tanulmányok 146/1983

MTA Számitástechnikai és Automatizálási Kutató Intézet Budapest

MAGYAR TUDOMÁNYOS AKADÉMIA
SZÁMITÁSTECHNIKAI ÉS AUTOMATIZÁLÁSI KUTATÓ INTÉZETE

NUMERICAL METHODS FOR COMPUTATION OF THE GENERALIZED INVERSE OF RECTANGULAR MATRICES

Irta:

VARGA GYULA

Tanulmányok 146/1983

A kiadásért felelős:

DR VÁMOS TIBOR

Osztályvezető:

BALLA KATALIN

ISBN 963 311 160 9

ISSN 0324-2951

NUMERICAL METHODS FOR COMPUTATION
OF THE GENERALIZED INVERSE OF RECTANGULAR MATRICES

INTRODUCTION

Recently a number of books and papers deal with the generalized inverse of rectangular matrices, and with its applications in the numerical analysis. The ground of its importance is that it is connected with the problems of linear parameter estimation and fitting based on the principle of the least squares. Various direct and iterative numerical methods are known for the computation of the GI of rectangular matrices based on different properties of its concept introduced by Moore [1] and Penrose [2]. This latter has shown, that, for any m by n $(m, n > 0)$ complex matrix A, there exists a unique n by m GI, denoted by A^+, which satisfies the following 4 relations (Penrose's Lemmas):

$$A A^+ A = A \qquad (A A^+)^\star = A A^+$$
$$A^+ A A^+ = A^+ \qquad (A^+ A)^\star = A^+ A$$

where $\star$ denotes the conjugate transpose.

Rao and Mitra [3] have shown, that if the matrix of the linear system of equations $A\underline{x} = \underline{b}$ is rectangular, then $\underline{x} = A^+\underline{b}$ is the vector of least Euclidean norm, which minimizes the Euclidean norm of $A\underline{x} - \underline{b}$. (This vector is called the (pseudo) normal solution of the system of equations $A\underline{x} = \underline{b}$.)

This result gives a possibility to write the GI of a matrix in explicit form. If A is a full-rank m by n matrix $(r(A) = \min(m, n))$, then there are 3 possible cases:

(1) $m > n$ overdetermined system, no solution

(2) $m = n$ regular system, 1 solution

(3) $m < n$ underdetermined system, ∞ solutions.

In the first case, there is only one vector, which minimizes the function $f(\underline{x}) = | A\underline{x} - \underline{b} |$, it is the result of an ordinary unconstrained minimization problem, $\underline{x} = (A^\star A)^{-1} A^\star \underline{b}$, and so $A^+ = (A^\star A)^{-1} A^\star$.

The second is the case of regular quadratic matrices: $A^+ = A^{-1}$.

In the third case there is only one vector, which minimizes $| \underline{x} |$ with the restriction, that $A\underline{x} = \underline{b}$. This minimization problem can be solved by applying the method of Lagrange multipliers. It gives $\underline{x} = A^\star(A A^\star)^{-1} \underline{b}$, and so $A^+ = A^\star(A A^\star)^{-1}$.

However, if A is not of full-rank, then it can be factorized into the product of full-rank matrix pairs of one of the following types:

$$(1) - (2), \quad (1) - (3), \quad (2) - (3) .$$

Though this factorization is not unique, the above mentioned minimization property of the GI ensures the uniqueness of the result, which can be obtained using the known rule for inversion of the product of two regular quadratic matrices ($(BC)^{-1} = C^{-1} B^{-1}$), which is valid for the calculation of the GI of the product of full-rank rectangular matrices too ($(BC)^+ = C^+ B^+$).

The aim of this treatise is to introduce to the numerical methods and their subroutines for the computation of the GI of rectangular matrices, contained by the program library of the machine CDC 3300 of the Hungarian Academy of Sciences. For this purpose first a brief survey will be given about the basic properties of the GI, that are used in the different computing methods.

This treatise does not contain the numerical comparison of the computing methods to be discussed, an exhaustive comparison can be found e.g. in [4].

BASIC PROPERTIES OF THE GI

In the following, some basic properties of the GI will be described without proof. They follow from the Penrose's Lemmas or from the explicit form of the GI.

(1) The rank of the GI of a rectangular matrix is equal to the rank of the given matrix.

(2) The GI of the (conjugate) transpose of a matrix is equal to the (conjugate) transpose of the GI for complex and real matrices respectively.

(3) The GI of the GI of a matrix is equal to the matrix in question.

(4) If $m < n$ for a full-rank rectangular matrix A, then $AA^+ = I$, however if $m > n$, then $A^+A = I$, otherwise, for not full-rank matrices, these products are projector matrices.

(5) The GI of a scalar c (1 by 1 matrix) is a scalar c^+ :

$$c^+ = \begin{cases} 1/c & \text{if } c \neq 0 \\ 0 & \text{otherwise} \end{cases} .$$

(6) The GI of a column vector $\underline{r}$ (n by 1 matrix) is a row vector defined as follows:

$$\underline{r}^+ = \begin{cases} \underline{r}^T/(r^T r) & \text{if } \underline{r} \neq \underline{0} \\ \underline{r}^T & \text{otherwise} \end{cases} .$$

in particular

$$\begin{bmatrix} 1 \\ \underline{x} \end{bmatrix}^+ = \frac{\left[1, \underline{x}^T\right]}{1 + \underline{x}^T \underline{x}}$$

(7) This latter property can be generalized for rectangular block matrices in the following form:

$$\begin{bmatrix} I \\ X \end{bmatrix}^+ = \left[I + X^T X\right]^{-1} \left[I, X^T\right] .$$

(8) If O is a zero matrix and

$$B = \begin{bmatrix} A \\ O \end{bmatrix}$$

then

$$B^+ = \begin{bmatrix} A^+, O^T \end{bmatrix} .$$

(9) If O_1 and O_2 are zero matrices and

$$C = \begin{bmatrix} A & O_1 \\ O_2 & B \end{bmatrix} ,$$

then

$$C^+ = \begin{bmatrix} A^+ & O_2^T \\ O_1^T & B^+ \end{bmatrix} .$$

(10) If $A=BC$, and either B or C is an orthogonal matrix, then $A^+= C^+B^+$, even when the other factor is not of full rank.

(11) If, as the result of a perturbation, the rank of the matrix reduces, then the change of the entries of the GI is not continuous [5]. E.g.

$$A = \begin{bmatrix} 1 & x \\ 2 & 0 \\ 1 & 0 \end{bmatrix} ,$$

if $x \neq 0$, then $r(A) = 2$, and

$$A^+ = \begin{bmatrix} 0 & 0.4 & 0.2 \\ 1/x & -0.4/x & -0.2/x \end{bmatrix} ,$$

if $x \longrightarrow 0$, then $A^+ \longrightarrow \infty$,

if $x = 0$, then $r(A) = 1$, and

$$A^+ = \begin{bmatrix} 1/6 & 1/3 & 1/6 \\ 0 & 0 & 0 \end{bmatrix} .$$

NUMERICAL METHODS FOR COMPUTATION OF THE GI

Further some numerical methods will be introduced for computation of the GI. Though the Penrose's Lemmas concern complex rectangular matrices, there exists no special method for them. However the methods for real ones can be applied either using complex arithmetic, or constructing a 2m by 2n real matrix

$$Z = \begin{bmatrix} B & -C \\ C & B \end{bmatrix}$$

from the complex m by n matrix $A = B + j\,C$, and considering, that if $A^+ = U + j\,V$, then

$$Z^+ = \begin{bmatrix} U & -V \\ V & U \end{bmatrix}$$

and its blocks yield the GI of Z. (This latter method requires more memory, but the other one is slower.)

In the matter of the GI of real rectangular matrices, the following FORTRAN subroutines will be introduced:

(1) GENINF (for computation of the GI of a full-rank real rectangular matrix by means of the Householder orthogonalization)

(2) GENINR (for computation of the GI of an arbitrary real rectangular matrix by means of the Householder orthogonalization)

(3) SVDINV (for the same case as in (2) by means of the singular value decomposition)

(4) GREINV (for the same case as in (2) by means of the Greville recursive algorithm)

(5) GINV (for the same case as in (2) by means of the Gram-Schmidt orthogonalization)

(6) NOREQU (for computation of the (pseudo) normal solution of a linear system of equations with real rectangular matrix and real right hand side vector by means of the Gaussian elimination and the Cholesky decomposition without explicit computation of the GI of the matrix of the system of equations).

In cases (1) and (5), the rank of the matrix in question is known, in the remaining cases, it is computed and stored by an output parameter.

BRIEF DESCRIPTION OF THE SUBROUTINES

(1) GENINF performs Householder orthogonal decomposition [5] on the matrix A in n steps, resulting in the form $A = QBP$, where Q is the product of elementary orthogonal matrices of the form $I - \underline{u}\,\underline{u}^T/h$, constructed by means of the columns of A, B is an m by n matrix whose lower trapezoidal part has zero entries, and P is a permutation matrix needed because of the successive choice of columns of maximum norm. The GI of A is $A^+ = P^T B^+ Q^T$ (B^+ can be easily computed by virtue of property 8). In case of rank deficiency error indication is given.

(2) GENINR performs Householder orthogonal decomposition on the matrix A in r steps (the number of successful steps gives the rank of A), resulting in the form $A = QBP$, where Q is of the same type as above, B is an m by n matrix having nonzero entries in its upper trapezoidal part only, P is the same type as above. One more orthogonal decomposition gives the equation $B = CR$, where C is a block matrix consisting of four blocks: the upper left hand block is a lower triangular matrix, the other blocks are zero matrices. The complete decomposition of A has the form $A = QCRP$. Because of property 10, the GI of A is $A^+ = P^T R^T C^+ Q^T$. (C^+ can be easily computed by means of property 9). In case of zero rank, error indication is given.

(3) SVDINV uses the singular value decomposition technique [7] for the matrix A, resulting in the form $A = U \Sigma V^T$ [8], where U and V are orthogonal matrices, and Σ is an m by n matrix having nonzero entries in its "main diagonal" only (the number of nonzero entries equals the rank of A). From the decomposed form of the matrix, the GI can be computed in the form $A^+ = V \Sigma^+ U^T$ (by virtue of the properties 9 and 10).

(4) GREINV applies the recursive Greville algorithm [9] for computation of the GI of an m by n matrix A in the following way: Let $A_1 = \underline{a}_1$ (the first

column of A), compute its GI A_1^+ by virtue of property 6. Further let $A_k = [A_{k-1}, \underline{a}_k]$ $(k = 2, \ldots, n)$ be in partitioned form. Seek for its GI in the form $A^+ = \left[B_k^T, \underline{b}_k^T \right]^T$. Requiring the fulfilment of Penrose's Lemmas, it gives $B_k = A_{k-1}^+ - \underline{d}_k \underline{b}_k$, where $\underline{d}_k = A_{k-1}^+ \underline{a}_k$. For computation of $\underline{b}_k$, there are two cases. Let namely $\underline{c}_k = \underline{a}_k - A_{k-1} \underline{d}_k$, then if

(i) $\underline{c}_k \neq 0$, i.e. $r(A_k) > r(A_{k-1})$, then $\underline{b}_k = \underline{c}_k^+$,

(ii) $\underline{c}_k = 0$, then we have $\underline{b}_k = (1 + d_k^T d_k)^{-1} \underline{d}_k^T A_{k-1}$.

After the last step of the recursion we have $A^+ = A_n^+$.

(5) GINV [10] can be applied for computation of the GI, if the matrix is partitioned in advance in the form $A = [R, S]$, where R contains the linearly independent columns of A, and S contains the linear combinations. Because of this form, there exists a unique factorization $S = RU$, and A can be written in the form $A = R\,[I, U]$. Due to the properties 2 and 7, one can write the GI of A in the form $A^+ = [I, U]^T \cdot (I + U\,U^T)^{-1} \cdot R^+$. Using the modified Gram-Schmidt orthogonalization technique, one can write the GI of A in the following form:

$$A^+ = \begin{bmatrix} \left[I - (UP)(UP)^T \right] ZQ^T \\ P(UP)^T ZQ^T \end{bmatrix} ,$$

where $R^+ = Z\,Q^T$ (obtained by means of Gram-Schmidt orthogonal decomposition of R), and $(I + U\,U^T)^{-1} = P\,P^T$.

(6) NOREQU performs Gaussian elimination with complete pivoting on the m by n matrix of the linear system of equations $A\underline{x} = \underline{b}$ to obtain its LU factorized form. The system of equations can be written in the form $LU\underline{y} = \underline{d}$, where $\underline{y} = P\,\underline{x}$ and $\underline{d} = P\,\underline{b}$ (P is a permutation matrix). Since L and U are full-rank matrices, the (pseudo) normal solution of the system of equations can be obtained by premultiplication of $\underline{d}$ by the GI of L and U (without explicitly computing them) and finally by P^T. So we have $\underline{x}_N = P^T U^+ L^+ \underline{d}$.

NOTES

(1) The requirement of memory and the size of arrays needed to the described procedures can be found in the list of parameters and in the comments of the FORTRAN subroutines of the numerical methods in question.

It is essential to underline, that the actual dimensioning of the arrays contained by the list of parameters of the subroutines takes place in the main program written by the user.

(2) It is quite impossible to give the number of operations needed to the execution of the procedures, because it depends on the rank of the rectangular matrix to be inverted. For each basic numerical technique (Gaussian elimination, Cholesky decomposition, Honseholder or Gram-Schmidt orthogonalization, QR triangularization) that are used in the described procedures, the number of operation can be found in the literature. Therefore a complete testing was not made, only a verification of the described methods is given here.

(3) For the verification of GENINF the full-rank test matrix

$$A = \begin{bmatrix} 1 & 2 & -3 \\ 4 & -5 & 8 \\ 3 & 2 & 1 \\ 8 & -1 & 6 \end{bmatrix}$$

vas used, resulting in the known GI

$$A^{+} = 0.01 \cdot \begin{bmatrix} 28 & 2 & -16 & 14 \\ -35 & -15 & 45 & -5 \\ -39 & -1 & 33 & -7 \end{bmatrix} .$$

The remaining subroutines (GENINR, SVDINV, GREINV and GINV) were verified by using the matrix

$$B = \begin{bmatrix} 2 & 5 & -7 & 0 \\ 3 & -2 & 4 & 5 \\ 1 & 1 & -1 & 1 \\ -2 & 8 & -11 & -5 \\ 1 & -1 & -3 & -3 \end{bmatrix},$$

that resulted in the GI

$$B^{+} = 0.1 \begin{bmatrix} 0.99579395 & 0.09994797 & -0.58776342 & 0.50797849 \\ 0.81266441 & 0.08737317 & -0.13059000 & 0.76944758 \\ 0.33200359 & 0.11750932 & -0.13427572 & 0.31523719 \\ -0.57453820 & 0.66299540 & -0.08108577 & 0.00737143 \\ 1.58933888 & -1.75353395 & -0.96059897 & -1.12479403 \end{bmatrix}^{T},$$

and the rank $r(B) = 3$ by all the four subroutines. Finally NOREQU was verified by solving the overdetermined system of equations $B\,\underline{x} = \underline{c}$, where B is the above rectangular matrix, and $\underline{c} = [1, 0, 4, -7, 3]^{T}$, resulting in the (pseudo) normal solution

$$\underline{x}_N = [1.11135923, -0.933158442, -0.343906282, -0.16570549]^{T}$$

that was corroborated by postmultiplying B^{+} by the right hand side vector $\underline{c}$.

LIST OF THE SUBROUTINES

```
      SUBROUTINE GENINF(NDIM,N,M,A,V,C,U,JEL,EPS,IER)
C  COMPUTE THE MOORE-PENROSE GENERALIZED INVERSE OF AN N BY M
C  REAL FULL RANK MATRIX BY MEANS OF THE HOUSEHOLDER
C  ORTHOGONAL DECOMPOSITION
C     PARAMETERS:
C  NDIM MAXIMUM NUMBER OF ROWS OF A
C  N NUMBER OF ROWS OF A
C  M NUMBER OF COLUMNS OF A (M.LE.N)
C  A N BY M ARRAY OF THE MATRIX TO BE INVERTED, REPLACED BY THE
C     TRANSPOSE OF THE RESULTANT GENERALIZED INVERSE
C  V N BY N ARRAY OF THE ORTHOGONAL MATRIX
C  C WORKING VECTOR OF LENGTH N
C  U WORKING VECTOR OF LENGTH N
C  JEL PERMUTATION VECTOR OF LENGTH M
C  EPS ABSOLUTE ERROR BOUND
C  IER ERROR CODE
C     IER=0 NORMAL TERMINATION
C     IER=1 RANK DEFICIENCY
      DIMENSION A(NDIM,1),V(NDIM,1),C(1),U(1).JEL(1)
      IER=0
      DO 1 I=1,N
      DO 1 J=1,N
    1 V(I,J)=(I/J)*(J/I)
      DO 2 IR=1,M
      IR1=IR+1
      SI=0.
      DO 17 IS=IR,M
      SIGMA=0.
      DO 3 I=IR,N
      C(I)=A(I,IS)
      S=C(I)
    3 SIGMA=SIGMA+S*S
      IF(SIGMA.LE.SI) GO TO 17
      SI=SIGMA
      IT=IS
      DO 40 I=IR,N
   40 U(I)=C(I)
   17 CONTINUE
      SIGMA=SI
      IF(SIGMA.GT.EPS) GO TO 4
      IER=1
      RETURN
    4 JEL(IR)=IT
      IF(IR.EQ.IT) GO TO 33
      DO 32 I=1,N
      S=A(I,IR)
      A(I,IR)=A(I,IT)
   32 A(I,IT)=S
   33 S=-SQRT(SIGMA)*SIGN(1.,A(IR,IR))
      U(IR)=U(IR)-S
      H=SIGMA-A(IR,IR)*S
      DO 6 K=IR1,M
```

```
      F=0.
      DO 5 J=IR,N
    5 F=F+U(J)*A(J,K)
    6 C(K)=F/H
      A(IR,IR)=S
      DO 7 I=IR,N
      DO 7 K=IR1,M
    7 A(I,K)=A(I,K)-U(I)*C(K)
      DO 8 K=1,N
      F=0.
      DO 9 J=IR,N
    9 F=U(J)*V(J,K)+F
    8 C(K)=F/H
      DO 10 I=IR,N
      DO 10 K=1,N
   10 V(I,K)=V(I,K)-U(I)*C(K)
    2 CONTINUE
      DO 11 J=1,M
      L=M+1-J
      A(L,L)=1./A(L,L)
      L1=L-1
      DO 12 KI=1,L1
      K=L-KI
      K1=K+1
      F=0.
      DO 13 I=K1,L
   13 F=F-A(K,I)*A(I,L)
   12 A(K,L)=F/A(K,K)
   11 CONTINUE
      DO 14 K=1,N
      DO 14 I=1,M
      F=0.
      DO 15 J=I,M
   15 F=F+A(I,J)*V(J,K)
   14 V(I,K)=F
      DO 20 IJ=1,M
      I=M+1-IJ
      K=JEL(I)
      IF(I-K) 22,20,22
   22 DO 21 J=1,N
      S=V(I,J)
      V(I,J)=V(K,J)
   21 V(K,J)=S
   20 CONTINUE
      DO 24 I=1,M
      DO 24 J=1,N
   24 A(J,I)=V(I,J)
      RETURN
      END
```

```
      SUBROUTINE GENINR(NDIM,MDIM,N,M,A,V,W,C,U,JEL,EPS,IER)
C   COMPUTE THE MOORE-PENROSE GENERALIZED INVERSE OF AN N BY M
C   REAL MATRIX BY MEANS OF THE HOUSEHOLDER
C   ORTHOGONAL DECOMPOSITION
C     PARAMETERS:
C   NDIM MAXIMUM NUMBER OF ROWS OF A
C   MDIM MAXIMUM NUMBER OF COLUMNS OF A
C   N NUMBER OF ROWS OF A
C   M NUMBER OF COLUMNS OF A ( OUTPUT: RANK OF A)
C   A N BY M ARRAY OF THE MATRIX TO BE INVERTED, REPLACED BY THE
C     TRANSPOSE OF THE RESULTANT GENERALIZED INVERSE
C   V N BY N ARRAY OF AN ORTHOGONAL MATRIX (LEFT HAND FACTOR)
C   W M BY M ARRAY OF AN ORTHOGONAL MATRIX (RIGHT HAND FACTOR)
C   C WORKING VECTOR OF LENGTH N
C   U WORKING VECTOR OF LENGTH N
C   JEL PERMUTATION VECTOR OF LENGTH M
C   EPS ABSOLUTE ERROR BOUND
C   IER ERROR CODE
C     IER=0 NORMAL TERMINATION
C     IER=1 ZERO RANK
      DIMENSION A(NDIM,1),V(NDIM,1),W(MDIM,1),C(1),U(1),JEL(1)
      MIN=M
      IF(N.LT.M) MIN=N
      IM=0
      IER=0
      DO 1 I=1,N
      DO 1 J=1,N
    1 V(I,J)=(I/J)*(J/I)
      DO 38 I=1,M
      DO 38 J=1,M
   38 W(I,J)=(I/J)*(J/I)
      DO 2 IR=1,MIN
      IR1=IR+1
      SI=0.
      DO 39 IS=IR,M
      SIGMA=0.
      DO 3 I=IR,N
      C(I)=A(I,IS)
      S=C(I)
    3 SIGMA=SIGMA+S*S
      IF(SIGMA.LE.SI) GO TO 39
      SI=SIGMA
      IT=IS
      DO 40 I=IR,N
   40 U(I)=C(I)
   39 CONTINUE
      IF(SI.GE.EPS) GO TO 4
      GO TO 17
    4 JEL(IR)=IT
      SIGMA=SI
      IF(IR.EQ.IT) GO TO 18
      DO 19 I=1,N
```

```
      F=A(I,IR)
      A(I,IR)=A(I,IT)
   19 A(I,IT)=F
   18 IM=IM+1
      S=-SQRT(SIGMA)*SIGN(1.,A(IR,IR))
      U(IR)=U(IR)-S
      H=SIGMA-A(IR,IR)*S
      DO 6 K=IR1,M
      F=0.
      DO 5 J=IR,N
    5 F=F+U(J)*A(J,K)
    6 C(K)=F/H
      A(IR,IR)=S
      DO 14 I=IR1,N
   14 A(I,IR)=0.
      DO 7 I=IR,N
      DO 7 K=IR1,M
    7 A(I,K)=A(I,K)-U(I)*C(K)
      DO 8 K=1,N
      F=0.
      DO 9 J=IR,N
    9 F=F+U(J)*V(J,K)
    8 C(K)=F/H
      DO 10 I=IR,N
      DO 10 K=1,N
   10 V(I,K)=V(I,K)-U(I)*C(K)
    2 CONTINUE
   17 IF(IM.GT.0) GO TO 20
      IER=1
      RETURN
   20 DO 25 IR=1,IM
      IR1=IR+1
      U(IR)=A(IR,IR)
      F=0.
      DO 26 I=IR1,M
      U(I)=A(IR,I)
      S=U(I)
   26 F=F+S*S
      IF(F.LT.EPS) GO TO 25
      F=F+U(IR)*U(IR)
      S=-SQRT(F)*SIGN(1.,A(IR,IR))
      U(IR)=U(IR)-S
      H=-U(IR)*S
      DO 28 I=IR1,IM
      F=0.
      DO 29 K=IR,M
   29 F=F+A(I,K)*U(K)
   28 C(I)=F/H
      A(IR,IR)=S
      DO 15 I=IR1,M
   15 A(IR,I)=0.
      DO 30 I=IR1,IM
```

```
   DO 30 K=IR,M
30 A(I,K)=A(I,K)-C(I)*U(K)
   DO 31 I=1,M
   F=0.
   DO 32 K=IR,M
32 F=F+W(I,K)*U(K)
31 C(I)=F/H
   DO 33 I=1,M
   DO 33 K=IR,M
33 W(I,K)=W(I,K)-C(I)*U(K)
25 CONTINUE
   DO 11 I=1,IM
   A(I,I)=1./A(I,I)
   I1=I+1
   DO 12 L=I1,IM
   L1=L-1
   F=0.
   DO 13 J=I,L1
13 F=F-A(L,J)*A(J,I)
12 A(L,I)=F/A(L,L)
11 CONTINUE
   DO 34 I=1,M
   DO 34 J=1,IM
   F=0.
   DO 35 K=J,IM
35 F=F+W(I,K)*A(K,J)
34 W(I,J)=F
   DO 36 I=1,M
   DO 36 J=1,N
   F=0.
   DO 37 K=1,IM
37 F=F+W(I,K)*V(K,J)
36 A(J,I)=F
   DO 22 IJ=1,IM
   I=IM+1-IJ
   K=JEL(I)
   IF(I.EQ.K) GO TO 22
   DO 23 J=1,N
   S=A(J,I)
   A(J,I)=A(J,K)
23 A(J,K)=S
22 CONTINUE
   M=IM
   RETURN
   END
```

```
      SUBROUTINE SVDINV(MDIM,NDIM,M,N,EPS,TOL,U,V,Q,E)
C   COMPUTE THE MOORE-PENROSE GENERALIZED INVERSE OF AN M BY N
C   REAL MATRIX BY MEANS OF THE SINGULAR VALUE DECOMPOSITION
C     PARAMATERS:
C   MDIM MAXIMUM NUMBER OF ROWS OF U
C   NDIM MAXIMUM NUMBER OF COLUMNS OF U
C   M NUMBER OF ROWS OF U
C   N NUMBER OF COLUMNS OF U, (M.GE.N) (OUTPUT:RANK OF U)
C   EPS SMALLEST POSITIVE NUMBER FOR WHICH 1+EPS NE 1
C   TOL MACHINE TOLERANCE = ETA/EPS, ETA IS THE SMALLEST
C     POSITIVE NUMBER REPRESENTABLE IN THE MACHINE
C   U M BY N ARRAY OF THE MATRIX TO BE INVERTED, REPLACED
C     BY THE TRANSPOSE OF THE RESULTANT GENERALIZED INVERSE
C   V N BY N WORKING MATRIX
C   Q WORKING VECTOR OF LENGTH N
C   E WORKING VECTOR OF LENGTH N
      DIMENSION U(MDIM,1),V(NDIM,1),Q(1),E(1)
      G=0.
      X=0.
      DO 2 I=1,N
      E(I)=G
      S=0.
      L=I+1
      DO 3 J=I,M
      T=U(J,I)
    3 S=S+T*T
      IF(S-TOL) 4,5,5
    4 G=0.
      GO TO 6
    5 F=U(I,I)
      G=-SQRT(S)
      IF(F.LT.0.) G=SQRT(S)
      H=F*G-S
      U(I,I)=F-G
      DO 9 J=L,N
      S=0.
      DO 8 K=I,M
    8 S=S+U(K,I)*U(K,J)
      F=S/H
      DO 9 K=I,M
    9 U(K,J)=U(K,J)+F*U(K,I)
    6 Q(I)=G
      S=0.
      DO 10 J=L,N
      T=U(I,J)
   10 S=S+T*T
      IF(S-TOL) 11,12,12
   11 G=0.
      GO TO 13
   12 F=U(I,I+1)
      G=-SQRT(S)
      IF(F.LT.0.) G=SQRT(S)
```

```
      H=F*G-S
      U(I,I+1)=F-G
      DO 14 J=L,N
   14 E(J)=U(I,J)/H
      DO 15 J=L,M
      S=0.
      DO 16 K=L,N
   16 S=S+U(J,K)*U(I,K)
      DO 17 K=L,N
   17 U(J,K)=U(J,K)+S*E(K)
   15 CONTINUE
   13 Y=ABS(Q(I))+ABS(E(I))
      IF(Y.GT.X) X=Y
    2 CONTINUE
      DO 19 I1=1,N
      I=N+1-I1
      IF(G.EQ.0.) GO TO 20
      H=U(I,I+1)*G
      DO 21 J=L,N
   21 V(J,I)=U(I,J)/H
      DO 22 J=L,N
      S=0.
      DO 23 K=L,N
   23 S=S+U(I,K)*V(K,J)
      DO 24 K=L,N
   24 V(K,J)=V(K,J)+S*V(K,I)
   22 CONTINUE
   20 DO 25 J=L,N
      V(I,J)=0.
   25 V(J,I)=0.
      V(I,I)=1.
      G=E(I)
   19 L=I
      DO 27 I1=1,N
      I=N+1-I1
      L=I+1
      G=Q(I)
      DO 28 J=L,N
   28 U(I,J)=0.
      IF(G.EQ.0.) GO TO 29
      H=U(I,I)*G
      DO 30 J=L,N
      S=0.
      DO 31 K=L,M
   31 S=S+U(K,I)*U(K,J)
      F=S/H
      DO 32 K=I,M
   32 U(K,J)=U(K,J)+F*U(K,I)
   30 CONTINUE
      DO 33 J=I,M
   33 U(J,I)=U(J,I)/G
      GO TO 34
```

```
29 DO 35 J=I,M
35 U(J,I)=0.
34 U(I,I)=U(I,I)+1.
27 CONTINUE
   EPS=EPS*X
   DO 36 I1=1,N
   K=N+1-I1
37 DO 38 L2=1,K
   L=K+1-L2
   IF(ABS(E(L)).LE.EPS) GO TO 40
   IF(ABS(Q(L-1)).LE.EPS) GO TO 39
38 CONTINUE
39 C=0.
   S=1.
   L1=L-1
   DO 41 I=L,K
   F=S*E(I)
   E(I)=C*E(I)
   IF(ABS(F).LE.EPS) GO TO 40
   G=Q(I)
   H=SQRT(F*F+G*G)
   Q(I)=H
   C=G/H
   S=-F/H
   DO 42 J=1,M
   Y=U(J,L1)
   Z=U(J,I)
   U(J,L1)=Y*C+Z*S
42 U(J,I)=-Y*S+Z*C
41 CONTINUE
40 Z=Q(K)
   IF(L.EQ.K) GO TO 43
   X=Q(L)
   Y=Q(K-1)
   G=E(K-1)
   H=E(K)
   F=((Y-Z)*(Y+Z)+(G-H)*(G+H))/(2.*H*Y)
   G=SQRT(F*F+1.)
   F1=F+G
   IF(F.LT.0.) F1=F-G
   F=((X-Z)*(X+Z)+H*(Y/F1-H))/X
   C=1.
   S=1.
   LPLUS=L+1
   DO 44 I=LPLUS,K
   G=E(I)
   Y=Q(I)
   H=S*G
   G=C*G
   Z=SQRT(F*F+H*H)
   E(I-1)=Z
   C=F/Z
```

```
      S=H/Z
      F=X*C+G*S
      G=-X*S+G*C
      H=Y*S
      Y=Y*C
      DO 46 J=1,N
      X=V(J,I-1)
      Z=V(J,I)
      V(J,I-1)=X*C+Z*S
   46 V(J,I)=-X*S+Z*C
      Z=SQRT(F*F+H*H)
      Q(I-1)=Z
      C=F/Z
      S=H/Z
      F=C*G+S*Y
      X=-S*G+C*Y
      DO 47 J=1,M
      Y=U(J,I-1)
      Z=U(J,I)
      U(J,I-1)=Y*C+Z*S
   47 U(J,I)=-Y*S+Z*C
   44 CONTINUE
      E(L)=0.
      E(K)=F
      Q(K)=X
      GO TO 37
   43 IF(Z.GE.0.) GO TO 36
      Q(K)=-Z
      DO 48 J=1,N
   48 V(J,K)=-V(J,K)
   36 CONTINUE
      IR=0
      DO 49 I=1,N
      IF(Q(I).LT.EPS) Q(I)=0.
      IF(Q(I).EQ.0.) GO TO 49
      Q(I)=1./Q(I)
      IR=IR+1
   49 CONTINUE
      DO 50 J=1,N
      X=Q(J)
      DO 50 I=1,N
   50 V(I,J)=V(I,J)*X
      DO 52 I=1,M
      DO 51 J=1,N
   51 E(J)=U(I,J)
      DO 52 J=1,N
      X=0.
      DO 53 K=1,N
   53 X=X+V(J,K)*E(K)
   52 U(I,J)=X
      N=IR
      RETURN
      END
```

```
      SUBROUTINE GREINV(MDIM,M,N,A,Z,B,D,E,EPS)
C  COMPUTE THE MOORE-PENROSE GENERALIZED INVERSE OF AN M BY N
C  REAL MATRIX BY MEANS OF THE GREVILLE RECURSIVE ALGORITHM
C     PARAMETERS:
C  MDIM MAXIMUM NUMBER OF ROWS OF A
C  M NUMBER OF ROWS OF A
C  N NUMBER OF COLUMNS OF A (OUTPUT:RANK OF A)
C  A M BY N ARRAY OF THE MATRIX TO BE INVERTED
C  Z M BY N ARRAY OF THE TRANSPOSE OF THE RESULTANT
C     GENERALIZED INVERSE
C  B WORKING VECTOR OF LENGTH M
C  D WORKING VECTOR OF LENGTH N
C  E WORKING VECTOR OF LENGTH M
C  EPS ABSOLUTE ERROR BOUND
      DIMENSION A(MDIM,1),Z(MDIM,1),B(1),D(1),E(1)
      IR=0
      S=0.
      DO 1 I=1,M
      T=A(I,1)
    1 S=S+T*T
      IF(S.LT.EPS) GO TO 3
      DO 2 I=1,M
    2 Z(I,1)=A(I,1)/S
      IR=IR+1
      GO TO 5
    3 DO 4 I=1,M
    4 Z(I,1)=0.
    5 DO 17 K=2,N
      L=K-1
      DO 6 I=1,L
      T=0.
      DO 7 J=1,M
    7 T=T+Z(J,I)*A(J,K)
    6 D(I)=T
      S=0.
      DO 9 I=1,M
      T=0.
      DO 8 J=1,L
    8 T=T+A(I,J)*D(J)
      E(I)=A(I,K)-T
      T=E(I)
    9 S=S+T*T
      IF(S.LT.EPS) GO TO 11
      IR=IR+1
      DO 10 J=1,M
   10 B(J)=E(J)/S
      GO TO 18
   11 S=1.
      DO 12 I=1,L
      T=D(I)
   12 S=S+T*T
      DO 13 J=1,M
```

```
      T=0.
      DO 14 I=1,L
   14 T=T+D(I)*Z(J,I)
   13 B(J)=T/S
   18 DO 15 J=1,M
      DO 16 I=1,L
   16 Z(J,I)=Z(J,I) -D(I)*B(J)
   15 Z(J,K)=B(J)
   17 CONTINUE
      N=IR
      RETURN
      END
```

```
      SUBROUTINE GINV(NDIM,NR,NC,A,U,AFLAG,ATEMP,TOL)
C   COMPUTE THE MOORE-PENROSE GENERALIZED INVERSE OF AN NR BY NC
C   REAL MATRIX BY MEANS OF THE GRAM-SCHMIDT ORTHOGONALIZATION
C      PARAMETERS:
C   NDIM MAXIMUM NUMBER OF ROWS OF A
C   NR NUMBER OF ROWS OF A
C   NC NUMBER OF COLUMNS OF A (NC.LE.NR)
C   A NR BY NC ARRAY OF THE MATRIX TO BE INVERTED, REPLACED
C      BY THE TRANSPOSE OF THE RESULTANT GENERALIZED INVERSE
C   U NC BY NC WORKING MATRIX
C   AFLAG WORKING VECTOR OF LENGTH NC
C   ATEMP WORKING VECTOR OF LENGTH NC
C   TOL' RELATIVE ERROR BOUND
      DIMENSION A(NDIM,1),U(NDIM,1),AFLAG(1),ATEMP(1)
      DO 10 I=1,NC
      DO 10 J=1,NC
   10 U(I,J)=(I/J)*(J/I)
      FAC=DOT(NDIM,NR,A,1,1)
      FAC=1./SQRT(FAC)
      DO 15 I=1,NR
   15 A(I,1)=A(I,1)*FAC
      DO 20 I=1,NC
   20 U(I,1)=U(I,1)*FAC
      AFLAG(1)=1.
      DO 100 J=2,NC
      DOT1=DOT(NDIM,NR,A,J,J)
      JM1=J-1
      DO 50 L=1,2
      DO 30 K=1,JM1
   30 ATEMP(K)=DOT(NDIM,NR,A,J,K)
      DO 45 K=1,JM1
      DO 35 I=1,NR
   35 A(I,J)=A(I,J)-ATEMP(K)*A(I,K)*AFLAG(K)
      DO 40 I=1,NC
   40 U(I,J)=U(I,J)-ATEMP(K)*U(I,K)
   45 CONTINUE
   50 CONTINUE
      DOT2=DOT(NDIM,NR,A,J,J)
      IF(DOT2/DOT1-TOL) 55,55,70
   55 DO 60 I=1,JM1
      ATEMP(I)=0.
      DO 60 K=1,I
   60 ATEMP(I)=ATEMP(I)+U(K,I)*U(K,J)
      DO 65 I=1,NR
      A(I,J)=0.
      DO 65 K=1,JM1
   65 A(I,J)=A(I,J)-A(I,K)*ATEMP(K)*AFLAG(K)
      AFLAG(J)=0.
      FAC=DOT(NDIM,NC,U,J,J)
      FAC=1./SQRT(FAC)
      GO TO 75
   70 AFLAG(J)=1.
```

```
      FAC=1./SQRT(DOT2)
   75 DO 80 I=1,NR
   80 A(I,J)=A(I,J)*FAC
      DO 85 I=1,NC
   85 U(I,J)=U(I,J)*FAC
  100 CONTINUE
      DO 130 J=1,NC
      DO 130 I=1,NR
      FAC=0.
      DO 120 K=J,NC
  120 FAC=FAC+A(I,K)*U(J,K)
  130 A(I,J)=FAC
      RETURN
      END
      FUNCTION DOT(NDIM,NR,A,JC,KC)
C  COMPUTE THE INNER PRODUCT OF TWO COLUMNS OF A MATRIX
      DIMENSION A(NDIM,1)
      DOT=0.
      DO 5 I=1,NR
    5 DOT=DOT+A(I,JC)*A(I,KC)
      RETURN
      END
```

```
      SUBROUTINE NOREQU(MDIM,M,N,A,B,D,IND,EPS)
C   COMPUTE THE (PSEUDO)NORMAL SOLUTION OF A LINEAR SYSTEM OF
C   EQUATIONS WITH M BY N REAL RECTANGULAR MATRIX AND REAL
C   RIGHT HAND SIDE VECTOR BY MEANS OF THE GAUSSIAN ELIMINATION
C   AND THE CHOLESKY DECOMPOSITION
C     PARAMETERS:
C   MDIM MAXIMUM NUMBER OF ROWS OF A
C   M NUMBER OF ROWS OF A
C   N NUMBER OF COLUMNS OF A (OUTPUT:RANK OF A)
C   A M BY N ARRAY OF THE MATRIX OF THE LINEAR SYSTEM OF
C     EQUATIONS TO BE SOLVED
C   B THE RIGHT HAND SIDE VECTOR OF LENGTH MAX(M.N)
C     REPLACED BY THE (PSEUDO)NORMAL SOLUTION
C   D WORKING VECTOR OF LENGTH M
C   IND PERMUTATION VECTOR OF LENGTH N
C   EPS ABSOLUTE ERROR BOUND
      DIMENSION A(MDIM,1),B(1),D(1),IND(1)
      ISM=0
      K=0
      MIN=MIN0(M,N)
      DO 13 I=1,N
   13 IND(I)=I
      DO 1 IK=1,MIN
      S=0.
      DO 2 L=IK,M
      DO 2 I=IK,N
      R=ABS(A(L,I))
      IF(R-S) 2,2,3
    3 S=R
      L1=L
      I1=I
    2 CONTINUE
      IF(S.LT.EPS) GO TO 5
      K=IK
      IF(I1.EQ.K) GO TO 6
      IC=IND(I1)
      IND(I1)=IND(K)
      IND(K)=IC
      DO 10 J=1,M
      X=A(J,K)
      A(J,K)=A(J,I1)
   10 A(J,I1)=X
    6 IF(L1.EQ.K) GO TO 9
      DO 7 J=1,N
      X=A(K,J)
      A(K,J)=A(L1,J)
    7 A(L1,J)=X
      X=B(K)
      B(K)=B(L1)
      B(L1)=X
    9 H=A(K,K)
      K1=K+1
```

```
      DO 8 I=K1,M
      X=A(I,K)/H
      A(I,K)=X
      DO 8 J=K1,N
    8 A(I,J)=A(I,J)-X*A(K,J)
    1 CONTINUE
    5 DO 38 I=1,K
      I1=I+1
      Y=B(I)
      DO 39 J=I1,M
   39 Y=Y+A(J,I)*B(J)
   38 D(I)=Y
      DO 50 J=1,K
      J1=J+1
      X=1.
      DO 51 I=J1,M
      X1=A(I,J)
   51 X=X+X1*X1
   50 B(J)=X
      K2=K-1
      DO 52 J=1,K2
      J1=J+1
      DO 52 L=J1,K
      L1=L+1
      X=A(L,J)
      DO 53 I=L1,M
   53 X=X+A(I,L)*A(I,J)
   52 A(L,J)=X
      GO TO 99
   98 DO 14 I=1,K
      X=0.
      DO 15 I1=I,N
      X1=A(I,I1)
   15 X=X+X1*X1
   14 B(I)=X
      K1=K-1
      DO 16 I=1,K1
      I2=I+1
      DO 16 J=I2,K
      X=0.
      DO 17 I1=J,N
   17 X=X+A(I,I1)*A(J,I1)
   16 A(J,I)=X
   99 ISM=ISM+1
      DO 18 I=1,K
      X=B(I)
      I2=I-1
      DO 20 L=1,I2
      X1=A(I,L)
   20 X=X-X1*X1
      X=SQRT(X)
      B(I)=X
```

```
      I1=I+1
      DO 19 J=I1,K
      Y=A(J,I)
      DO 21 L=1,I2
   21 Y=Y-A(I,L)*A(J,L)
   19 A(J,I)=Y/X
   18 CONTINUE
      DO 22 I=1,K
      I2=I-1
      X=D(I)
      DO 23 J=1,I2
   23 X=X-A(I,J)*D(J)
   22 D(I)=X/B(I)
      DO 24 IC=1,K
      I=K+1-IC
      X=D(I)
      I1=I+1
      DO 25 J=I1,K
   25 X=X-A(J,I)*D(J)
   24 D(I)=X/B(I)
      IF(ISM.EQ.1) GO TO 98
      DO 26 I=1,N
      L=I
      K1=MIN0(L,K)
      X=0.
      DO 27 J=1,K1
   27 X=X+A(J,I)*D(J)
      L=IND(I)
   26 B(L)=X
      N=K
      RETURN
      END
```

REFERENCES

[1] Moore, E.H., General Analysis Part I,
Mem. Amer. Phil. Soc. Vol.1, 197-209 (1935)

[2] Penrose, R., A Generalized Inverse for Matrices
Proc. Camb. Phil. Soc. 51, 406-413 (1955)

[3] Rao, C.R. - Mitra, S.K., Generalized Inverse of Matrices and its Applications,
John Wiley and Sons Inc. 1971.

[4] Nobuo Shinozaki, Masaaki Sibuya, Kunio Tanabe, Numerical Algorithms for the Moore-Penrose Inverse of a Matrix: Direct Methods
Annals Inst. Statist. Math. 24. 193-203 (1972)

[5] Steward, G.W., On the Continuity of the Generalized Inverse,
Siam J. on Appl.Math. Vol. 17.33-45 (1967)

[6] Householder, A.S., Unitary Triangularization of a Nonsymmetric Matrix,
J. Assoc. Comp. Mach. 5. 339-342 (1958)

[7] Golub, G.H., Reinsch, C., Singular Value Decomposition and Least Squares Solutions,
Num. Math. 14, 403-420 (1970)

[8] Forsythe, G.E., Moler, C.B., Computer Solution of Linear Algebraic Systems
Engl. Cliffs, New Jersey: Prentice Hall 1967.

[9] Greville, T.N.E., Some Applications of the Pseudoinverse of a Matrix,
Siam Rev. 2., 15-22 (1960)

[10] Rust, B., Burruss, W.R., Schneeberger, C., A Simple Algorithm for Computing the Generalized Inverse of a Matrix
CACM Vol.9. Nr.5. 381-387 (1966)

C O N T E N T S