G02CFF – 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

G02CFF re-orders the elements in two vectors (typically vectors of means and standard deviations), and the rows and columns in two matrices (typically either matrices of sums of squares and cross-products of deviations from means and Pearson product-moment correlation coefficients, or matrices of sums of squares and cross-products about zero and correlation-like coefficients).

2 Specification

```
 SUBROUTINE G02CFF(N, KORDER, XBAR, STD, SSP, ISSP, R, IR, KWORK,
1                  IFAIL)
 INTEGER           N, KORDER(N), ISSP, IR, KWORK(N), IFAIL
 real              XBAR(N), STD(N), SSP(ISSP,N), R(IR,N)
```

3 Description

Input to the routine consists of:

(a) A list of the order in which the n variables are to be arranged on exit:

$$i_1, i_2, i_3, \ldots, i_n$$

(b) A vector of means:

$$(\bar{x}_1, \bar{x}_2, \bar{x}_3, \ldots, \bar{x}_n)$$

(c) A vector of standard deviations:

$$(s_1, s_2, s_3, \ldots, s_n)$$

(d) A matrix of sums of squares and cross-products of deviations from means:

$$\begin{pmatrix} S_{11} & S_{12} & S_{13} & . & . & . & S_{1n} \\ S_{21} & S_{22} & & & & & . \\ S_{31} & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ S_{n1} & S_{n2} & . & & . & . & S_{nn} \end{pmatrix}$$

(e) A matrix of correlation coefficients:

$$\begin{pmatrix} R_{11} & R_{12} & R_{13} & . & . & . & R_{1n} \\ R_{21} & R_{22} & & & & & . \\ R_{31} & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ R_{n1} & R_{n2} & . & & . & . & R_{nn} \end{pmatrix}$$

On exit from the routine, these same vectors and matrices are re-ordered, in the manner specified, and contain the following information:

(i) The vector of means:

$$(\bar{x}_{i_1}, \bar{x}_{i_2}, \bar{x}_{i_3}, \ldots, .., \bar{x}_{i_n})$$

(ii) The vector of standard deviations:

$$(s_{i_1}, s_{i_2}, s_{i_3,i_n})$$

(iii) The matrix of sums of squares and cross-products of deviations from means:

$$\begin{pmatrix} S_{i_1i_1} & S_{i_1i_2} & S_{i_1i_3} & . & . & . & S_{i_1i_n} \\ S_{i_2i_1} & S_{i_2i_2} & & & & & . \\ S_{i_3i_1} & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ S_{i_ni_1} & S_{i_ni_2} & . & & . & . & S_{i_ni_n} \end{pmatrix}$$

(iv) The matrix of correlation coefficients:

$$\begin{pmatrix} R_{i_1i_1} & R_{i_1i_2} & R_{i_1i_3} & . & . & . & R_{i_1i_n} \\ R_{i_2i_1} & R_{i_2i_2} & & & & & . \\ R_{i_3i_1} & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ . & & & & & & . \\ R_{i_ni_1} & R_{i_ni_2} & . & & . & . & R_{i_ni_n} \end{pmatrix}$$

Note. For sums of squares of cross-products of deviations about zero and correlation-like coefficients S_{ij} and R_{ij} should be replaced by $\tilde{S}_{ij}$ and $\tilde{R}_{ij}$ in the description of the input and output above.

4 References

None.

5 Parameters

1: N — INTEGER *Input*

On entry: the number of variables n, in the input data.

Constraint: $\mathrm{N} \geq 2$.

2: KORDER(N) — INTEGER array *Input*

On entry: KORDER(i) must be set to the number of the original variable which is to be the ith variable in the re-arranged data, for $i = 1, 2, \ldots, n$.

Constraint: $1 \leq \mathrm{KORDER}(i) \leq \mathrm{N}$, for $i = 1, 2, \ldots, n$.

3: XBAR(N) — ***real*** array *Input/Output*

On entry: XBAR(i) must be set to the mean of variable i, for $i = 1, 2, \ldots, n$.

On exit: XBAR(i) contains the mean of variable k where $k = \mathrm{KORDER}(i)$, for $i = 1, 2, \ldots, n$.

4: STD(N) — ***real*** array *Input/Output*

On entry: STD(i) must be set to the standard deviation of variable i, for $i = 1, 2, \ldots, n$.

On exit: STD(i) contains the standard deviation of variable k where $k = \mathrm{KORDER}(i)$, for $i = 1, 2, \ldots, n$.

5: SSP(ISSP,N) — ***real*** array *Input/Output*

On entry: SSP(i, j) must be set to the sum of cross-products of deviations from means S_{ij} (or about zero $\tilde{S}_{ij}$) for variables i and j, for $i, j = 1, 2, \ldots, n$.

On exit: SSP(i, j) contains the sum of cross-products of deviations from means S_{kl} (or about zero $\tilde{S}_{kl}$) for variables k and l, where $k = \mathrm{KORDER}(i)$, and $l = \mathrm{KORDER}(j)$, $i, j = 1, 2, \ldots, n$.

6: ISSP — INTEGER *Input*

On entry: the first dimension of the array SSP as declared in the (sub)program from which G02CFF is called.

Constraint: ISSP $\geq$ N.

7: R(IR,N) — ***real*** array *Input/Output*

On entry: R(i,j) must be set to the Pearson product-moment correlation coefficient R_{ij} (or the correlation-like coefficient $\tilde{R}_{ij}$) for variables i and j, for $i,j = 1,2,\ldots,n$.

On exit: R(i,j) contains the Pearson product-moment correlation coefficient R_{kl} (or the correlation-like coefficient $\tilde{R}_{kl}$) for variables k and l, where k = KORDER(i) and l = KORDER(j), for $i,j = 1,2,\ldots,n$.

8: IR — INTEGER *Input*

On entry: the first dimension of the array R as declared in the (sub)program from which G02CFF is called.

Constraint: IR $\geq$ N.

9: KWORK(N) — INTEGER array *Workspace*

10: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

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

For this routine, because the values of output parameters may be useful even if IFAIL $\neq 0$ on exit, users are recommended to set IFAIL to -1 before entry. **It is then essential to test the value of IFAIL on exit**. To suppress the output of an error message when soft failure occurs, set IFAIL to 1.

6 Error Indicators and Warnings

Errors or warnings specified by the routine:

IFAIL = 1

On entry, N < 2.

IFAIL = 2

On entry, ISSP $<$ N,
or IR $<$ N.

IFAIL = 3

On entry, KORDER$(i) < 1$,
or KORDER$(i) >$ N for some $i = 1,2,\ldots,n$.

IFAIL = 4

On entry, there is not a one-to-one correspondence between the old variables and the new variables; at least one of the original variables is not included in the new set, and consequently at least one other variable has been included more than once.

7 Accuracy

Not applicable.

8 Further Comments

The time taken by the routine depends on n and the amount of re-arrangement involved.

The routine is intended primarily for use when a set of variables is to be re-ordered for use in a regression, and is described accordingly. There is however no reason why the routine should not also be used to re-order vectors and matrices which contain any other non-statistical information; the matrices need not be symmetric.

The routine may be used either with sums of squares and cross-products of deviations from means and Pearson product-moment correlation coefficients in connection with a regression involving a constant, or with sums of squares and cross-products about zero and correlation-like coefficients in connection with a regression with no constant.

9 Example

The following program reads in the means, standard deviations, sums of squares and cross-products, and correlation coefficients for three variables. The vectors and matrices are re-ordered so that they contain the means, standard deviations, sums of squares and cross-products, and correlation coefficients for the first, third and second variables (in that order). Finally the re-ordered vectors and matrices are printed.

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.

```
*     G02CFF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          N, ISSP, ICORR
      PARAMETER        (N=3,ISSP=N,ICORR=N)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J
*     .. Local Arrays ..
      real             CORR(ICORR,N), SSP(ISSP,N), STD(N), XM(N)
      INTEGER          IORDER(N), KW(N)
*     .. External Subroutines ..
      EXTERNAL         G02CFF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G02CFF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) (XM(I),I=1,N), (STD(I),I=1,N),
     +  ((SSP(I,J),J=1,N),I=1,N), ((CORR(I,J),J=1,N),I=1,N)
      WRITE (NOUT,*)
      WRITE (NOUT,99999) 'Original vector XM   :   ', (XM(I),I=1,N)
      WRITE (NOUT,*)
      WRITE (NOUT,99999) 'Original vector STD  :   ', (STD(I),I=1,N)
      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Original matrix SSP  :'
      WRITE (NOUT,99998) ((SSP(I,J),J=1,N),I=1,N)
      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Original matrix CORR :'
      WRITE (NOUT,99998) ((CORR(I,J),J=1,N),I=1,N)
      WRITE (NOUT,*)
      IORDER(1) = 1
      IORDER(2) = 3
```

```
      IORDER(3) = 2
      IFAIL = 1
*
      CALL G02CFF(N,IORDER,XM,STD,SSP,ISSP,CORR,ICORR,KW,IFAIL)
*
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99997) 'Routine fails, IFAIL =', IFAIL
      ELSE
         WRITE (NOUT,99996) 'New vector XM   :   ', (XM(I),I=1,N)
         WRITE (NOUT,*)
         WRITE (NOUT,99996) 'New vector STD  :   ', (STD(I),I=1,N)
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'New matrix SSP  :'
         WRITE (NOUT,99995) ((SSP(I,J),J=1,N),I=1,N)
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'New matrix CORR :'
         WRITE (NOUT,99995) ((CORR(I,J),J=1,N),I=1,N)
      END IF
      STOP
*
99999 FORMAT (1X,A,3F10.4)
99998 FORMAT (1X,3F10.4)
99997 FORMAT (1X,A,I2)
99996 FORMAT (1X,A,3F10.4)
99995 FORMAT (1X,3F10.4)
      END
```

9.2 Program Data

```
G02CFF Example Program Data
5.4000    5.8000    2.8000
4.9800    5.0695    1.9240
99.2000   -57.6000  6.4000
-57.6000  102.8000  -29.2000
6.4000    -29.2000  14.8000
1.0000    -0.5704   0.1670
-0.5704   1.0000    -0.7486
0.1670    -0.7486   1.0000
```

9.3 Program Results

```
 G02CFF Example Program Results

 Original vector XM   :       5.4000    5.8000    2.8000

 Original vector STD  :       4.9800    5.0695    1.9240

 Original matrix SSP  :
    99.2000  -57.6000    6.4000
   -57.6000  102.8000  -29.2000
     6.4000  -29.2000   14.8000

 Original matrix CORR :
     1.0000   -0.5704    0.1670
    -0.5704    1.0000   -0.7486
     0.1670   -0.7486    1.0000
```

```
New vector XM   :       5.4000    2.8000    5.8000

New vector STD  :       4.9800    1.9240    5.0695

New matrix SSP  :
   99.2000    6.4000  -57.6000
    6.4000   14.8000  -29.2000
  -57.6000  -29.2000  102.8000

New matrix CORR :
    1.0000    0.1670   -0.5704
    0.1670    1.0000   -0.7486
   -0.5704   -0.7486    1.0000
```