NAG Toolbox

nag_specfun_beta_incomplete (s14cc)

1 Purpose

nag_specfun_beta_incomplete (s14cc) computes values for the incomplete beta function $I_x(a,b)$ and its complement $1 - I_x(a,b)$.

2 Syntax

```
[w, w1, ifail] = nag_specfun_beta_incomplete(a, b, x)
[w, w1, ifail] = s14cc(a, b, x)
```

3 Description

nag_specfun_beta_incomplete (s14cc) evaluates the incomplete beta function and its complement in the normalized form

$$\begin{aligned} I_x(a,b) &= \frac{1}{B(a,b)}\int_0^x t^{a-1}(1-t)^{b-1}dt \\ 1-I_x(a,b) &= I_y(b,a), \text{ where } y = 1-x, \end{aligned}$$

with

$0 \leq x \leq 1$,

$a \geq 0$ and $b \geq 0$,

and the beta function $B(a,b)$ is defined as $B(a,b) = \displaystyle\int_0^1 t^{a-1}(1-t)^{b-1}dt = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ where $\Gamma(y)$ is the gamma function.

Several methods are used to evaluate the functions depending on the arguments a, b and x. The methods include Wise's asymptotic expansion (see Wise (1950)) when $a > b$, continued fraction derived by DiDonato and Morris (1992) when a, $b > 1$, and power series when $b \leq 1$ or $b \times x \leq 0.7$. When both a and b are large, specifically a, $b \geq 15$, the DiDonato and Morris (1992) asymptotic expansion is employed for greater efficiency.

Once either $I_x(a,b)$ or $I_y(b,a)$ is computed, the other is obtained by subtraction from 1. In order to avoid loss of relative precision in this subtraction, the smaller of $I_x(a,b)$ and $I_y(b,a)$ is computed first.

nag_specfun_beta_incomplete (s14cc) is derived from BRATIO in DiDonato and Morris (1992).

4 References

DiDonato A R and Morris A H (1992) Algorithm 708: Significant digit computation of the incomplete beta function ratios *ACM Trans. Math. Software* **18** 360–373

Wise M E (1950) The incomplete beta function as a contour integral and a quickly converging series for its inverse *Biometrika* **37** 208–218

5 Parameters

5.1 Compulsory Input Parameters

1: **a** – REAL (KIND=nag_wp)

The argument a of the function.

Constraint: $\mathbf{a} \geq 0.0$.

2: **b** – REAL (KIND=nag_wp)

The argument b of the function.

Constraints:

$\mathbf{b} \geq 0.0$;
either $\mathbf{b} \neq 0.0$ or $\mathbf{a} \neq 0.0$.

3: **x** – REAL (KIND=nag_wp)

x, upper limit of integration.

Constraints:

$0.0 \leq \mathbf{x} \leq 1.0$;
either $\mathbf{x} \neq 0.0$ or $\mathbf{a} \neq 0.0$;
either $1 - \mathbf{x} \neq 0.0$ or $\mathbf{b} \neq 0.0$.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

1: **w** – REAL (KIND=nag_wp)

The value of the incomplete beta function $I_x(a, b)$ evaluated from zero to x.

2: **w1** – REAL (KIND=nag_wp)

The value of the complement of the incomplete beta function $1 - I_x(a, b)$, i.e., the incomplete beta function evaluated from x to one.

3: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

Constraint: $\mathbf{a} \geq 0.0$.

Constraint: $\mathbf{b} \geq 0.0$.

ifail $= 2$

On entry, **a** and **b** were zero.
Constraint: **a** or **b** must be nonzero.

ifail $= 3$

Constraint: $0.0 \leq \mathbf{x} \leq 1.0$.

ifail $= 4$

On entry, **x** and **a** were zero.
Constraint: **x** or **a** must be nonzero.

ifail $= 5$

On entry, $1.0 - \mathbf{x}$ and **b** were zero.
Constraint: $1.0 - \mathbf{x}$ or **b** must be nonzero.

ifail $= -99$

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

ifail $= -399$

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

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

nag_specfun_beta_incomplete (s14cc) is designed to maintain relative accuracy for all arguments. For very tiny results (of the order of ***machine precision*** or less) some relative accuracy may be lost – loss of three or four decimal places has been observed in experiments. For other arguments full relative accuracy may be expected.

8 Further Comments

None.

9 Example

This example reads values of the arguments a and b from a file, evaluates the function and its complement for 10 different values of x and prints the results.

9.1 Program Text

```
function s14cc_example

fprintf('s14cc example results\n\n');

a = 5.3;
b = 10.1;
fprintf('\n   a     b     x          Ix(a,b)          1-Ix(a,b)\n');
for x = 0.01:0.01:0.1
  [w, w1, ifail] = s14cc(a, b, x);
  fprintf('%6.2f%6.2f%6.2f%17.4e%17.4e\n', a, b, x, w, w1);
end
```

9.2 Program Results

```
s14cc example results

   a     b     x          Ix(a,b)          1-Ix(a,b)
  5.30 10.10  0.01       6.4755e-08       1.0000e+00
  5.30 10.10  0.02       2.3613e-06       1.0000e+00
  5.30 10.10  0.03       1.8734e-05       9.9998e-01
  5.30 10.10  0.04       7.9575e-05       9.9992e-01
  5.30 10.10  0.05       2.3997e-04       9.9976e-01
```

```
  5.30 10.10  0.06       5.8255e-04       9.9942e-01
  5.30 10.10  0.07       1.2174e-03       9.9878e-01
  5.30 10.10  0.08       2.2797e-03       9.9772e-01
  5.30 10.10  0.09       3.9249e-03       9.9608e-01
  5.30 10.10  0.10       6.3236e-03       9.9368e-01
```