NAG Toolbox for MATLAB

g01na

1 Purpose

g01na computes the cumulants and moments of quadratic forms in Normal variates.

2 Syntax

```
[rkum, rmom, ifail] = g01na(mom, mean, a, emu, sigma, l, 'n', n)
```

3 Description

Let x have an n-dimensional multivariate Normal distribution with mean μ and variance-covariance matrix Σ. Then for a symmetric matrix A, g01na computes up to the first 12 moments and cumulants of the quadratic form $Q = x^{\mathrm{T}}Ax$. The sth moment (about the origin) is defined as

$$E(Q^s),$$

where E denotes expectation. The sth moment of Q can also be found as the coefficient of $t^s/s!$ in the expansion of $E\left(e^{Qt}\right)$. The sth cumulant is defined as the coefficient of $t^s/s!$ in the expansion of $\log\left(E\left(e^{Qt}\right)\right)$.

The function is based on the function CUM written by Magnus and Pesaran 1993 and based on the theory given by Magnus 1978, Magnus 1979 and Magnus 1986.

4 References

Magnus J R 1978 The moments of products of quadratic forms in Normal variables *Statist. Neerlandica* **32** 201–210

Magnus J R 1979 The expectation of products of quadratic forms in Normal variables: the practice *Statist. Neerlandica* **33** 131–136

Magnus J R 1986 The exact moments of a ratio of quadratic forms in Normal variables *Ann. Économ. Statist.* **4** 95–109

Magnus J R and Pesaran B 1993 The evaluation of cumulants and moments of quadratic forms in Normal variables (CUM): Technical description *Comput. Statist.* **8** 39–45

Magnus J R and Pesaran B 1993 The evaluation of moments of quadratic forms and ratios of quadratic forms in Normal variables: Background, motivation and examples *Comput. Statist.* **8** 47–55

5 Parameters

5.1 Compulsory Input Parameters

1: **mom – string**

Indicates if moments are computed in addition to cumulants.

mom = 'C'

Only cumulants are computed.

mom = 'M'

Moments are computed in addition to cumulants.

Constraint: **mom** = 'C' or 'M'.

2: **mean – string**

μ, indicates if the mean is zero.

mean = 'Z'

μ is zero.

mean = 'M'

The value of μ is supplied in **emu**.

Constraint: **mean** = 'Z' or 'M'.

3: **a(lda,n) – double array**

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

The n by n symmetric matrix A. Only the lower triangle is referenced.

4: **emu(∗) – double array**

Note: the dimension of the array **emu** must be at least **n** if **mean** = 'M', and at least 1 otherwise.

If **mean** = 'M', **emu** must contain the n elements of the vector μ.

If **mean** = 'Z', **emu** is not referenced.

5: **sigma(ldsig,n) – double array**

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

The n by n variance-covariance matrix Σ. Only the lower triangle is referenced.

Constraint: the matrix Σ must be positive-definite.

6: **l – int32 scalar**

the required number of cumulants, and moments if specified.

Constraint: $1 \leq \mathbf{l} \leq 12$.

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

n, the dimension of the quadratic form.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldsig, wk

5.4 Output Parameters

1: **rkum(l) – double array**

The **l** cumulants of the quadratic form.

2: **rmom(∗) – double array**

Note: the dimension of the array **rmom** must be at least **l** if **mom** = 'M', and at least 1 otherwise.

If **mom** = 'M', the **l** moments of the quadratic form.

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, **n** ≤ 1,
or **l** < 1,
or **l** > 12,
or **lda** $<$ **n**,
or **ldsig** $<$ **n**,
or **mom** $\neq$ 'C' or 'M',
or **mean** $\neq$ 'M' or 'Z'.

ifail $= 2$

On entry, the matrix Σ is not positive-definite.

7 Accuracy

In a range of tests the accuracy was found to be a modest multiple of ***machine precision***. See Magnus and Pesaran 1993.

8 Further Comments

None.

9 Example

```
mom = 'M';
mean = 'M';
a = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0;
     0.5, 0, 0, 0, 0, 0, 0, 0, 0, 0;
     0, 0.5, 0, 0, 0, 0, 0, 0, 0, 0;
     0, 0, 0.5, 0, 0, 0, 0, 0, 0, 0;
     0, 0, 0, 0.5, 0, 0, 0, 0, 0, 0;
     0, 0, 0, 0, 0.5, 0, 0, 0, 0, 0;
     0, 0, 0, 0, 0, 0.5, 0, 0, 0, 0;
     0, 0, 0, 0, 0, 0, 0.5, 0, 0, 0;
     0, 0, 0, 0, 0, 0, 0, 0.5, 0, 0;
     0, 0, 0, 0, 0, 0, 0, 0, 0.5, 0];
emu = [0.8;
     0.6400000000000001;
     0.5120000000000001;
     0.4096000000000001;
     0.3276800000000001;
     0.2621440000000001;
     0.2097152000000001;
     0.1677721600000001;
     0.1342177280000001;
     0.1073741824000001];
sigma = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0;
     0.8, 1.64, 0, 0, 0, 0, 0, 0, 0, 0;
     0.64, 1.312, 2.0496, 0, 0, 0, 0, 0, 0, 0;
     0.512, 1.0496, 1.63968, 2.311744, 0, 0, 0, 0, 0, 0;
     0.4096, 0.83968, 1.311744, 1.8493952, 2.479516160000001, 0, 0, 0, 0,
0;
     0.32768, 0.671744, 1.0493952, 1.47951616, 1.983612928, 2.5868903424,
```

```
0, 0, 0, 0;
          0.262144,  0.5373952,  0.83951616,  1.183612928,  1.5868903424,
2.06951227392, 2.655609819136, 0, 0, 0;
        0.2097152,  0.42991616,  0.671612928,  0.9468903424,  1.26951227392,
1.655609819136, ...
     2.1244878553088, 2.69959028424704, 0, 0;
           0.1677721600000001,  0.3439329280000002,  0.5372903424000002,
0.7575122739200002, ...
                1.015609819136,    1.3244878553088,    1.699590284247041,
2.159672227397632, 2.727737781918106, 0;
           0.1342177280000001,  0.2751463424000002,  0.4298322739200002,
0.6060098191360002, ...
            0.8124878553088003,   1.05959028424704,   1.359672227397632,
1.727737781918106, 2.182190225534485, 2.745752180427588];
l = int32(4);
[rkum, rmom, ifail] = g01na(mom, mean, a, emu, sigma, l)
```

```
rkum =
   1.0e+06 *
    0.0000
    0.0004
    0.0161
    1.1700
rmom =
   1.0e+06 *
    0.0000
    0.0007
    0.0399
    3.4039
ifail =
           0
```