NAG Toolbox

nag_rand_dist_beta (g05sb)

1 Purpose

nag_rand_dist_beta (g05sb) generates a vector of pseudorandom numbers taken from a beta distribution with parameters a and b.

2 Syntax

```
[state, x, ifail] = nag_rand_dist_beta(n, a, b, state)
[state, x, ifail] = g05sb(n, a, b, state)
```

3 Description

The beta distribution has PDF (probability density function)

$$f(x) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1}(1-x)^{b-1} \quad \text{if } 0 \leq x \leq 1;\ a, b > 0,$$

$$f(x) = 0 \qquad \text{otherwise.}$$

One of four algorithms is used to generate the variates depending on the values of a and b. Let α be the maximum and β be the minimum of a and b. Then the algorithms are as follows:

(i) if $\alpha < 0.5$, Johnk's algorithm is used, see for example Dagpunar (1988). This generates the beta variate as $u_1^{1/a} / \left(u_1^{1/a} + u_2^{1/b} \right)$, where u_1 and u_2 are uniformly distributed random variates;

(ii) if $\beta > 1$, the algorithm BB given by Cheng (1978) is used. This involves the generation of an observation from a beta distribution of the second kind by the envelope rejection method using a log-logistic target distribution and then transforming it to a beta variate;

(iii) if $\alpha > 1$ and $\beta < 1$, the switching algorithm given by Atkinson (1979) is used. The two target distributions used are $f_1(x) = \beta x^{\beta}$ and $f_2(x) = \alpha(1-x)^{\beta-1}$, along with the approximation to the switching argument of $t = (1-\beta)/(\alpha+1-\beta)$;

(iv) in all other cases, Cheng's BC algorithm (see Cheng (1978)) is used with modifications suggested by Dagpunar (1988). This algorithm is similar to BB, used when $\beta > 1$, but is tuned for small values of a and b.

One of the initialization functions nag_rand_init_repeat (g05kf) (for a repeatable sequence if computed sequentially) or nag_rand_init_nonrepeat (g05kg) (for a non-repeatable sequence) must be called prior to the first call to nag_rand_dist_beta (g05sb).

4 References

Atkinson A C (1979) A family of switching algorithms for the computer generation of beta random variates *Biometrika* **66** 141–5

Cheng R C H (1978) Generating beta variates with nonintegral shape parameters *Comm. ACM* **21** 317–322

Dagpunar J (1988) *Principles of Random Variate Generation* Oxford University Press

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

5 Parameters

5.1 Compulsory Input Parameters

1: **n** – INTEGER

n, the number of pseudorandom numbers to be generated.

Constraint: **n** ≥ 0.

2: **a** – REAL (KIND=nag_wp)

a, the parameter of the beta distribution.

Constraint: **a** > 0.0.

3: **b** – REAL (KIND=nag_wp)

b, the parameter of the beta distribution.

Constraint: **b** > 0.0.

4: **state**$(:)$ – INTEGER array

Note: the actual argument supplied **must** be the array **state** supplied to the initialization routines nag_rand_init_repeat (g05kf) or nag_rand_init_nonrepeat (g05kg).

Contains information on the selected base generator and its current state.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

1: **state**$(:)$ – INTEGER array

Contains updated information on the state of the generator.

2: **x**$($**n**$)$ – REAL (KIND=nag_wp) array

The n pseudorandom numbers from the specified beta distribution.

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: **n** ≥ 0.

ifail $= 2$

Constraint: **a** > 0.0.

ifail $= 3$

Constraint: **b** > 0.0.

ifail $= 4$

On entry, **state** vector has been corrupted or not initialized.

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

Not applicable.

8 Further Comments

To generate an observation, y, from the beta distribution of the second kind from an observation, x, generated by nag_rand_dist_beta (g05sb) the transformation, $y = x/(1-x)$, may be used.

9 Example

This example prints a set of five pseudorandom numbers from a beta distribution with parameters $a = 2.0$ and $b = 2.0$, generated by a single call to nag_rand_dist_beta (g05sb), after initialization by nag_rand_init_repeat (g05kf).

9.1 Program Text

```
function g05sb_example

fprintf('g05sb example results\n\n');

% Initialize the base generator to a repeatable sequence
seed  = [nag_int(1762543)];
genid = nag_int(1);
subid = nag_int(1);
[state, ifail] = g05kf( ...
                        genid, subid, seed);

% Number of variates
n = nag_int(5);

% Parameters
a = 2;
b = 2;

% Generate variates from beta distribution
[state, x, ifail] = g05sb( ...
                           n, a, b, state);

disp('Variates');
disp(x);
```

9.2 Program Results

```
g05sb example results

Variates
    0.5977
    0.6818
    0.1797
    0.4174
    0.4987
```