

## nag\_random\_gamma (g05ffc)

### 1. Purpose

**nag\_random\_gamma (g05ffc)** generates a vector of pseudo-random variates from a gamma distribution with parameters  $a$  and  $b$ .

### 2. Specification

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

void nag_random_gamma(double a, double b, Integer n, double x[],
                     NagError *fail)
```

### 3. Description

The gamma distribution has PDF (probability density function):

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

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

One of three algorithms is used to generate the variates depending upon the value of  $a$ :

If  $a < 1$  a switching algorithm described by Dagpunar (1988) (called G6), is used. The target distributions are  $f_1(x) = cax^{a-1}/t^a$  and  $f_2(x) = (1-c)e^{-(x-t)}$ , where  $c = t/(t+ae^{-t})$ , and the switching parameter,  $t$ , is taken as  $1-a$ . This is similar to GS algorithm of Ahrens and Dieter (1974) in which  $t = 1$ .

If  $a = 1$  the gamma distribution reduces to the exponential distribution and the method based on the logarithmic transformation of a uniform random variate is used.

If  $a > 1$  the algorithm given by Best (1978) is used. This is based on using a Student's  $t$ -distribution with two degrees of freedom as the target distribution in an envelope rejection method.

### 4. Parameters

**a**

Input: the parameter,  $a$ , of the gamma distribution.  
Constraint: **a** > 0.0.

**b**

Input: the parameter,  $b$ , of the gamma distribution.  
Constraint: **b** > 0.0.

**n**

Input: the number,  $n$ , of pseudo-random numbers to be generated.  
Constraint: **n** ≥ 1.

**x[n]**

Output: the  $n$  pseudo-random variates from the specified gamma distribution.

**fail**

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

### 5. Error Indications and Warnings

**NE\_REAL\_ARG\_LE**

On entry, **a** must not be less than or equal to 0.0: **a** =  $\langle value \rangle$ .  
On entry, **b** must not be less than or equal to 0.0: **b** =  $\langle value \rangle$ .

**NE\_INT\_ARG\_LE**

On entry, **n** must not be less than or equal to 0: **n** =  $\langle value \rangle$ .

## 6. Further Comments

To generate an observation from the  $\chi^2$  distribution with  $v$  degrees of freedom generate an observation from a gamma distribution with parameters  $a = v/2$ ,  $b = 2$ .

To generate an observation,  $y$ , from a Student's  $t$ -distribution with degrees of freedom  $v$  generate an observation,  $x$ , from a gamma distribution with parameters  $a = v/2$  and  $b = 2$  and an observation,  $z$ , from a standard Normal distribution (see nag\_random\_normal (g05ddc)) and use the transformation  $y = z/\sqrt{x}$ .

### 6.1. Accuracy

Not applicable.

### 6.2. References

Ahrens J H and Dieter U (1974) Computer Methods for Sampling from Gamma, Beta, Poisson and Binomial Distributions *Comput.* **12** 223–46.

Best D J (1978) Letter to the Editor *Appl. Statist.* **29** 181.

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

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

## 7. See Also

nag\_random\_init\_repeatable (g05cbc)  
nag\_random\_init\_nonrepeatable (g05ccc)  
nag\_random\_normal (g05ddc)

## 8. Example

The example program prints a set of five pseudo-random variates from a gamma distribution with parameters  $a = 5.0$  and  $b = 1.0$ , generated by nag\_random\_gamma after initialisation by nag\_random\_init\_repeatable (g05cbc).

### 8.1. Program Text

```
/* nag_random_gamma(g05ffc) Example Program
 *
 * Copyright 1991 Numerical Algorithms Group.
 *
 * Mark 2, 1991.
 */

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

#define N 5

main()
{
    Integer j;
    double a = 5.0;
    double b = 1.0;
    double x[N];

    Vprintf("g05ffc Example Program Results\n");
    g05cbc((Integer)0);
    Vprintf("Beta Dist --- a=%2.1f, b=%2.1f\n",a,b);
    g05ffc(a, b, (Integer)N, x, NAGERR_DEFAULT);
    for (j=0; j<(Integer)N; j++)
        Vprintf("%10.4f\n", x[j]);
    exit(EXIT_SUCCESS);
}
```

### 8.2. Program Data

None.

**8.3. Program Results**

```
g05ffc Example Program Results
Beta Dist --- a=5.0, b=1.0
  6.7603
  2.9943
  8.3800
  4.5740
  4.9672
```

---