nag_gamma_dist (g01efc)

1. Purpose

nag_gamma_dist (g01efc) returns the lower or upper tail probability of the gamma distribution, with parameters α and β.

2. Specification

```
#include <nag.h>
#include <nagg01.h>

double nag_gamma_dist(Nag_TailProbability tail, double g, double a, double b,
       NagError *fail)
```

3. Description

The lower tail probability for the gamma distribution with parameters α and β, $P(G \leq g)$, is defined by

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

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

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

4. Parameters

tail
Input: indicates whether the upper or lower tail probability is required.
If **tail** = **Nag_LowerTail**, the lower tail probability is returned, i.e., $P(G \leq g : \alpha, \beta)$.
If **tail** = **Nag_UpperTail**, the upper tail probability is returned, i.e., $P(G \geq g : \alpha, \beta)$.
Constraint: **tail** = **Nag_LowerTail** or **Nag_UpperTail**.

g
Input: the value of the gamma variate, g.
Constraint: **g** ≥ 0.0.

a
Input: the parameter α of the gamma distribution.
Constraint: **a** > 0.0.

b
Input: the parameter β of the gamma distribution.
Constraint: **b** > 0.0.

fail
The NAG error parameter, see the Essential Introduction to the NAG C Library.

5. Error Indications and Warnings

On any of the error conditions listed below except **NE_ALG_NOT_CONV** nag_gamma_dist returns 0.0.

NE_BAD_PARAM
On entry, parameter **tail** had an illegal value.

NE_REAL_ARG_LT
On entry, **g** must not be less than 0.0: **g** = ⟨*value*⟩.

NE_REAL_ARG_LE
On entry, **a** must not be less than or equal to 0.0: **a** = ⟨*value*⟩.
On entry, **b** must not be less than or equal to 0.0: **b** = ⟨*value*⟩.

NE_ALG_NOT_CONV
The algorithm has failed to converge in ⟨*value*⟩ iterations.
The probability returned should be a reasonable approximation to the solution.

6. Further Comments

The time taken by the function varies slightly with the input parameters g, α and β.

6.1. Accuracy

The result should have a relative accuracy of ***machine precision***. There are rare occasions when the relative accuracy attained is somewhat less than ***machine precision*** but the error should not exceed more than 1 or 2 decimal places. Note also that there is a limit of 18 decimal places on the achievable accuracy.

6.2. References

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

7. See Also

None.

8. Example

The following example reads in values for several gamma distributions, computes and prints the lower probabilities for each case, until the end of data is reached.

8.1. Program Text

```
/* nag_gamma_dist(g01efc) Example Program
 *
 * Copyright 1990 Numerical Algorithms Group.
 *
 * Mark 1, 1990.
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nagg01.h>

main()
{
  double a, b, g, p;

  /* Skip heading in data file */
  Vscanf("%*[^\n]");
  Vprintf("g01efc Example Program Results\n");
  Vprintf(" Gamma deviate    Alpha     Beta    Lower tail prob.\n\n");
  while (scanf("%lf %lf %lf", &g, &a, &b) != EOF)
    {
      p = g01efc(Nag_LowerTail, g, a, b, NAGERR_DEFAULT);
      Vprintf(" %9.2f%13.2f%9.2f%14.4f\n", g, a, b, p);
    }
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
g01efc Example Program Data
 15.5   4.0   2.0
```

```
 0.5   4.0   1.0
10.0   1.0   2.0
 5.0   2.0   2.0
```

8.3. Program Results

```
g01efc Example Program Results
 Gamma deviate    Alpha     Beta    Lower tail prob.

      15.50         4.00     2.00        0.9499
       0.50         4.00     1.00        0.0018
      10.00         1.00     2.00        0.9933
       5.00         2.00     2.00        0.7127
```