NAG Toolbox for MATLAB

g01dd

1 Purpose

g01dd calculates Shapiro and Wilk's W statistic and its significance level for testing Normality.

2 Syntax

```
[a, w, pw, ifail] = g01dd(x, calwts, a, 'n', n)
```

3 Description

g01dd calculates Shapiro and Wilk's **w** statistic and its significance level for any sample size between 3 and 2000. It is an adaptation of the Applied Statistics Algorithm AS 181, see Royston 1982. The full description of the theory behind this algorithm is given in Royston 1982.

Given a set of observations $x_1, x_2, \ldots, x_n$ sorted into either ascending or descending order (m01ca may be used to sort the data) this function calculates the value of Shapiro and Wilk's **w** statistic defined as:

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_i\right)^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2},$$

where $\bar{x} = \frac{1}{n}\sum_{1}^{n} x_i$ is the sample mean and a_i, for $i = 1, 2, \ldots, n$, are a set of 'weights' whose values depend only on the sample size n.

On exit, the values of a_i, for $i = 1, 2, \ldots, n$ are only of interest should you wish to call the function again to calculate **w** and its significance level for a different sample of the same size.

It is recommended that the function is used in conjunction with a Normal $(Q - Q)$ plot of the data. Functions g01da and g01db can be used to obtain the required Normal scores.

4 References

Royston J P 1982 Algorithm AS181: The W test for normality *Appl. Statist.* **31** 176–180

Royston J P 1982 An extension of Shapiro and Wilk's W test for normality to large samples *Appl. Statist.* **31** 115–124

Royston J P 1986 A remark on AS181: The W test for normality *Appl. Statist.* **35** 232–234

5 Parameters

5.1 Compulsory Input Parameters

1: **x**(**n**) – **double array**

The ordered sample values, x_i; for $i = 1, 2, \ldots, n$.

2: **calwts – logical scalar**

Must be set to **true** if you wish g01dd to calculate the elements of **a**.

calwts should be set to **false** if you have saved the values in **a** from a previous call to g01dd.

If in doubt, set **calwts** equal to **true**.

3: **a(n) – double array**

If **calwts** has been set to **false** then before entry **a** must contain the n weights as calculated in a previous call to g01dd, otherwise **a** need not be set.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the arrays **x**, **a**. (An error is raised if these dimensions are not equal.)

n, the sample size.

Constraint: $3 \leq \mathbf{n} \leq 2000$.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **a(n) – double array**

The n weights required to calculate **w**.

2: **w – double scalar**

The value of the statistic, **w**.

3: **pw – double scalar**

The significance level of **w**.

4: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, $\mathbf{n} < 3$.

ifail $= 2$

On entry, $\mathbf{n} > 2000$.

ifail $= 3$

On entry, the elements in **x** are not in ascending or descending order or are all equal.

7 Accuracy

There may be a loss of significant figures for large n.

8 Further Comments

The time taken by g01dd depends roughly linearly on the value of n.

For very small samples the power of the test may not be very high.

The contents of the array **a** should not be modified between calls to g01dd for a given sample size, unless **calwts** is reset to **true** before each call of g01dd.

The Shapiro and Wilk **w** test is very sensitive to ties. If the data has been rounded the test can be improved by using Sheppard's correction to adjust the sum of squares about the mean. This produces an adjusted value of **w**,

$$WA = W\frac{\sum x_{(i)} - \bar{x}^2}{\left\{\sum_{i=1}^{n} x_{(i)} = \bar{x}^2 - \frac{n-1}{12}\omega^2\right\}},$$

where ω is the rounding width. WA can be compared with a standard Normal distribution, but a further approximation is given by Royston 1986.

9 Example

```
x = [0.11;
     0.21;
     0.42;
     0.46;
     0.47;
     0.71;
     0.93;
     1.52;
     3.14;
     3.24;
     4.43;
     4.55;
     4.61;
     4.75;
     4.97;
     6.66;
     7.87;
     7.95;
     9.53;
     10.14];
calwts = true;
a = zeros(20, 1);
[aOut, w, pw, ifail] = g01dd(x, calwts, a)
```

```
aOut =
    0.4758
    0.3183
    0.2557
    0.2082
    0.1685
    0.1335
    0.1014
    0.0712
    0.0423
    0.0140
    0.0620
    0.1869
    0.3149
    0.4483
    0.5903
    0.7454
    0.9210
    1.1310
    1.4076
    1.8675
w =
    0.8992
pw =
    0.0408
```

```
ifail =

           0
```