

NAG Toolbox

nag_rand_sample_wgt (g05ne)

1 Purpose

nag_rand_sample_wgt (g05ne) selects a pseudorandom sample, without replacement and allowing for unequal probabilities.

2 Syntax

```
[isampl, state, ifail] = nag_rand_sample_wgt(order, wt, pop, ipop, m, state,
'n', n)
[isampl, state, ifail] = g05ne(order, wt, pop, ipop, m, state, 'n', n)
```

3 Description

nag_rand_sample_wgt (g05ne) selects m elements from either the set of values $(1, 2, \dots, n)$ or a supplied population vector of length n . The probability of selecting the i th element is proportional to a user-supplied weight, w_i . Each element will appear at most once in the sample, i.e., the sampling is done without replacement.

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_sample_wgt (g05ne).

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **order** – CHARACTER(1)

A flag indicating the sorted status of the **wt** vector.

order = 'A'

wt is sorted in ascending order,

order = 'D'

wt is sorted in descending order,

order = 'U'

wt is unsorted and nag_rand_sample_wgt (g05ne) will sort the weights prior to using them.

Irrespective of the value of **order**, no checks are made on the sorted status of **wt**, e.g., it is possible to supply **order** = 'A', even when **wt** is not sorted. In such cases the **wt** array will not be sorted internally, but nag_rand_sample_wgt (g05ne) will still work correctly except, possibly, in cases of extreme weight values.

It is usually more efficient to specify a value of **order** that is consistent with the status of **wt**.

Constraint: **order** = 'A', 'D' or 'U'.

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

w_i , the relative probability weights. These weights need not sum to 1.0.

Constraints:

$\mathbf{wt}(i) \geq 0.0$, for $i = 1, 2, \dots, \mathbf{n}$;
at least \mathbf{m} values must be nonzero.

3: **pop** – CHARACTER(1)

A flag indicating whether a population to be sampled has been supplied.

pop = 'D'

the population is assumed to be the integers $(1, 2, \dots, \mathbf{n})$ and **ipop** is not referenced,

pop = 'S'

the population must be supplied in **ipop**.

Constraint: **pop** = 'D' or 'S'.

4: **ipop(:)** – INTEGER array

The dimension of the array **ipop** must be at least \mathbf{n} if **pop** = 'S'

The population to be sampled. If **pop** = 'D' then the population is assumed to be the set of values $(1, 2, \dots, \mathbf{n})$ and the array **ipop** is not referenced. Elements of **ipop** with the same value are not combined, therefore if $\mathbf{wt}(i) \neq 0$, $\mathbf{wt}(j) \neq 0$ and $i \neq j$ then there is a nonzero probability that the sample will contain both **ipop**(i) and **ipop**(j). If **ipop**(i) = **ipop**(j) then that value can appear in **isampl** more than once.

5: **m** – INTEGER

m , the size of the sample required.

Constraint: $0 \leq \mathbf{m} \leq \mathbf{n}$.

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

1: **n** – INTEGER

Default: the dimension of the array **wt**.

n , the size of the population.

Constraint: $\mathbf{n} \geq 1$.

5.3 Output Parameters

1: **isampl(m)** – INTEGER array

The selected sample.

2: **state(:)** – INTEGER array

Contains updated information on the state of the generator.

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

On entry, **order** = $\langle value \rangle$ was an illegal value.

On entry, **order** had an illegal value.

ifail = 2

On entry, at least one weight was less than zero.

ifail = 3

On entry, **pop** had an illegal value.

ifail = 5

Constraint: $n \geq 1$.

ifail = 7

Constraint: $0 \leq m \leq n$.

ifail = 8

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

ifail = 21

Constraint: must be at least **m** nonzero weights.

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

nag_rand_sample_wgt (g05ne) internally allocates $(n + 1)$ doubles and **n** integers.

Although it is possible to use nag_rand_sample_wgt (g05ne) to sample using equal probabilities, by setting all elements of the input array **wt** to the same positive value, it is more efficient to use nag_rand_sample (g05nd). To sample with replacement, nag_rand_int_general (g05td) can be used when the probabilities are unequal and nag_rand_int_uniform (g05tl) when the probabilities are equal.

9 Example

This example samples from a population of 25.

9.1 Program Text

```
function g05ne_example

fprintf('g05ne example results\n\n');

genid = nag_int(3);
subid = nag_int(0);
seed = [nag_int(1762543)];
% Initialise the generator to a repeatable sequence
[state, ifail] = g05kf( ...
    genid, subid, seed);

% Data to sample with weights
ipop = [nag_int(171); 52; 172; 139; 196;
        125; 36; 70; 25; 86;
        76; 37; 185; 40; 90;
        27; 79; 118; 142; 127;
        101; 22; 41; 199; 59];
wt = [ 85.54; 71.78; 118.13; 13.68; 153.60;
       165.35; 122.35; 35.87; 151.78; 128.33;
       178.27; 183.37; 165.81; 101.41; 145.16;
       42.01; 59.08; 17.53; 87.14; 69.20;
       31.13; 60.26; 21.00; 85.06; 119.73];

% Generate the sample size m without replacement, unequal weights
m = nag_int(10);
pop = 's';
order = 'u';
[isampl, state, ifail] = g05ne( ...
    order, wt, pop, ipop, m, state);

% Display the results
disp(isampl);
```

9.2 Program Results

```
g05ne example results
```

```

125
 41
185
 40
 37
196
 22
 25
 76
172
```
