

## NAG Toolbox

### nag\_rand\_int\_binomial (g05ta)

#### 1 Purpose

nag\_rand\_int\_binomial (g05ta) generates a vector of pseudorandom integers from the discrete binomial distribution with parameters  $m$  and  $p$ .

#### 2 Syntax

```
[r, state, x, ifail] = nag_rand_int_binomial(mode, n, m, p, r, state)
[r, state, x, ifail] = g05ta(mode, n, m, p, r, state)
```

#### 3 Description

nag\_rand\_int\_binomial (g05ta) generates  $n$  integers  $x_i$  from a discrete binomial distribution, where the probability of  $x_i = I$  is

$$P(x_i = I) = \frac{m!}{I!(m-I)!} p^I \times (1-p)^{m-I}, \quad I = 0, 1, \dots, m,$$

where  $m \geq 0$  and  $0 \leq p \leq 1$ . This represents the probability of achieving  $I$  successes in  $m$  trials when the probability of success at a single trial is  $p$ .

The variates can be generated with or without using a search table and index. If a search table is used then it is stored with the index in a reference vector and subsequent calls to nag\_rand\_int\_binomial (g05ta) with the same parameter values can then use this reference vector to generate further variates.

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\_int\_binomial (g05ta).

#### 4 References

Kendall M G and Stuart A (1969) *The Advanced Theory of Statistics (Volume 1)* (3rd Edition) Griffin  
 Knuth D E (1981) *The Art of Computer Programming (Volume 2)* (2nd Edition) Addison–Wesley

#### 5 Parameters

##### 5.1 Compulsory Input Parameters

1: **mode** – INTEGER

A code for selecting the operation to be performed by the function.

**mode** = 0

Set up reference vector only.

**mode** = 1

Generate variates using reference vector set up in a prior call to nag\_rand\_int\_binomial (g05ta).

**mode** = 2

Set up reference vector and generate variates.

**mode** = 3

Generate variates without using the reference vector.

*Constraint:* **mode** = 0, 1, 2 or 3.

2: **n** – INTEGER

$n$ , the number of pseudorandom numbers to be generated.

*Constraint:* **n**  $\geq$  0.

3: **m** – INTEGER

$m$ , the number of trials of the distribution.

*Constraint:* **m**  $\geq$  0.

4: **p** – REAL (KIND=nag\_wp)

$p$ , the probability of success of the binomial distribution.

*Constraint:*  $0.0 \leq \mathbf{p} \leq 1.0$ .

5: **r**( $lr$ ) – REAL (KIND=nag\_wp) array

$lr$ , the dimension of the array, must satisfy the constraint

if **mode** = 0 or 2,

$$lr > \min(\mathbf{m}, \text{int}[\mathbf{m} \times \mathbf{p} + 7.15 \times \sqrt{\mathbf{m} \times \mathbf{p} \times (1 - \mathbf{p})} + 1]) \\ - \max(0, \text{int}[\mathbf{m} \times \mathbf{p} - 7.15 \times \sqrt{\mathbf{m} \times \mathbf{p} \times (1 - \mathbf{p})} - 7.15]) + 8;$$

if **mode** = 1,  $lr$  must remain unchanged from the previous call to nag\_rand\_int\_binomial (g05ta).

.

If **mode** = 1, the reference vector from the previous call to nag\_rand\_int\_binomial (g05ta).

If **mode** = 3, **r** is not referenced.

6: **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: **r**( $lr$ ) – REAL (KIND=nag\_wp) array

If **mode**  $\neq$  3, the reference vector.

2: **state**(:) – INTEGER array

Contains updated information on the state of the generator.

3: **x**(**n**) – INTEGER array

The  $n$  pseudorandom numbers from the specified binomial distribution.

4: **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: **mode** = 0, 1, 2 or 3.

**ifail** = 2

Constraint: **n**  $\geq$  0.

**ifail** = 3

Constraint: **m**  $\geq$  0.

**ifail** = 4

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

**ifail** = 5

On entry, some of the elements of the array **r** have been corrupted or have not been initialized.  
**p** or **m** is not the same as when **r** was set up in a previous call.

**ifail** = 6

On entry, *lr* is too small when **mode** = 0 or 2.

**ifail** = 7

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

None.

## 9 Example

This example prints 20 pseudorandom integers from a binomial distribution with parameters  $m = 6000$  and  $p = 0.8$ , generated by a single call to `nag_rand_int_binomial` (g05ta), after initialization by `nag_rand_init_repeat` (g05kf).

## 9.1 Program Text

```
function g05ta_example

fprintf('g05ta 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(20);

% Parameters
m = nag_int(6000);
p = 0.8;
r = zeros(500, 1);

% r is a reasonable size, so ...
mode = nag_int(2);

% Generate variates from a binomial distribution
[r, state, x, ifail] = g05ta( ...
    mode, n, m, p, r, state);

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

## 9.2 Program Results

```
g05ta example results

Variates
    4811
    4761
    4821
    4826
    4761
    4800
    4791
    4825
    4800
    4814
    4749
    4780
    4810
    4750
    4807
    4782
    4778
    4877
    4840
    4802
```

---