NAG C Library Function Document

nag_rngs_students_t (g05lbc)

1 Purpose

nag_rngs_students_t (g05lbc) generates a vector of pseudo-random numbers taken from a Student’s \( t \)-distribution with \( \nu \) degrees of freedom.

2 Specification

```c
void nag_rngs_students_t (Integer df, Integer n, double x[], Integer igen,
   Integer iseed[], NagError *fail)
```

3 Description

The distribution has PDF (probability density function)

\[
f(x) = \frac{\frac{\nu-1}{2}}{(\frac{\nu}{2}-1)!\sqrt{\pi\nu}(1 + \frac{x^2}{\nu})^{\frac{\nu+1}{2}}},
\]

nag_rngs_students_t (g05lbc) calculates the values

\[
y_i = \frac{\nu}{z_i}, \quad i = 1, \ldots, n
\]

where the \( y_i \) are generated by nag_rngs_normal (g05lac) from a Normal distribution with mean 0 and variance 1.0, and the \( z_i \) are generated by nag_rngs_gamma (g05lfc) from a gamma distribution with parameters \( \frac{1}{2} \nu \) and 2 (i.e., from a \( \chi^2 \) distribution with \( \nu \) degrees of freedom).

One of the initialisation functions nag_rngs_init_repeatable (g05kbc) (for a repeatable sequence if computed sequentially) or nag_rngs_init_nonrepeatable (g05kcc) (for a non-repeatable sequence) must be called prior to the first call to nag_rngs_students_t (g05lbc).

4 References


5 Parameters

1: \( df \) – Integer

\( \text{Input} \)

\( \text{On entry:} \) the number of degrees of freedom, \( \nu \), of the distribution.

\( \text{Constraint:} \ df \geq 1. \)

2: \( n \) – Integer

\( \text{Input} \)

\( \text{On entry:} \) the number, \( n \), of pseudo-random numbers to be generated.

\( \text{Constraint:} \ n \geq 0. \)

3: \( x[\text{dim}] \) – double

\( \text{Output} \)

\( \text{Note:} \) the dimension, \( \text{dim} \), of the array \( x \) must be at least \( \max(1, n) \).

\( \text{On exit:} \) the \( n \) pseudo-random numbers from the specified Student’s \( t \)-distribution.
igen – Integer

*Input*

On entry: must contain the identification number for the generator to be used to return a pseudo-
random number and should remain unchanged following initialisation by a prior call to one of the
functions nag_rngs_init_repeatable (g05kbc) or nag_rngs_init_nonrepeatable (g05kcc).


*Input/Output*

On entry: contains values which define the current state of the selected generator.
On exit: contains updated values defining the new state of the selected generator.

fail – NagError *

*Input/Output*

The NAG error parameter (see the Essential Introduction).

6 Error Indicators and Warnings

**NE_INT**

On entry, n = (value).
Constraint: n ≥ 0.
On entry, df = (value).
Constraint: df ≥ 1.

**NE_BAD_PARAM**

On entry, parameter (value) had an illegal value.

**NE_INTERNAL_ERROR**

An internal error has occurred in this function. Check the function call and any array sizes. If the
call is correct then please consult NAG for assistance.

7 Accuracy

Not applicable.

8 Further Comments

The time taken by nag_rngs_students_t (g05lbc) increases with \( \nu \).

9 Example

The example program prints 5 pseudo-random numbers from a Student’s \( t \)-distribution with five degrees of
freedom, generated by a single call to nag_rngs_students_t (g05lbc), after initialisation by
nag_rngs_init_repeatable (g05kbc).

9.1 Program Text

```c
/* nag_rngs_students_t(g05lbc) Example Program.
 * Copyright 2001 Numerical Algorithms Group.
 */
#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagg05.h>

int main(void)
```


{
  /* Scalars */
  Integer i, igen, n;
  Integer exit_status=0;
  NagError fail;

  /* Arrays */
  double *x=0;
  Integer iseed[4];

  INIT_FAIL(fail);
  Vprintf("g05lbc Example Program Results\n\n");

  n = 5;
  /* Allocate memory */
  if ( !(x = NAG_ALLOC(n, double)) )
  {
    Vprintf("Allocation failure\n");
    exit_status = -1;
    goto END;
  }

  /* Initialise the seed to a repeatable sequence */
  iseed[0] = 1762543;
  iseed[1] = 9324783;
  iseed[2] = 42344;
  iseed[3] = 742355;

  /* igen identifies the stream. */
  igen = 1;
  g05kbc(&igen, iseed);
  g05lbc(5, n, x, igen, iseed, &fail);
  if (fail.code != NE_NOERROR)
  {
    Vprintf("Error from g05lbc.\n%s\n", fail.message);
    exit_status = 1;
    goto END;
  }
  for (i = 0; i < n; ++i)
  {
    Vprintf("%10.4f\n", x[i]);
  }

  END:
  if (x) NAG_FREE(x);
  return exit_status;
}

9.2 Program Data
None.

9.3 Program Results

g05lbc Example Program Results

2.3726
-0.8473
-0.0452
-1.3595
-0.5932