NAG Library Routine Document

G02GPF

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

G02GPF allows prediction from a generalized linear model fit via G02GAF, G02GBF, G02GCF or G02GDF.

2 Specification

```
 SUBROUTINE G02GPF(ERRFN, LINK, MEAN, OFFSET, WEIGHT, N, X, LDX, M, ISX,
1                  IP, T, OFF, WT, S, A, B, COV, VFOBS, ETA, SEETA, PRED,
2                  SEPRED, IFAIL)
 INTEGER           N, LDX, M, ISX(M), IP, IFAIL
 double precision  X(LDX,M), T(*), OFF(*), WT(*), S, A, B(IP),
1                  COV(IP*(IP+1)/2), ETA(N), SEETA(N), PRED(N), SEPRED(N)
 LOGICAL           VFOBS
 CHARACTER*1       ERRFN, LINK, MEAN, OFFSET, WEIGHT
```

3 Description

A generalized linear model consists of the following elements:

(i) A suitable distribution for the dependent variable y.

(ii) A linear model, with linear predictor $\eta = X\beta$, where X is a matrix of independent variables and β a column vector of p parameters.

(iii) A link function $g(.)$ between the expected value of y and the linear predictor, that is $E(y) = \mu = g(\eta)$.

In order to predict from a generalized linear model, that is estimate a value for the dependent variable, y, given a set of independent variables X, the matrix X must be supplied, along with values for the parameters β and their associated variance-covariance matrix, C. Suitable values for β and C are usually estimated by first fitting the prediction model to a training dataset with known responses, using for example G02GAF, G02GBF, G02GCF or G02GDF. The predicted variable, and its standard error can then be obtained from:

$$\hat{y} = g^{-1}(\eta), \qquad \text{se}(\hat{y}) = \sqrt{\left(\frac{\delta g^{-1}(x)}{\delta x}\right)_{\eta}}\,\text{se}(\eta) + I_{\text{fobs}}\,\text{Var}(y)$$

where

$$\eta = o + X\beta, \qquad \text{se}(\eta) = \text{diag}\,\sqrt{XCX^{\text{T}}},$$

o is a vector of offsets and $I_{\text{fobs}} = 0$, if the variance of future observations is not taken into account, and 1 otherwise. Here diag A indicates the diagonal elements of matrix A.

If required, the variance for the ith future observation, $\text{Var}(y_i)$, can be calculated as:

$$\text{Var}(y_i) = \frac{\phi V(\theta)}{w_i}$$

where w_i is a weight, ϕ is the scale (or dispersion) parameter, and $V(\theta)$ is the variance function. Both the scale parameter and the variance function depend on the distribution used for the y, with:

Poisson $\quad V(\theta) = \mu_i,\ \phi = 1$

binomial $\quad V(\theta) = \frac{\mu_i(t_i - \mu_i)}{t_i},\ \phi = 1$

Normal $V(\theta) = 1$

gamma $V(\theta) = \mu_i^2$

In the cases of a Normal and gamma error structure, the scale parameter (ϕ), is supplied by you. This value is usually obtained from the routine used to fit the prediction model. In many cases, for a Normal error structure, $\phi = \hat{\sigma}^2$, i.e., the estimated variance.

4 References

McCullagh P and Nelder J A (1983) *Generalized Linear Models* Chapman and Hall

5 Parameters

1: ERRFN – CHARACTER*1 *Input*

On entry: indicates the distribution used to model the dependent variable, y.

ERRFN = 'B'
The binomial distribution is used.

ERRFN = 'G'
The gamma distribution is used.

ERRFN = 'N'
The Normal (Gaussian) distribution is used.

ERRFN = 'P'
The Poisson distribution is used.

Constraint: ERRFN = 'B', 'G', 'N' or 'P'.

2: LINK – CHARACTER*1 *Input*

On entry: indicates which link function to be used.

LINK = 'C'
A complementary log-log link is used.

LINK = 'E'
An exponent link is used.

LINK = 'G'
A logistic link is used.

LINK = 'I'
An identity link is used.

LINK = 'L'
A log link is used.

LINK = 'P'
A probit link is used.

LINK = 'R'
A reciprocal link is used.

LINK = 'S'
A square root link is used.

Details on the functional form of the different links can be found in the G02 Chapter Introduction.

Constraints:

if ERRFN = 'B', LINK = 'C', 'G' or 'P';
otherwise LINK = 'E', 'I', 'L', 'R' or 'S'.

3: MEAN – CHARACTER*1 *Input*

On entry: indicates if a mean term is to be included.

MEAN = 'M'
A mean term, intercept, will be included in the model.

MEAN = 'Z'
The model will pass through the origin, zero-point.

Constraint: MEAN = 'M' or 'Z'.

4: OFFSET – CHARACTER*1 *Input*

On entry: indicates if an offset is required.

OFFSET = 'Y'
An offset must be supplied in OFF.

OFFSET = 'N'
OFF is not referenced.

Constraint: OFFSET = 'Y' or 'N'.

5: WEIGHT – CHARACTER*1 *Input*

On entry: if VFOBS = .TRUE. indicates if weights are used, otherwise WEIGHT is not referenced.

WEIGHT = 'U'
No weights are used.

WEIGHT = 'W'
Weights are used and must be supplied in WT.

Constraint: if VFOBS = .TRUE., WEIGHT = 'U' or 'W'.

6: N – INTEGER *Input*

On entry: n, the number of observations.

Constraint: N ≥ 1.

7: X(LDX,M) – ***double precision*** array *Input*

On entry: X(i,j) must contain the ith observation for the jth independent variable, for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$.

8: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G02GPF is called.

Constraint: LDX $\geq$ N.

9: M – INTEGER *Input*

On entry: m, the total number of independent variables.

Constraint: M ≥ 1.

10: ISX(M) – INTEGER array *Input*

On entry: indicates which independent variables are to be included in the model.

If ISX$(j) > 0$, the jth independent variable is included in the regression model.

Constraints:

ISX(j) ≥ 0, for $i = 1, 2, \ldots, \mathrm{M}$;
if MEAN = 'M', exactly IP $-$ 1 values of ISX must be > 0;
if MEAN = 'Z', exactly IP values of ISX must be > 0.

11: IP – INTEGER *Input*

On entry: the number of independent variables in the model, including the mean or intercept if present.

Constraint: IP > 0.

12: T($*$) – ***double precision*** array *Input*

Note: the dimension of the array must be at least N if ERRFN = 'B', and at least 1 otherwise.

On entry: if ERRFN = 'B', T(i) must contain the binomial denominator, t_i, for the ith observation.

Otherwise T is not referenced.

Constraint: if ERRFN = 'B', T(i) ≥ 0.0, for $i = 1, 2, \ldots, n$.

13: OFF($*$) – ***double precision*** array *Input*

Note: the dimension of the array must be at least N if OFFSET = 'Y', and at least 1 otherwise.

On entry: if OFFSET = 'Y', OFF(i) must contain the offset o_i, for the ith observation.

Otherwise OFF is not referenced.

14: WT($*$) – ***double precision*** array *Input*

Note: the dimension of the array must be at least N if WEIGHT = 'W' and VFOBS = .TRUE., and at least 1 otherwise.

On entry: if WEIGHT = 'W' and VFOBS = .TRUE., WT(i) must contain the weight, w_i, for the ith observation.

If the variance of future observations is not included in the standard error of the predicted variable, WT is not referenced.

Constraint: if VFOBS = .TRUE. and WEIGHT = 'W', WT(i) $\geq 0.$, for $i = 1, 2, \ldots, i$.

15: S – ***double precision*** *Input*

On entry: if ERRFN = 'N' or 'G' and VFOBS = .TRUE., the scale parameter, ϕ.

Otherwise S is not referenced and $\phi = 1$.

Constraint: if ERRFN = 'N' or 'G' and VFOBS = .TRUE., S > 0.0.

16: A – ***double precision*** *Input*

On entry: if LINK = 'E', A must contain the power of the exponential.

If LINK $\neq$ 'E', A is not referenced.

Constraint: if LINK = 'E', A $\neq 0.0$.

17: B(IP) – ***double precision*** array *Input*

On entry: the model parameters, β.

If MEAN = 'M', B(1) must contain the mean parameter and B($i + 1$) the coefficient of the variable contained in the jth independent X, where ISX(j) is the ith positive value in the array ISX.

If MEAN = 'Z', B(i) must contain the coefficient of the variable contained in the jth independent X, where ISX(j) is the ith positive value in the array ISX.

18: COV(IP × (IP + 1)/2) – ***double precision*** array *Input*

On entry: the upper triangular part of the variance-covariance matrix, C, of the model parameters. This matrix should be supplied packed by column, i.e., the covariance between parameters β_i and β_j, that is the values stored in B(i) and B(j), should be supplied in COV($j \times (j-1)/2 + i$), for $i = 1, 2, \ldots,$ IP and $j = i, i+1, \ldots,$ IP.

Constraint: the matrix represented in COV must be a valid variance-covariance matrix.

19: VFOBS – LOGICAL *Input*

On entry: if VFOBS = .TRUE., the variance of future observations is included in the standard error of the predicted variable (i.e., $I_{\text{fobs}} = 1$), otherwise $I_{\text{fobs}} = 0$.

20: ETA(N) – ***double precision*** array *Output*

On exit: the linear predictor, η.

21: SEETA(N) – ***double precision*** array *Output*

On exit: the standard error of the linear predictor, se(η).

22: PRED(N) – ***double precision*** array *Output*

On exit: the predicted value, $\hat{y}$.

23: SEPRED(N) – ***double precision*** array *Output*

On exit: the standard error of the predicted value, se($\hat{y}$). If PRED(i) could not be calculated, then G02GPF returns IFAIL = 22, and SEPRED(i) is set to -99.0.

24: 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: G02GPF 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, ERRFN $\neq$ 'B', 'P', 'G' or 'N'.

IFAIL = 2

On entry, ERRFN = 'B' and LINK $\neq$ 'G', 'P' or 'C' or ERRFN $\neq$ 'B' and LINK $\neq$ 'E', 'I', 'L', 'R' or 'S'.

IFAIL = 3

On entry, MEAN $\neq$ 'M' or 'Z'.

IFAIL $= 4$

On entry, OFFSET $\neq$ 'Y' or 'N'.

IFAIL $= 5$

On entry, VFOBS $=$ .TRUE. and WEIGHT $\neq$ 'U' or 'W'.

IFAIL $= 6$

On entry, N < 1.

IFAIL $= 8$

On entry, LDX $<$ N.

IFAIL $= 9$

On entry, M ≤ 0.

IFAIL $= 10$

On entry, number of nonzero elements in ISX is not consistent with IP.

IFAIL $= 11$

On entry, IP < 1.

IFAIL $= 12$

On entry, ERRFN $=$ 'B' and T$(i) < 0.0$ for at least one $i = 1, 2, \ldots, n$.

IFAIL $= 14$

On entry, VFOBS $=$ .TRUE., WEIGHT $=$ 'W' and WT$(i) < 0.0$ for at least one $i = 1, 2, \ldots, n$.

IFAIL $= 15$

On entry, VFOBS $=$ .TRUE., ERRFN $=$ 'G' or 'N' and S ≤ 0.0.

IFAIL $= 16$

On entry, LINK $=$ 'E' and A $= 0.0$.

IFAIL $= 18$

On entry, supplied covariance matrix has at least one diagonal element < 0.0.

IFAIL $= 22$

On exit, at least one predicted value could not be calculated as required. SEPRED is set to -99.0 for affected predicted values.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

The model

$$y = \frac{1}{\beta_1 + \beta_2 x} + \epsilon$$

is fitted to a training dataset with five observations. The resulting model is then used to predict the response for two new observations.

9.1 Program Text

```
*     G02GPF Example Program Text
*     Mark 22 Revised. NAG Copyright 2008.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, MMAX, LDX, LDV, PNMAX
      PARAMETER        (NMAX=5,MMAX=2,LDX=NMAX,LDV=NMAX,PNMAX=5)
*     .. Local Scalars ..
      DOUBLE PRECISION A, EPS, RSS, S, TOL
      INTEGER          I, IDF, IFAIL, IP, IPRINT, IRANK, J, M, MAXIT, N
      LOGICAL          VFOBS
      CHARACTER        ERROR, LINK, MEAN, OFFSET, WEIGHT
*     .. Local Arrays ..
      DOUBLE PRECISION B(MMAX), COV((MMAX*MMAX+MMAX)/2), ETA(PNMAX),
     +                 FWT(NMAX), OFF(PNMAX), PRED(PNMAX), SE(MMAX),
     +                 SEETA(PNMAX), SEPRED(PNMAX), T(1), V(LDV,7+MMAX),
     +                 WK((MMAX*MMAX+3*MMAX+22)/2), WT(PNMAX),
     +                 X(LDX,MMAX), Y(NMAX)
      INTEGER          ISX(MMAX)
*     .. External Subroutines ..
      EXTERNAL         G02GAF, G02GPF
*     .. Executable Statements ..
      CONTINUE
      WRITE (NOUT,*) 'G02GPF Example Program Results'
*
*     Skip headings in data file
      READ (NIN,*)
      READ (NIN,*)
*     Read in training data for model that will be used for prediction
      READ (NIN,*) LINK, MEAN, OFFSET, WEIGHT, N, M, S, IPRINT
      IF (N.GT.NMAX .OR. M.GE.MMAX) THEN
         WRITE (NOUT,99994)
         GO TO 200
      END IF
      IF (WEIGHT.EQ.'W' .OR. WEIGHT.EQ.'w') THEN
         DO 20 I = 1, N
            READ (NIN,*) (X(I,J),J=1,M), Y(I), FWT(I)
   20    CONTINUE
      ELSE
         DO 40 I = 1, N
            READ (NIN,*) (X(I,J),J=1,M), Y(I)
   40    CONTINUE
      END IF
      READ (NIN,*) (ISX(J),J=1,M)
*     Calculate IP
      IP = 0
      DO 60 J = 1, M
         IF (ISX(J).GT.0) THEN
            IP = IP + 1
         END IF
   60 CONTINUE
      IF (MEAN.EQ.'M' .OR. MEAN.EQ.'m') THEN
         IP = IP + 1
      END IF
      IF (LINK.EQ.'E' .OR. LINK.EQ.'e') THEN
         READ (NIN,*) A
      END IF
*     Set control parameters
```

```
      EPS = 0.000001D0
      TOL = 0.00005D0
      MAXIT = 10
      IFAIL = 1
*     Call routine to fit model to training data
      CALL G02GAF(LINK,MEAN,OFFSET,WEIGHT,N,X,LDX,M,ISX,IP,Y,FWT,S,A,
     +            RSS,IDF,B,IRANK,SE,COV,V,LDV,TOL,MAXIT,IPRINT,EPS,WK,
     +            IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99996) IFAIL
         IF (IFAIL.LT.6) THEN
            GO TO 200
         END IF
      END IF
*     Display parameter estimates for training data
      WRITE (NOUT,*)
      WRITE (NOUT,99999) RSS, IDF
      WRITE (NOUT,*)
      WRITE (NOUT,*) '      Estimate     Standard error'
      WRITE (NOUT,*)
      DO 80 I = 1, IP
         WRITE (NOUT,99998) B(I), SE(I)
   80 CONTINUE
*
*     Skip second lot of headings in data file
      READ (NIN,*)
*     Read in data to predict from and check array sizes
      READ (NIN,*) N, VFOBS, OFFSET, WEIGHT
      IF (N.GT.PNMAX) THEN
         WRITE (NOUT,*) 99994
         GO TO 200
      END IF
      IF (OFFSET.EQ.'Y' .AND. WEIGHT.EQ.'W') THEN
         DO 100 I = 1, N
            READ (NIN,*) (X(I,J),J=1,M), OFF(I), WT(I)
  100    CONTINUE
      ELSE IF (OFFSET.EQ.'Y') THEN
         DO 120 I = 1, N
            READ (NIN,*) (X(I,J),J=1,M), OFF(I)
  120    CONTINUE
      ELSE IF (WEIGHT.EQ.'W') THEN
         DO 140 I = 1, N
            READ (NIN,*) (X(I,J),J=1,M), WT(I)
  140    CONTINUE
      ELSE
         DO 160 I = 1, N
            READ (NIN,*) (X(I,J),J=1,M)
  160    CONTINUE
      END IF
*     Using G02GAF to fit training model, so error structure is normal
      ERROR = 'N'
*     Call prediction routine
      IFAIL = 1
      CALL G02GPF(ERROR,LINK,MEAN,OFFSET,WEIGHT,N,X,LDX,M,ISX,IP,T,OFF,
     +            WT,S,A,B,COV,VFOBS,ETA,SEETA,PRED,SEPRED,IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99995) IFAIL
         IF (IFAIL.NE.22) THEN
            GO TO 200
         END IF
      END IF
*     Display predicted values
      WRITE (NOUT,*)
      WRITE (NOUT,*) '  I      ETA         SE(ETA)      Predicted  ',
     +  ' SE(Predicted)'
      WRITE (NOUT,*)
      DO 180 I = 1, N
         WRITE (NOUT,99997) I, ETA(I), SEETA(I), PRED(I), SEPRED(I)
  180 CONTINUE
*
  200 CONTINUE
```

```
*
99999 FORMAT (1X,'Residual sum of squares = ',E12.4,' Degrees of freed',
     +       'om = ',I2)
99998 FORMAT (1X,2F14.4)
99997 FORMAT (1X,I3,')',4(F10.5,4X))
99996 FORMAT (1X,/1X,' ** G02GAF returned with IFAIL = ',I5)
99995 FORMAT (1X,/1X,' ** G02GPF returned with IFAIL = ',I5)
99994 FORMAT (1X,' ** Problem size too large, increase array limits')
      END
```

9.2 Program Data

```
G02GPF Example Program Data
Training Data
'R' 'M' 'N' 'U' 5 1  0.0 0  : LINK,MEAN,OFFSET,WEIGHT,N,M,S,IPRINT
1.0 25.0
2.0 10.0
3.0  6.0
4.0  4.0
5.0  3.0                    : End of X, Y
1                           : ISX
Prediction Data
2 .TRUE. 'N' 'U'            : N,VFOBS,OFFSET,WEIGHT
32.0
18.0                        : End of X
```

9.3 Program Results

```
 G02GPF Example Program Results

 Residual sum of squares =   0.3872E+00 Degrees of freedom =  3

       Estimate     Standard error

        -0.0239         0.0028
         0.0638         0.0026

   I       ETA          SE(ETA)       Predicted   SE(Predicted)

   1)   2.01807        0.08168        0.49552        0.35981
   2)   1.12472        0.04476        0.88911        0.36098
```