NAG Library Routine Document

G03ZAF

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

G03ZAF produces standardized values (z-scores) for a data matrix.

2 Specification

```
SUBROUTINE G03ZAF (N, M, X, LDX, NVAR, ISX, S, E, Z, LDZ, IFAIL)
INTEGER            N, M, LDX, NVAR, ISX(M), LDZ, IFAIL
REAL (KIND=nag_wp) X(LDX,M), S(M), E(M), Z(LDZ,NVAR)
```

3 Description

For a data matrix, X, consisting of n observations on p variables, with elements x_{ij}, G03ZAF computes a matrix, Z, with elements z_{ij} such that:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}, \quad i = 1, 2, \ldots, n; \quad j = 1, 2, \ldots, p,$$

where μ_j is a location shift and σ_j is a scaling factor. Typically, μ_j will be the mean and σ_j will be the standard deviation of the jth variable and therefore the elements in column j of Z will have zero mean and unit variance.

4 References

None.

5 Arguments

1: N – INTEGER *Input*

On entry: n, the number of observations in the data matrix.

Constraint: N $\geq$ 1.

2: M – INTEGER *Input*

On entry: the number of variables in the data array X.

Constraint: M $\geq$ NVAR.

3: X(LDX, M) – REAL (KIND=nag_wp) array *Input*

On entry: X(i, j) must contain the ith sample point for the jth variable, x_{ij}, for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots,$ M.

4: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which G03ZAF is called.

Constraint: LDX $\geq$ N.

5: NVAR – INTEGER *Input*

On entry: p, the number of variables to be standardized.

Constraint: NVAR ≥ 1.

6: ISX(M) – INTEGER array *Input*

On entry: ISX(j) indicates whether or not the observations on the jth variable are included in the matrix of standardized values.

If ISX(j) $\neq 0$, the observations from the jth variable are included.

If ISX(j) $= 0$, the observations from the jth variable are not included.

Constraint: ISX(j) $\neq 0$ for NVAR values of j.

7: S(M) – REAL (KIND=nag_wp) array *Input*

On entry: if ISX(j) $\neq 0$, S(j) must contain the scaling (standard deviation), σ_j, for the jth variable.

If ISX(j) $= 0$, S(j) is not referenced.

Constraint: if ISX(j) $\neq 0$, S(j) > 0.0, for $j = 1, 2, \ldots, \mathrm{M}$.

8: E(M) – REAL (KIND=nag_wp) array *Input*

On entry: if ISX(j) $\neq 0$, E(j) must contain the location shift (mean), μ_j, for the jth variable.

If ISX(j) $= 0$, E(j) is not referenced.

9: Z(LDZ, NVAR) – REAL (KIND=nag_wp) array *Output*

On exit: the matrix of standardized values (z-scores), Z.

10: LDZ – INTEGER *Input*

On entry: the first dimension of the array Z as declared in the (sub)program from which G03ZAF is called.

Constraint: LDZ $\geq$ N.

11: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation 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 argument, 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 < 1,
or NVAR < 1,
or M $<$ NVAR,
or LDX $<$ N,
or LDZ $<$ N.

IFAIL $= 2$

On entry, there are not precisely NVAR elements of ISX $\neq 0$.

IFAIL $= 3$

On entry, ISX(j) $\neq 0$ and S(j) ≤ 0.0 for some j.

IFAIL $= -99$

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

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -399$

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

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

7 Accuracy

Standard accuracy is achieved.

8 Parallelism and Performance

G03ZAF is not threaded in any implementation.

9 Further Comments

Means and standard deviations may be obtained using G01ATF or G02BXF.

10 Example

A 4 by 3 data matrix is input along with location and scaling values. The first and third columns are scaled and the results printed.

10.1 Program Text

```
    Program g03zafe

!     G03ZAF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: g03zaf, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                          :: i, ifail, ldx, ldz, m, n, nvar
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: e(:), s(:), x(:,:), z(:,:)
      Integer, Allocatable             :: isx(:)
!     .. Intrinsic Procedures ..
      Intrinsic                        :: count
!     .. Executable Statements ..
      Write (nout,*) 'G03ZAF Example Program Results'
      Write (nout,*)
      Flush (nout)

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

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

      ldx = n
      Allocate (x(ldx,m),isx(m),e(m),s(m))

!     Read in data
      Read (nin,*)(x(i,1:m),i=1,n)

!     Read in variable inclusion flags
      Read (nin,*) isx(1:m)

!     Read in shift and scaling
      Read (nin,*) e(1:m)
      Read (nin,*) s(1:m)

!     Calculate NVAR
      nvar = count(isx(1:m)/=0)

      ldz = n
      Allocate (z(ldz,nvar))

!     Standardize data
      ifail = 0
      Call g03zaf(n,m,x,ldx,nvar,isx,s,e,z,ldz,ifail)

!     Display results
      ifail = 0
      Call x04caf('General',' ',n,nvar,z,ldz,'Standardized Values',ifail)

    End Program g03zafe
```

10.2 Program Data

```
G03ZAF Example Program Data
4 3                :: N, M
15.0 0.0 1500.0
12.0 1.0 1000.0
18.0 2.0 1200.0
14.0 3.0  500.0    :: End of X
 1      0    1     :: ISX
14.75 0.0 1050.0   :: E
 2.50 0.0  420.3   :: S
```

10.3 Program Results

```
 G03ZAF Example Program Results

 Standardized Values
              1          2
 1      0.1000     1.0707
 2     -1.1000    -0.1190
 3      1.3000     0.3569
 4     -0.3000    -1.3086
```