NAG Toolbox for MATLAB

g05qb

1 Purpose

g05qb generates a random correlation matrix with given eigenvalues.

2 Syntax

```
[c, iseed, ifail] = g05qb(d, igen, iseed, 'n', n, 'eps', eps)
```

3 Description

Given n eigenvalues, $\lambda_1, \lambda_2, \ldots, \lambda_n$, such that

$$\sum_{i=1}^{n} \lambda_i = n$$

and

$$\lambda_i \geq 0, \qquad i = 1, 2, \ldots, n,$$

g05qb will generate a random correlation matrix, C, of dimension n, with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$.

The method used is based on that described by Lin and Bendel 1985. Let D be the diagonal matrix with values $\lambda_1, \lambda_2, \ldots, \lambda_n$ and let A be a random orthogonal matrix generated by g05qa then the matrix $C_0 = ADA^{\mathrm{T}}$ is a random covariance matrix with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. The matrix C_0 is transformed into a correlation matrix by means of $n-1$ elementary rotation matrices P_i such that $C = P_{n-1}P_{n-2}\ldots P_1 C_0 P_1^{\mathrm{T}} \ldots P_{n-2}^{\mathrm{T}} P_{n-1}^{\mathrm{T}}$. The restriction on the sum of eigenvalues implies that for any diagonal element of $C_0 > 1$, there is another diagonal element < 1. The P_i are constructed from such pairs, chosen at random, to produce a unit diagonal element corresponding to the first element. This is repeated until all diagonal elements are 1 to within a given tolerance ϵ.

The randomness of C should be interpreted only to the extent that A is a random orthogonal matrix and C is computed from A using the P_i which are chosen as arbitrarily as possible.

One of the initialization functions g05kb (for a repeatable sequence if computed sequentially) or g05kc (for a non-repeatable sequence) must be called prior to the first call to g05qb.

4 References

Lin S P and Bendel R B 1985 Algorithm AS213: Generation of population correlation on matrices with specified eigenvalues *Appl. Statist.* **34** 193–198

5 Parameters

5.1 Compulsory Input Parameters

1: **d(n) – double array**

The n eigenvalues, λ_i, for $i = 1, 2, \ldots, n$.

Constraints:

$\mathbf{d}(i) \geq 0.0$, for $i = 1, 2, \ldots, n$;

$\sum_{i=1}^{n} \mathbf{d}(i) = n$ to within **eps**.

2: **igen – int32 scalar**

Must contain the identification number for the generator to be used to return a pseudo-random number and should remain unchanged following initialization by a prior call to g05kb or g05kc.

3: **iseed**(4) **– int32 array**

Contains values which define the current state of the selected generator.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the arrays **d**, **c**. (An error is raised if these dimensions are not equal.)

n, the dimension of the correlation matrix to be generated.

Constraint: $\mathbf{n} \geq 1$.

2: **eps – double scalar**

ϵ, the maximum acceptable error in the diagonal elements.

Constraint: $\mathbf{eps} \geq \mathbf{n} \times$ ***machine precision*** (see Chapter X02).

Suggested value: $\mathbf{eps} = 0.00001$.

Default: 0.00001

5.3 Input Parameters Omitted from the MATLAB Interface

ldc, wk

5.4 Output Parameters

1: **c**$(\mathbf{ldc},\mathbf{n})$ **– double array**

A random correlation matrix, C, of dimension n.

2: **iseed**(4) **– int32 array**

Contains updated values defining the new state of the selected generator.

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, $\mathbf{n} < 1$,
or $\mathbf{ldc} < \mathbf{n}$.
or $\mathbf{eps} < \mathbf{n} \times$ ***machine precision***.

ifail $= 2$

On entry, $\mathbf{d}(i) < 0.0$ for some i,
or $\sum_{i=1}^{n} \mathbf{d}(i) \neq n$ to within **eps**.

ifail = 3

The error in a diagonal element is greater than **eps**. The value of **eps** should be increased. Otherwise the program could be rerun with a different value used for the seed of the random number generator, see g05kb or g05kc.

7 Accuracy

The maximum error in a diagonal element is given by **eps**.

8 Further Comments

The time taken by g05qb is approximately proportional to n^2.

9 Example

```
d = [0.7;
     0.9;
     1.4];
igen = int32(1);
iseed = [int32(1762543);
     int32(9324783);
     int32(423446);
     int32(742355)];
[c, iseedOut, ifail] = g05qb(d, igen, iseed)
```

```
c =
    1.0000    0.2043   -0.1058
    0.2043    1.0000   -0.2776
   -0.1058   -0.2776    1.0000
iseedOut =
    12622195
     6767553
    16277754
    11529009
ifail =
           0
```