G10ZAF – NAG Fortran Library Routine Document

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

G10ZAF orders and weights data which is entered unsequentially, weighted or unweighted.

2 Specification

```
 SUBROUTINE G10ZAF(WEIGHT, N, X, Y, WT, NORD, XORD, YORD, WTORD,
1                  RSS, IWRK, IFAIL)
 INTEGER           N, NORD, IWRK(N), IFAIL
 real              X(N), Y(N), WT(*), XORD(N), YORD(N), WTORD(N),
1                  RSS
 CHARACTER*1       WEIGHT
```

3 Description

Given a set of observations (x_i, y_i) for $i = 1, 2, \ldots, n$, with corresponding weights w_i, G10ZAF rearranges the observations so that the x_i's are in ascending order.

For any equal x_i in the ordered set, say $x_j = x_{j+1} = \ldots = x_{j+k}$ a single observation x_j is returned with a corresponding y' and w', calculated as:

$$w' = \sum_{l=0}^{k} w_{i+l}$$

and

$$y' = \frac{\sum_{l=0}^{k}}{w}_{i+l} y_{i+l} w'.$$

Observations with zero weight are ignored. If no weights are supplied by the user, then unit weights are assumed; that is $w_i = 1$ for $i = 1, 2, \ldots, n$.

In addition, the within group sum of squares is computed for the tied observations using West's algorithm [2].

4 References

[1] Draper N R and Smith H (1985) *Applied Regression Analysis* Wiley (2nd Edition)

[2] West D H D (1979) Updating mean and variance estimates: An improved method *Comm. ACM* **22** 532–535

5 Parameters

1: WEIGHT — CHARACTER*1 *Input*

On entry: indicates whether user-defined weights are to be used.

If WEIGHT = 'W' user-defined weights are to be used and must be supplied in WT.
If WEIGHT = 'U' the data is treated as unweighted.

Constraint: WEIGHT = 'W' or 'U'.

2: N — INTEGER *Input*

On entry: the number of observations, n.

Constraint: N ≥ 1.

3: X(N) — ***real*** array *Input*

On entry: the values, x_i for $i = 1, 2, \ldots, n$.

4: Y(N) — ***real*** array *Input*

On entry: the values, y_i for $i = 1, 2, \ldots, n$.

5: WT(∗) — ***real*** array *Input*

Note: the dimension of the array WT must be at least 1 if WEIGHT = 'U' and N if WEIGHT = 'W'.

On entry: if WEIGHT = 'W' then WT must contain n weights, w_i, for $i = 1, 2, \ldots, n$. If WEIGHT = 'U' then WT is not referenced.

Constraint: if WEIGHT = 'W' then WT(i) ≥ 0.0 for $i = 1, 2, \ldots, n$, and at least one WT(i) > 0.0 for some i.

6: NORD — INTEGER *Output*

On exit: the number of distinct observations.

7: XORD(N) — ***real*** array *Output*

On exit: the first NORD elements contain the ordered and distinct x_i.

8: YORD(N) — ***real*** array *Output*

On exit: the first NORD elements contain the values y corresponding to the values in XORD.

9: WTORD(N) — ***real*** array *Output*

On exit: the first NORD elements contain the values w' corresponding to the values of XORD and YORD.

10: RSS — ***real*** *Output*

On exit: the within group sum of squares for tied observations.

11: IWRK(N) — INTEGER array *Workspace*

12: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

On exit: IFAIL = 0 unless the routine detects an error (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 detected by the routine:

IFAIL = 1

On entry, WEIGHT ≠ 'W' or 'U',
or N < 1.

IFAIL = 2

On entry, WEIGHT = 'W' and at least one element of WT is < 0.0, or all elements of WT are 0.0.

7 Accuracy

For a discussion on the accuracy of the algorithm for computing mean and variance see West [2].

8 Further Comments

The routine may be used to compute the pure error sum of squares in simple linear regression along with G02DAF, see Draper and Smith [1].

9 Example

A set of unweighted observations are input and G10ZAF used to produce a set of strictly increasing weighted observations.

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.

```
*     G10ZAF Example Program Text
*     Mark 16 Release. NAG Copyright 1992.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX
      PARAMETER        (NMAX=100)
*     .. Local Scalars ..
      real             RSS
      INTEGER          I, IFAIL, N, NORD
      CHARACTER        WEIGHT
*     .. Local Arrays ..
      real             WT(NMAX), WTORD(NMAX), X(NMAX), XORD(NMAX),
     +                 Y(NMAX), YORD(NMAX)
      INTEGER          IWRK(NMAX)
*     .. External Subroutines ..
      EXTERNAL         G10ZAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G10ZAF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
         READ (NIN,*) WEIGHT
         DO 20 I = 1, N
            READ (NIN,*) X(I), Y(I)
   20    CONTINUE
         IFAIL = 0
*
         CALL G10ZAF(WEIGHT,N,X,Y,WT,NORD,XORD,YORD,WTORD,RSS,IWRK,
     +               IFAIL)
*
*        Print results
*
         WRITE (NOUT,*)
         WRITE (NOUT,99999) NORD
         WRITE (NOUT,99998) RSS
         WRITE (NOUT,*)
         WRITE (NOUT,99997)
         DO 40 I = 1, NORD
            WRITE (NOUT,99996) XORD(I), YORD(I), WTORD(I)
   40    CONTINUE
      END IF
```

```
      STOP
*
99999 FORMAT (1X,'Number of distinct observations = ',I6)
99998 FORMAT (1X,'Residual sum of squares = ',F13.5)
99997 FORMAT (13X,'X                 Y                 WT')
99996 FORMAT (5X,F13.5,5X,F13.5,5X,F13.5)
      END
```

9.2 Program Data

```
G10ZAF Example Program Data
10
'U'
1.0 4.0
3.0 4.0
5.0 1.0
5.0 2.0
3.0 5.0
4.0 3.0
9.0 4.0
6.0 9.0
9.0 7.0
9.0 4.0
```

9.3 Program Results

```
G10ZAF Example Program Results

Number of distinct observations =      6
Residual sum of squares =       7.00000

           X                 Y                 WT
     1.00000           4.00000           1.00000
     3.00000           4.50000           2.00000
     4.00000           3.00000           1.00000
     5.00000           1.50000           2.00000
     6.00000           9.00000           1.00000
     9.00000           5.00000           3.00000
```