NAG Toolbox for MATLAB

g11ba

1 Purpose

g11ba computes a table from a set of classification factors using a selected statistic.

2 Syntax

```
[table, ncells, ndim, idim, icount, auxt, ifail] = g11ba(stat, update,
weight, isf, lfac, ifac, y, wt, table, ncells, icount, auxt, 'n', n,
'nfac', nfac, 'maxt', maxt)
```

3 Description

A data set may include both classification variables and general variables. The classification variables, known as factors, take a small number of values known as levels. For example, the factor sex would have the levels male and female. These can be coded as 1 and 2 respectively. Given several factors, a multi-way table can be constructed such that each cell of the table represents one level from each factor. For example, the two factors sex and habitat, habitat having three levels (inner-city, suburban and rural) define the 2×3 contingency table

Sex	**Habitat**		
	Inner-city	Suburban	Rural
Male			
Female			

For each cell statistics can be computed. If a third variable in the data set was age, then for each cell the average age could be computed:

Sex	**Habitat**		
	Inner-city	Suburban	Rural
Male	25.5	30.3	35.6
Female	23.2	29.1	30.4

That is the average age for all observations for males living in rural areas is 35.6. Other statistics can also be computed: the number of observations, the total, the variance, the largest value and the smallest value.

g11ba computes a table for one of the selected statistics. The factors have to be coded with levels $1, 2, \ldots$. Weights can be used to eliminate values from the calculations, e.g., if they represent 'missing values'. There is also the facility to update an existing table with the addition of new observations.

4 References

John J A and Quenouille M H 1977 *Experiments: Design and Analysis* Griffin

Kendall M G and Stuart A 1969 *The Advanced Theory of Statistics (Volume 1)* (3rd Edition) Griffin

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: **stat – string**

Indicates which statistic is to be computed for the table cells.

If **stat** = 'N', the number of observations for each cell.

If **stat** = 'T', the total for the variable in **y** for each cell.

If **stat** = 'A', the average (mean) for the variable in **y** for each cell.

If **stat** = 'V', the variance for the variable in **y** for each cell.

If **stat** = 'L', the largest value for the variable in **y** for each cell.

If **stat** = 'S', the smallest value for the variable in **y** for each cell.

Constraint: **stat** = 'N', 'T', 'A', 'V', 'L' or 'S'.

2: **update – string**

Indicates if an existing table is to be updated by further observation.

If **update** = 'I', the table cells will be initialized to zero before tabulations take place.

If **update** = 'U', the table input in **table** will be updated. The parameters **ncells**, **table**, **icount** and **auxt** must remain unchanged from the previous call to g11ba.

Constraint: **update** = 'I' or 'U'.

3: **weight – string**

Indicates if weights are to be used.

weight = 'U'

Weights are not used and unit weights are assumed.

weight = 'W' or 'V'

Weights are used and must be supplied in **wt**. The only difference between **weight** = 'W' and **weight** = 'V' is if the variance is computed.

weight = 'W'

The divisor for the variance is the sum of the weights minus one and if **weight** = 'V', the divisor is the number of observations with nonzero weights minus one. The former is useful if the weights represent the frequency of the observed values.

If **stat** = 'T' or 'A', the weighted total or mean is computed respectively.

If **stat** = 'N', 'L' or 'S', the only effect of weights is to eliminate values with zero weights from the computations.

Constraint: **weight** = 'U', 'V' or 'W'.

4: **isf**(**nfac**) **– int32 array**

Indicates which factors in **ifac** are to be used in the tabulation.

If **isf**$(i) > 0$ the ith factor in **ifac** is included in the tabulation.

Note that if **isf**$(i) \leq 0$, for $i = 1, 2, \ldots,$ **nfac** then the statistic for the whole sample is calculated and returned in a 1×1 table.

5: **lfac**(**nfac**) – **int32 array**

The number of levels of the classifying factors in **ifac**.

Constraint: **lfac**$(i) \geq 2$ if **isf**$(i) > 0$, for $i = \mathrm{Ai}, \mathrm{NaN}, \ldots, \mathrm{Ai}$.

6: **ifac**(**ldf**,**nfac**) – **int32 array**

ldf, the first dimension of the array, must be at least **n**.

The **nfac** coded classification factors for the **n** observations.

Constraint: $1 \leq$ **ifac**$(i,j) \leq$ **lfac**(j), for $i = 1, 2, \ldots,$ **n** and $j = 1, 2, \ldots,$ **nfac**.

7: **y**(**n**) – **double array**

The variable to be tabulated. If **stat** = 'N', **y** is not referenced.

8: **wt**$(*)$ – **double array**

Note: the dimension of the array **wt** must be at least **n** if **weight** = 'W' or 'V', and at least 1 otherwise.

If **weight** = 'W' or 'V', **wt** must contain the **n** weights. Otherwise **wt** is not referenced.

Constraint: **wt**$(i) \geq 0.0$ if **weight** = 'W' or 'V', for $i = \mathrm{Ai}, \mathrm{NaN}, \ldots, \mathrm{Ai}$.

9: **table**(**maxt**) – **double array**

If **update** = 'U', **table** must be unchanged from the previous call to g11ba, otherwise **table** need not be set.

10: **ncells** – **int32 scalar**

If **update** = 'U', **ncells** must be unchanged from the previous call to g11ba, otherwise **ncells** need not be set.

11: **icount**(**maxt**) – **int32 array**

If **update** = 'U', **icount** must be unchanged from the previous call to g11ba, otherwise **icount** need not be set.

12: **auxt**$(*)$ – **double array**

Note: the dimension of the array **auxt** must be at least **ncells** if **stat** = 'A', $2 \times$ **ncells** if **stat** = 'V', and at least 1 otherwise.

If **update** = 'U', **auxt** must be unchanged from the previous call to g11ba, otherwise **auxt** need not be set.

5.2 Optional Input Parameters

1: **n** – **int32 scalar**

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

the number of observations.

Constraint: **n** ≥ 2.

2: **nfac** – **int32 scalar**

Default: The dimension of the arrays **isf**, **lfac**, **ifac**, **idim**. (An error is raised if these dimensions are not equal.)

the number of classifying factors in **ifac**.

Constraint: **nfac** ≥ 1.

3: **maxt – int32 scalar**

the maximum size of the table to be computed.

Constraint: **maxt** $\geq$ product of the levels of the factors included in the tabulation.

5.3 Input Parameters Omitted from the MATLAB Interface

ldf, iwk

5.4 Output Parameters

1: **table**(**maxt**) **– double array**

The computed table. The **ncells** cells of the table are stored so that for any two factors the index relating to the factor referred to later in **lfac** and **ifac** changes faster. For further details see Section 8.

2: **ncells – int32 scalar**

The number of cells in the table.

3: **ndim – int32 scalar**

The number of factors defining the table.

4: **idim**(**nfac**) **– int32 array**

The first **ndim** elements contain the number of levels for the factors defining the table.

5: **icount**(**maxt**) **– int32 array**

A table containing the number of observations contributing to each cell of the table, stored identically to **table**. Note if **stat** = 'N' this is the same as is returned in **table**.

6: **auxt**(∗) **– double array**

Note: the dimension of the array **auxt** must be at least **ncells** if **stat** = 'A', 2 × **ncells** if **stat** = 'V', and at least 1 otherwise.

If **stat** = 'A' or 'V', the first **ncells** values hold the table containing the sum of the weights for the observations contributing to each cell, stored identically to **table**.

If **stat** = 'V', the second set of **ncells** values hold the table of cell means. Otherwise **auxt** is not referenced.

7: **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, **n** < 2,
or **nfac** < 1,
or **ldf** < **n**,
or **update** ≠ 'I' or 'U',
or **weight** ≠ 'U', 'W' or 'V',
or **stat** ≠ 'N', 'T', 'A', 'V', 'L' or 'S'.

ifail = 2

On entry, **isf**$(i) > 0$ and **lfac**$(i) < 2$, for some i,
or **ifac**$(i,j) < 1$, for some i,j,
or **ifac**$(i,j) >$ **lfac**(j) for some i,j,
or **maxt** is too small,
or **weight** = 'W' or 'V' and **wt**$(i) < 0.0$, for some i.

ifail = 3

stat = 'V' and the divisor for the variance is ≤ 0.0.

ifail = 4

update = 'U' and at least one of **ncells**, **table**, **auxt** or **icount** have been changed since previous call to g11ba.

7 Accuracy

Only applicable when **stat** = 'V'. In this case a one pass algorithm is used as described by West 1979.

8 Further Comments

The tables created by g11ba and stored in **table**, **icount** and, depending on **stat**, also in **auxt** are stored in the following way. Let there be n factors defining the table with factor k having l_k levels, then the cell defined by the levels $i_1, i_2, \ldots, i_n$ of the factors is stored in the mth cell given by

$$m = 1 + \sum_{k=1}^{n} [(i_k - 1)c_k],$$

where $c_j = \prod_{k=j+1}^{n} l_k$, for $j = 1, 2, \ldots, n-1$ and $c_n = 1$.

9 Example

```
stat = 'A';
update = 'I';
weight = 'U';
isf = [int32(0);
     int32(1);
     int32(1)];
lfac = [int32(3);
     int32(3);
     int32(6)];
ifac = [int32(1), int32(1), int32(1);
     int32(1), int32(2), int32(1);
     int32(1), int32(3), int32(1);
     int32(1), int32(1), int32(2);
     int32(1), int32(2), int32(2);
     int32(1), int32(3), int32(2);
     int32(1), int32(1), int32(3);
     int32(1), int32(2), int32(3);
     int32(1), int32(3), int32(3);
     int32(1), int32(1), int32(4);
     int32(1), int32(2), int32(4);
     int32(1), int32(3), int32(4);
     int32(1), int32(1), int32(5);
     int32(1), int32(2), int32(5);
     int32(1), int32(3), int32(5);
     int32(1), int32(1), int32(6);
     int32(1), int32(2), int32(6);
     int32(1), int32(3), int32(6);
```

```
     int32(2), int32(1), int32(1);
     int32(2), int32(2), int32(1);
     int32(2), int32(3), int32(1);
     int32(2), int32(1), int32(2);
     int32(2), int32(2), int32(2);
     int32(2), int32(3), int32(2);
     int32(2), int32(1), int32(3);
     int32(2), int32(2), int32(3);
     int32(2), int32(3), int32(3);
     int32(2), int32(1), int32(4);
     int32(2), int32(2), int32(4);
     int32(2), int32(3), int32(4);
     int32(2), int32(1), int32(5);
     int32(2), int32(2), int32(5);
     int32(2), int32(3), int32(5);
     int32(2), int32(1), int32(6);
     int32(2), int32(2), int32(6);
     int32(2), int32(3), int32(6);
     int32(3), int32(1), int32(1);
     int32(3), int32(2), int32(1);
     int32(3), int32(3), int32(1);
     int32(3), int32(1), int32(2);
     int32(3), int32(2), int32(2);
     int32(3), int32(3), int32(2);
     int32(3), int32(1), int32(3);
     int32(3), int32(2), int32(3);
     int32(3), int32(3), int32(3);
     int32(3), int32(1), int32(4);
     int32(3), int32(2), int32(4);
     int32(3), int32(3), int32(4);
     int32(3), int32(1), int32(5);
     int32(3), int32(2), int32(5);
     int32(3), int32(3), int32(5);
     int32(3), int32(1), int32(6);
     int32(3), int32(2), int32(6);
     int32(3), int32(3), int32(6)];
y = [274;
     361;
     253;
     325;
     317;
     339;
     326;
     402;
     336;
     379;
     345;
     361;
     352;
     334;
     318;
     339;
     393;
     358;
     350;
     340;
     203;
     397;
     356;
     298;
     382;
     376;
     355;
     418;
     387;
     379;
     432;
     339;
     293;
     322;
```

```
     417;
     342;
     82;
     297;
     133;
     306;
     352;
     361;
     220;
     333;
     270;
     388;
     379;
     274;
     336;
     307;
     266;
     389;
     333;
     353];
wt = [];
table = zeros(18,1);
ncells = int32(0);
icount = [int32(8186736);
     int32(-16);
     int32(-1232427584);
     int32(-1208182168);
     int32(7);
     int32(8183760);
     int32(8142643);
     int32(10581210);
     int32(-1208182228);
     int32(8183732);
     int32(-1208182816);
     int32(0);
     int32(-1081228592);
     int32(8123961);
     int32(8264206);
     int32(10581250);
     int32(-1081228624);
     int32(32)];
auxt = zeros(36,1);
[tableOut, ncellsOut, ndim, idim, icountOut, auxtOut, ifail] = ...
      g11ba(stat, update, weight, isf, lfac, ifac, y, wt, table, ncells,
icount, auxt)
```

```
tableOut =
  235.3333
  342.6667
  309.3333
  395.0000
  373.3333
  350.0000
  332.6667
  341.6667
  370.3333
  370.3333
  326.6667
  381.0000
  196.3333
  332.6667
  320.3333
  338.0000
  292.3333
  351.0000
ncellsOut =
          18
ndim =
           2
idim =
```

```
            3
            6
            0
icountOut =
            3
            3
            3
            3
            3
            3
            3
            3
            3
            3
            3
            3
            3
            3
            3
            3
auxtOut =
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     3
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
ifail =
            0
```