

NAG Library Routine Document

G01FEF

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

G01FEF returns the deviate associated with the given lower tail probability of the beta distribution, via the routine name.

2 Specification

```
FUNCTION G01FEF (P, A, B, TOL, IFAIL)
REAL (KIND=nag_wp) G01FEF
INTEGER IFAIL
REAL (KIND=nag_wp) P, A, B, TOL
```

3 Description

The deviate, β_p , associated with the lower tail probability, p , of the beta distribution with parameters a and b is defined as the solution to

$$P(B \leq \beta_p : a, b) = p = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^{\beta_p} B^{a-1} (1-B)^{b-1} dB, \quad 0 \leq \beta_p \leq 1; a, b > 0.$$

The algorithm is a modified version of the Newton–Raphson method, following closely that of Cran *et al.* (1977).

An initial approximation, β_0 , to β_p is found (see Cran *et al.* (1977)), and the Newton–Raphson iteration

$$\beta_i = \beta_{i-1} - \frac{f(\beta_{i-1})}{f'(\beta_{i-1})},$$

where $f(\beta) = P(B \leq \beta : a, b) - p$ is used, with modifications to ensure that β remains in the range $(0, 1)$.

4 References

Cran G W, Martin K J and Thomas G E (1977) Algorithm AS 109. Inverse of the incomplete beta function ratio *Appl. Statist.* **26** 111–114

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

5 Arguments

- | | | |
|----|--|--------------|
| 1: | P – REAL (KIND=nag_wp) | <i>Input</i> |
| | <i>On entry:</i> p , the lower tail probability from the required beta distribution. | |
| | <i>Constraint:</i> $0.0 \leq P \leq 1.0$. | |
| 2: | A – REAL (KIND=nag_wp) | <i>Input</i> |
| | <i>On entry:</i> a , the first parameter of the required beta distribution. | |
| | <i>Constraint:</i> $0.0 < A \leq 10^6$. | |

- 3: B – REAL (KIND=nag_wp) *Input*
On entry: b , the second parameter of the required beta distribution.
Constraint: $0.0 < B \leq 10^6$.
- 4: TOL – REAL (KIND=nag_wp) *Input*
On entry: the relative accuracy required by you in the result. If G01FEF is entered with TOL greater than or equal to 1.0 or less than $10 \times \textit{machine precision}$ (see X02AJF), then the value of $10 \times \textit{machine precision}$ is used instead.
- 5: 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, because for this routine the values of the output arguments may be useful even if $IFAIL \neq 0$ on exit, the recommended value is -1 . **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).

Note: G01FEF may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the routine:

If on exit $IFAIL = 1$ or 2, then G01FEF returns 0.0.

$IFAIL = 1$

On entry, $P < 0.0$,
 or $P > 1.0$.

$IFAIL = 2$

On entry, $A \leq 0.0$,
 or $A > 10^6$,
 or $B \leq 0.0$,
 or $B > 10^6$.

$IFAIL = 3$

There is doubt concerning the accuracy of the computed result. 100 iterations of the Newton–Raphson method have been performed without satisfying the accuracy criterion (see Section 7). The result should be a reasonable approximation of the solution.

$IFAIL = 4$

Requested accuracy not achieved when calculating beta probability. The result should be a reasonable approximation to the correct solution. You should try setting TOL larger.

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 required precision, given by TOL, should be achieved in most circumstances.

8 Parallelism and Performance

G01FEF is not threaded in any implementation.

9 Further Comments

The typical timing will be several times that of G01EEF and will be very dependent on the input argument values. See G01EEF for further comments on timings.

10 Example

This example reads lower tail probabilities for several beta distributions and calculates and prints the corresponding deviates until the end of data is reached.

10.1 Program Text

```

Program g01fef

!      G01FEF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
      Use nag_library, Only: g01fef, nag_wp
!      .. Implicit None Statement ..
      Implicit None
!      .. Parameters ..
      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
      Real (Kind=nag_wp)          :: a, b, p, tol, x
      Integer                     :: ifail
!      .. Executable Statements ..
      Write (nout,*) 'G01FEF Example Program Results'
      Write (nout,*)

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

!      Display titles
      Write (nout,*) ' Probability      A          B      Deviate'
      Write (nout,*)

!      Use default tolerance

```

```

      tol = 0.0E0_nag_wp

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

!    Calculate deviates (inverse CDF)
  ifail = -1
  x = g01fef(p,a,b,tol,ifail)
  If (ifail/=0) Then
    If (ifail/=3 .And. ifail/=4) Then
      Exit d_lp
    End If
  End If

!    Display results
  Write (nout,99999) p, a, b, x
End Do d_lp

99999 Format (1X,F9.4,2F10.3,F10.4,A,I1)
End Program g01fef

```

10.2 Program Data

G01FEF Example Program Data

0.5000	1.0	2.0	:P A B
0.9900	1.5	1.5	:P A B
0.2500	20.0	10.0	:P A B

10.3 Program Results

G01FEF Example Program Results

Probability	A	B	Deviates
0.5000	1.000	2.000	0.2929
0.9900	1.500	1.500	0.9672
0.2500	20.000	10.000	0.6105
