NAG Library Routine Document

G05PHF

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

G05PHF generates a realisation of a univariate time series from an autoregressive moving average (ARMA) model. The realisation may be continued or a new realisation generated at subsequent calls to G05PHF.

2 Specification

```
SUBROUTINE G05PHF (MODE, N, XMEAN, IP, PHI, IQ, THETA, AVAR, R, LR, STATE,      &
                   VAR, X, IFAIL)

INTEGER             MODE, N, IP, IQ, LR, STATE(*), IFAIL
REAL (KIND=nag_wp) XMEAN, PHI(IP), THETA(IQ), AVAR, R(LR), VAR, X(N)
```

3 Description

Let the vector x_t, denote a time series which is assumed to follow an autoregressive moving average (ARMA) model of the form:

$$x_t - \mu = \phi_1(x_{t-1} - \mu) + \phi_2(x_{t-2} - \mu) + \cdots + \phi_p(x_{t-p} - \mu) + \epsilon_t - \theta_1\epsilon_{t-1} - \theta_2\epsilon_{t-2} - \cdots - \theta_q\epsilon_{t-q}$$

where ϵ_t, is a residual series of independent random perturbations assumed to be Normally distributed with zero mean and variance σ^2. The parameters $\{\phi_i\}$, for $i = 1, 2, \ldots, p$, are called the autoregressive (AR) parameters, and $\{\theta_j\}$, for $j = 1, 2, \ldots, q$, the moving average (MA) parameters. The parameters in the model are thus the p ϕ values, the q θ values, the mean μ and the residual variance σ^2.

G05PHF sets up a reference vector containing initial values corresponding to a stationary position using the method described in Tunnicliffe–Wilson (1979). The routine can then return a realisation of $x_1, x_2, \ldots, x_n$. On a successful exit, the recent history is updated and saved in the reference vector R so that G05PHF may be called again to generate a realisation of $x_{n+1}, x_{n+2}, \ldots$, etc. See the description of the parameter MODE in Section 5 for details.

One of the initialization routines G05KFF (for a repeatable sequence if computed sequentially) or G05KGF (for a non-repeatable sequence) must be called prior to the first call to G05PHF.

4 References

Knuth D E (1981) *The Art of Computer Programming (Volume 2)* (2nd Edition) Addison–Wesley

Tunnicliffe–Wilson G (1979) Some efficient computational procedures for high order ARMA models *J. Statist. Comput. Simulation* **8** 301–309

5 Parameters

1: MODE – INTEGER *Input*

On entry: a code for selecting the operation to be performed by the routine.

MODE = 0
Set up reference vector only.

MODE = 1
Generate terms in the time series using reference vector set up in a prior call to G05PHF.

MODE = 2
Set up reference vector and generate terms in the time series.

Constraint: MODE = 0, 1 or 2.

2: N – INTEGER *Input*

On entry: n, the number of observations to be generated.

Constraint: N $\geq$ 0.

3: XMEAN – REAL (KIND=nag_wp) *Input*

On entry: the mean of the time series.

4: IP – INTEGER *Input*

On entry: p, the number of autoregressive coefficients supplied.

Constraint: IP $\geq$ 0.

5: PHI(IP) – REAL (KIND=nag_wp) array *Input*

On entry: the autoregressive coefficients of the model, $\phi_1, \phi_2, \ldots, \phi_p$.

6: IQ – INTEGER *Input*

On entry: q, the number of moving average coefficients supplied.

Constraint: IQ $\geq$ 0.

7: THETA(IQ) – REAL (KIND=nag_wp) array *Input*

On entry: the moving average coefficients of the model, $\theta_1, \theta_2, \ldots, \theta_q$.

8: AVAR – REAL (KIND=nag_wp) *Input*

On entry: σ^2, the variance of the Normal perturbations.

Constraint: AVAR $\geq$ 0.0.

9: R(LR) – REAL (KIND=nag_wp) array *Communication Array*

On entry: if MODE = 1, the reference vector from the previous call to G05PHF.

On exit: the reference vector.

10: LR – INTEGER *Input*

On entry: the dimension of the array R as declared in the (sub)program from which G05PHF is called.

Constraint: LR $\geq$ IP + IQ + 6 + max(IP, IQ + 1).

11: STATE(*) – INTEGER array *Communication Array*

Note: the actual argument supplied must be the array STATE supplied to the initialization routines G05KFF or G05KGF.

On entry: contains information on the selected base generator and its current state.

On exit: contains updated information on the state of the generator.

12: VAR – REAL (KIND=nag_wp) *Output*

On exit: the proportion of the variance of a term in the series that is due to the moving-average (error) terms in the model. The smaller this is, the nearer is the model to non-stationarity.

13: X(N) – REAL (KIND=nag_wp) array *Output*

On exit: contains the next n observations from the time series.

14: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

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, if you are 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.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

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, MODE $\neq 0$, 1 or 2.

IFAIL $= 2$

On entry, N < 0.

IFAIL $= 4$

On entry, IP < 0.

IFAIL $= 5$

PHI does not define a stationary autoregressive process.

IFAIL $= 6$

On entry, IQ < 0.

IFAIL $= 8$

On entry, AVAR < 0.0.

IFAIL $= 9$

Either R has been corrupted or the value of IP or IQ is not the same as when R was set up in a previous call to G05PHF with MODE $= 0$ or 2.

IFAIL $= 10$

On entry, LR $<$ IP $+$ IQ $+ 6 +$ max(IP, IQ $+ 1$).

IFAIL $= 11$

On entry, STATE vector was not initialized or has been corrupted.

7 Accuracy

Any errors in the reference vector's initial values should be very much smaller than the error term; see Tunnicliffe–Wilson (1979).

8 Further Comments

The time taken by G05PHF is essentially of order $(\mathrm{IP})^2$.

Note: The reference vector, R, contains a copy of the recent history of the series. If attempting to re-initialize the series by calling G05KFF or G05KGF a call to G05PHF with MODE $= 0$ must also be made. In the repeatable case the calls to G05PHF should be performed in the same order (at the same point(s) in simulation) every time G05KFF is used. When the generator state is saved and restored using the parameter STATE, the time series reference vector must be saved and restored as well.

The ARMA model for a time series can also be written as:

$$(x_n - E) = A_1(x_{n-1} - E) + \cdots + A_{NA}(x_{n-NA} - E) + B_1 a_n + \cdots + B_{NB} a_{n-NB+1}$$

where

x_n is the observed value of the time series at time n,

NA is the number of autoregressive parameters, A_i,

NB is the number of moving average parameters, B_i,

E is the mean of the time series,

and

a_t is a series of independent random Standard Normal perturbations.

This is the form used in G05PHF. This is related to the form given in Section 3 by:

$B_1^2 = \sigma^2$,

$B_{i+1} = -\theta_i \sigma = -\theta_i B_1, \qquad i = 1, 2, \ldots, q$,

$NB = q + 1$,

$E = \mu$,

$A_i = \phi_i, \qquad i = 1, 2, \ldots, p$,

$NA = p$.

9 Example

This example generates values for an autoregressive model given by

$$x_t = 0.4x_{t-1} + 0.2x_{t-2} + \epsilon_t$$

where ϵ_t is a series of independent random Normal perturbations with variance 1.0. The random number generators are initialized by G05KFF and then G05PHF is called to initialize a reference vector and generate a sample of ten observations.

9.1 Program Text

```
    Program g05phfe

!     G05PHF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: g05kff, g05phf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
```

```
!     .. Parameters ..
      Integer, Parameter               :: lseed = 1, nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: avar, var, xmean
      Integer                          :: genid, ifail, ip, iq, lr, lstate,    &
                                          mode, n, subid
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: phi(:), r(:), theta(:), x(:)
      Integer                          :: seed(lseed)
      Integer, Allocatable             :: state(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: max
!     .. Executable Statements ..
      Write (nout,*) 'G05PHF Example Program Results'
      Write (nout,*)

!     Skip heading in data file
      Read (nin,*)

!     Read in the base generator information and seed
      Read (nin,*) genid, subid, seed(1)

!     Initial call to initialiser to get size of STATE array
      lstate = 0
      Allocate (state(lstate))
      ifail = 0
      Call g05kff(genid,subid,seed,lseed,state,lstate,ifail)

!     Reallocate STATE
      Deallocate (state)
      Allocate (state(lstate))

!     Initialize the generator to a repeatable sequence
      ifail = 0
      Call g05kff(genid,subid,seed,lseed,state,lstate,ifail)

!     Read in sample size
      Read (nin,*) n

!     Read in number of coefficients
      Read (nin,*) ip, iq

      lr = ip + iq + 6 + max(ip,iq+1)
      Allocate (phi(ip),theta(iq),x(n),r(lr))

!     Read in mean
      Read (nin,*) xmean

!     Read in autoregressive coefficients
      If (ip>0) Then
        Read (nin,*) phi(1:ip)
      End If

!     Read in moving average coefficients
      If (iq>0) Then
        Read (nin,*) theta(1:iq)
      End If

!     Read in variance
      Read (nin,*) avar

!     Using a single call to G05PHF, so set up reference vector
!     and generate values in one go
      mode = 2

      ifail = 0
      Call g05phf(mode,n,xmean,ip,phi,iq,theta,avar,r,lr,state,var,x,ifail)
```

```
!     Display the variates
      Write (nout,99999) x(1:n)

99999 Format (1X,F12.4)
    End Program g05phfe
```

9.2 Program Data

```
G05PHF Example Program Data
1  1  1762543     :: GENID,SUBID,SEED(1)
10                :: N
2  0              :: IP,IQ
0.0               :: XMEAN
0.4  0.2          :: PHI
1.0               :: AVAR
```

9.3 Program Results

```
 G05PHF Example Program Results

      -1.7103
      -0.4042
      -0.1845
      -1.5004
      -1.1946
      -1.8184
      -1.0895
       1.6408
       1.3555
       1.1908
```