NAG Library Routine Document

G01HAF

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

G01HAF returns the lower tail probability for the bivariate Normal distribution, via the routine name.

2 Specification

```
FUNCTION G01HAF (X, Y, RHO, IFAIL)
REAL (KIND=nag_wp) G01HAF
INTEGER            IFAIL
REAL (KIND=nag_wp) X, Y, RHO
```

3 Description

For the two random variables (X, Y) following a bivariate Normal distribution with

$$E[X] = 0, \quad E[Y] = 0, \quad E\left[X^2\right] = 1, \quad E\left[Y^2\right] = 1 \quad \text{and} \quad E[XY] = \rho,$$

the lower tail probability is defined by:

$$P(X \leq x, Y \leq y : \rho) = \frac{1}{2\pi\sqrt{1-\rho^2}} \int_{-\infty}^{y} \int_{-\infty}^{x} \exp\left(-\frac{(X^2 - 2\rho XY + Y^2)}{2(1-\rho^2)}\right) dX dY.$$

For a more detailed description of the bivariate Normal distribution and its properties see Abramowitz and Stegun (1972) and Kendall and Stuart (1969). The method used is described by Genz (2004).

4 References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

Genz A (2004) Numerical computation of rectangular bivariate and trivariate Normal and t probabilities *Statistics and Computing* **14** 151–160

Kendall M G and Stuart A (1969) *The Advanced Theory of Statistics (Volume 1)* (3rd Edition) Griffin

5 Parameters

1: X – REAL (KIND=nag_wp) *Input*

On entry: x, the first argument for which the bivariate Normal distribution function is to be evaluated.

2: Y – REAL (KIND=nag_wp) *Input*

On entry: y, the second argument for which the bivariate Normal distribution function is to be evaluated.

3: RHO – REAL (KIND=nag_wp) *Input*

On entry: ρ, the correlation coefficient.

Constraint: $-1.0 \leq \text{RHO} \leq 1.0$.

4: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction 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 parameter, 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, RHO < -1.0,
or RHO > 1.0.

If on exit IFAIL $= 1$ then G01HAF returns zero.

IFAIL $= -99$

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

See Section 3.8 in the Essential Introduction for further information.

IFAIL $= -399$

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

See Section 3.7 in the Essential Introduction for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

7 Accuracy

Accuracy of the hybrid algorithm implemented here is discussed in Genz (2004). This algorithm should give a maximum absolute error of less than 5×10^{-16}.

8 Parallelism and Performance

Not applicable.

9 Further Comments

The probabilities for the univariate Normal distribution can be computed using S15ABF and S15ACF.

10 Example

This example reads values of x and y for a bivariate Normal distribution along with the value of ρ and computes the lower tail probabilities.

10.1 Program Text

```
    Program g01hafe

!     G01HAF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: g01haf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: prob, rho, x, y
      Integer                          :: ifail
!     .. Executable Statements ..
      Write (nout,*) 'G01HAF Example Program Results'
      Write (nout,*)

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

!     Display titles
      Write (nout,*) '     X       Y      RHO    PROB'
      Write (nout,*)

d_lp: Do
        Read (nin,*,Iostat=ifail) x, y, rho
        If (ifail/=0) Then
          Exit d_lp
        End If

!       Calculate probability
        ifail = 0
        prob = g01haf(x,y,rho,ifail)

!       Display results
        Write (nout,99999) x, y, rho, prob
      End Do d_lp

99999 Format (1X,3F8.3,F8.4)
    End Program g01hafe
```

10.2 Program Data

```
G01HAF Example Program Data
  1.7  23.1   0.0         :X Y RHO
  0.0   0.0   0.1         :X Y RHO
  3.3  11.1   0.54        :X Y RHO
  9.1   9.1   0.17        :X Y RHO
```

10.3 Program Results

```
 G01HAF Example Program Results

      X       Y      RHO    PROB

    1.700  23.100   0.000  0.9554
    0.000   0.000   0.100  0.2659
    3.300  11.100   0.540  0.9995
    9.100   9.100   0.170  1.0000
```