NAG Toolbox for MATLAB

g02bt

1 Purpose

g02bt updates the sample means and sums of squares and cross-products, or sums of squares and cross-products of deviations about the mean, for a new observation. The data may be weighted.

2 Syntax

```
[sw, xbar, c, ifail] = g02bt(mean, wt, x, incx, sw, xbar, c, 'm', m)
```

3 Description

g02bt is an adaptation of West's WV2 algorithm; see West 1979. This function updates the weighted means of variables and weighted sums of squares and cross-products or weighted sums of squares and cross-products of deviations about the mean for observations on m variables X_j, for $j = 1, 2, \ldots, m$. For the first $i-1$ observations let the mean of the jth variable be $\bar{x}_j(i-1)$, the cross-product about the mean for the jth and kth variables be $c_{jk}(i-1)$ and the sum of weights be W_{i-1}. These are updated by the ith observation, x_{ij}, for $j = 1, 2, \ldots, m$, with weight w_i as follows:

$$W_i = W_{i-1} + w_i, \qquad \bar{x}_j(i) = \bar{x}_j(i-1) + \frac{w_i}{W_i}\big(x_j - \bar{x}_j(i-1)\big), \qquad j = 1, 2, \ldots, m$$

and

$$c_{jk}(i) = c_{jk}(i-1) + \frac{w_i}{W_i}\big(x_j - \bar{x}_j(i-1)\big)\big(x_k - \bar{x}_k(i-1)\big)W_{i-1}, \qquad j = 1, 2, \ldots, m; k = j, j+1, 2, \ldots, m.$$

The algorithm is initialized by taking $\bar{x}_j(1) = x_{1j}$, the first observation and $c_{ij}(1) = 0.0$.

For the unweighted case $w_i = 1$ and $W_i = i$ for all i.

4 References

Chan T F, Golub G H and Leveque R J 1982 *Updating Formulae and a Pairwise Algorithm for Computing Sample Variances* Compstat, Physica-Verlag

West D H D 1979 Updating mean and variance estimates: An improved method *Comm. ACM* **22** 532–555

5 Parameters

5.1 Compulsory Input Parameters

1: **mean – string**

Indicates whether g02bt is to calculate sums of squares and cross-products, or sums of squares and cross-products of deviations about the mean.

mean = 'M'

The sums of squares and cross-products of deviations about the mean are calculated.

mean = 'Z'

The sums of squares and cross-products are calculated.

Constraint: **mean** = 'M' or 'Z'.

2: **wt – double scalar**

The weight to use for the current observation, w_i.

For unweighted means and cross-products set **wt** $= 1.0$. The use of a suitable negative value of **wt**, e.g., $-w_i$ will have the effect of deleting the observation.

3: **x**(**m** $\times$ **incx**) – **double array**

x$((j-1)$**incx** $+1)$ must contain the value of the jth variable for the current observation, $j = 1, 2, \ldots, m$.

4: **incx** – **int32 scalar**

The increment of **x**. Two situations are common.

If **incx** $= 1$, the data values are to be found in consecutive locations in **x**, i.e., in a column.

If **incx** $= LDX$, for some positive integer LDX, the data values are to be found as a row of an array with first dimension LDX.

Constraint: **incx** > 0.

5: **sw** – **double scalar**

The sum of weights for the previous observations, W_{i-1}.

sw $= 0.0$

The update procedure is initialized.

sw $+$ **wt** $= 0.0$

All elements of **xbar** and **c** are set to zero.

Constraint: **sw** ≥ 0.0 and **sw** $+$ **wt** ≥ 0.0.

6: **xbar**(**m**) – **double array**

If **sw** $= 0.0$, **xbar** is initialized, otherwise **xbar**(j) must contain the weighted mean of the jth variable for the previous $(i-1)$ observations, $\bar{x}_j(i-1)$, for $j = 1, 2, \ldots, m$.

7: **c**$((\mathbf{m} \times \mathbf{m} + \mathbf{m})/\mathbf{2})$ – **double array**

If **sw** $\neq 0.0$, **c** must contain the upper triangular part of the matrix of weighted sums of squares and cross-products or weighted sums of squares and cross-products of deviations about the mean. It is stored packed form by column, i.e., the cross-product between the jth and kth variable, $k \geq j$, is stored in **c**$(k \times (k-1)/2 + j)$.

5.2 Optional Input Parameters

1: **m** – **int32 scalar**

Default: The dimension of the array **xbar**.

m, the number of variables.

Constraint: **m** ≥ 1.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **sw** – **double scalar**

Contains the updated sum of weights, W_i.

2: **xbar(m) – double array**

xbar(j) contains the weighted mean of the jth variable, $\bar{x}_j(i)$, for $j = 1, 2, \ldots, m$.

3: **c((m × m + m)/2) – double array**

The update sums of squares and cross-products stored as on input.

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, **m** < 1,
or **incx** < 1.

ifail $= 2$

On entry, **sw** < 0.0.

ifail $= 3$

On entry, (**sw** + **wt**) < 0.0, the current weight causes the sum of weights to be less than 0.0.

ifail $= 4$

On entry, **mean** $\neq$ 'M' or 'Z'.

7 Accuracy

For a detailed discussion of the accuracy of this method see Chan *et al.* 1982 and West 1979.

8 Further Comments

g02bt may be used to update the results returned by g02bu.

g02bw may be used to calculate the correlation matrix from the matrix of sums of squares and cross-products of deviations about the mean and the matrix may be scaled using Missing 'id' to produce a variance-covariance matrix.

9 Example

```
mean = 'M';
wt = 0.13;
x = [9.1231;
     3.7011;
     4.523];
incx = int32(1);
sw = 0;
xbar = zeros(3,1);
c = zeros(6,1);
[swOut, xbarOut, cOut, ifail] = g02bt(mean, wt, x, incx, sw, xbar, c)
```

```
swOut =
    0.1300
xbarOut =
    9.1231
```

```
    3.7011
    4.5230
cOut =
     0
     0
     0
     0
     0
     0
ifail =
           0
```