```
comment Calculation of resultant lengths and mean directions;

if abs(n) < 10-9 then ifault := 1 else
  begin
  p[7] := c := c / n; p[8] := s := s / n;
  p[9] := a2 := a2 / n; p[10] := b2 := b2 / n;
  p[5] := rb := sqrt(c X c + s X s);
  p[6] := r2b := sqrt(a2 X a2 + b2 X b2); p[4] := rb X n;
  if rb < 10-8 then ifault := 4
  else p[2] := if s ≥ 0.0 then arccos(c / rb)
  else 6.2831853072 - arccos(c / rb);
  if r2b < 10-8 then ifault := ifault + 3
  else p[3] := if b2 ≥ 0.0 then arccos(a2 / r2b)
  else 6.2831853072 - arccos(a2 / r2b);

  comment Calculation of maximum likelihood estimate of concentration
  parameter. The values of the lower and upper bound parameters in
  HYP are chosen in such a way as to provide reasonable initial
  bounds, whereas still ensuring that an error exit can never occur;

  if rb > 0.99999999 then ifault := 2 else
  if rb < 0.005 then
  p[11] := rb X (rb X rbX (5.0 X rb X rb + 6.0) + 12.0) / 6.0 else
  if rb > 0.995 then p[11] := 1.0 / (rb X (rb X (rb - 4.0) + 3.0))
  else
    begin
    if rb < 0.94 then
      begin
      c := 2.0 X rb; s := 10.0 X rb
      end
    else
      begin
      c := 8.6; s := 101.0
      end;
    HYP(a2, d, c, s, 10-8, exit); p[11] := a2
    end
  end;
exit:
  end circstat
```

Algorithm AS 82

The Determinant of an Orthogonal Matrix

By J. C. GOWER

Rothamsted Experimental Station

Keywords: ORTHOGONAL MATRICES; ROTATION; REFLECTION; HOUSEHOLDER TRANSFORMS; MULTIVARIATE ANALYSIS

LANGUAGE

ISO Fortran

DESCRIPTION AND PURPOSE

The determinant $|\mathbf{Q}|$ of an orthogonal matrix $\mathbf{Q}$ of order n is either plus or minus one. If it is plus one, $\mathbf{Q}$ represents a rotation in n dimensions; if minus one, $\mathbf{Q}$ represents a product of a rotation with a reflection (Vitali, 1928). Orthogonal matrices are important in multivariate analysis (for example in principal components analysis, and in Procrustes fitting problems, Schonemann and Carroll, 1970; Gower 1971, 1975) and sometimes it is necessary to distinguish a pure rotation from a rotation with a reflection. Eigenvector and singular value decomposition subroutines could probably include this information with little extra programming, but do not do so. This algorithm evaluates $|\mathbf{Q}|$ using a sequence of Householder transforms (see, for example, Wilkinson, 1965, p. 235). If $\mathbf{H}_1$ is a Householder transform that annihilates the first column of $\mathbf{Q}$ except for the diagonal element, then

$$\mathbf{H}_1 \mathbf{Q} = \begin{pmatrix} e_1 & \mathbf{0}' \\ \mathbf{0} & \mathbf{Q}_1 \end{pmatrix}. \tag{1}$$

The first element e_1 is ± 1 because $\mathbf{H}_1 \mathbf{Q}$ is orthogonal and therefore the sum of squares of every column is unity. The remainder of the first row is zero because all columns are orthogonal to the first. The (i, j)th element of the orthogonal matrix $\mathbf{Q}_1$ is given by

$$q_{ij}^{(1)} = q_{ij} \pm q_{i1} q_{1j}/(1 \mp q_{11}). \tag{2}$$

In (2) if the upper (lower) set of signs is taken, e_1 is positive (negative). The signs are chosen so that $1 \mp q_{11}$ is furthest from zero, as this gives best accuracy.

This process can be continued until the right-hand side is diagonal, giving

$$\mathbf{H}_n \mathbf{H}_{n-1} \dots \mathbf{H}_1 \mathbf{Q} = \operatorname{diag}(e_1, e_2, \dots, e_n), \tag{3}$$

where $e_i = \pm 1$, $i = 1, 2, \dots, n$. Hence,

$$|\mathbf{Q}| = \prod_{i=1}^{n} e_i |\mathbf{H}_i|. \tag{4}$$

If at any stage the leading diagonal element $q_{ii}^{(i-1)}$ is ± 1 then $\mathbf{H}_i = \mathbf{I}$ and $\mathbf{Q}_i$ is unchanged; otherwise Householder transforms with $|\mathbf{H}_i| = -1$ are used. Thus $\mathbf{Q}$ is expressed as the product of elementary transforms, each of whose determinants is known to be $+1$ or -1. The product terms are easily accumulated in the course of the algorithm.

Structure

SUBROUTINE DETQ(A, N, NN, D, IFAULT)

Formal parameters

A	Real array (*NN*)	input:	orthogonal matrix with N^2 elements stored by rows or columns as a one-way array
		output:	*A* is overwritten
N	Integer	input:	order of matrix *A*
NN	Integer	input:	N^2 the size of *A* stored as a one-way array
D	Real	output:	$\|A\|$
IFAULT	Integer	output:	= 0 if no error detected = 1 if an error described under *failure indication* is detected

Failure Indication and Restriction

If $\mathbf{Q}$ is orthogonal no diagonal element $q_{ii}^{(i-1)}$ can exceed unity in modulus and the penultimate transformation $\mathbf{H}_{n-1}$ must give a scalar $\mathbf{Q}_{n-1}$ whose element $q_{nn}^{(n-1)}$ is ± 1, except for possible round-off. If either check fails **Q** cannot be orthogonal and *IFAULT* is set to 1. These checks are necessary but not sufficient for orthogonality, so if the check does not fail *IFAULT* is set to zero but **Q** need not be orthogonal. Any more stringent test needs to check that every first row of $\mathbf{H}_{i+1}\mathbf{Q}_i$ is of the form $(\pm 1, \mathbf{0}')$ but this would appreciably add to the complexity of the algorithm. For complete safety the algorithm should be used only when **Q** is known to be orthogonal. In this case the check can fail only because of accumulated errors. *IFAULT* is also set to 1 if $N \leqslant 0$ or if $NN \neq N^2$. *D* is not set by the subroutine if a fault is detected.

Time and Accuracy

About $\frac{1}{3}n^3$ real multiplications, $\frac{1}{3}n^3$ real additions/subtractions and another $\frac{1}{3}n^3$ integer additions are needed. If **Q** is orthogonal, accuracy should be good, as throughout the computation all elements remain in the range $-1 \leqslant q_{ij}^{(k)} \leqslant 1$ and division is always by a number whose modulus lies between 1 and 2. This means that the computation of (2) does not require double-precision working. Determinants of Helmuth matrices up to order 101 were tested successfully. In tests, ± 1 is replaced by $\pm(1 \pm TOL)$ and *TOL* set in a *DATA* statement to 0·0001. The setting of *IFAULT* reveals those cases where there is any doubt about the accuracy.

References

Gower, J. C. (1971). Statistical methods of comparing different multivariate analyses of the same data. In *Mathematics in the Archaeological and Historical Sciences* (F. R. Hodson, D. G. Kendall and P. Tautu, eds), pp. 137–149. Edinburgh: University Press.

—— (1975). Generalised Procrustes analysis. *Psychometrika*, **40**, in the press.

Schonemann, P. H. and Carroll, R. M. (1970). Fitting one matrix to another under choice of a central dilation and a rigid motion. *Psychometrika*, **35**, 245–256.

Vitali, G. (1928). Sulle sostituzioni lineari ortogonali. *Boll. Un. Mat. Ital.*, **7**, 1–7.

Wilkinson, J. H. (1965). *The Algebraic Eigenvalue Problem*. London: Oxford University Press.

```
      SUBROUTINE DETQ(A, N, NN, D, IFAULT)
C
C        ALGORITHM AS 82  APPL. STATIST. (1975) VOL.24, NO.1
C
C        FINDS THE DETERMINANT OF AN ORTHOGONAL MATRIX
C
      REAL A(NN)
      INTEGER P, Q, R, S
C
      DATA TOL /0.0001/
C
      IF ((N .LE. 0) .OR. (NN .NE. N * N)) GOTO 4
      IFAULT = 0
      E = 1.0
      N1 = N + 1
      R = 1
      DO 3 K = 2, N1
      Q = R
C
C        EVALUATE 1/(1 - Q(1, 1)) OR -1/(1 + Q(1, 1))
C
      X = A(R)
      Y = SIGN(1.0, X)
      E = E * Y
      Y = -1.0 / (X + Y)
C
C        CHECK FOR UNIT DIAGONAL ELEMENT
```

```
      X = ABS(X) - 1.0
      IF (ABS(X) .LT. TOL) GOTO 2
C
C        CHECK FOR NON-ORTHOGONALITY
C
      IF ((X .GT. 0.0) .OR. (K .EQ. N1)) GOTO 4
C
C        EVALUATE Q(R)
C
      DO 1 I = K, N
      Q = Q + N
      X = A(Q) * Y
      P = R
      S = Q
      DO 1 J = K, N
      P = P + 1
      S = S + 1
      A(S) = A(S) + X * A(P)
    1 CONTINUE
    2 R = R + N1
    3 CONTINUE
      D = E
      RETURN
    4 IFAULT = 1
      RETURN
      END
```

Algorithm AS 83

Complex Discrete Fast Fourier Transform

By Donald M. Monro

Engineering in Medicine Laboratory, Imperial College, London

Keywords: FAST FOURIER TRANSFORM; DISCRETE FOURIER TRANSFORM; FOURIER SERIES; SPECTRAL ANALYSIS

Language

ISO Fortran

Description and Purpose

The purpose of this algorithm is the efficient evaluation of the complex discrete Fourier transform (DFT) of a sequence of length N, $X = \{X_0, X_1, \ldots, X_{n-1}\}$, defined for this procedure as

$$Y_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k \exp(-j2\pi kn/N), \quad n = 0, 1, \ldots, N-1, \tag{1}$$

where $\mathbf{Y} = \{Y_0, Y_1, \ldots, Y_{N-1}\}$ is the transformed sequence. The inverse transform is

$$X_k = \sum_{n=0}^{N-1} Y_n \exp(j2\pi kn/N), \quad k = 0, 1, \ldots, n-1. \tag{2}$$