NAG Toolbox for MATLAB

g05ea

1 Purpose

g05ea sets up a reference vector for a multivariate Normal distribution with mean vector a and covariance matrix C, so that g05ez may be used to generate pseudo-random vectors.

2 Syntax

```
[r, ifail] = g05ea(a, c, eps, nr, 'n', n)
```

3 Description

When the covariance matrix is nonsingular (i.e., strictly positive-definite), the distribution has probability density function

$$f(x) = \sqrt{\frac{|C^{-1}|}{(2\pi)^n}} \exp\left\{-(x-a)^T C^{-1}(x-a)\right\}$$

where n is the number of dimensions, C is the covariance matrix, a is the vector of means and x is the vector of positions.

Covariance matrices are symmetric and positive semi-definite. Given such a matrix C, there exists a lower triangular matrix L such that $LL^{\mathrm{T}} = C$. L is not unique, if C is singular.

g05ea decomposes C to find such an L. It then stores n, a and L in the reference vector r for later use by g05ez. g05ez generates a vector x of independent standard Normal pseudo-random numbers. It then returns the vector $a + Lx$, which has the required multivariate Normal distribution.

It should be noted that this function will work with a singular covariance matrix C, provided C is positive semi-definite, despite the fact that the above formula for the probability density function is not valid in that case. Wilkinson 1965 should be consulted if further information is required.

4 References

Knuth D E 1981 *The Art of Computer Programming (Volume 2)* (2nd Edition) Addison–Wesley

Wilkinson J H 1965 *The Algebraic Eigenvalue Problem* Oxford University Press, Oxford

5 Parameters

5.1 Compulsory Input Parameters

1: **a**(**n**) – **double array**

a, the vector of means of the distribution.

2: **c**(**ldc**,**n**) – **double array**

ldc, the first dimension of the array, must be at least **n**.

The covariance matrix of the distribution. Only the upper triangle need be set.

Constraint: **c** must be positive semi-definite to ***machine precision***

3: **eps – double scalar**

The maximum error in any element of **c**, relative to the largest element of **c**.

Constraint: $0.0 \leq \mathbf{eps} \leq 0.1/\mathbf{n}$.

If **eps** is less than ***machine precision***, ***machine precision*** is used

4: **nr – int32 scalar**

Constraint: $\mathbf{nr} \geq ((\mathbf{n}+1) \times (\mathbf{n}+2))/2$.

5.2 Optional Input Parameters

1: **n – int32 scalar**

n, the number of dimensions of the distribution.

Constraint: $\mathbf{n} > 0$.

5.3 Input Parameters Omitted from the MATLAB Interface

ldc

5.4 Output Parameters

1: **r(nr) – double array**

The reference vector for subsequent use by g05ez.

2: **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, $\mathbf{n} < 1$.

ifail $= 2$

On entry, $\mathbf{nr} < ((\mathbf{n}+1) \times (\mathbf{n}+2))/2$.

ifail $= 3$

On entry, $\mathbf{ldc} < \mathbf{n}$.

ifail $= 4$

On entry, $\mathbf{eps} < 0.0$,
or $\mathbf{eps} > 0.1/\mathbf{n}$.

ifail $= 5$

The covariance matrix C is not positive semi-definite to accuracy **eps**.

7 Accuracy

The maximum absolute error in LL^{T}, and hence in the covariance matrix of the resulting vectors, is less than $(n \times \max(\mathbf{eps}, \epsilon) + (n+3)\epsilon/2)$ times the maximum element of C, where ϵ is the ***machine precision***. Under normal circumstances, the above will be small compared to sampling error.

8 Further Comments

The time taken by g05ea is of order n^3.

It is recommended that the diagonal elements of C should not differ too widely in order of magnitude. This may be achieved by scaling the variables if necessary. The actual matrix decomposed is $C + E = LL^{\mathrm{T}}$, where E is a diagonal matrix with small positive diagonal elements. This ensures that, even when C is singular, or nearly singular, the Cholesky Factor L corresponds to a positive-definite covariance matrix that agrees with C within a tolerance determined by **eps**.

9 Example

```
a = [1;
     2];
c = [2, 1;
     1, 3];
eps = 0.01;
nr = int32(6);
g05za('O');
g05cb(int32(0));
[r, ifail] = g05ea(a, c, eps, nr)
```

```
r =
    2.5000
    1.0000
    2.0000
    1.4353
    0.6967
    1.6045
ifail =
           0
```