Module 20.6: nag_gamma_dist

Probabilities and Deviate for a Gamma Distribution

`nag_gamma_dist` provides procedures for computing probabilities and the deviate for a gamma distribution.

Contents

Procedure: nag_gamma_prob

1 Description

nag_gamma_prob calculates the lower or upper tail probability for a gamma distribution with shape parameter a and scale parameter b.

2 Usage

```
USE nag_gamma_dist
[value =] nag_gamma_prob(tail, g, a, b  [, optional arguments])
```

The function result is a scalar of type real(kind=*wp*).

3 Arguments

3.1 Mandatory Arguments

tail — character(len=1), intent(in)

Input: the type of tail probability to be returned:

if tail = 'L' or 'l', the lower tail probability is returned;

if tail = 'U' or 'u', the upper tail probability is returned.

Constraints: tail = 'L', 'l', 'U' or 'u'.

g — real(kind=*wp*), intent(in)

Input: the value of the gamma variate.

Constraints: $g \geq 0.0$.

a — real(kind=*wp*), intent(in)

Input: the shape parameter of the gamma distribution.

Constraints: $0.0 < a \leq 10^6$.

b — real(kind=*wp*), intent(in)

Input: the scale parameter of the gamma distribution.

Constraints: $b > 0.0$.

3.2 Optional Argument

error — type(nag_error), intent(inout), optional

The NAG *fl*90 error-handling argument. See the Essential Introduction, or the module document nag_error_handling (1.2). You are recommended to omit this argument if you are unsure how to use it. If this argument is supplied, it *must* be initialized by a call to nag_set_error before this procedure is called.

4 Error Codes

Fatal errors (error%level = 3):

error%code	Description
301	An input argument has an invalid value.

Warnings (error%level = 1):

error%code	Description
101	The solution did not converge in 600 iterations. The probability returned should still be a reasonable approximation to the required solution.

5 Examples of Usage

A complete example of the use of this procedure appears in Example 1 of this module document.

6 Further Comments

6.1 Mathematical Background

Let g be a gamma distributed variate with shape parameter a and scale parameter b. The lower tail probability $P(G \leq g : a, b)$ is defined by

$$P(G \leq g; a, b) = \frac{1}{b^a \Gamma(a)} \int_0^g G^{a-1} e^{-G/b} \, dG, \quad a > 0.0, \quad b > 0.0.$$

The mean of the distribution is ab and its variance is ab^2. The transformation $Z = G/b$ is applied to yield the following incomplete gamma function in normalised form:

$$P(G \leq g; a, b) = P(Z \leq g/b : a, 1.0) = \frac{1}{\Gamma(a)} \int_0^{g/b} Z^{a-1} e^{-Z} \, dZ.$$

6.2 Accuracy

The result should have a relative accuracy of `EPSILON(1.0_wp)`. There are rare occasions when the relative accuracy attained is somewhat less than `EPSILON(1.0_wp)` but the error should not exceed more than 1 or 2 decimal place(s).

6.3 Timing

The time taken by the procedure varies slightly with the arguments `g`, `a`, and `b`.

Procedure: nag_gamma_deviate

1 Description

`nag_gamma_deviate` returns the deviate associated with the lower tail probability of a gamma distribution with shape parameter a and scale parameter b.

2 Usage

```
USE nag_gamma_dist
[value =] nag_gamma_deviate(p, a, b  [, optional arguments])
```

The function result is a scalar of type real(kind=wp).

3 Arguments

3.1 Mandatory Arguments

p — real(kind=wp), intent(in)

Input: the lower tail probability of the gamma distribution.

Constraints: $0.0 \leq$ `p` < 1.0.

a — real(kind=wp), intent(in)

Input: the shape parameter of the gamma distribution.

Constraints: $0.0 <$ `a` $\leq 10^6$.

b — real(kind=wp), intent(in)

Input: the scale parameter of the gamma distribution.

Constraints: `b` > 0.0.

3.2 Optional Arguments

Note. Optional arguments must be supplied by keyword, not by position. The order in which they are described below may differ from the order in which they occur in the argument list.

tol — real(kind=wp), intent(in), optional

Input: the relative accuracy which you want for the result.

Default: `tol` $= 50 \times \delta$, where $\delta = \max(10^{-18}$,`EPSILON(1.0_wp)`$)$.

Note: if `tol` is $< 50 \times \delta$ or `tol` ≥ 1.0, the default value is used.

error — type(nag_error), intent(inout), optional

The NAG *fl*90 error-handling argument. See the Essential Introduction, or the module document `nag_error_handling` (1.2). You are recommended to omit this argument if you are unsure how to use it. If this argument is supplied, it *must* be initialized by a call to `nag_set_error` before this procedure is called.

4 Error Codes

Fatal errors (error%level = 3):

error%code	Description
301	An input argument has an invalid value.

Failures (error%level = 2):

error%code	Description
201	The value of `p` is too close to 0.0 or 1.0 for the deviate to be computed. The result is set to 0.0.
202	The series to calculate the gamma probabilities has failed to converge. The result is set to 0.0, however this is an unlikely error.

Warnings (error%level = 1):

error%code	Description
101	The accuracy of the result is doubtful. This is because 100 iterations of the underlying method have been performed without satisfying the accuracy criterion. Nevertheless, the result should be a reasonable approximation to the correct solution, a larger value for `tol` should probably be used.

5 Examples of Usage

A complete example of the use of this procedure appears in Example 1 of this module document.

6 Further Comments

6.1 Mathematical Background

Given the lower tail probability p of a gamma distribution with parameters a and b, the deviate g_p associated with p is defined as the solution to

$$P(G \le g_p : a, b) = p = \frac{1}{b^a \Gamma(a)} \int_0^{g_p} e^{-G/b} G^{a-1} \, dG, \quad 0 \le g_p < \infty, \quad a, b > 0.$$

6.2 Algorithmic Detail

The method used is described by Best and Roberts [1], making use of the relationship between the gamma distribution and the χ^2-distribution.

Let $y = 2\frac{g_p}{b}$. The required y is found from the Taylor series expansion

$$y = y_0 + \sum_r \frac{C_r(y_0)}{r!} \left(\frac{E}{\phi(y_0)} \right)^r$$

where y_0 is a starting approximation,

$$C_1(u) = 1, \quad C_{r+1}(u) = \left(r\Psi + \frac{d}{du} \right) C_r(u),$$

$$\Psi = \frac{1}{2} - \frac{a-1}{u}, \quad E = p - \int_0^{y_0} \phi(u) \, du \quad \text{and} \quad \phi(u) = \frac{1}{2^a \Gamma(a)} e^{-u/2} u^{a-1}.$$

For most values of p and a the starting value

$$y_{01} = 2a \left(z \sqrt{\frac{1}{9a}} + 1 - \frac{1}{9a} \right)^3$$

is used, where z is the deviate associated with a lower tail probability of p for the standard Normal distribution.

For p close to zero,

$$y_{02} = \left(pa2^a\Gamma\left(a\right)\right)^{1/a}$$

is used.
For large values of p, when $y_{01} > 4.4a + 6.0$

$$y_{03} = -2\left(\ln(1-p) - (a-1)\ln\left(\frac{1}{2}y_{01}\right) + \ln\left(\Gamma\left(a\right)\right)\right)$$

is found to be a better starting value than y_{01}.
For small a ($a \leq 0.16$), p is expressed in terms of an approximation to the exponential integral and y_{04} is found by Newton–Raphson iterations.
Seven terms of the Taylor series are used to refine the starting approximation, repeating the process, if necessary, until the required accuracy is obtained.

6.3 Accuracy

In most cases the relative accuracy of the results should be as specified by `tol`. However, for very small values of `a` or very small values of `p`, there may be some loss of accuracy.

Example 1: Calculation of probabilities and the deviate for a Gamma distribution

This example program shows how `nag_gamma_prob` returns the lower tail probability or upper tail probability for a beta distribution with parameters a and b. It also shows how `nag_gamma_deviate` calculates the deviate (g_calculated) associated with a given lower tail probability.

1 Program Text

Note. The listing of the example program presented below is double precision. Single precision users are referred to Section 5.2 of the Essential Introduction for further information.

```
PROGRAM nag_gamma_dist_ex01

  ! Example Program Text for nag_gamma_dist
  ! NAG fl90, Release 3. NAG Copyright 1997.

  ! .. Use Statements ..
  USE nag_examples_io, ONLY : nag_std_out, nag_std_in
  USE nag_gamma_dist, ONLY : nag_gamma_prob, nag_gamma_deviate
  ! .. Implicit None Statement ..
  IMPLICIT NONE
  ! .. Intrinsic Functions ..
  INTRINSIC KIND
  ! .. Parameters ..
  INTEGER, PARAMETER :: wp = KIND(1.0D0)
  ! .. Local Scalars ..
  REAL (wp) :: a, b, g, g_calculated, prob, probl
  CHARACTER (1) :: tail
  ! .. Executable Statements ..

  WRITE (nag_std_out,*) 'Example Program Results for nag_gamma_dist_ex01'

  READ (nag_std_in,*)          ! Skip heading in data file

  WRITE (nag_std_out,*)
  WRITE (nag_std_out,*) 'tail   Gamma_variate     a       b &
   &      Probability    Gamma_calculated'
  WRITE (nag_std_out,*)

  DO
    READ (nag_std_in,*,end=20) tail, g, a, b

    prob = nag_gamma_prob(tail,g,a,b)

    probl = prob
    IF (tail=='u' .OR. tail=='U') probl = 1.0_wp - prob

    g_calculated = nag_gamma_deviate(probl,a,b)

    WRITE (nag_std_out,'(1X,A,F15.2,2F10.1,5x,F10.4,F15.4)') tail, g, a, &
     b, prob, g_calculated

  END DO
20 CONTINUE

END PROGRAM nag_gamma_dist_ex01
```

2 Program Data

```
Example Program Data for nag_gamma_dist_ex01
  'L'  15.5   4.0   2.0           :tail, g, a, b
  'U'   0.5   4.0   1.0
  'U'  10.0   1.0   2.0
  'L'   5.0   2.0   2.0
```

3 Program Results

```
 Example Program Results for nag_gamma_dist_ex01

 tail   Gamma_variate     a       b       Probability    Gamma_calculated

 L          15.50       4.0     2.0          0.9499        15.5000
 U           0.50       4.0     1.0          0.9982         0.5000
 U          10.00       1.0     2.0          0.0067        10.0000
 L           5.00       2.0     2.0          0.7127         5.0000
```

Additional Examples

Not all example programs supplied with NAG *fl*90 appear in full in this module document. The following additional examples, associated with this module, are available.

`nag_gamma_dist_ex02`

Calculation of the deviate associated with a given lower tail probability for a gamma distribution with known parameters.

References

[1] Best D J and Roberts D E (1975) Algorithm AS91. The percentage points of the χ^2 distribution *Appl. Statist.* **24** 385–388