NAG Fortran Library Routine Document

G07EAF

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

G07EAF computes a rank based (nonparametric) estimate and confidence interval for the location parameter of a single population.

2 Specification

```
 SUBROUTINE G07EAF(METHOD, N, X, CLEVEL, THETA, THETAL, THETAU, ESTCL,
1                  WLOWER, WUPPER, WRK, IWRK, IFAIL)
 INTEGER           N, IWRK(3*N), IFAIL
 real              X(N), CLEVEL, THETA, THETAL, THETAU, ESTCL, WLOWER,
1                  WUPPER, WRK(4*N)
 CHARACTER*1       METHOD
```

3 Description

Consider a vector of independent observations, $x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ with unknown common symmetric density $f(x_i - \theta)$. G07EAF computes the Hodges–Lehmann location estimator (see Lehmann (1975)) of the centre of symmetry θ, together with an associated confidence interval. The Hodges–Lehmann estimate is defined as

$$\hat{\theta} = \text{median}\left\{\frac{x_i + x_j}{2}, 1 \leq i \leq j \leq n\right\}.$$

Let $m = (n(n+1))/2$ and let a_k, for $k = 1, 2, \ldots, m$ denote the m ordered averages $(x_i + x_j)/2$ for $1 \leq i \leq j \leq n$. Then

if m is odd, $\hat{\theta} = a_k$ where $k = (m+1)/2$,

if m is even, $\hat{\theta} = (a_k + a_{k+1})/2$ where $k = m/2$.

This estimator arises from inverting the one-sample Wilcoxon signed-rank test statistic, $W(x - \theta_0)$, for testing the hypothesis that $\theta = \theta_0$. Effectively $W(x - \theta_0)$ is a monotonically decreasing step function of θ_0 with

$$\text{mean}(W) = \mu = \frac{n(n+1)}{4},$$

$$\text{var}(W) = \sigma^2 = \frac{n(n+1)(2n+1)}{24}.$$

The estimate $\hat{\theta}$ is the solution to the equation $W(x - \hat{\theta}) = \mu$; two methods are available for solving this equation. These methods avoid the computation of all the ordered averages a_k; this is because for large n both the storage requirements and the computation time would be excessive.

The first is an exact method based on a set partitioning procedure on the set of all ordered averages $(x_i + x_j)/2$ for $i \leq j$. This is based on the algorithm proposed by Monahan (1984).

The second is an iterative algorithm, based on the Illinois method which is a modification of the *regula falsi* method, see McKean and Ryan (1977). This algorithm has proved suitable for the function $W(x - \theta_0)$ which is asymptotically linear as a function of θ_0.

The confidence interval limits are also based on the inversion of the Wilcoxon test statistic.

Given a desired percentage for the confidence interval, $1 - \alpha$, expressed as a proportion between 0 and 1, initial estimates for the lower and upper confidence limits of the Wilcoxon statistic are found from

$$W_l = \mu - 0.5 + (\sigma \Phi^{-1}(\alpha/2))$$

and

$$W_u = \mu + 0.5 + (\sigma \Phi^{-1}(1 - \alpha/2)),$$

where Φ^{-1} is the inverse cumulative Normal distribution function.

W_l and W_u are rounded to the nearest integer values. These estimates are then refined using an exact method if $n \leq 80$, and a Normal approximation otherwise, to find W_l and W_u satisfying

$$P(W \leq W_l) \leq \alpha/2$$
$$P(W \leq W_l + 1) > \alpha/2$$

and

$$P(W \geq W_u) \leq \alpha/2$$
$$P(W \geq W_u - 1) > \alpha/2.$$

Let $W_u = m - k$; then $\theta_l = a_{k+1}$. This is the largest value θ_l such that $W(x - \theta_l) = W_u$.

Let $W_l = k$; then $\theta_u = a_{m-k}$. This is the smallest value θ_u such that $W(x - \theta_u) = W_l$.

As in the case of $\hat{\theta}$, these equations may be solved using either the exact or the iterative methods to find the values θ_l and θ_u.

Then (θ_l, θ_u) is the confidence interval for θ. The confidence interval is thus defined by those values of θ_0 such that the null hypothesis, $\theta = \theta_0$, is not rejected by the Wilcoxon signed-rank test at the $(100 \times \alpha)\%$ level.

4 References

Lehmann E L (1975) *Nonparametrics: Statistical Methods Based on Ranks* Holden-Day

Marazzi A (1987) Subroutines for robust estimation of location and scale in ROBETH *Cah. Rech. Doc. IUMSP, No. 3 ROB 1* Institut Universitaire de Médecine Sociale et Préventive, Lausanne

McKean J W and Ryan T A (1977) Algorithm 516: An algorithm for obtaining confidence intervals and point estimates based on ranks in the two-sample location problem *ACM Trans. Math. Software* **10** 183–185

Monahan J F (1984) Algorithm 616: Fast computation of the Hodges–Lehman location estimator *ACM Trans. Math. Software* **10** 265–270

5 Parameters

1: METHOD – CHARACTER*1 *Input*

On entry: specifies the method to be used.

If METHOD = 'E', the exact algorithm is used.

If METHOD = 'A', the iterative algorithm is used.

Constraint: METHOD = 'E' or 'A'.

2: N – INTEGER *Input*

On entry: the sample size, n.

Constraint: N $\geq$ 2.

3: X(N) – ***real*** array *Input*

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

4: CLEVEL – ***real*** *Input*

On entry: the confidence interval desired.

For example, for a 95% confidence interval set CLEVEL = 0.95.

Constraint: 0.0 < CLEVEL < 1.0.

5: THETA – ***real*** *Output*

On exit: the estimate of the location, $\hat{\theta}$.

6: THETAL – ***real*** *Output*

On exit: the estimate of the lower limit of the confidence interval, θ_l.

7: THETAU – ***real*** *Output*

On exit: the estimate of the upper limit of the confidence interval, θ_u.

8: ESTCL – ***real*** *Output*

On exit: an estimate of the actual percentage confidence of the interval found, as a proportion between (0.0,1.0).

9: WLOWER – ***real*** *Output*

On exit: the upper value of the Wilcoxon test statistic, W_u, corresponding to the lower limit of the confidence interval.

10: WUPPER – ***real*** *Output*

On exit: the lower value of the Wilcoxon test statistic, W_l, corresponding to the upper limit of the confidence interval.

11: WRK(4*N) – ***real*** array *Workspace*
12: IWRK(3*N) – INTEGER array *Workspace*

13: 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 (see Section 6).

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, for users 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.**

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, METHOD ≠ ’E’ or ’A’,
or N < 2,
or CLEVEL ≤ 0.0,
or CLEVEL ≥ 1.0.

IFAIL = 2

There is not enough information to compute a confidence interval since the whole sample consists of identical values.

IFAIL = 3

For at least one of the estimates $\hat{\theta}$, θ_l and θ_u, the underlying iterative algorithm (when METHOD = 'A') failed to converge. This is an unlikely exit but the estimate should still be a reasonable approximation.

7 Accuracy

The routine should produce results accurate to 5 significant figures in the width of the confidence interval; that is the error for any one of the three estimates should be less than 0.00001 × (THETAU − THETAL).

8 Further Comments

The time taken increases with the sample size n.

9 Example

The following program calculates a 95% confidence interval for θ, a measure of symmetry of the sample of 50 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.

```
*     G07EAF 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             CLEVEL, ESTCL, THETA, THETAL, THETAU, WLOWER,
     +                 WUPPER
      INTEGER          I, IFAIL, N
*     .. Local Arrays ..
      real             WRK(4*NMAX), X(NMAX)
      INTEGER          IWRK(3*NMAX)
*     .. External Subroutines ..
      EXTERNAL         G07EAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G07EAF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.1 .OR. N.GT.NMAX) THEN
         WRITE (NOUT,99999) N
      ELSE
         READ (NIN,*) (X(I),I=1,N)
         READ (NIN,*) CLEVEL
         IFAIL = 0
*
         CALL G07EAF('Exact',N,X,CLEVEL,THETA,THETAL,THETAU,ESTCL,
     +               WLOWER,WUPPER,WRK,IWRK,IFAIL)
*
         WRITE (NOUT,*)
         WRITE (NOUT,*) ' Location estimator     Confidence Interval '
         WRITE (NOUT,*)
         WRITE (NOUT,99998) THETA, '( ', THETAL, ' , ', THETAU, ' )'
```

```
         WRITE (NOUT,*)
         WRITE (NOUT,*) ' Corresponding Wilcoxon statistics'
         WRITE (NOUT,*)
         WRITE (NOUT,99997) ' Lower : ', WLOWER
         WRITE (NOUT,99997) ' Upper : ', WUPPER
      END IF
      STOP
*
99999 FORMAT (1X,'N is less than 2 or greater than NMAX : N = ',I8)
99998 FORMAT (3X,F10.4,12X,A,F6.4,A,F6.4,A)
99997 FORMAT (A,F8.2)
      END
```

9.2 Program Data

```
G07EAF Example Program Data
 40
 -0.23   0.35  -0.77   0.35   0.27   -0.72   0.08  -0.40  -0.76   0.45
  0.73   0.74   0.83  -0.87   0.21    0.29  -0.91  -0.04   0.82  -0.38
 -0.31   0.24  -0.47  -0.68  -0.77   -0.86  -0.59   0.73   0.39  -0.44
  0.63  -0.22  -0.07  -0.43  -0.21   -0.31   0.64  -1.00  -0.86  -0.73
  0.95
```

9.3 Program Results

```
 G07EAF Example Program Results

  Location estimator     Confidence Interval

      -0.1300            ( -.3300 , 0.0350 )

  Corresponding Wilcoxon statistics

 Lower :   556.00
 Upper :   264.00
```