

         INTEGER N, IDIM
         REAL*8 A(IDIM,N), AI(IDIM,N)

C  TNVRT2 IS A MODIFICATION OF INVRT2 TO INVERT AN
C  ORIGINAL MATRIX THAT IS UPPER TRIANGULAR.
C  CALLS TO DECMP2 AND IMPRV2 ARE OMITTED.

      EXTERNAL TSOLV2
         INTEGER I, J
         REAL*8 E(100), X(100)

         DO 3 I=1,N
            E(I) = 0.0D0
    3    CONTINUE
         DO 1 I=1,N
            E(I) = 1.0D0
            CALL TSOLV2 (N, A, IDIM, E, X, &10)
            DO 2 J=1,N
               AI(J,I) = X(J)
    2       CONTINUE
            E(I) = 0.0D0
    1    CONTINUE
         RETURN

   10    RETURN 1

C     LAST CARD OF SUBROUTINE TNVRT2.
      END
```

```
      SUBROUTINE TSOLV2 (N, LU, IDIM, B, X, *)

          INTEGER N, IDIM
          REAL*8 LU( IDIM,N), B(N), X(N)

C   SOLVES LU*X = B, WHERE LU IS UPPER TRIANGULAR.
C   TSOLV2 IS A MODIFICATION OF THE USUAL SOLVE2, WHICH FOLLOWS
C   DECMP2.
C
C
          INTEGER I, J, IP, IP1, IM1, NP1, IBACK
          REAL*8 SUM
          NP1 = N + 1

      IF (LU(N,N).EQ.0.0D0)  RETURN 1
         X(N) = B(N)/LU(N,N)
      IF(N.EQ.1)  RETURN

C  BACK SUBSTITUTION

         DO 4 IBACK =2,N
            I = NP1 - IBACK

C           I GOES FROM (N-1) TO 1 .

            IP1 = I+1
            SUM = 0.0D0
            DO 3 J=IP1,N
               SUM = SUM + LU(I,J)*X(J)
    3       CONTINUE
      IF (LU(I,I).EQ.0.0D0)  RETURN 1
        X(I) = (B(I)-SUM)/LU(I,I)
    4 CONTINUE
         RETURN
C     LAST CARD OF SUBROUTINE TSOLV2.
      END
```

```
      SUBROUTINE HCOMP1(MDIM,M,N,NTOTAL,A,U)
C
      INTEGER MDIM,M,N
      DOUBLE PRECISION A(MDIM,N),U(M)
C
C   HOUSEHOLDER REDUCTION OF THE FIRST N COLUMNS OF A
    TO UPPER TRANGULAR FORM.
C   HCOMP1 IS TAKEN FROM CLEVE MOLER'S SUBROUTINE HECOMP,
C   WHICH REDUCES ALL COLUMNS OF THE MATRIX TO TRIANGULAR FORM.
C   HCOMP1 WAS DESIGNED FOR A SPECIAL PURPOSE, TO MAKE THE FIRST
C   N COLUMNS UPPER TRIANGULAR, WHILE APPLYING THE TRANSFORMATIONS
C   TO NTOTAL COLUMNS.
C
C
C
C   MDIM= DECLARED ROW DIMENSION OF A
C   M= NUMBER OF ROWS
C   N= NUMBER OF COLUMNS OF A WHOSE UPPER PORTIONS ARE TO
C   BE REDUCED TO TRIANGULAR FORM.  AFTER REDUCTION, THE
C   MATRIX A WILL CONTAIN AN N BY N UPPER TRIANGULAR MATRIX
C   IN ITS UPPER LEFT CORNER.
C   NTOTAL= TOTAL NUMBER OF COLUMNS OF A.  THE TRANSFORMATIONS
C   WILL BE APPLIED TO ALL COLUMNS.
C   FOR STANDARD LEAST-SQUARES PROBLEMS, NTOTAL=N.
C
    A= M BY NTOTAL MATRIX WITH M.GE.NTOTAL
C      INPUT, MATRIX TO BE REDUCED
C      OUTPUT, REDUCED MATRIX AND INFORMATION ABOUT REDUCTION
C   U= VECTOR OF LENGTH M
C      INPUT, IGNORED
C      OUTPUT, INFORMATION ABOUT REDUCTION
C
C
C
      DOUBLE PRECISION ALPHA,BETA,GAMMA,DSQRT
C
      DO 6 K=1,N
C
C   FIND REFLECTION THAT ZEROS A(I,K),I=K+1,...,M
C
      ALPHA=0.0
      DO 1 I=K,M
         U(I)=A(I,K)
         ALPHA=ALPHA+U(I)**2
1     CONTINUE
```

```
      ALPHA=DSQRT(ALPHA)
      IF(U(K).LT.0.0)  ALPHA=-ALPHA
      U(K)=U(K)+ALPHA
      BETA=ALPHA*U(K)
      A(K,K)=-ALPHA
      IF(BETA.EQ.0.0  .OR.  K.EQ.NTOTAL)  GO TO 6
C
C
C  APPLY REFLECTION TO REMAINING COLUMNS OF A
C
      KP1=K+1
      DO 4 J=KP1,NTOTAL
         GAMMA=0.0
         DO 2 I=K,M
            GAMMA=GAMMA+U(I)*A(I,J)
2        CONTINUE
         GAMMA=GAMMA / BETA
         DO 3 I=K, M
            A(I,J)=A(I,J) - GAMMA * U(I)
3        CONTINUE
4     CONTINUE
6     CONTINUE
      RETURN
C
C TRIANGULAR  RESULT  IS STOREDINA(I J ),..,'E.J
C  VECTORS DEFINING REFLECTIONS ARE STORED IN U AND REST OF A
C
      END
```