

NAG Library Routine Document

G01FAF

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

G01FAF returns the deviate associated with the given probability of the standard Normal distribution, via the routine name.

2 Specification

```
FUNCTION G01FAF (TAIL, P, IFAIL)
REAL (KIND=nag_wp) G01FAF
INTEGER          IFAIL
REAL (KIND=nag_wp) P
CHARACTER(1)    TAIL
```

3 Description

The deviate, x_p associated with the lower tail probability, p , for the standard Normal distribution is defined as the solution to

$$P(X \leq x_p) = p = \int_{-\infty}^{x_p} Z(X) dX,$$

where

$$Z(X) = \frac{1}{\sqrt{2\pi}} e^{-X^2/2}, \quad -\infty < X < \infty.$$

The method used is an extension of that of Wichura (1988). p is first replaced by $q = p - 0.5$.

(a) If $|q| \leq 0.3$, x_p is computed by a rational Chebyshev approximation

$$x_p = s \frac{A(s^2)}{B(s^2)},$$

where $s = \sqrt{2\pi}q$ and A , B are polynomials of degree 7.

(b) If $0.3 < |q| \leq 0.42$, x_p is computed by a rational Chebyshev approximation

$$x_p = \text{sign } q \left(\frac{C(t)}{D(t)} \right),$$

where $t = |q| - 0.3$ and C , D are polynomials of degree 5.

(c) If $|q| > 0.42$, x_p is computed as

$$x_p = \text{sign } q \left[\left(\frac{E(u)}{F(u)} \right) + u \right],$$

where $u = \sqrt{-2 \times \log(\min(p, 1-p))}$ and E , F are polynomials of degree 6.

For the upper tail probability $-x_p$ is returned, while for the two tail probabilities the value x_{p^*} is returned, where p^* is the required tail probability computed from the input value of p .

4 References

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

Hastings N A J and Peacock J B (1975) *Statistical Distributions* Butterworth

Wichura (1988) Algorithm AS 241: the percentage points of the Normal distribution *Appl. Statist.* **37** 477–484

5 Arguments

1: TAIL – CHARACTER(1) *Input*

On entry: indicates which tail the supplied probability represents.

TAIL = 'L'

The lower probability, i.e., $P(X \leq x_p)$.

TAIL = 'U'

The upper probability, i.e., $P(X \geq x_p)$.

TAIL = 'S'

The two tail (significance level) probability, i.e., $P(X \geq |x_p|) + P(X \leq -|x_p|)$.

TAIL = 'C'

The two tail (confidence interval) probability, i.e., $P(X \leq |x_p|) - P(X \leq -|x_p|)$.

Constraint: TAIL = 'L', 'U', 'S' or 'C'.

2: P – REAL (KIND=nag_wp) *Input*

On entry: p , the probability from the standard Normal distribution as defined by TAIL.

Constraint: $0.0 < P < 1.0$.

3: 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:

If on exit IFAIL \neq 0, then G01FAF returns 0.0.

IFAIL = 1

On entry, TAIL \neq 'L', 'U', 'S' or 'C'.

IFAIL = 2

On entry, $P \leq 0.0$,
or $P \geq 1.0$.

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

The accuracy is mainly limited by the *machine precision*.

8 Parallelism and Performance

G01FAF is not threaded in any implementation.

9 Further Comments

None.

10 Example

Four values of TAIL and P are input and the deviates calculated and printed.

10.1 Program Text

```

Program g01faf
!      G01FAF Example Program Text
!      Mark 26 Release. NAG Copyright 2016.
!
!      .. Use Statements ..
      Use nag_library, Only: g01faf, nag_wp
!      .. Implicit None Statement ..
      Implicit None
!      .. Parameters ..
      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
      Real (Kind=nag_wp)          :: dev, p
      Integer                     :: ifail
      Character (1)               :: tail
!      .. Executable Statements ..
      Write (nout,*) 'G01FAF Example Program Results'
      Write (nout,*)
!
!      Skip heading in data file
      Read (nin,*)
!
!      Display titles

```

```

        Write (nout,*) ' Tail      Probability      Deviate '
        Write (nout,*)

d_lp: Do
    Read (nin,*,Iostat=ifail) tail, p
    If (ifail/=0) Then
        Exit d_lp
    End If

!       Calculate the deviate (inverse CDF)
        ifail = 0
        dev = g01faf(tail,p,ifail)

!       Display results
        Write (nout,99999) tail, p, dev
    End Do d_lp

99999 Format (3X,A1,8X,F6.3,8X,F7.4)
    End Program g01faf

```

10.2 Program Data

G01FAF Example Program Data

```

'L' 0.975
'U' 0.025
'C' 0.95
'S' 0.05

```

10.3 Program Results

G01FAF Example Program Results

Tail	Probability	Deviante
L	0.975	1.9600
U	0.025	1.9600
C	0.950	1.9600
S	0.050	1.9600
