NAG Library Routine Document

G13FCF

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

G13FCF estimates the parameters of a univariate regression-type II AGARCH(p, q) process.

2 Specification

```
 SUBROUTINE G13FCF(DIST, YT, X, LDX, NUM, IP, IQ, NREG, MN, NPAR, THETA,
1                  SE, SC, COVR, LDCOVR, HP, ET, HT, LGF, COPTS, MAXIT,
2                  TOL, WORK, LWORK, IFAIL)
 INTEGER           LDX, NUM, IP, IQ, NREG, MN, NPAR, LDCOVR, MAXIT,
1                  LWORK, IFAIL
 double precision  YT(NUM), X(LDX,*), THETA(NPAR), SE(NPAR), SC(NPAR),
1                  COVR(LDCOVR,NPAR), HP, ET(NUM), HT(NUM), LGF, TOL,
2                  WORK(LWORK)
 LOGICAL           COPTS(2)
 CHARACTER*1       DIST
```

3 Description

A univariate regression-type II AGARCH(p, q) process, with q coefficients α_i, for $i = 1, 2, \ldots, q$, p coefficients, β_i, for $i = 1, 2, \ldots, p$, and k linear regression coefficients b_i, for $i = 1, 2, \ldots, k$, can be represented by:

$$y_t = b_o + x_t^{\mathrm{T}} b + \epsilon_t \tag{1}$$

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \left(|\epsilon_{t-i}| + \gamma \epsilon_{t-i} \right)^2 + \sum_{i=1}^{p} \beta_i h_{t-i}, \qquad t = 1, 2, \ldots, T. \tag{2}$$

where $\epsilon_t \mid \psi_{t-1} = N(0, h_t)$ or $\epsilon_t \mid \psi_{t-1} = S_t(df, h_t)$. Here S_t is a standardized Student's t-distribution with df degrees of freedom and variance h_t, T is the number of terms in the sequence, y_t denotes the endogenous variables, x_t the exogenous variables, b_o the regression mean, b the regression coefficients, ϵ_t the residuals, h_t the conditional variance, and ψ_t the set of all information up to time t.

G13FCF provides an estimate for the parameter vector $\theta = \left(b_o, b^{\mathrm{T}}, \omega^{\mathrm{T}}\right)$ where $b^{\mathrm{T}} = (b_1, \ldots, b_k)$, $\omega^{\mathrm{T}} = \left(\alpha_0, \alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_p, \gamma\right)$ when DIST = 'N' and $\omega^{\mathrm{T}} = \left(\alpha_0, \alpha_1, \ldots, \alpha_q, \beta_1, \ldots, \beta_p, \gamma, df\right)$ when DIST = 'T'.

MN and NREG can be used to simplify the GARCH(p, q) expression in (1) as follows:

No Regression and No Mean

$y_t = \epsilon_t$,

MN $= 0$,

NREG $= 0$ and

θ is a $(p + q + 2)$ vector when DIST = 'N' and a $(p + q + 3)$ vector when DIST = 'T'.

No Regression

$y_t = b_o + \epsilon_t$,

MN $= 1$,

NREG $= 0$ and

θ is a $(p+q+3)$ vector when DIST = 'N' and a $(p+q+4)\times 1$ vector when DIST = 'T'.

Note: if the $y_t = \mu + \epsilon_t$, where μ is known (not to be estimated by G13FCF) then (1) can be written as ${y_t}^\mu = \epsilon_t$, where ${y_t}^\mu = y_t - \mu$. This corresponds to the case **No Regression and No Mean**, with y_t replaced by $y_t - \mu$.

No Mean

$y_t = x_t^{\mathrm{T}} b + \epsilon_t$,

MN $= 0$,

NREG $= k$ and

θ is a $(p+q+k+2)$ vector when DIST = 'N' and a $(p+q+k+3)$ vector when DIST = 'T'.

4 References

Bollerslev T (1986) Generalised autoregressive conditional heteroskedasticity *Journal of Econometrics* **31** 307–327

Engle R (1982) Autoregressive conditional heteroskedasticity with estimates of the variance of United Kingdom inflation *Econometrica* **50** 987–1008

Engle R and Ng V (1993) Measuring and Testing the Impact of News on Volatility *Journal of Finance* **48** 1749–1777

Hamilton J (1994) *Time Series Analysis* Princeton University Press

5 Parameters

1: DIST – CHARACTER*1 *Input*

On entry: the type of distribution to use for e_t.

DIST = 'N'
A Normal distribution is used.

DIST = 'T'
A Student's t-distribution is used.

2: YT(NUM) – ***double precision*** array *Input*

On entry: the sequence of observations, y_t, for $t = 1, 2, \ldots, T$.

3: X(LDX,$*$) – ***double precision*** array *Input*

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

On entry: row t of X must contain the time dependent exogenous vector x_t, where $x_t^{\mathrm{T}} = \left({x_t}^1, \ldots, {x_t}^k\right)$, for $t = 1, 2, \ldots, T$.

4: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G13FCF is called.

Constraint: LDX $\geq$ NUM.

5: NUM – INTEGER *Input*

On entry: T, the number of terms in the sequence.

Constraints:

NUM $\geq \max(\text{IP}, \text{IQ})$;
NUM $\geq$ NREG + MN.

6: IP – INTEGER *Input*

On entry: the number of coefficients, β_i, for $i = 1, 2, \ldots, p$.

Constraint: IP ≥ 0 (see also NPAR).

7: IQ – INTEGER *Input*

On entry: the number of coefficients, α_i, for $i = 1, 2, \ldots, q$.

Constraint: IQ ≥ 1 (see also NPAR).

8: NREG – INTEGER *Input*

On entry: k, the number of regression coefficients.

Constraint: NREG ≥ 0 (see also NPAR).

9: MN – INTEGER *Input*

On entry: if MN = 1, the mean term b_0 will be included in the model.

Constraint: MN = 0 or 1.

10: NPAR – INTEGER *Input*

On entry: the number of parameters to be included in the model. NPAR = 2 + IQ + IP + MN + NREG when DIST = 'N' and NPAR = 3 + IQ + IP + MN + NREG when DIST = 'T'.

Constraint: NPAR < 20.

11: THETA(NPAR) – ***double precision*** array *Input/Output*

On entry: the initial parameter estimates for the vector θ.

The first element must contain the coefficient α_o and the next IQ elements must contain the coefficients α_i, for $i = 1, 2, \ldots, q$.

The next IP elements must contain the coefficients β_j, for $j = 1, 2, \ldots, p$.

The next element must contain the asymmetry parameter γ.

If DIST = 'T', the next element must contain df, the number of degrees of freedom of the Student's t-distribution.

If MN = 1, the next element contains the mean term b_o.

If COPTS(2) = .FALSE., the remaining NREG elements are taken as initial estimates of the linear regression coefficients b_i, for $i = 1, 2, \ldots, k$.

On exit: the estimated values $\hat{\theta}$ for the vector θ.

The first element contains the coefficient α_o, the next IQ elements contain the coefficients α_i, for $i = 1, 2, \ldots, q$.

The next IP elements are the coefficients β_j, for $j = 1, 2, \ldots, p$.

The next element contains the estimate for the asymmetry parameter γ.

If DIST = 'T', the next element contains an estimate for df, the number of degrees of freedom of the Student's t-distribution.

If MN = 1, the next element contains an estimate for the mean term b_o.

The final NREG elements are the estimated linear regression coefficients b_i, for $i = 1, 2, \ldots, k$.

12: SE(NPAR) – ***double precision*** array *Output*

On exit: the standard errors for $\hat{\theta}$.

The first element contains the standard error for α_o and the next IQ elements contain the standard errors for α_i, for $i = 1, 2, \ldots, q$.

The next IP elements are the standard errors for β_j, for $j = 1, 2, \ldots, p$.

The next element contains the standard error for γ.

If DIST = 'T', the next element contains the standard error for df, the number of degrees of freedom of the Student's t-distribution.

If MN = 1, the next element contains the standard error for b_o.

The final NREG elements are the standard errors for b_j, for $j = 1, 2, \ldots, k$.

13: SC(NPAR) – ***double precision*** array *Output*

On exit: the scores for $\hat{\theta}$.

The first element contains the score for α_o and the next IQ elements contain the score for α_i, for $i = 1, 2, \ldots, q$.

The next IP elements are the scores for β_j, for $j = 1, 2, \ldots, p$.

The next element contains the score for γ.

If DIST = 'T', the next element contains the score for df, the number of degrees of freedom of the Student's t-distribution.

If MN = 1, the next element contains the score for b_o.

The final NREG elements are the scores for b_j, for $j = 1, 2, \ldots, k$.

14: COVR(LDCOVR,NPAR) – ***double precision*** array *Output*

On exit: the covariance matrix of the parameter estimates $\hat{\theta}$, that is the inverse of the Fisher Information Matrix.

15: LDCOVR – INTEGER *Input*

On entry: the first dimension of the array COVR as declared in the (sub)program from which G13FCF is called.

Constraint: LDCOVR $\geq$ NPAR.

16: HP – ***double precision*** *Input/Output*

On entry: if COPTS(2) = .FALSE., HP is the value to be used for the pre-observed conditional variance; otherwise HP is not referenced.

On exit: if COPTS(2) = .TRUE., HP is the estimated value of the pre-observed of the conditional variance.

17: ET(NUM) – ***double precision*** array *Output*

On exit: the estimated residuals, ϵ_t, for $t = 1, 2, \ldots, T$.

18: HT(NUM) – ***double precision*** array *Output*

On exit: the estimated conditional variances, h_t, for $t = 1, 2, \ldots, T$.

19: LGF – ***double precision*** *Output*

On exit: the value of the log-likelihood function at $\hat{\theta}$.

20: COPTS(2) – LOGICAL array *Input*

On entry: the options to be used by G13FCF.

COPTS(1) = .TRUE.
Stationary conditions are enforced, otherwise they are not.

COPTS(2) = .TRUE.
The routine provides initial parameter estimates of the regression terms, otherwise these are to be provided by you.

21: MAXIT – INTEGER *Input*

On entry: the maximum number of iterations to be used by the optimization routine when estimating the GARCH(p, q) parameters. If MAXIT is set to 0, the standard errors, score vector and variance-covariance are calculated for the input value of θ in THETA; however the value of θ is not updated.

Constraint: MAXIT ≥ 0.

22: TOL – ***double precision*** *Input*

On entry: the tolerance to be used by the optimization routine when estimating the GARCH(p, q) parameters.

23: WORK(LWORK) – ***double precision*** array *Workspace*
24: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which G13FCF is called.

Constraint: LWORK $\geq$ (NREG + 3) $\times$ NUM + NPAR + 403.

25: 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, because for this routine the values of the output parameters may be useful even if IFAIL $\neq 0$ on exit, the recommended value is -1. **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).

Note: G13FCF may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the routine:

IFAIL = 1

On entry, NREG < 0,
or MN > 1,
or MN < 0,
or IQ < 1,
or IP < 0,
or NPAR ≥ 20,
or LDCOVR $<$ NPAR,
or LDX $<$ NUM,

or DIST $\neq$ 'N', and DIST $\neq$ 'T',
or MAXIT < 0,
or NUM $<$ NREG + MN,
or NPAR has an invalid value NUM $< \max(\mathrm{IP}, \mathrm{IQ})$.

IFAIL $= 2$

On entry, LWORK $< (\mathrm{NREG} + 3) \times \mathrm{NUM} + 3$.

IFAIL $= 3$

The matrix X is not full rank.

IFAIL $= 4$

The information matrix is not positive-definite.

IFAIL $= 5$

The maximum number of iterations has been reached.

IFAIL $= 6$

The log-likelihood cannot be optimized any further.

IFAIL $= 7$

No feasible model parameters could be found.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example uses G05PEF to generate 1500 data points, with known process parameters θ for the following two time series:

(i) A GARCH$(1,1)$ sequence with Normally distributed residuals.

(ii) A GARCH$(1,1)$ sequence with Student's t-distributed residuals.

Here G05PEF is initially called, with the output discarded, to eliminate 'start-up effects' in these sequences. The process parameter estimates, $\hat{\theta}$, are then obtained using G13FCF, and compared with their true values, θ. Finally a four step ahead volatility estimate is computed using G13FDF.

9.1 Program Text

```
*     G13FCF Example Program Text
*     Mark 22 Revised. NAG Copyright 2006.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
      INTEGER          NPARMX, LDCOVR, NUM, MSEED, MSTATE, NUM1, NREGMX,
     +                 LRVEC
      PARAMETER        (NPARMX=10,LDCOVR=NPARMX,NUM=1500,MSEED=1,
     +                 MSTATE=200,NUM1=3000,NREGMX=10,LRVEC=40)
      DOUBLE PRECISION ZERO
      PARAMETER        (ZERO=0.0D0)
*     .. Local Scalars ..
      DOUBLE PRECISION FAC1, GAMMA, HP, LGF, MEAN, TOL, XTERM
```

```
      INTEGER          D, DF, GENID, I, IFAIL, IP, IQ, J, K, LDX, LSEED,
     +                 LSTATE, LWORK, MAXIT, MN, NPAR, NPAR2, NREG, NT,
     +                 SUBID
      LOGICAL          FCALL
      CHARACTER        DIST
*     .. Local Arrays ..
      DOUBLE PRECISION BX(10), COVR(LDCOVR,NPARMX), CVAR(100),
     +                 ETM(NUM1), HT(NUM1+10), HTM(NUM1), PARAM(NPARMX),
     +                 RVEC(LRVEC), SC(NPARMX), SE(NPARMX),
     +                 THETA(NPARMX), WORK(NUM1*3+NPARMX+NREGMX*NUM1+20*
     +                 20+1), X(NUM1,10), YT(NUM1+10)
      INTEGER          SEED(MSEED), STATE(MSTATE)
      LOGICAL          COPTS(2)
*     .. External Subroutines ..
      EXTERNAL         G05KFF, G05PEF, G13FCF, G13FDF
*     .. Intrinsic Functions ..
      INTRINSIC        DBLE, SIN
*     .. Executable Statements ..
      IFAIL = 0
      WRITE (NOUT,*) 'G13FCF Example Program Results'
      WRITE (NOUT,*)
*     Initialize the seed
      SEED(1) = 111
*     GENID and SUBID identify the base generator
      GENID = 1
      SUBID = 1
*     Initialize the generator to a repeatable sequence
      LSTATE = MSTATE
      LSEED = MSEED
      CALL G05KFF(GENID,SUBID,SEED,LSEED,STATE,LSTATE,IFAIL)
*
*     Initialize the time dependent, exogenous, variables
      LDX = NUM1
      BX(1) = 1.5D0
      BX(2) = 2.5D0
      BX(3) = 3.0D0
      DO 20 I = 1, NUM
         FAC1 = DBLE(I)*0.01D0
         X(I,1) = 0.01D0 + 0.7D0*SIN(FAC1)
         X(I,2) = 0.5D0 + FAC1*0.1D0
         X(I,3) = 1.0D0
   20 CONTINUE
*
*     Show two examples, normally distributed errors,
*     and student t errors
      DO 180 D = 1, 2
         IF (D.EQ.1) THEN
*           Errors distributed as a normal distribution
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Normal distribution'
            WRITE (NOUT,*)
            DIST = 'N'
*           Initialize the series parameters required
*           for generation of some test data
            MEAN = 3.0D0
            MN = 1
            NREG = 2
            GAMMA = -0.4D0
            IP = 1
            IQ = 1
            PARAM(1) = 0.08D0
            PARAM(2) = 0.2D0
            PARAM(3) = 0.7D0
         ELSE
*           Errors distributed as a students t distribution
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Students T distribution'
            WRITE (NOUT,*)
            DIST = 'T'
*           Initialize the series parameters required
*           for generation of some test data
```

```
            DF = 4
            MEAN = 3.0D0
            MN = 1
            NREG = 2
            GAMMA = -0.4D0
            IP = 1
            IQ = 1
            PARAM(1) = 0.1D0
            PARAM(2) = 0.1D0
            PARAM(3) = 0.8D0
         END IF
*
*        Start of data generation
*        Generate the errors
         FCALL = .TRUE.
         CALL G05PEF(DIST,NUM,IP,IQ,PARAM,GAMMA,DF,HT,YT,FCALL,RVEC,
     +               LRVEC,STATE,IFAIL)
*        Calculate the observed data
         DO 60 I = 1, NUM
            XTERM = ZERO
            DO 40 K = 1, NREG
               XTERM = XTERM + X(I,K)*BX(K)
   40       CONTINUE
            IF (MN.EQ.1) THEN
               YT(I) = MEAN + XTERM + YT(I)
            ELSE
               YT(I) = XTERM + YT(I)
            END IF
   60    CONTINUE
*        End of data generation
*
*        Calculate the number of parameters
         NPAR = IQ + IP + 1
         NPAR2 = NPAR + MN + NREG + 1
         IF (DIST.EQ.'T') NPAR2 = NPAR2 + 1
         LWORK = NREG*NUM + 3*NUM + NPAR2 + 403
*
*        Set the model fitting options
         COPTS(1) = .TRUE.
         COPTS(2) = .TRUE.
         MAXIT = 50
         TOL = 1.0D-5
*
*        Generate initial values for the alpha_i and beta_j parameters
         DO 80 I = 1, NPAR
            THETA(I) = PARAM(I)*0.5D0
   80    CONTINUE
*        Generate initial values for the asymmetry parameter, gamma
         THETA(NPAR+1) = GAMMA*0.5D0
*        Generate initial values for DF estimate
         J = NPAR + 2
         IF (DIST.EQ.'T') THEN
            THETA(J) = DF*0.65D0
            J = J + 1
         END IF
*        Generate the initial values for b_0
         IF (MN.EQ.1) THEN
            THETA(J) = MEAN*0.5D0
            J = J + 1
         END IF
*        Generate the initial values for the linear regression
*        coefficients b_i, if required
         IF ( .NOT. COPTS(2)) THEN
            DO 100 I = 1, NREG
               THETA(I+J-1) = BX(I)*0.5D0
  100       CONTINUE
         END IF
*
*        Call the analysis routine
         CALL G13FCF(DIST,YT,X,LDX,NUM,IP,IQ,NREG,MN,NPAR2,THETA,SE,SC,
     +               COVR,LDCOVR,HP,ETM,HTM,LGF,COPTS,MAXIT,TOL,WORK,
```

```
     +              LWORK,IFAIL)
*
*        Output the results
         WRITE (NOUT,*) '         Parameter       Standard',
     +     '        Correct'
         WRITE (NOUT,*) '         estimates       errors  ',
     +     '        values'
*        Output the coefficient alpha_0
         WRITE (NOUT,99999) THETA(1), SE(1), PARAM(1)
*        Output the coefficients alpha_i
         DO 120 I = 2, IQ
            WRITE (NOUT,99999) THETA(I), SE(I), PARAM(I)
  120    CONTINUE
*        Output the coefficients beta_j
         DO 140 I = IQ + 1, NPAR
            WRITE (NOUT,99999) THETA(I), SE(I), PARAM(I)
  140    CONTINUE
*        Output the estimated asymmetry parameter, gamma
         WRITE (NOUT,99999) THETA(NPAR+1), SE(NPAR+1), GAMMA
         J = NPAR + 2
*        Output the estimated degrees of freedom, df
         IF (DIST.EQ.'T') THEN
            WRITE (NOUT,99999) THETA(J), SE(J), DBLE(DF)
            J = J + 1
         END IF
*        Output the estimated mean term, b_0
         IF (MN.EQ.1) THEN
            WRITE (NOUT,99999) THETA(J), SE(J), MEAN
            J = J + 1
         END IF
*        Output the estimated linear regression coefficients, b_i
         DO 160 I = 1, NREG
            WRITE (NOUT,99999) THETA(I+J-1), SE(I+J-1), BX(I)
  160    CONTINUE
*
*        Calculate the volatility forecast
         NT = 4
         CALL G13FDF(NUM,NT,IP,IQ,THETA,GAMMA,CVAR,HTM,ETM,IFAIL)
*        Display the volatility forecast
         WRITE (NOUT,*)
         WRITE (NOUT,99998) 'Volatility forecast = ', CVAR(NT)
         WRITE (NOUT,*)
  180 CONTINUE
*
99999 FORMAT (1X,3F16.4)
99998 FORMAT (1X,A,F12.4)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 G13FCF Example Program Results


 Normal distribution

           Parameter       Standard        Correct
           estimates       errors          values
           0.0921          0.0147          0.0800
           0.2283          0.0324          0.2000
           0.6576          0.0326          0.7000
          -0.4175          0.0682         -0.4000
           2.9803          0.0568          3.0000
           1.5345          0.0362          1.5000
           2.5316          0.0426          2.5000

 Volatility forecast =       0.4676
```

```
Students T distribution

          Parameter       Standard        Correct
          estimates       errors          values
          3.6941          0.4209          0.1000
          0.4531          0.3415          0.1000
          0.5459          0.7323          0.8000
         -1.0000          2.2991         -0.4000
          2.0702          0.1390          4.0000
          2.9509          0.0725          3.0000
          1.4607          0.0493          1.5000
          2.5211          0.0593          2.5000

Volatility forecast =      19.2438
```