

nag_ref_vec_discrete_pdf_cdf (g05exc)

1. Purpose

nag_ref_vec_discrete_pdf_cdf (g05exc) sets up the reference vector **R** for a discrete distribution with probability density function (PDF) or cumulative distribution function (CDF) **p**.

2. Specification

```
#include <nag.h>
#include <nagg05.h>
```

```
void nag_ref_vec_discrete_pdf_cdf(double p[], Integer np, Integer sizep,
    Nag_DiscreteDistrib distf, double **r, NagError *fail)
```

3. Description

nag_ref_vec_discrete_pdf_cdf sets up a reference vector for use in **nag_return_discrete (g05exc)** according to information supplied by the user in **p**. This may either be the PDF or CDF of the distribution.

4. Parameters

p[np]

Input: the PDF or CDF of the distribution.

np

Input: the dimension of the array **p** as declared in the function from which **nag_ref_vec_discrete_pdf_cdf** is called. **np** is used in the calculation of the length of the reference vector.

Constraint: **np** $\geq$ 1.

sizep

Input: the value of the variate, assumed to be a whole number, to which the probability in **p[0]** corresponds.

distf

Input: **distf** indicates the type of information contained in **p**. If **distf** is **Nag_CDF**, **p** contains a cumulative distribution function (CDF); if **distf** is **Nag_PDF**, **p** contains a probability density function (PDF).

Constraint: **distf** = **Nag_PDF** or **Nag_CDF**.

r

Output: reference vector for which memory will be allocated internally. If no memory is allocated to **r** (e.g. when an input error is detected) then **r** will be NULL on return.

fail

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

5. Error Indications and Warnings

NE_BAD_PARAM

On entry, parameter **distf** had an illegal value.

NE_INT_ARG_LT

On entry, **np** must not be less than 1: **np** = $\langle value \rangle$.

NE_ALLOC_FAIL

Memory allocation failed.

NE_PROB_NOT_ONE

The total probability is not 1, may be due to rounding errors, total probability = $\langle value \rangle$.

NE_ALL_PROB_ZERO

All the probabilities in array **p** are zero.

NE_NEG_PROB

At least one of the probabilities in array **p** is negative, $\mathbf{p}[\langle value \rangle] = \langle value \rangle$.

NE_NOT_INCREASING

distf = **Nag_CDF** and the sequence **p** is not increasing: $\mathbf{p}[i - 1] = \langle value \rangle$, $\mathbf{p}[i] = \langle value \rangle$.

6. Further Comments**6.1. Accuracy**

Not applicable.

6.2. References

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

7. See Also

nag_random_init_repeatable (g05cbc)
 nag_random_init_nonrepeatable (g05ccc)
 nag_return_discrete (g05eyc)

8. Example

The example program sets up a reference vector for a distribution whose CDF, $f(n)$, is defined as follows:

n	$f(n)$
0	0.0
1	0.1
2	0.2
3	0.4
4	0.5
5	0.6
6	0.8
7	0.9
8	1.0
9	1.0

It then prints the first five pseudo-random numbers generated by nag_ref_vec_discrete_pdf_cdf, after initialisation by nag_random_init_repeatable (g05cbc).

8.1. Program Text

```
/* nag_ref_vec_discrete_pdf_cdf(g05exc) Example Program
 *
 * Copyright 1991 Numerical Algorithms Group.
 *
 * Mark 2, 1991.
 *
 * Mark 3 revised, 1994.
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nagg05.h>

#define NP 10

main()
{
    static double p[NP] = {0.0, 0.1, 0.2, 0.4, 0.5, 0.6, 0.8, 0.9, 1.0, 1.0};
    Integer i, x;
    double *r;
```

```
Vprintf("g05exc Example Program Results\n");
g05cbc((Integer)0);
g05exc(p, (Integer)NP, (Integer)0, Nag_CDF, &r, NAGERR_DEFAULT);
for (i=1; i<=5; i++)
{
    x = g05eyc(r);
    Vprintf("%5ld\n",x);
}
NAG_FREE(r);
exit(EXIT_SUCCESS);
}
```

8.2. Program Data

None.

8.3. Program Results

```
g05exc Example Program Results
6
3
3
3
7
```
