

# NAG Library Routine Document

## G05RYF

**Note:** before using this routine, please read the Users' Note for your implementation to check the interpretation of *bold italicised* terms and other implementation-dependent details.

### 1 Purpose

G05RYF sets up a reference vector and generates an array of pseudorandom numbers from a multivariate Student's  $t$  distribution with  $\nu$  degrees of freedom, mean vector  $a$  and covariance matrix  $\frac{\nu}{\nu-2}C$ .

### 2 Specification

```

SUBROUTINE G05RYF (MODE, N, DF, M, XMU, C, LDC, R, LR, STATE, X, LDX,      &
                  IFAIL)
INTEGER              MODE, N, DF, M, LDC, LR, STATE(*), LDX, IFAIL
REAL (KIND=nag_wp) XMU(M), C(LDC,M), R(LR), X(LDX,M)

```

### 3 Description

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

$$f(x) = \frac{\Gamma\left(\frac{\nu+m}{2}\right)}{(\pi\nu)^{m/2} \Gamma(\nu/2) |C|^{1/2}} \left[ 1 + \frac{(x-a)^T C^{-1} (x-a)}{\nu} \right]^{-\frac{(\nu+m)}{2}}$$

where  $m$  is the number of dimensions,  $\nu$  is the degrees of freedom,  $a$  is the vector of means,  $x$  is the vector of positions and  $\frac{\nu}{\nu-2}C$  is the covariance matrix.

The routine returns the value

$$x = a + \sqrt{\frac{\nu}{s}} z$$

where  $z$  is generated by G05SKF from a Normal distribution with mean zero and covariance matrix  $C$  and  $s$  is generated by G05SDF from a  $\chi^2$ -distribution with  $\nu$  degrees of freedom.

One of the initialization routines G05KFF (for a repeatable sequence if computed sequentially) or G05KGF (for a non-repeatable sequence) must be called prior to the first call to G05RYF.

### 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 Arguments

1: MODE – INTEGER *Input*  
*On entry:* a code for selecting the operation to be performed by the routine.  
 MODE = 0  
     Set up reference vector only.  
 MODE = 1  
     Generate variates using reference vector set up in a prior call to G05RYF.

MODE = 2

Set up reference vector and generate variates.

*Constraint:* MODE = 0, 1 or 2.

- 2: N – INTEGER *Input*  
*On entry:*  $n$ , the number of random variates required.  
*Constraint:*  $N \geq 0$ .
- 3: DF – INTEGER *Input*  
*On entry:*  $\nu$ , the number of degrees of freedom of the distribution.  
*Constraint:*  $DF \geq 3$ .
- 4: M – INTEGER *Input*  
*On entry:*  $m$ , the number of dimensions of the distribution.  
*Constraint:*  $M > 0$ .
- 5: XMU(M) – REAL (KIND=nag\_wp) array *Input*  
*On entry:*  $a$ , the vector of means of the distribution.
- 6: C(LDC, M) – REAL (KIND=nag\_wp) array *Input*  
*On entry:* matrix which, along with DF, defines the covariance of the distribution. Only the upper triangle need be set.  
*Constraint:* C must be positive semidefinite to *machine precision*.
- 7: LDC – INTEGER *Input*  
*On entry:* the first dimension of the array C as declared in the (sub)program from which G05RYF is called.  
*Constraint:*  $LDC \geq M$ .
- 8: R(LR) – REAL (KIND=nag\_wp) array *Input/Output*  
*On entry:* if MODE = 1, the reference vector as set up by G05RYF in a previous call with MODE = 0 or 2.  
*On exit:* if MODE = 0 or 2, the reference vector that can be used in subsequent calls to G05RYF with MODE = 1.
- 9: LR – INTEGER *Input*  
*On entry:* the dimension of the array R as declared in the (sub)program from which G05RYF is called. If MODE = 1, it must be the same as the value of LR specified in the prior call to G05RYF with MODE = 0 or 2.  
*Constraint:*  $LR \geq M \times (M + 1) + 2$ .
- 10: STATE(\*) – INTEGER array *Communication Array*  
**Note:** the actual argument supplied **must** be the array STATE supplied to the initialization routines G05KFF or G05KGF.  
*On entry:* contains information on the selected base generator and its current state.  
*On exit:* contains updated information on the state of the generator.

- 11: X(LDX, M) – REAL (KIND=nag\_wp) array *Output*  
*On exit:* the array of pseudorandom multivariate Student's  $t$  vectors generated by the routine, with  $X(i, j)$  holding the  $j$ th dimension for the  $i$ th variate.
- 12: LDX – INTEGER *Input*  
*On entry:* the first dimension of the array X as declared in the (sub)program from which G05RYF is called.  
*Constraint:*  $LDX \geq N$ .
- 13: IFAIL – INTEGER *Input/Output*  
*On entry:* IFAIL must be set to 0,  $-1$  or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation for details.  
 For environments where it might be inappropriate to halt program execution when an error is detected, the value  $-1$  or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this argument, the recommended value is 0. **When the value  $-1$  or 1 is used it is essential to test the value of IFAIL on exit.**  
*On exit:* IFAIL = 0 unless the routine detects an error or a warning has been flagged (see Section 6).

## 6 Error Indicators and Warnings

If on entry IFAIL = 0 or  $-1$ , explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

On entry, MODE =  $\langle value \rangle$ .

Constraint: MODE = 0, 1 or 2.

IFAIL = 2

On entry, N =  $\langle value \rangle$ .

Constraint:  $N \geq 0$ .

IFAIL = 3

On entry, DF =  $\langle value \rangle$ .

Constraint:  $DF \geq 3$ .

IFAIL = 4

On entry, M =  $\langle value \rangle$ .

Constraint:  $M > 0$ .

IFAIL = 6

On entry, the covariance matrix  $C$  is not positive semidefinite to *machine precision*.

IFAIL = 7

On entry, LDC =  $\langle value \rangle$  and M =  $\langle value \rangle$ .

Constraint:  $LDC \geq M$ .

IFAIL = 8

M is not the same as when R was set up in a previous call.  
Previous value of M =  $\langle value \rangle$  and M =  $\langle value \rangle$ .

IFAIL = 9

On entry, LR is not large enough, LR =  $\langle value \rangle$ : minimum length required =  $\langle value \rangle$ .

IFAIL = 10

On entry, STATE vector has been corrupted or not initialized.

IFAIL = 12

On entry, LDX =  $\langle value \rangle$  and N =  $\langle value \rangle$ .  
Constraint: LDX  $\geq$  N.

IFAIL = -99

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -399

Your licence key may have expired or may not have been installed correctly.

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -999

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

## 7 Accuracy

Not applicable.

## 8 Parallelism and Performance

G05RYF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

G05RYF makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

## 9 Further Comments

The time taken by G05RYF is of order  $nm^3$ .

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  $C + E = LL^T$ , 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 *machine precision*.

## 10 Example

This example prints ten pseudorandom observations from a multivariate Student's  $t$ -distribution with ten degrees of freedom, means vector

$$\begin{bmatrix} 1.0 \\ 2.0 \\ -3.0 \\ 0.0 \end{bmatrix}$$

and C matrix

$$\begin{bmatrix} 1.69 & 0.39 & -1.86 & 0.07 \\ 0.39 & 98.01 & -7.07 & -0.71 \\ -1.86 & -7.07 & 11.56 & 0.03 \\ 0.07 & -0.71 & 0.03 & 0.01 \end{bmatrix},$$

generated by G05RYF. All ten observations are generated by a single call to G05RYF with  $\text{MODE} = 2$ . The random number generator is initialized by G05KFF.

### 10.1 Program Text

```

Program g05ryfe

!      G05RYF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
Use nag_library, Only: g05kff, g05ryf, nag_wp, x04caf
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: lseed = 1, nin = 5, nout = 6
!      .. Local Scalars ..
Integer                    :: df, genid, i, ifail, ldc, ldx, lr, &
                           lstate, m, mode, n, subid
!      .. Local Arrays ..
Real (Kind=nag_wp), Allocatable :: c(:,,:), r(:), x(:,,:), xmu(:)
Integer                    :: seed(lseed)
Integer, Allocatable        :: state(:)
!      .. Executable Statements ..
Write (nout,*) 'G05RYF Example Program Results'
Write (nout,*)
Flush (nout)

!      Skip heading in data file
Read (nin,*)

!      Read in the base generator information and seed
Read (nin,*) genid, subid, seed(1)

!      Initial call to initializer to get size of STATE array
lstate = 0
Allocate (state(lstate))
ifail = 0
Call g05kff(genid,subid,seed,lseed,state,lstate,ifail)

!      Reallocate STATE
Deallocate (state)
Allocate (state(lstate))

!      Initialize the generator to a repeatable sequence
ifail = 0
Call g05kff(genid,subid,seed,lseed,state,lstate,ifail)

!      Read in sample size and number of dimensions
Read (nin,*) n, m

```

```

      ldc = m
      ldx = n
      lr = m*(m+1) + 2
      Allocate (x(ldx,m),c(ldc,m),r(lr),xmu(m))

!      Read in degrees of freedom
      Read (nin,*) df

!      Read in vector of means
      Read (nin,*) xmu(1:m)

!      Read in upper triangle portion of the covariance matrix
      Do i = 1, m
        Read (nin,*) c(i,i:m)
      End Do

!      Using a single call to G05RYF, so set up reference vector
!      and generate values in one go
      mode = 2

!      Generate variates
      ifail = 0
      Call g05ryf(mode,n,df,m,xmu,c,ldc,r,lr,state,x,ldx,ifail)

!      Display the variates
      ifail = 0
      Call x04caf('General',' ',n,m,x,ldx,'Variates',ifail)

      End Program g05ryfe

```

## 10.2 Program Data

```

G05RYF Example Program Data
1 1 1762543      :: GENID,SUBID,SEED(1)
10 4           :: N,M
10            :: DF
1.0 2.0 -3.0 0.0 :: XMU
1.69 0.39 -1.86 0.07
      98.01 -7.07 -0.71
          11.56 0.03
              0.01 :: End of C (upper triangular part)

```

## 10.3 Program Results

G05RYF Example Program Results

```

Variates
      1           2           3           4
1      1.4957    -15.6226    -3.8101     0.1294
2      -1.0827     -6.7473     0.6696    -0.0391
3       2.1369     6.3861    -5.7413     0.0140
4       2.2481    -16.0417    -1.0982     0.1641
5      -0.2550     3.5166    -0.2541    -0.0592
6       0.9731    -4.3553    -4.4181     0.0043
7       0.7098    -3.4281     1.1741     0.0586
8       1.8827    23.2619     1.5140    -0.0704
9       0.9904    22.7479     0.1811    -0.0893
10      1.5026     2.7753    -2.2805    -0.0112

```

---