NAG Library Routine Document

G02CFF

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 reorders 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, LDSSP, R, LDR, KWORK,        &
                   IFAIL)

INTEGER            N, KORDER(N), LDSSP, LDR, KWORK(N), IFAIL
REAL (KIND=nag_wp) XBAR(N), STD(N), SSP(LDSSP,N), R(LDR,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 reordered, 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:

$$(s_{i_1}, s_{i_2}, s_{i_3}, \ldots s_{i_n}).$$

(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: n, the number of variables in the input data.

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

3: XBAR(N) – REAL (KIND=nag_wp) 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 $k =$ KORDER(i), for $i = 1, 2, \ldots, n$.

4: STD(N) – REAL (KIND=nag_wp) 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 $k =$ KORDER(i), for $i = 1, 2, \ldots, n$.

5: SSP(LDSSP, N) – REAL (KIND=nag_wp) 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 $i = 1,2,\ldots,n$ and $j = 1,2,\ldots,n$.

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 $k =$ KORDER(i), and $l =$ KORDER(j), $i,j = 1,2,\ldots,n$.

6: LDSSP – INTEGER *Input*

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

Constraint: LDSSP $\geq$ N.

7: R(LDR, N) – REAL (KIND=nag_wp) 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 $i = 1,2,\ldots,n$ and $j = 1,2,\ldots,n$.

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 $i = 1,2,\ldots,n$ and $j = 1,2,\ldots,n$.

8: LDR – INTEGER *Input*

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

Constraint: LDR $\geq$ N.

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

10: 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.

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.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (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 or warnings detected by the routine:

IFAIL $= 1$

On entry, N < 2.

IFAIL $= 2$

On entry, LDSSP $<$ N,
or LDR $<$ 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.

IFAIL $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.8 in the Essential Introduction for further information.

IFAIL $= -399$

Your licence key may have expired or may not have been installed correctly.

See Section 3.7 in the Essential Introduction for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

7 Accuracy

Not applicable.

8 Parallelism and Performance

Not applicable.

9 Further Comments

The time taken by G02CFF 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 reordered for use in a regression, and is described accordingly. There is however no reason why the routine should not also be used to reorder 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.

10 Example

This example reads in the means, standard deviations, sums of squares and cross-products, and correlation coefficients for three variables. The vectors and matrices are reordered 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 reordered vectors and matrices are printed.

10.1 Program Text

```
    Program g02cffe

!     G02CFF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: g02cff, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                          :: i, ifail, ldr, ldssp, n
      Character (80)                   :: fmt
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: r(:,:), ssp(:,:), std(:), xbar(:)
      Integer, Allocatable             :: korder(:), kwork(:)
!     .. Executable Statements ..
      Write (nout,*) 'G02CFF Example Program Results'
      Write (nout,*)

!     Skip heading in data file
      Read (nin,*)

!     Read in the problem size
      Read (nin,*) n

      ldr = n
      ldssp = n
      Allocate (r(ldr,n),ssp(ldssp,n),std(n),xbar(n),korder(n),kwork(n))

!     Read in data
      Read (nin,*) xbar(1:n)
      Read (nin,*) std(1:n)
      Read (nin,*)(ssp(i,1:n),i=1,n)
      Read (nin,*)(r(i,1:n),i=1,n)

!     Read in the reordering vector
      Read (nin,*) korder(1:n)

!     Format for displaying vectors
      Write (fmt,99999) '(1x,A,', n, '(1X,F10.4))'

!     Display data
      Write (nout,fmt) 'Original vector XBAR :   ', xbar(1:n)
      Write (nout,*)
      Write (nout,fmt) 'Original vector STD  :   ', std(1:n)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04caf('General',' ',n,n,ssp,ldssp,'Original matrix SSP  :',ifail)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04caf('General',' ',n,n,r,ldr,'Original matrix R    :',ifail)
      Write (nout,*)

!     Reorder the results
      ifail = 0
      Call g02cff(n,korder,xbar,std,ssp,ldssp,r,ldr,kwork,ifail)

!     Display results
      Write (nout,fmt) 'New vector XBAR :   ', xbar(1:n)
      Write (nout,*)
      Write (nout,fmt) 'New vector STD  :   ', std(1:n)
      Write (nout,*)
      Flush (nout)
      ifail = 0
```

```
      Call x04caf('General',' ',n,n,ssp,ldssp,'New matrix SSP  :',ifail)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04caf('General',' ',n,n,r,ldr,'New matrix R    :',ifail)

99999 Format (A,I0,A)
    End Program g02cffe
```

10.2 Program Data

```
G02CFF Example Program Data
3                               :: N
  5.4000     5.8000     2.8000 :: XBAR
  4.9800     5.0695     1.9240 :: STD
 99.2000   -57.6000     6.4000
-57.6000   102.8000   -29.2000
  6.4000   -29.2000    14.8000 :: End of SSP
  1.0000    -0.5704     0.1670
 -0.5704     1.0000    -0.7486
  0.1670    -0.7486     1.0000 :: End of R
1  3  2                         :: KORDER
```

10.3 Program Results

```
 G02CFF Example Program Results

 Original vector XBAR :        5.4000     5.8000     2.8000

 Original vector STD  :        4.9800     5.0695     1.9240

 Original matrix SSP  :
             1          2          3
 1     99.2000   -57.6000     6.4000
 2    -57.6000   102.8000   -29.2000
 3      6.4000   -29.2000    14.8000

 Original matrix R    :
          1       2       3
 1   1.0000 -0.5704  0.1670
 2  -0.5704  1.0000 -0.7486
 3   0.1670 -0.7486  1.0000

 New vector XBAR :        5.4000     2.8000     5.8000

 New vector STD  :        4.9800     1.9240     5.0695

 New matrix SSP  :
             1          2          3
 1     99.2000     6.4000   -57.6000
 2      6.4000    14.8000   -29.2000
 3    -57.6000   -29.2000   102.8000

 New matrix R    :
          1       2       3
 1   1.0000  0.1670 -0.5704
 2   0.1670  1.0000 -0.7486
 3  -0.5704 -0.7486  1.0000
```