NAG Fortran Library Routine Document

G05LPF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

G05LPF generates a vector of pseudo-random numbers from a von Mises distribution with concentration parameter κ.

2 Specification

```
SUBROUTINE G05LPF(VK, N, X, IGEN, ISEED, IFAIL)
INTEGER           N, IGEN, ISEED(4), IFAIL
real              VK, X(*)
```

3 Description

The von Mises distribution is a symmetric distribution used in the analysis of circular data. The probability density function of this distribution on the circle with mean direction $\mu_0 = 0$ and concentration parameter kappa, κ, can be written as:

$$f(\theta) = \frac{e^{\kappa \cos \theta}}{2\pi I_0(\kappa)},$$

where θ is reduced modulo 2π so that $-\pi \leq \theta < \pi$ and $\kappa \geq 0$. For very small κ the distribution is almost the uniform distribution, whereas for $\kappa \rightarrow \infty$ all the probability is concentrated at one point.

The n variates, $\theta_1, \theta_2, \ldots, \theta_n$, are generated using an envelope rejection method with a wrapped Cauchy target distribution as proposed by Best and Fisher (1979) and described by Dagpunar (1988).

One of the initialisation routines G05KBF (for a repeatable sequence if computed sequentially) or G05KCF (for a non-repeatable sequence) must be called prior to the first call to G05LPF.

4 References

Best D J and Fisher N I (1979) Efficient simulation of the von Mises distribution *Appl. Statist.* **28** 152–157

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

Mardia K V (1972) *Statistics of Directional Data* Academic Press

5 Parameters

1: VK – ***real*** *Input*

On entry: the concentration parameter, κ, of the required von Mises distribution.

Constraint: VK > 0.0.

2: N – INTEGER *Input*

On entry: the number, n, of pseudo-random numbers to be generated.

Constraint: N ≥ 0.

3: X(*) – ***real*** array *Output*

Note: the dimension of the array X must be at least $\max(1, \mathrm{N})$.

On exit: the n pseudo-random numbers from the specified von Mises distribution.

4: 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 routines G05KBF or G05KCF.

5: ISEED(4) – INTEGER array *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.

6: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, for users not familiar with this parameter the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, VK ≤ 0.0.

IFAIL $= 2$

On entry, N < 0.

7 Accuracy

Not applicable.

8 Further Comments

For a given number of random variates the generation time increases slightly with increasing κ.

If VK is supplied too large (i.e., VK $>$ SQRT(X02ALF())) then floating point overflow will occur in internal calculation.

9 Example

The example program prints the first five pseudo-random real numbers from a von Mises distribution with $\kappa = 1.0$, generated by a single call to G05LPF, after initialisation by G05KBF.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     G05LPF Example Program Text
*     Mark 20 Release. NAG Copyright 2001.
*     .. Parameters ..
      INTEGER          NOUT, M
      PARAMETER        (NOUT=6,M=5)
*     .. Local Scalars ..
      INTEGER          IFAIL, IGEN
*     .. Local Arrays ..
      real             X(M)
      INTEGER          ISEED(4)
*     .. External Subroutines ..
      EXTERNAL         G05KBF, G05LPF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G05LPF Example Program Results'
      WRITE (NOUT,*)
*     Initialise the seed to a repeatable sequence
      ISEED(1) = 1762543
      ISEED(2) = 9324783
      ISEED(3) = 42344
      ISEED(4) = 742355
*     IGEN identifies the stream.
      IGEN = 1
      CALL G05KBF(IGEN,ISEED)
*
      IFAIL = 0
      CALL G05LPF(1.0e0,M,X,IGEN,ISEED,IFAIL)
*
      WRITE (NOUT,99999) X
      STOP
*
99999 FORMAT (1X,F10.4)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 G05LPF Example Program Results

    -1.1339
    -2.5880
    -0.6178
     0.0519
    -0.9584
```