1.3 求解对称正定方程组的共轭梯度法

对称正定方程组的 LU 分解方法求解，又叫平方根法，下面介绍共轭梯度法

共轭梯度法求 AX=B 的递推计算公式如下，实际上是计算 R=B-AX 的最小值

取 **P_i** 为梯度算符。**（适用于偏微分方程和优化问题，其中矩阵 A 是正定对称稀疏矩阵）**

取初值 X_0=（0, 0, 0,……,0）, $R_0=P_0=B$

对于 i=0,1,2,…….n-1 做如下计算：

$$X_{i+1} = X_i + \alpha_i P_i$$

其中 $\alpha_i = \dfrac{(P_i, B)}{(P_i, AP_i)}$

$$R_{i+1} = B - AX_{i+1}, \quad P_{i+1} = R_{i+1} - \beta_i P_i$$

其中 $\beta_i = \dfrac{(R_{i+1}, AP_i)}{(P_i, AP_i)}$

上述过程一直做到$||R_i||<\varepsilon$ 或者 i=n-1 为止

例子：

$$A=\begin{bmatrix}5 & 7 & 6 & 5\\ 7 & 10 & 8 & 7\\ 6 & 8 & 10 & 9\\ 5 & 7 & 9 & 10\end{bmatrix}\qquad B=\begin{bmatrix}23\\ 32\\ 33\\ 31\end{bmatrix}$$

求AX=B

结果：

X[0]=1.0

X[1]=1.0

X[2]=1.0

X[3]=1.0

主程序名： AGRAD0 。子程序名：AGRAD，BRMUL

```
PROGRAM AGRAD0
DIMENSION A(4,4),B(4),X(4),P(4),R(4),S(4),Q(4)
    DOUBLE PRECISION A,B,X,P,R,S,Q
    DATA A/5.0,7.0,6.0,5.0,7.0,10.0,8.0,7.0,6.0,8.0,10.0,9.0,5.0,7.0,9.0,10.0/
    DATA B/23.0,32.0,33.0,31.0/
    EPS=0.000001
    CALL AGRAD(A,4,B,EPS,X,P,R,S,Q)
    WRITE(*,*)
    WRITE(*,10) (X(I),I=1,4)
    WRITE(*,*)
10  FORMAT(1X,D15.6)
    END PROGRAM
```

运行结果:

X(1)=1.000000e+00

X(2)=1.000000e+00

X(3)=1.000000e+00

X(4)=1.000000e+00

```
SUBROUTINE AGRAD(A,N,B,EPS,X,P,R,S,Q)
   DIMENSION A(N,N),B(N),X(N),P(N),R(N),S(N),Q(N)
   DOUBLE PRECISION A,B,X,P,R,S,Q,ALPHA,BETA,D,E
   DO 10 I=1,N
      X(I)=0.0
      P(I)=B(I)
      R(I)=B(I)
10 CONTINUE
```

```
      I=1
20    CALL BRMUL(A,P,N,N,1,S)
      D=0.0
      E=0.0
      DO 30 K=1,N
         D=D+P(K)*B(K)
         E=E+P(K)*S(K)
30    CONTINUE
      ALPHA=D/E
      DO 40 K=1,N
40    X(K)=X(K)+ALPHA*P(K)
      CALL BRMUL(A,X,N,N,1,Q)
      D=0.0
      DO 50 K=1,N
         R(K)=B(K)-Q(K)
```

```
      D=D+R(K)*S(K)
50  CONTINUE
    BETA=D/E
    D=0.0
    DO 55 K=1,N
55  D=D+R(K)*R(K)
    D=SQRT(D)
    IF (D.LT.EPS) RETURN
    DO 60 K=1,N
60  P(K)=R(K)-BETA*P(K)
    I=I+1
    IF (I.LE.N) GOTO 20
    RETURN
    END SUBROUTINE
```

```
      SUBROUTINE BRMUL(A,B,M,N,K,C)
      DIMENSION A(M,N),B(N,K),C(M,K)
      DOUBLE PRECISION A,B,C
      DO 50 I=1,M
      DO 50 J=1,K
         C(I,J)=0.0
         DO 10 L=1,N
            C(I,J)=C(I,J)+A(I,L)*B(L,J)
10       CONTINUE
50    CONTINUE
      RETURN
      END SUBROUTINE
```

1.4 求解超定方程组的最小二乘法

范数的定义： $\|E\|_p = \left(\sum_{i=1}^{n} |E_i|^2\right)^{\frac{1}{p}}$

下面简单介绍一下**“最小二乘法”**

考虑超定方程组（超定指未知数小于方程个数）：

$$\sum_{j=1}^{n} X_{ij}\beta_j = y_i, i = 1, 2, 3, \ldots, m$$

其中， m 代表样本数， n 代表参数维度，将上式向量化得到：

$$X\beta = y$$

$$X = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & & \vdots \\ X_{m1} & X_{m2} & \cdots & X_{mn} \end{bmatrix}, \beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}, y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

为了求解 β 的最佳估计值 $\hat{\beta}$ ，可采用最小二乘法，问题转化如下：

$$S(\beta) = ||X\beta - y||$$

$$\hat{\beta} = argmin(S(\beta))$$

最小二乘法就是利用

$$\sum_i |y_i - \overline{y}|^2 < \varepsilon$$

的方法

通过对 $S(\beta)$ 进行微分求最值，可得：

$$X^T X\beta = X^T y$$

如果矩阵 $X^T X$ 非奇异，则 β 有唯一解：

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

因此问题的关键是如何求矩阵的逆阵的问题

但是这里由于要求矩阵倒数，计算上存在一定困难，这里如果采用QR分解可以使得问题简单很多。

首先对A进行QR分解，即$A = QR$,其中$QQ^T = I$,R为上三角矩阵

$$x = (A^T A)^{-1} A^T b$$

$$(QR)^T (QR)x = (QR)^T b$$

$$R^T Q^T QRx = R^T Q^T b$$

$$Rx = Q^T b$$

$$x = R^{-1} Qb$$

其中 R为上三角矩阵,求逆相对容易很多，规避了直接对$(A^T A)^{-1}$求逆复杂度高的问题。

化解到上式也可以对倒数第二式子 $Rx=Q^Tb$ 采用最小二乘法计算，下面就介绍采用 QR 分解的方法采用最小二乘法求解

1.5 求线性最小二乘问题的豪斯荷尔德变换法

一、功能

用豪斯荷尔德(Householder)变换求解线性最小二乘问题。

二、方法说明

设超定方程组为 $AX=B$,其中 A 为 $m\times n(m\geqslant n)$ 列线性无关的矩阵,X 为 n 维列向量,B 为 m 维列向量。

用豪斯荷尔德变换将 A 进行 QR 分解。即

$$A = QR$$

其中 Q 为 $m\times m$ 的正交矩阵,R 为上三角矩阵。

具体分解下下面章节

设 $E=B-AX$,用 Q^T 乘上式两端得

$$Q^TE = Q^TB - Q^TAX = Q^TB - RX$$

因为 Q^T 为正交矩阵,所以有

$$\| E \|_2^2 = \| Q^TE \|_2^2 = \| Q^TB - RX \|_2^2$$

若令

$$Q^TB = \begin{bmatrix} C \\ D \end{bmatrix}, RX = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} X$$

其中 C 为 n 维列向量，D 为 $m-n$ 维列向量，R_1 为 $n\times n$ 上三角方阵，0 为 $(m-n)\times n$ 的零矩阵。则有

$$\| E \|_2^2 = \| C - R_1 X \|_2^2 + \| D \|_2^2$$

显然，当 X 满足 $R_1X=C$ 时，$\| E \|_2^2$ 将取最小值。

由上所述，求解线性最小二乘问题 $AX=B$ 的步骤如下：

(1) 对 A 进行 QR 分解。即 $A=QR$，其中 Q 为 $m\times m$ 的正交矩阵，R 为右上三角矩阵。且令

$$R=\begin{bmatrix} R_1 \\ 0 \end{bmatrix}$$

其中 R_1 为 $n\times n$ 的上三角方阵。

(2) 计算

$$\begin{bmatrix} C \\ D \end{bmatrix} = Q^T B$$

其中 C 为 n 维列向量。

(3) 利用回代求解方程组 $R_1X=C$

计算如下超定方程的解

$$
\begin{aligned}
x_0 + x_1 - x_2 &= 2 \\
2x_0 + x_1 \quad &= -3 \\
x_0 - x_1 \quad &= 1 \\
-x_0 + 2x_1 + x_2 &= 4
\end{aligned}
$$

主程序 AGMQR0。 子程序为： AGMQR，BMAQR

PROGRAM AGMQR0

```
DIMENSION A(4,3),B(4),Q(4,4),C(3)
DOUBLE PRECISION A,B,Q,C
DATA A/1.0,2.0,1.0,-1.0,1.0,1.0,-1.0,2.0,-1.0,0.0,0.0,1.0/
DATA B/2.0,-3.0,1.0,4.0/
M=4
```

```
 N=3
 CALL AGMQR(A,M,N,B,Q,L,C)
 IF (L.NE.0) THEN
    WRITE(*,*)
    WRITE(*,10) (I,B(I),I=1,N)
    WRITE(*,*)
    WRITE(*,20)
    WRITE(*,30) ((Q(I,J),J=1,M),I=1,M)
    WRITE(*,*)
    WRITE(*,40)
    WRITE(*,50) ((A(I,J),J=1,N),I=1,M)
    WRITE(*,*)
ELSE
 WRITE(*,*)    'MAT A is Singular matrix:'
END IF
```

```
10 FORMAT(1X,'X(',I2,' )=',D15.6)
20 FORMAT(1X,'MAT Q IS:')
30 FORMAT(1X,4D15.6)
40 FORMAT(1X,'MAT R IS:')
50 FORMAT(1X,3D15.6)
   END PROGRAM
```

运行结果：

```
X(1)= -1.190476e+00
X(2)=    9.523810e-01
X(3)= -6.666667e-01

MAT Q IS

MAT R IS:
```

```
SUBROUTINE AGMQR(A,M,N,B,Q,L,C)
      DIMENSION A(M,N),B(M),Q(M,M),C(N)
      DOUBLE PRECISION A,B,Q,C,D
      CALL BMAQR(A,M,N,Q,L)
      IF (L.EQ.0) RETURN
      DO 20 I=1,N
         D=0.0
         DO 10 J=1,M
10       D=D+Q(J,I)*B(J)
         C(I)=D
20    CONTINUE
      B(N)=C(N)/A(N,N)
      DO 40 I=N-1,1,-1
         D=0.0
         DO 30 J=I+1,N
```

```
30      D=D+A(I,J)*B(J)
        B(I)=(C(I)-D)/A(I,I)
40  CONTINUE
    RETURN
    END SUBROUTINE


    SUBROUTINE BMAQR(A,M,N,Q,L)
    DIMENSION A(M,N),Q(M,M)
    DOUBLE PRECISION A,Q,ALPHA,T,U
    IF (M.LT.N) THEN
       L=0
       WRITE(*,40)
       RETURN
    END IF
```

```
40 FORMAT(1X,'    FAIL')
   DO 10 I=1,M
   DO 10 J=1,M
      Q(I,J)=0.0
      IF (I.EQ.J) Q(I,J)=1.0
10 CONTINUE
   NN=N
   IF (M.EQ.N) NN=M-1
   DO 200 K=1,NN
      U=0.0
      DO 20 I=K,M
         IF (ABS(A(I,K)).GT.U) U=ABS(A(I,K))
20    CONTINUE
      ALPHA=0.0
      DO 30 I=K,M
```

```
      T=A(I,K)/U
      ALPHA=ALPHA+T*T
30    CONTINUE
      IF (A(K,K).GT.0.0) U=-U
      ALPHA=U*SQRT(ALPHA)
      IF (ABS(ALPHA)+1.0.EQ.1.0) THEN
         L=0
         WRITE(*,40)
         RETURN
      END IF
      U=SQRT(2.0*ALPHA*(ALPHA-A(K,K)))
      IF (U+1.0.NE.1.0) THEN
         A(K,K)=(A(K,K)-ALPHA)/U
         DO 50 I=K+1,M
50       A(I,K)=A(I,K)/U
```

```
      DO 80 J=1,M
        T=0.0
        DO 60 L=K,M
60      T=T+A(L,K)*Q(L,J)
        DO 70 I=K,M
70      Q(I,J)=Q(I,J)-2.0*T*A(I,K)
80    CONTINUE
      DO 110 J=K+1,N
        T=0.0
        DO 90 L=K,M
90      T=T+A(L,K)*A(L,J)
        DO 100 I=K,M
100        A(I,J)=A(I,J)-2.0*T*A(I,K)
110     CONTINUE
      A(K,K)=ALPHA
```

```
          DO 120 I=K+1,M
120          A(I,K)=0.0
        END IF
200     CONTINUE
     L=1
     DO 210 I=1,M-1
     DO 210 J=I+1,M
        T=Q(I,J)
        Q(I,J)=Q(J,I)
        Q(J,I)=T
210     CONTINUE
     RETURN
```

END SUBROUTINE

1.6 求线性最小二乘问题的广义逆法

利用广义逆求超定方程组 AX=B 的最小二乘解，A 为 m×n（m≥n）的矩阵

方法说明：

首先对矩阵 A 进行奇异值分解（详细分解见矩阵一节）

$$A = U\begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} V^T$$

U 是 m×m 列正交矩阵，V 都是 n×n 列正交矩阵。

然后利用奇异值分解计算 A 的广义逆 A^+

$$A^+ = V\begin{bmatrix} \Sigma^{-1} & 0 \\ 0 & 0 \end{bmatrix} U^T$$

利用广义逆求超定方程组的最小二乘解，即

$$X = A^+ B$$

利用子程序完成特定例子的计算。

主程序名：AGMIV0.　　子程序：需要 AGMIV, BGINV,BMUAV,SSS 四个子程序

```
PROGRAM AGMIV0
  DIMENSION A(4,3),U(4,4),V(3,3),B(4),C(3,4),X(3)
  DIMENSION S(5),E(5),WORK(5)
  DOUBLE PRECISION A,U,V,B,C,X,S,E,WORK
  DATA A/1.0,2.0,1.0,-1.0,1.0,1.0,-1.0,2.0,-1.0,2*0.0,1.0/
  DATA B/2.0,-3.0,1.0,4.0/
  M=4
  N=3
  KA=5      ! Ka=max（m,n）+1
  EPS=0.000001
  WRITE(*,*)
  WRITE(*,10)
10 FORMAT(1X,'MAT A IS:')
  WRITE(*,200) ((A(I,J),J=1,N),I=1,M)
  CALL AGMIV(M,N,A,B,C,X,L,EPS,U,V,KA,S,E,WORK)
```

```
! 对矩阵 A 进行奇异值分解的处理，返回值是非零表示正常返回，如果是零表示分解失败，
!  该程序调用了子程序 BGINV 完成奇异值分解的工作
   IF (L.EQ.0) THEN
     WRITE(*,*)
     WRITE(*,50)
50   FORMAT(1X,'MAT A+ IS:')
     WRITE(*,60) ((C(I,J),J=1,M),I=1,N)
60   FORMAT(1X,4D13.6)
     WRITE(*,*)
          WRITE(*,*) 'THE SOLUTION TO THE LEAST SQUARES PROBLEM IS:'
          WRITE(*,100) (I,X(I),I=1,N)
100     FORMAT(1X,'X(',I2,')=',D15.6)
   END IF
   WRITE(*,*)
   CALL BGINV(N,M,C,A,L,EPS,U,V,KA,S,E,WORK)
```

```
      IF (L.EQ.0) THEN
        WRITE(*,70)
70      FORMAT(1X,'MAT A++ IS:')
        WRITE(*,200) ((A(I,J),J=1,N),I=1,M)
200        FORMAT(1X,3D15.6)
        WRITE(*,*)
      END IF
      END PROGRAM
```

运行结果是：

X(1)= -1.190476e+00

X(2)=   9.523810e-01

X(3)= -6.666667e-01

MAT A++ IS：

显示一个 3*4 的矩阵

SUBROUTINE AGMIV(M,N,A,B,AA,X,L,EPS,U,V,KA,S,E,WORK)

!EPS 奇异值分解的精度控制。A是超定方程的系数矩阵，AA是广义逆矩阵，B是超定方程右边的常数向量。M，N表示维度X是存放超定方程最小二乘解。U是奇异值分解的左边的正交U矩阵，V是奇异值分解的右边正交V矩阵。Ka=max（m,n）+1。 S,E,WORK是ka维度的工作数组。

```
      DIMENSION A(M,N),U(M,M),V(N,N),B(M),AA(N,M),X(N)
      DIMENSION S(KA),E(KA),WORK(KA)
      DOUBLE PRECISION A,U,V,B,AA,X,S,E,WORK
      CALL BMUAV(A,M,N,U,V,L,EPS,KA,S,E,WORK)
      IF (L.EQ.0) THEN
        K=1
10      IF (A(K,K).NE.0.0) THEN
          K=K+1
          IF (K.LE.MIN(M,N)) GOTO 10
        END IF
        K=K-1
```

```
      IF (K.NE.0) THEN
        DO 40 I=1,N
        DO 40 J=1,M
          AA(I,J)=0.0
          DO 30 II=1,K
30        AA(I,J)=AA(I,J)+V(II,I)*U(J,II)/A(II,II)
40      CONTINUE
      END IF
      DO 80 I=1,N
        X(I)=0.0
        DO 70 J=1,M
70      X(I)=X(I)+AA(I,J)*B(J)
80    CONTINUE
    END IF
    RETURN
```

```
      END SUBROUTINE

!  奇异值分解是否可行的判断子程序，其中调用了真正做奇异值分解的子程序BMUAV
SUBROUTINE BGINV(M,N,A,AA,L,EPS,U,V,KA,S,E,WORK)
      DIMENSION A(M,N),U(M,M),V(N,N),AA(N,M)
      DIMENSION S(KA),E(KA),WORK(KA)
      DOUBLE PRECISION A,U,V,AA,S,E,WORK
      CALL BMUAV(A,M,N,U,V,L,EPS,KA,S,E,WORK)
      IF (L.EQ.0) THEN
        K=1
10      IF (A(K,K).NE.0.0) THEN
          K=K+1
          IF (K.LE.MIN(M,N)) GOTO 10
        END IF
        K=K-1
```

```
        IF (K.NE.0) THEN
          DO 40 I=1,N
          DO 40 J=1,M
            AA(I,J)=0.0
            DO 30 II=1,K
30          AA(I,J)=AA(I,J)+V(II,I)*U(J,II)/A(II,II)
40        CONTINUE
        END IF
      END IF
      RETURN
      END
```

！ 奇异值分解的子程序

SUBROUTINE BMUAV(A,M,N,U,V,L,EPS,KA,S,E,WORK)

```
      DIMENSION A(M,N),U(M,M),V(N,N),S(KA),E(KA),WORK(KA)
```

```
      DOUBLE PRECISION A,U,V,S,E,D,WORK,DD,F,G,CS,SN,      SHH,SK,EK,B,C,SM,SM1,EM1
      IT=60
      K=N
      IF (M-1.LT.N) K=M-1
      L=M
      IF (N-2.LT.M) L=N-2
      IF (L.LT.0) L=0
      LL=K
      IF (L.GT.K) LL=L
      IF (LL.GE.1) THEN
        DO 150 KK=1,LL
          IF (KK.LE.K) THEN
            D=0.0
            DO 10 I=KK,M
10          D=D+A(I,KK)*A(I,KK)
```

```
            S(KK)=SQRT(D)
            IF (S(KK).NE.0.0) THEN
                  IF (A(KK,KK).NE.0.0) S(KK)=SIGN(S(KK),A(KK,KK))
              DO 20 I=KK,M
20            A(I,KK)=A(I,KK)/S(KK)
              A(KK,KK)=1.0+A(KK,KK)
            END IF
            S(KK)=-S(KK)
          END IF
          IF (N.GE.KK+1) THEN
            DO 50 J=KK+1,N
              IF ((KK.LE.K).AND.(S(KK).NE.0.0)) THEN
                D=0.0
                DO 30 I=KK,M
30              D=D+A(I,KK)*A(I,J)
```

```
                D=-D/A(KK,KK)
                DO 40 I=KK,M
40              A(I,J)=A(I,J)+D*A(I,KK)
             END IF
             E(J)=A(KK,J)
50        CONTINUE
       END IF
       IF (KK.LE.K) THEN
          DO 60 I=KK,M
60        U(I,KK)=A(I,KK)
       END IF
       IF (KK.LE.L) THEN
          D=0.0
          DO 70 I=KK+1,N
70        D=D+E(I)*E(I)
```

```
      E(KK)=SQRT(D)
      IF (E(KK).NE.0.0) THEN
         IF (E(KK+1).NE.0.0) E(KK)=SIGN(E(KK),E(KK+1))
         DO 80 I=KK+1,N
80       E(I)=E(I)/E(KK)
         E(KK+1)=1.0+E(KK+1)
      END IF
      E(KK)=-E(KK)
      IF ((KK+1.LE.M).AND.(E(KK).NE.0.0)) THEN
         DO 90 I=KK+1,M
90       WORK(I)=0.0
         DO 110 J=KK+1,N
            DO 100 I=KK+1,M
100            WORK(I)=WORK(I)+E(J)*A(I,J)
110         CONTINUE
```

```
              DO 130 J=KK+1,N
                DO 120 I=KK+1,M
120               A(I,J)=A(I,J)-WORK(I)*E(J)/E(KK+1)
130             CONTINUE
            END IF
            DO 140 I=KK+1,N
140           V(I,KK)=E(I)
          END IF
150       CONTINUE
      END IF
      MM=N
      IF (M+1.LT.N) MM=M+1
      IF (K.LT.N) S(K+1)=A(K+1,K+1)
      IF (M.LT.MM) S(MM)=0.0
      IF (L+1.LT.MM) E(L+1)=A(L+1,MM)
```

```
      E(MM)=0.0
      NN=M
      IF (M.GT.N) NN=N
      IF (NN.GE.K+1) THEN
        DO 190 J=K+1,NN
          DO 180 I=1,M
180         U(I,J)=0.0
          U(J,J)=1.0
190     CONTINUE
      END IF
      IF (K.GE.1) THEN
        DO 250 LL=1,K
          KK=K-LL+1
          IF (S(KK).NE.0.0) THEN
            IF (NN.GE.KK+1) THEN
```

```
          DO 220 J=KK+1,NN
            D=0.0
            DO 200 I=KK,M
200           D=D+U(I,KK)*U(I,J)/U(KK,KK)
            D=-D
            DO 210 I=KK,M
210           U(I,J)=U(I,J)+D*U(I,KK)
220         CONTINUE
        END IF
        DO 225 I=KK,M
225       U(I,KK)=-U(I,KK)
        U(KK,KK)=1.0+U(KK,KK)
        IF (KK-1.GE.1) THEN
          DO 230 I=1,KK-1
230         U(I,KK)=0.0
```

```
          END IF
        ELSE
          DO 240 I=1,M
240         U(I,KK)=0.0
          U(KK,KK)=1.0
        END IF
250    CONTINUE
    END IF
    DO 300 LL=1,N
      KK=N-LL+1
      IF ((KK.LE.L).AND.(E(KK).NE.0.0)) THEN
        DO 280 J=KK+1,N
          D=0.0
          DO 260 I=KK+1,N
260         D=D+V(I,KK)*V(I,J)/V(KK+1,KK)
```

```
            D=-D
            DO 270 I=KK+1,N
270             V(I,J)=V(I,J)+D*V(I,KK)
280          CONTINUE
        END IF
        DO 290 I=1,N
290       V(I,KK)=0.0
        V(KK,KK)=1.0
300     CONTINUE
      DO 305 I=1,M
      DO 305 J=1,N
305     A(I,J)=0.0
      M1=MM
      IT=60
310     IF (MM.EQ.0) THEN
```

```
      L=0
      IF (M.GE.N) THEN
        I=N
      ELSE
        I=M
      END IF
      DO 315 J=1,I-1
        A(J,J)=S(J)
        A(J,J+1)=E(J)
315       CONTINUE
      A(I,I)=S(I)
      IF (M.LT.N) A(I,I+1)=E(I)
      DO 314 I=1,N-1
        DO 313 J=I+1,N
          D=V(I,J)
```

```
          V(I,J)=V(J,I)
          V(J,I)=D
313        CONTINUE
314      CONTINUE
      RETURN
    END IF
    IF (IT.EQ.0) THEN
      L=MM
      IF (M.GE.N) THEN
        I=N
      ELSE
        I=M
      END IF
      DO 316 J=1,I-1
        A(J,J)=S(J)
```

```
            A(J,J+1)=E(J)
316         CONTINUE
          A(I,I)=S(I)
          IF (M.LT.N) A(I,I+1)=E(I)
          DO 318 I=1,N-1
            DO 317 J=I+1,N
              D=V(I,J)
              V(I,J)=V(J,I)
              V(J,I)=D
317           CONTINUE
318         CONTINUE
          RETURN
        END IF
        KK=MM
320     KK=KK-1
```

```
      IF (KK.NE.0) THEN
        D=ABS(S(KK))+ABS(S(KK+1))
        DD=ABS(E(KK))
        IF (DD.GT.EPS*D) GOTO 320
        E(KK)=0.0
      END IF
      IF (KK.EQ.MM-1) THEN
        KK=KK+1
        IF (S(KK).LT.0.0) THEN
          S(KK)=-S(KK)
          DO 330 I=1,N
330         V(I,KK)=-V(I,KK)
        END IF
335     IF (KK.NE.M1) THEN
          IF (S(KK).LT.S(KK+1)) THEN
```

```
      D=S(KK)
      S(KK)=S(KK+1)
      S(KK+1)=D
      IF (KK.LT.N) THEN
        DO 340 I=1,N
          D=V(I,KK)
          V(I,KK)=V(I,KK+1)
          V(I,KK+1)=D
340       CONTINUE
      END IF
      IF (KK.LT.M) THEN
        DO 350 I=1,M
          D=U(I,KK)
          U(I,KK)=U(I,KK+1)
          U(I,KK+1)=D
```

```
350          CONTINUE
        END IF
        KK=KK+1
        GOTO 335
      END IF
    END IF
    IT=60
    MM=MM-1
    GOTO 310
  END IF
  KS=MM+1
360   KS=KS-1
  IF (KS.GT.KK) THEN
    D=0.0
    IF (KS.NE.MM) D=D+ABS(E(KS))
```

```
    IF (KS.NE.KK+1) D=D+ABS(E(KS-1))
    DD=ABS(S(KS))
    IF (DD.GT.EPS*D) GOTO 360
    S(KS)=0.0
END IF
IF (KS.EQ.KK) THEN
    KK=KK+1
    D=ABS(S(MM))
    IF (ABS(S(MM-1)).GT.D) D=ABS(S(MM-1))
    IF (ABS(E(MM-1)).GT.D) D=ABS(E(MM-1))
    IF (ABS(S(KK)).GT.D) D=ABS(S(KK))
    IF (ABS(E(KK)).GT.D) D=ABS(E(KK))
    SM=S(MM)/D
    SM1=S(MM-1)/D
    EM1=E(MM-1)/D
```

```
SK=S(KK)/D

EK=E(KK)/D

B=((SM1+SM)*(SM1-SM)+EM1*EM1)/2.0

C=SM*EM1

C=C*C

SHH=0.0

IF ((B.NE.0.0).OR.(C.NE.0.0)) THEN

   SHH=SQRT(B*B+C)

   IF (B.LT.0.0) SHH=-SHH

   SHH=C/(B+SHH)

END IF

F=(SK+SM)*(SK-SM)-SHH

G=SK*EK

DO 400 I=KK,MM-1

   CALL SSS(F,G,CS,SN)
```

```
      IF (I.NE.KK) E(I-1)=F
      F=CS*S(I)+SN*E(I)
      E(I)=CS*E(I)-SN*S(I)
      G=SN*S(I+1)
      S(I+1)=CS*S(I+1)
      IF ((CS.NE.1.0).OR.(SN.NE.0.0)) THEN
        DO 370 J=1,N
          D=CS*V(J,I)+SN*V(J,I+1)
          V(J,I+1)=-SN*V(J,I)+CS*V(J,I+1)
          V(J,I)=D
370       CONTINUE
      END IF
      CALL SSS(F,G,CS,SN)
      S(I)=F
      F=CS*E(I)+SN*S(I+1)
```

```
      S(I+1)=-SN*E(I)+CS*S(I+1)
      G=SN*E(I+1)
      E(I+1)=CS*E(I+1)
      IF (I.LT.M) THEN
        IF ((CS.NE.1.0).OR.(SN.NE.0.0)) THEN
          DO 380 J=1,M
            D=CS*U(J,I)+SN*U(J,I+1)
            U(J,I+1)=-SN*U(J,I)+CS*U(J,I+1)
            U(J,I)=D
380         CONTINUE
        END IF
      END IF
400   CONTINUE
    E(MM-1)=F
    IT=IT-1
```

```
    GOTO 310
END IF
IF (KS.EQ.MM) THEN
    KK=KK+1
    F=E(MM-1)
    E(MM-1)=0.0
    DO 420 LL=KK,MM-1
        I=MM+KK-LL-1
        G=S(I)
        CALL SSS(G,F,CS,SN)
        S(I)=G
        IF (I.NE.KK) THEN
            F=-SN*E(I-1)
            E(I-1)=CS*E(I-1)
        END IF
```

```
      IF ((CS.NE.1.0).OR.(SN.NE.0.0)) THEN
        DO 410 J=1,N
          D=CS*V(J,I)+SN*V(J,MM)
          V(J,MM)=-SN*V(J,I)+CS*V(J,MM)
          V(J,I)=D
410       CONTINUE
      END IF
420   CONTINUE
    GOTO 310
  END IF
  KK=KS+1
  F=E(KK-1)
  E(KK-1)=0.0
  DO 450 I=KK,MM
    G=S(I)
```

```
      CALL SSS(G,F,CS,SN)
      S(I)=G
      F=-SN*E(I)
      E(I)=CS*E(I)
      IF ((CS.NE.1.0).OR.(SN.NE.0.0)) THEN
        DO 430 J=1,M
          D=CS*U(J,I)+SN*U(J,KK-1)
          U(J,KK-1)=-SN*U(J,I)+CS*U(J,KK-1)
          U(J,I)=D
430       CONTINUE
      END IF
450   CONTINUE
    GOTO 310
    END
```

```
SUBROUTINE SSS(F,G,CS,SN)

DOUBLE PRECISION F,G,CS,SN,D,R

IF ((ABS(F)+ABS(G)).EQ.0.0) THEN

   CS=1.0

   SN=0.0

   D=0.0

ELSE

   D=SQRT(F*F+G*G)

   IF (ABS(F).GT.ABS(G)) D=SIGN(D,F)

   IF (ABS(G).GE.ABS(F)) D=SIGN(D,G)

   CS=F/D

   SN=G/D

END IF

R=1.0

IF (ABS(F).GT.ABS(G)) THEN
```

```
    R=SN
ELSE
    IF (CS.NE.0.0) R=1.0/CS
END IF
F=D
G=R
RETURN
END
```