NAG Library Routine Document

G07GAF

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

G07GAF identifies outlying values using Peirce's criterion.

2 Specification

```
SUBROUTINE G07GAF (N, P, Y, MEAN, VAR, IOUT, NIOUT, LDIFF, DIFF, LLAMB,    &
                   IFAIL)

INTEGER            N, P, IOUT(N), NIOUT, LDIFF, IFAIL
REAL (KIND=nag_wp) Y(N), MEAN, VAR, DIFF(LDIFF), LLAMB(LDIFF)
```

3 Description

G07GAF flags outlying values in data using Peirce's criterion. Let

y denote a vector of n observations (for example the residuals) obtained from a model with p parameters,

m denote the number of potential outlying values,

μ and σ^2 denote the mean and variance of y respectively,

$\tilde{y}$ denote a vector of length $n - m$ constructed by dropping the m values from y with the largest value of $|y_i - \mu|$,

$\tilde{\sigma}^2$ denote the (unknown) variance of $\tilde{y}$,

λ denote the ratio of $\tilde{\sigma}$ and σ with $\lambda = \frac{\tilde{\sigma}}{\sigma}$.

Peirce's method flags y_i as a potential outlier if $|y_i - \mu| \geq x$, where $x = \sigma^2 z$ and z is obtained from the solution of

$$R^m = \lambda^{m-n} \frac{m^m (n-m)^{n-m}}{n^n} \tag{1}$$

where

$$R = 2 \exp\left(\left(\frac{z^2 - 1}{2} \right) (1 - \Phi(z)) \right) \tag{2}$$

and Φ is the cumulative distribution function for the standard Normal distribution.

As $\tilde{\sigma}^2$ is unknown an assumption is made that the relationship between $\tilde{\sigma}^2$ and σ^2, hence λ, depends only on the sum of squares of the rejected observations and the ratio estimated as

$$\lambda^2 = \frac{n - p - mz^2}{n - p - m}$$

which gives

$$z^2 = 1 + \frac{n - p - m}{m} \left(1 - \lambda^2\right) \tag{3}$$

A value for the cutoff x is calculated iteratively. An initial value of $R = 0.2$ is used and a value of λ is estimated using equation (1). Equation (3) is then used to obtain an estimate of z and then equation (2) is used to get a new estimate for R. This process is then repeated until the relative change in z between consecutive iterations is $\leq \sqrt{\epsilon}$, where ϵ is ***machine precision***.

By construction, the cutoff for testing for $m+1$ potential outliers is less than the cutoff for testing for m potential outliers. Therefore Peirce's criterion is used in sequence with the existence of a single potential outlier being investigated first. If one is found, the existence of two potential outliers is investigated etc.

If one of a duplicate series of observations is flagged as an outlier, then all of them are flagged as outliers.

4 References

Gould B A (1855) On Peirce's criterion for the rejection of doubtful observations, with tables for facilitating its application *The Astronomical Journal* **45**

Peirce B (1852) Criterion for the rejection of doubtful observations *The Astronomical Journal* **45**

5 Parameters

1: N – INTEGER *Input*

On entry: n, the number of observations.

Constraint: $\mathrm{N} \geq 3$.

2: P – INTEGER *Input*

On entry: p, the number of parameters in the model used in obtaining the y. If y is an observed set of values, as opposed to the residuals from fitting a model with p parameters, then p should be set to 1, i.e., as if a model just containing the mean had been used.

Constraint: $1 \leq \mathrm{P} \leq \mathrm{N} - 2$.

3: Y(N) – REAL (KIND=nag_wp) array *Input*

On entry: y, the data being tested.

4: MEAN – REAL (KIND=nag_wp) *Input*

On entry: if $\mathrm{VAR} > 0.0$, MEAN must contain μ, the mean of y, otherwise MEAN is not referenced and the mean is calculated from the data supplied in Y.

5: VAR – REAL (KIND=nag_wp) *Input*

On entry: if $\mathrm{VAR} > 0.0$, VAR must contain σ^2, the variance of y, otherwise the variance is calculated from the data supplied in Y.

6: IOUT(N) – INTEGER array *Output*

On exit: the indices of the values in Y sorted in descending order of the absolute difference from the mean, therefore $|\mathrm{Y}(\mathrm{IOUT}(i-1)) - \mu| \geq |\mathrm{Y}(\mathrm{IOUT}(i)) - \mu|$, for $i = 2, 3, \ldots, \mathrm{N}$.

7: NIOUT – INTEGER *Output*

On exit: the number of potential outliers. The indices for these potential outliers are held in the first NIOUT elements of IOUT. By construction there can be at most $\mathrm{N} - \mathrm{P} - 1$ values flagged as outliers.

8: LDIFF – INTEGER *Input*

On entry: the maximum number of values to be returned in arrays DIFF and LLAMB.

If $\mathrm{LDIFF} \leq 0$, arrays DIFF and LLAMB are not referenced.

9: DIFF(LDIFF) – REAL (KIND=nag_wp) array *Output*

On exit: DIFF(i) holds $|y-\mu|-\sigma^2 z$ for observation Y(IOUT(i)), for $i=1,2,\ldots,\min(\mathrm{LDIFF},\mathrm{NIOUT}+1,\mathrm{N}-\mathrm{P}-1)$.

10: LLAMB(LDIFF) – REAL (KIND=nag_wp) array *Output*

On exit: LLAMB(i) holds $\log(\lambda^2)$ for observation Y(IOUT(i)), for $i=1,2,\ldots,\min(\mathrm{LDIFF},\mathrm{NIOUT}+1,\mathrm{N}-\mathrm{P}-1)$.

11: 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 $=\langle value\rangle$.
Constraint: N ≥ 3.

IFAIL $=2$

On entry, P $=\langle value\rangle$ and N $=\langle value\rangle$.
Constraint: $1\leq \mathrm{P}\leq \mathrm{N}-2$.

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

One problem with Peirce’s algorithm as implemented in G07GAF is the assumed relationship between σ^2, the variance using the full dataset, and $\tilde{\sigma}^2$, the variance with the potential outliers removed. In some cases, for example if the data y were the residuals from a linear regression, this assumption may not hold as the regression line may change significantly when outlying values have been dropped resulting in a radically different set of residuals. In such cases G07GBF should be used instead.

10 Example

This example reads in a series of data and flags any potential outliers.

The dataset used is from Peirce’s original paper and consists of fifteen observations on the vertical semidiameter of Venus.

10.1 Program Text

```
    Program g07gafe

!     G07GAF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: g07gaf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: mean, var
      Integer                          :: i, ifail, ldiff, n, niout, p
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: diff(:), llamb(:), y(:)
      Integer, Allocatable             :: iout(:)
!     .. Executable Statements ..
      Write (nout,*) ’G07GAF Example Program Results’
      Write (nout,*)

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

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

      Allocate (y(n),iout(n),diff(ldiff),llamb(ldiff))

!     Read in data
      Read (nin,*) y(1:n)

!     Let routine calculate mean and variance
      mean = 0.0E0_nag_wp
      var = 0.0E0_nag_wp

!     Get a list of potential outliers
      ifail = 0
      Call g07gaf(n,p,y,mean,var,iout,niout,ldiff,diff,llamb,ifail)

!     Display results
      Write (nout,*) ’Number of potential outliers:’, niout
      If (ldiff>0) Then
        Write (nout,*) ’  No.  Index    Value       Diff    ln(lambda^2)’
      Else
```

```
        Write (nout,*) '  No.  Index    Value'
      End If
      Do i = 1, niout
        If (i>ldiff) Then
          Write (nout,99999) i, iout(i), y(iout(i))
        Else
          Write (nout,99998) i, iout(i), y(iout(i)), diff(i), llamb(i)
        End If
      End Do

99999 Format (1X,I4,2X,I4,1X,F10.2)
99998 Format (1X,I4,2X,I4,3(1X,F10.2))
    End Program g07gafe
```

10.2 Program Data

```
G07GAF Example Program Data
15 2 1 :: N,P,LDIFF
-0.30
 0.48
 0.63
-0.22
 0.18
-0.44
-0.24
-0.13
-0.05
 0.39
 1.01
 0.06
-1.40
 0.20
 0.10  :: Y
```

10.3 Program Results

```
 G07GAF Example Program Results

 Number of potential outliers: 2
   No.  Index    Value       Diff    ln(lambda^2)
    1    13      -1.40       0.31      -0.30
    2    11       1.01
```