NAG Library Routine Document

G05PMF

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

G05PMF simulates from an exponential smoothing model, where the model uses either single exponential, double exponential or a Holt–Winters method.

2 Specification

```
SUBROUTINE G05PMF(MODE, N, ITYPE, P, PARAM, INIT, VAR, R, STATE, E, EN,
1                 X, IFAIL)
INTEGER            MODE, N, ITYPE, P, STATE(lstate), EN, IFAIL
double precision   PARAM(*), INIT(*), VAR, R(*), E(EN), X(N)
```

3 Description

G05PMF returns $\{x_t : t = 1, 2, \ldots, n\}$, a realization of a time series from an exponential smoothing model defined by one of five smoothing functions:

Single Exponential Smoothing

$$\begin{aligned} x_t &= m_{t-1} + \epsilon_t \\ m_t &= \alpha x_t + (1 - \alpha) m_{t-1} \end{aligned}$$

Brown Double Exponential Smoothing

$$\begin{aligned} x_t &= m_{t-1} + \tfrac{r_{t-1}}{\alpha} + \epsilon_t \\ m_t &= \alpha x_t + (1 - \alpha) m_{t-1} \\ r_t &= \alpha(m_t - m_{t-1}) + (1 - \alpha) r_{t-1} \end{aligned}$$

Linear Holt Exponential Smoothing

$$\begin{aligned} x_t &= m_{t-1} + \phi r_{t-1} + \epsilon_t \\ m_t &= \alpha x_t + (1 - \alpha)(m_{t-1} + \phi r_{t-1}) \\ r_t &= \gamma(m_t - m_{t-1}) + (1 - \gamma)\phi r_{t-1} \end{aligned}$$

Additive Holt–Winters Smoothing

$$\begin{aligned} x_t &= m_{t-1} + \phi r_{t-1} + s_{t-1-p} + \epsilon_t \\ m_t &= \alpha\big(x_t - s_{t-p}\big) + (1 - \alpha)(m_{t-1} + \phi r_{t-1}) \\ r_t &= \gamma(m_t - m_{t-1}) + (1 - \gamma)\phi r_{t-1} \\ s_t &= \beta(x_t - m_t) + (1 - \beta) s_{t-p} \end{aligned}$$

Multiplicative Holt–Winters Smoothing

$$\begin{aligned} x_t &= (m_{t-1} + \phi r_{t-1}) \times s_{t-1-p} + \epsilon_t \\ m_t &= \alpha x_t / s_{t-p} + (1 - \alpha)(m_{t-1} + \phi r_{t-1}) \\ r_t &= \gamma(m_t - m_{t-1}) + (1 - \gamma)\phi r_{t-1} \\ s_t &= \beta x_t / m_t + (1 - \beta) s_{t-p} \end{aligned}$$

where m_t is the mean, r_t is the trend and s_t is the seasonal component at time t with p being the seasonal order. The errors, ϵ_t are either drawn from a normal distribution with mean zero and variance σ^2 or randomly sampled, with replacement, from a user supplied vector.

4 References

Chatfield C (1980) *The Analysis of Time Series* Chapman and Hall

5 Parameters

1: MODE – INTEGER *Input*

On entry: indicates if G05PMF is continuing from a previous call or, if not, how the initial values are computed.

MODE $= 0$
Values for m_0, r_0 and s_{-j}, for $j = 0, \ldots, p-1$, are supplied in INIT.

MODE $= 1$
G05PMF continues from a previous call using values that are supplied in R. R is not updated.

MODE $= 2$
G05PMF continues from a previous call using values that are supplied in R. R is updated.

Constraint: MODE $= 0$, 1 or 2.

2: N – INTEGER *Input*

On entry: the number of terms of the time series being generated.

Constraint: N ≥ 0.

3: ITYPE – INTEGER *Input*

On entry: the smoothing function.

ITYPE $= 1$
Single exponential.

ITYPE $= 2$
Brown's double exponential.

ITYPE $= 3$
Linear Holt.

ITYPE $= 4$
Additive Holt–Winters.

ITYPE $= 5$
Multiplicative Holt–Winters.

Constraint: ITYPE $= 1$, 2, 3, 4 or 5.

4: P – INTEGER *Input*

On entry: if ITYPE $= 4$ or 5, the seasonal order, p, otherwise P is not referenced.

Constraint: if ITYPE $= 4$ or 5, P > 1.

5: PARAM($*$) – ***double precision*** array *Input*

Note: the dimension of the array PARAM must be at least 1 if ITYPE $= 1$ or 2, 3 if ITYPE $= 3$ and at least 4 if ITYPE $= 4$ or 5.

On entry: the smoothing parameters.

If ITYPE $= 1$ or 2, PARAM$(1) = \alpha$ and any remaining elements of PARAM are not referenced.

If ITYPE $= 3$, PARAM$(1) = \alpha$, PARAM$(2) = \gamma$, PARAM$(3) = \phi$ and any remaining elements of PARAM are not referenced.

If ITYPE $= 4$ or 5, PARAM(1) $= \alpha$, PARAM(2) $= \gamma$, PARAM(3) $= \beta$ and PARAM(4) $= \phi$ and any remaining elements of PARAM are not referenced.

Constraints:

if ITYPE $= 1$, $0.0 \leq \alpha \leq 1.0$;
if ITYPE $= 2$, $0.0 < \alpha \leq 1.0$;
if ITYPE $= 3$, $0.0 \leq \alpha \leq 1.0$ and $0.0 \leq \gamma \leq 1.0$ and $\phi \geq 0.0$;
if ITYPE $= 4$ or 5, $0.0 \leq \alpha \leq 1.0$ and $0.0 \leq \gamma \leq 1.0$ and $0.0 \leq \beta \leq 1.0$ and $\phi \geq 0.0$.

6: INIT($*$) – ***double precision*** array *Input*

Note: the dimension of the array INIT must be at least 1 if ITYPE $= 1$, 2 if ITYPE $= 2$ or 3 and at least $2 + \mathrm{P}$ if ITYPE $= 4$ or 5.

On entry: if MODE $= 0$, the initial values for m_0, r_0 and s_{-j}, for $j = 0, 1, \ldots, p-1$, used to initialize the smoothing.

If ITYPE $= 1$, INIT(1) $= m_0$ and any remaining elements of INIT are not referenced.

If ITYPE $= 2$ or 3, INIT(1) $= m_0$ and INIT(2) $= r_0$ and any remaining elements of INIT are not referenced.

If ITYPE $= 4$ or 5, INIT(1) $= m_0$, INIT(2) $= r_0$ and INIT(3) to INIT($2 + p$) hold the values for s_{-j}, for $j = 0, 1, \ldots, p-1$. Any remaining elements of INIT are not referenced.

7: VAR – ***double precision*** *Input*

On entry: the variance, σ^2 of the Normal distribution used to generate the errors ϵ_i. If VAR ≤ 0.0 then Normally distributed errors are not used.

8: R($*$) – ***double precision*** array *Input/Output*

Note: the dimension of the array R must be at least 13 if ITYPE $= 1$, 2 or 3 and at least $13 + \mathrm{P}$ if ITYPE $= 4$ or 5.

On entry: if MODE $= 1$ or 2, R must contain the values as returned by a previous call to G05PMF, R need not be set otherwise.

On exit: if MODE $= 1$, R is unchanged. Otherwise, R contains the information on the current state of smoothing.

Constraint: if MODE $= 1$ or 2, R must have been initialized by at least one call to G05PMF or G13AMF with MODE $\neq 1$, and R must not have been changed since that call.

9: STATE($lstate$) – INTEGER array *Communication Array*

Note: $lstate =$ LSTATE, the dimension of STATE as supplied to the initialization routine 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.

10: E(EN) – ***double precision*** array *Input*

On entry: if EN > 0 and VAR ≤ 0.0, a vector from which the errors, ϵ_t are randomly drawn, with replacement.

If EN ≤ 0, E is not referenced.

11: EN – INTEGER *Input*

On entry: if EN > 0, then the length of the vector E.

If both VAR ≤ 0.0 and EN ≤ 0 then $\epsilon_t = 0.0$, for $t = 1, 2, \ldots, n$.

12: X(N) – ***double precision*** array *Output*

On exit: the generated time series, x_t, for $t = 1, 2, \dots, n$.

13: 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.

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, 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.**

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, ITYPE $\neq 1$, 2, 3, 4 or 5.

IFAIL $= 3$

On entry, ITYPE $= 4$ or 5 and P < 2.

IFAIL $= 4$

On entry, at least one of α, β or $\gamma > 0.0$ or > 1.0.

On entry, ITYPE $= 2$ and $\alpha = 0.0$.

On entry, $\phi < 0.0$.

IFAIL $= 5$

On entry, N < 0.

IFAIL $= 8$

On entry, MODE $= 1$ or 2 and the array R has not been initialized correctly.

IFAIL $= 9$

On entry, the array STATE has not been initialized correctly.

IFAIL $= 12$

ITYPE $= 5$ and model is unsuitable for multiplicative Holt–Winter.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example reads 11 observations from a time series relating to the rate of the earth's rotation about its polar axis and fits an exponential smoothing model using G13AMF.

G05PMF is then called multiple times to obtain simulated forecast confidence intervals.

9.1 Program Text

```
*     G05PMF Example Program Text
*     Mark 22 Revised. NAG Copyright 2008.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, NFMAX, PMAX, MSTATE, MSEED, NSMAX
      PARAMETER        (NMAX=200,NFMAX=12,PMAX=5,MSTATE=633,MSEED=1,
     +                 NSMAX=100)
      DOUBLE PRECISION ONE, TWO
      PARAMETER        (ONE=1.0D0,TWO=2.0D0)
*     .. Local Scalars ..
      DOUBLE PRECISION AD, ALPHA, DV, TMP, VAR, Z
      INTEGER          EN, GENID, I, IFAIL, ITYPE, IVAL, J, K, LSEED,
     +                 LSTATE, MODE, N, NF, NSIM, P, SMODE, SUBID
*     .. Local Arrays ..
      DOUBLE PRECISION BLIM(2,NFMAX), BSIM(NSMAX,NFMAX), E(1),
     +                 FSE(NFMAX), FV(NFMAX), GLIM(2,NFMAX),
     +                 GSIM(NSMAX,NFMAX), INIT(PMAX+2), PARAM(4), Q(2),
     +                 R(PMAX+13), RES(NMAX), TSIM1(NFMAX),
     +                 TSIM2(NFMAX), Y(NMAX), YHAT(NMAX)
      INTEGER          SEED(MSEED), STATE(MSTATE)
*     .. External Functions ..
      DOUBLE PRECISION G01FAF
      EXTERNAL         G01FAF
*     .. External Subroutines ..
      EXTERNAL         G01AMF, G05KFF, G05PMF, G13AMF
*     .. Executable Statements ..
      CONTINUE
      WRITE (NOUT,*) 'G05PMF Example Program Results'
*     Initialize the seed to a repeatable sequence
      SEED(1) = 1762543
*     GENID and SUBID identify the base generator.
      GENID = 1
      SUBID = 1
*     Initialize the generator to a repeatable sequence
      LSTATE = MSTATE
      LSEED = MSEED
      IFAIL = 1
      CALL G05KFF(GENID,SUBID,SEED,LSEED,STATE,LSTATE,IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99995) IFAIL
         GO TO 140
      END IF
*     Skip headings in data file
      READ (NIN,*)
*     Read in the initial arguments and check array sizes
      READ (NIN,*) MODE, ITYPE, N, NF, NSIM, ALPHA
      IF (N.GT.NMAX .OR. NF.GT.NFMAX .OR. NSIM.GT.NSMAX) THEN
         WRITE (NOUT,99991)
         GO TO 140
      END IF
      READ (NIN,*) (Y(I),I=1,N)
*     Read in the ITYPE dependent arguments (skipping headings)
      IF (ITYPE.EQ.1) THEN
         READ (NIN,*) PARAM(1)
         P = 0
```

```
            IVAL = 1
         ELSE IF (ITYPE.EQ.2) THEN
            READ (NIN,*) PARAM(1), PARAM(2)
            P = 0
            IVAL = 2
         ELSE IF (ITYPE.EQ.3) THEN
            READ (NIN,*) PARAM(1), PARAM(2), PARAM(3)
            P = 0
            IVAL = 2
         ELSE
            READ (NIN,*) PARAM(1), PARAM(2), PARAM(3), PARAM(4), P
            IVAL = P + 2
            IF (P.GT.PMAX) THEN
               WRITE (NOUT,99991)
               GO TO 140
            END IF
         END IF
*     Read in the MODE dependent arguments (skipping headings)
         IF (MODE.EQ.0) THEN
*        User supplied initial values
            READ (NIN,*) (INIT(I),I=1,IVAL)
         ELSE IF (MODE.EQ.1) THEN
*        Continuing from a previously saved R
            READ (NIN,*) (R(I),I=1,P+13)
         ELSE IF (MODE.EQ.2) THEN
*        Initial values calculated from first K observations
            READ (NIN,*) K
         END IF
*
*     Fit a smoothing model (parameter R in G05PMF and STATE in G13AMF
*     are in the same format)
         IFAIL = 1
         CALL G13AMF(MODE,ITYPE,P,PARAM,N,Y,K,INIT,NF,FV,FSE,YHAT,RES,DV,
        +            AD,R,IFAIL)
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99994) IFAIL
            GO TO 140
         END IF
*
*     Simulate forecast values from the model, and don't update R
         SMODE = 2
         VAR = DV*DV
         EN = N
*     Simulate NSIM forecasts
         DO 40 I = 1, NSIM
*        Simulations assuming gaussian errors
            IFAIL = 1
            CALL G05PMF(SMODE,NF,ITYPE,P,PARAM,INIT,VAR,R,STATE,E,0,TSIM1,
        +               IFAIL)
            IF (IFAIL.NE.0) THEN
               WRITE (NOUT,99993) IFAIL
               GO TO 140
            END IF
*        Bootstrapping errors
            IFAIL = 1
            CALL G05PMF(SMODE,NF,ITYPE,P,PARAM,INIT,0.0D0,R,STATE,RES,EN,
        +               TSIM2,IFAIL)
            IF (IFAIL.NE.0) THEN
               WRITE (NOUT,99993) IFAIL
               GO TO 140
            END IF
*        Copy and transpose the simulated values
            DO 20 J = 1, NF
               GSIM(I,J) = TSIM1(J)
               BSIM(I,J) = TSIM2(J)
   20       CONTINUE
   40    CONTINUE
*
*     Calculate CI based on the quantiles for each simulated forecast
         Q(1) = ALPHA/TWO
         Q(2) = ONE - Q(1)
```

```
      DO 60 I = 1, NF
         IFAIL = 1
         CALL G01AMF(NSIM,GSIM(1,I),2,Q,GLIM(1,I),IFAIL)
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99992) IFAIL
            GO TO 140
         END IF
         IFAIL = 1
         CALL G01AMF(NSIM,BSIM(1,I),2,Q,BLIM(1,I),IFAIL)
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99992) IFAIL
            GO TO 140
         END IF
   60 CONTINUE
*
*     Display the forecast values and associated prediction intervals
      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Initial values used:'
      DO 80 I = 1, IVAL
         WRITE (NOUT,99998) I, INIT(I)
   80 CONTINUE
      WRITE (NOUT,*)
      WRITE (NOUT,99999) 'Mean Deviation     = ', DV
      WRITE (NOUT,99999) 'Absolute Deviation = ', AD
      WRITE (NOUT,*)
      WRITE (NOUT,*) '         Observed      1-Step'
      WRITE (NOUT,*) ' Period   Values      Forecast      Residual'
      WRITE (NOUT,*)
      DO 100 I = 1, N
         WRITE (NOUT,99998) I, Y(I), YHAT(I), RES(I)
  100 CONTINUE
      WRITE (NOUT,*)
      WRITE (NOUT,*) '                                            '//
     +  '      Simulated CI         Simulated CI'
      WRITE (NOUT,*) 'Obs.  Forecast      Estimated CI   '//
     +  '     (Gaussian Errors)    (Bootstrap Errors)'
      Z = G01FAF('L',Q(2),IFAIL)
      DO 120 I = 1, NF
         TMP = Z*FSE(I)
         WRITE (NOUT,99997) N + I, FV(I), FV(I) - TMP, FV(I) + TMP,
     +     GLIM(1,I), GLIM(2,I), BLIM(1,I), BLIM(2,I)
  120 CONTINUE
      WRITE (NOUT,99996) 100.0D0*(ONE-ALPHA), '% CIs were produced'
*
  140 CONTINUE
*
99999 FORMAT (A,E12.4)
99998 FORMAT (I4,3(1X,F12.3))
99997 FORMAT (I3,7(1X,F10.3))
99996 FORMAT (1X,F5.1,A)
99995 FORMAT (1X,' ** G05KFF returned with IFAIL = ',I5)
99994 FORMAT (1X,' ** G13AMF returned with IFAIL = ',I5)
99993 FORMAT (1X,' ** G05PMF returned with IFAIL = ',I5)
99992 FORMAT (1X,' ** G01AMF returned with IFAIL = ',I5)
99991 FORMAT (1X,' ** Problem size too large, increase array limits')
      END
```

9.2 Program Data

```
G05PMF Example Program Data
2 3 11 5 100 0.05                               : MODE,ITYPE,N,NF,NSIM,ALPHA
180 135 213 181 148 204 228 225 198 200 187 : Y
0.01 1.0 1.0                                    : PARAM(1:3) (since ITYPE=3)
11                                              : K (since MODE=2)
```

9.3 Program Results

```
 G05PMF Example Program Results

 Initial values used:
   1      168.018
   2        3.800

Mean Deviation     =   0.2547E+02
Absolute Deviation =   0.2123E+02

         Observed      1-Step
  Period   Values      Forecast      Residual

   1      180.000      171.818        8.182
   2      135.000      175.782      -40.782
   3      213.000      178.848       34.152
   4      181.000      183.005       -2.005
   5      148.000      186.780      -38.780
   6      204.000      189.800       14.200
   7      228.000      193.492       34.508
   8      225.000      197.732       27.268
   9      198.000      202.172       -4.172
  10      200.000      206.256       -6.256
  11      187.000      210.256      -23.256

                                                 Simulated CI         Simulated CI
 Obs.  Forecast      Estimated CI          (Gaussian Errors)     (Bootstrap Errors)
 12    213.854    163.928    263.781    161.431    258.001    173.073    248.363
 13    217.685    167.748    267.622    172.660    262.100    177.311    252.638
 14    221.516    171.556    271.475    169.259    263.107    179.344    256.921
 15    225.346    175.347    275.345    180.721    272.776    183.672    260.804
 16    229.177    179.115    279.238    184.790    263.591    186.398    264.173
  95.0% CIs were produced
```