NAG Toolbox for MATLAB

g13dm

1 Purpose

g13dm calculates the sample cross-correlation (or cross-covariance) matrices of a multivariate time series.

2 Syntax

```
[wmean, r0, r, ifail] = g13dm(matrix, k, m, w, 'n', n)
```

3 Description

Let $W_t = (w_{1t}, w_{2t}, \ldots, w_{kt})^{\mathrm{T}}$, for $t = 1, 2, \ldots, n$, denote n observations of a vector of k time series. The sample cross-covariance matrix at lag l is defined to be the k by k matrix $\hat{C}(l)$, whose (i,j)th element is given by

$$\hat{C}_{ij}(l) = \frac{1}{n}\sum_{t=l+1}^{n} \left(w_{i(t-l)} - \bar{w}_i\right)\left(w_{jt} - \bar{w}_j\right), \qquad l = 0, 1, 2, \ldots, m; i = 1, 2, \ldots, k; j = 1, 2, \ldots, k,$$

where $\bar{w}_i$ and $\bar{w}_j$ denote the sample means for the ith and jth series respectively. The sample cross-correlation matrix at lag l is defined to be the k by k matrix $\hat{R}(l)$, whose (i,j)th element is given by

$$\hat{R}_{ij}(l) = \frac{\hat{C}_{ij}(l)}{\sqrt{\hat{C}_{ii}(0)\hat{C}_{jj}(0)}}, \qquad l = 0, 1, 2, \ldots, m; i = 1, 2, \ldots, k; j = 1, 2, \ldots, k.$$

The number of lags, m, is usually taken to be at most $n/4$.

If W_t follows a vector moving average model of order q, then it can be shown that the theoretical cross-correlation matrices $(R(l))$ are zero beyond lag q. In order to help spot a possible cut-off point, the elements of $\hat{R}(l)$ are usually compared to their approximate standard error of $1/\sqrt{n}$. For further details see, for example, Wei 1990.

The function uses a single pass through the data to compute the means and the cross-covariance matrix at lag zero. The cross-covariance matrices at further lags are then computed on a second pass through the data.

4 References

Wei W W S 1990 *Time Series Analysis: Univariate and Multivariate Methods* Addison–Wesley

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: **matrix – string**

Indicates whether the cross-covariance or cross-correlation matrices are to be computed.

matrix = 'V'
The cross-covariance matrices are computed.

matrix = 'R'
The cross-correlation matrices are computed.

Constraint: **matrix** = 'V' or 'R'.

2: **k – int32 scalar**

k, the dimension of the multivariate time series.

Constraint: $\mathbf{k} \geq 1$.

3: **m – int32 scalar**

m, the number of cross-correlation (or cross-covariance) matrices to be computed. If in doubt set $\mathbf{m} = 10$. However it should be noted that **m** is usually taken to be at most $\mathbf{n}/4$.

Constraint: $1 \leq \mathbf{m} < \mathbf{n}$.

4: **w(kmax,n) – double array**

$\mathbf{w}(i,t)$ must contain the observation w_{it}, for $i = 1,2,\ldots,k$; $t = 1,2,\ldots,n$.

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

n, the number of observations in the series.

Constraint: $\mathbf{n} \geq 2$.

5.3 Input Parameters Omitted from the MATLAB Interface

kmax

5.4 Output Parameters

1: **wmean(k) – double array**

The means, $\bar{w}_i$, for $i = 1,2,\ldots,k$.

2: **r0(kmax,k) – double array**

If $i \neq j$, then $\mathbf{r0}(i,j)$ contains an estimate of the (i,j)th element of the cross-correlation (or cross-covariance) matrix at lag zero, $\hat{R}_{ij}(0)$; if $i = j$, then if **matrix** = 'V', $\mathbf{r0}(i,i)$ contains the variance of the ith series, $\hat{C}_{ii}(0)$, and if **matrix** = 'R', $\mathbf{r0}(i,i)$ contains the standard deviation of the ith series, $\sqrt{\hat{C}_{ii}(0)}$.

If **ifail** = 2 and **matrix** = 'R', then on exit all the elements in **r0** whose computation involves the zero variance are set to zero.

3: **r(kmax,kmax,m) – double array**

$\mathbf{r}(i,j,l)$ contains an estimate of the (i,j)th element of the cross-correlation (or cross-covariance) at lag l, $\hat{R}_{ij}(l)$, for $l = 1,2,\ldots,m$; $i = 1,2,\ldots,k$; $j = 1,2,\ldots,k$.

If **ifail** = 2 and **matrix** = 'R', then on exit all the elements in **r** whose computation involves the zero variance are set to zero.

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, **matrix** ≠ 'V' or 'R',
or **k** < 1,
or **n** < 2,
or **m** < 1,
or **m** ≥ **n**,
or **kmax** < **k**.

ifail = 2

On entry, at least one of the k series is such that all its elements are practically equal giving zero (or near zero) variance. In this case if **matrix** = 'R' all the correlations in **r0** and **r** involving this variance are set to zero.

7 Accuracy

For a discussion of the accuracy of the one-pass algorithm used to compute the sample cross-covariances at lag zero see West 1979. For the other lags a two-pass algorithm is used to compute the cross-covariances; the accuracy of this algorithm is also discussed in West 1979. The accuracy of the cross-correlations will depend on the accuracy of the computed cross-covariances.

8 Further Comments

The time taken is roughly proportional to mnk^2.

9 Example

```
matrix = 'R';
k = int32(2);
m = int32(10);
w = [-1.49, -1.62, 5.2, 6.23, 6.21, 5.86, 4.09, 3.18, 2.62, 1.49, 1.17,
...
     0.85, -0.35, 0.24, 2.44, 2.58, 2.04, 0.4, 2.26, 3.34, 5.09, 5, 4.78,
...
      4.11, 3.45, 1.65, 1.29, 4.09, 6.32, 7.5, 3.89, 1.58, 5.21, 5.25,
4.93, ...
     7.38, 5.87, 5.81, 9.68, 9.07, 7.29, 7.84, 7.55, 7.32, 7.97, 7.76, 7,
8.35;
      7.34, 6.35, 6.96, 8.539999999999999, 6.62, 4.97, 4.55, 4.81, 4.75,
...
     4.76, 10.88, 10.01, 11.62, 10.36, 6.4, 6.24, 7.93, 4.04, 3.73, 5.6,
...
     5.35, 6.81, 8.27, 7.68, 6.65, 6.08, 10.25, 9.140000000000001, 17.75,
13.3, ...
     9.630000000000001, 6.8, 4.08, 5.06, 4.94, 6.65, 7.94, 10.76, 11.89,
...
      5.85, 9.01, 7.5, 10.02, 10.38, 8.15, 8.369999999999999, 10.73,
12.14];
[wmean, r0, r, ifail] = g13dm(matrix, k, m, w)
```

```
wmean =
    4.3702
    7.8675
r0 =
    2.8176    0.2493
    0.2493    2.8149
r =
```

```
(:,:,1) =
    0.7359    0.1743
    0.2114    0.5546
(:,:,2) =
    0.4557    0.0764
    0.0693    0.2605
(:,:,3) =
    0.3792    0.0138
    0.0260   -0.0381
(:,:,4) =
    0.3224    0.1100
    0.0933   -0.2359
(:,:,5) =
    0.3411    0.2694
    0.0872   -0.2501
(:,:,6) =
    0.3631    0.3436
    0.1323   -0.2265
(:,:,7) =
    0.2800    0.4254
    0.2069   -0.1285
(:,:,8) =
    0.2480    0.5217
    0.1970   -0.0846
(:,:,9) =
    0.2398    0.2664
    0.2537    0.0745
(:,:,10) =
    0.1619   -0.0197
    0.2667    0.0047
ifail =
           0
```