NAG Toolbox for MATLAB

g03fa

1 Purpose

g03fa performs a principal co-ordinate analysis also known as classical metric scaling.

2 Syntax

```
[x, eval, ifail] = g03fa(roots, n, d, ndim)
```

3 Description

For a set of n objects a distance matrix D can be calculated such that d_{ij} is a measure of how 'far apart' are objects i and j (see g03ea for example). Principal co-ordinate analysis or metric scaling starts with a distance matrix and finds points X in Euclidean space such that those points have the same distance matrix. The aim is to find a small number of dimensions, $k \ll (n-1)$, that provide an adequate representation of the distances.

The principal co-ordinates of the points are computed from the eigenvectors of the matrix E where $e_{ij} = -1/2\left(d_{ij}^2 - d_{i.}^2 - d_{.j}^2 - d_{..}^2\right)$ with $d_{i.}^2$ denoting the average of d_{ij}^2 over the suffix j . The eigenvectors are then scaled by multiplying by the square root of the value of the corresponding eigenvalue.

Provided that the ordered eigenvalues, λ_i, of the matrix E are all positive, $\sum_{i=1}^{k}\lambda_i / \sum_{i=1}^{n-1}\lambda_i$ shows how well the data is represented in k dimensions. The eigenvalues will be nonnegative if E is positive semi-definite. This will be true provided d_{ij} satisfies the inequality: $d_{ij} \leq d_{ik} + d_{jk}$ for all i, j, k. If this is not the case the size of the negative eigenvalue reflects the amount of deviation from this condition and the results should be treated cautiously in the presence of large negative eigenvalues. See Krzanowski 1990 for further discussion. g03fa provides the option for all eigenvalues to be computed so that the smallest eigenvalues can be checked.

4 References

Chatfield C and Collins A J 1980 *Introduction to Multivariate Analysis* Chapman and Hall

Gower J C 1966 Some distance properties of latent root and vector methods used in multivariate analysis *Biometrika* **53** 325–338

Krzanowski W J 1990 *Principles of Multivariate Analysis* Oxford University Press

5 Parameters

5.1 Compulsory Input Parameters

1: **roots – string**

Indicates if all the eigenvalues are to be computed or just the **ndim** largest.

roots = 'A'

All the eigenvalues are computed.

roots = 'L'

Only the largest **ndim** eigenvalues are computed.

Constraint: **roots** = 'A' or 'L'.

2: **n – int32 scalar**

n, the number of objects in the distance matrix.

Constraint: **n** $>$ **ndim**.

3: $\mathbf{d}(\mathbf{n} \times (\mathbf{n} - \mathbf{1})/\mathbf{2})$ **– double array**

The lower triangle of the distance matrix D stored packed by rows. That is $\mathbf{d}((i-1) \times (i-2)/2 + j)$ must contain d_{ij} for $i = 2, 3, \ldots, n; j = 1, 2, \ldots, i-1$.

Constraint: $\mathbf{d}(i) \geq 0.0, i = 1, 2, \ldots, n(n-1)/2$.

4: **ndim – int32 scalar**

k, the number of dimensions used to represent the data.

Constraint: **ndim** ≥ 1.

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

ldx, wk, iwk

5.4 Output Parameters

1: **x(ldx,ndim) – double array**

The ith row contains k co-ordinates for the ith point, $i = 1, 2, \ldots, n$.

2: **eval(n) – double array**

If **roots** = 'A', **eval** contains the n scaled eigenvalues of the matrix E.

If **roots** = 'L', **eval** contains the largest k scaled eigenvalues of the matrix E.

In both cases the eigenvalues are divided by the sum of the eigenvalues (that is, the trace of E).

3: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail = 1

On entry, **ndim** < 1,
or **n** $<$ **ndim**,
or **roots** $\neq$ 'A' or 'L',
or **ldx** $<$ **n**.

ifail = 2

On entry, $\mathbf{d}(i) < 0.0$ for some i, $i = 1, 2, \ldots, n(n-1)/2$,
or all elements of **d** = 0.0.

ifail = 3

There are less than **ndim** eigenvalues greater than zero. Try a smaller number of dimensions (**ndim**) or use non-metric scaling (g03fc).

ifail = 4

The computation of the eigenvalues or eigenvectors has failed. Seek expert help.

7 Accuracy

g03fa uses f08jf or f08jj to compute the eigenvalues and f08jk to compute the eigenvectors. These functions should be consulted for a discussion of the accuracy of the computations involved.

8 Further Comments

Alternative, non-metric, methods of scaling are provided by g03fc.

The relationship between principal co-ordinates and principal components, see g03fc, is discussed in Krzanowski 1990 and Gower 1966.

9 Example

```
roots = 'l';
n = int32(14);
d = [0.099;
     0.033;
     0.022;
     0.183;
     0.114;
     0.042;
     0.148;
     0.224;
     0.059;
     0.068;
     0.198;
     0.039;
     0.053;
     0.08500000000000001;
     0.051;
     0.462;
     0.266;
     0.322;
     0.435;
     0.268;
     0.025;
     0.628;
     0.442;
     0.444;
     0.406;
     0.24;
     0.129;
     0.014;
     0.113;
     0.07000000000000001;
     0.046;
     0.047;
     0.034;
     0.002;
     0.106;
     0.129;
     0.173;
     0.119;
     0.162;
     0.331;
     0.177;
     0.039;
     0.089;
     0.237;
     0.07099999999999999;
```

```
      0.434;
      0.419;
      0.339;
      0.505;
      0.469;
      0.39;
      0.315;
      0.349;
      0.151;
      0.43;
      0.762;
      0.633;
      0.781;
      0.7;
      0.758;
      0.625;
      0.469;
      0.618;
      0.44;
      0.538;
      0.607;
      0.53;
      0.389;
      0.482;
      0.579;
      0.597;
      0.498;
      0.374;
      0.5620000000000001;
      0.247;
      0.383;
      0.387;
      0.08400000000000001;
      0.586;
      0.435;
      0.55;
      0.53;
      0.552;
      0.509;
      0.369;
      0.471;
      0.234;
      0.346;
      0.456;
      0.09;
      0.038];
ndim = int32(2);
[x, eval, ifail] = g03fa(roots, n, d, ndim)
```

```
x =
    0.2408    0.2337
    0.1137    0.1168
    0.2394    0.0760
    0.2129    0.0605
    0.2495   -0.0693
    0.1487   -0.0778
   -0.0514   -0.1623
    0.0115   -0.3446
   -0.0039    0.0059
    0.0386   -0.0089
   -0.0421   -0.0566
   -0.5158    0.0291
   -0.3180    0.1501
   -0.3238    0.0475
eval =
    0.7871
    0.2808
         0
         0
         0
```

```
         0
         0
         0
         0
         0
         0
         0
         0
         0
ifail =
           0
```