NAG Toolbox for MATLAB

g13dp

1 Purpose

g13dp calculates the sample partial autoregression matrices of a multivariate time series. A set of likelihood ratio statistics and their significance levels are also returned. These quantities are useful for determining whether the series follows an autoregressive model and, if so, of what order.

2 Syntax

```
[maxlag, parlag, se, qq, x, pvalue, loglhd, ifail] = g13dp(k, z, m, 'n',
n)
```

3 Description

Let $W_t = (w_{1t}, w_{2t}, \ldots, w_{kt})^{\mathrm{T}}$, for $t = 1, 2, \ldots, n$, denote a vector of k time series. The partial autoregression matrix at lag l, P_l, is defined to be the last matrix coefficient when a vector autoregressive model of order l is fitted to the series. P_l has the property that if W_t follows a vector autoregressive model of order p then $P_l = 0$ for $l > p$.

Sample estimates of the partial autoregression matrices may be obtained by fitting autoregressive models of successively higher orders by multivariate least squares; see Tiao and Box 1981 and Wei 1990. These models are fitted using a QR algorithm based on the functions g02dc and g02df. They are calculated up to lag m, which is usually taken to be at most $n/4$.

The function also returns the asymptotic standard errors of the elements of $\hat{P}_l$ and an estimate of the residual variance-covariance matrix $\hat{\Sigma}_l$, for $l = 1, 2, \ldots, m$. If S_l denotes the residual sum of squares and cross-products matrix after fitting an AR(l) model to the series then under the null hypothesis $H_0 : P_l = 0$ the test statistic

$$X_l = -\left((n - m - 1) - \tfrac{1}{2} - lk\right) \log\left(\frac{|S_l|}{|S_{l-1}|}\right)$$

is asymptotically distributed as χ^2 with k^2 degrees of freedom. X_l provides a useful diagnostic aid in determining the order of an autoregressive model. (Note that $\hat{\Sigma}_l = S_l/(n - l)$.) The function also returns an estimate of the maximum of the log-likelihood function for each AR model that has been fitted.

4 References

Tiao G C and Box G E P 1981 Modelling multiple time series with applications *J. Am. Stat. Assoc.* **76** 802–816

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

5 Parameters

5.1 Compulsory Input Parameters

1: **k – int32 scalar**

k, the number of time series.

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

2: **z(kmax,n) – double array**

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

3: **m – int32 scalar**

m, the number of partial autoregression matrices to be computed. If in doubt set $\mathbf{m} = 10$.

Constraint: $\mathbf{m} \geq 1$ and $\mathbf{n} - \mathbf{m} - (\mathbf{k} \times \mathbf{m} + 1) \geq \mathbf{k}$.

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

n, the number of observations in the time series.

Constraint: $\mathbf{n} \geq 4$.

5.3 Input Parameters Omitted from the MATLAB Interface

kmax, work, lwork, iwork

5.4 Output Parameters

1: **maxlag – int32 scalar**

The maximum lag up to which partial autoregression matrices (along with their likelihood ratio statistics and their significance levels) have been successfully computed. On a successful exit **maxlag** will equal **m**. If $\mathbf{ifail} = 2$ on exit then **maxlag** will be less than **m**.

2: **parlag(kmax,kmax,m) – double array**

parlag(i,j,l) contains an estimate of the (i,j)th element of the partial autoregression matrix at lag l, $\hat{P}_l(ij)$, for $l = 1,2,\ldots,$**maxlag**; $i = 1,2,\ldots,k$; $j = 1,2,\ldots,k$.

3: **se(kmax,kmax,m) – double array**

se(i,j,l) contains an estimate of the standard error of the corresponding element in the array **parlag**.

4: **qq(kmax,kmax,m) – double array**

qq(i,j,l) contains an estimate of the (i,j)th element of the corresponding variance-covariance matrix $\hat{\Sigma}_l$, for $l = 1,2,\ldots,$**maxlag**; $i = 1,2,\ldots,k$; $j = 1,2,\ldots,k$.

5: **x(m) – double array**

x(l) contains X_l, the likelihood ratio statistic at lag l, for $l = 1,2,\ldots,$**maxlag**.

6: **pvalue(m) – double array**

pvalue(l) contains the significance level of the statistic in the corresponding element of **x**.

7: **loglhd(m) – double array**

loglhd(l) contains an estimate of the maximum of the log-likelihood function when an AR(l) model has been fitted to the series for $l = 1,2,\ldots,$**maxlag**.

8: **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, **k** < 1,
or **n** < 4,
or **kmax** < **k**,
or **m** < 1,
or **n** − **m** − (**k** × **m** + 1) < **k**,
or **lwork** is too small.

ifail = 2

The recursive equations used to compute the sample partial autoregression matrices have broken down at lag **maxlag** + 1. This exit could occur if the regression model is overparameterised. For your settings of k and n the value returned by **maxlag** is the largest permissible value of m for which the model is not overparameterised. All output quantities in the arrays **parlag**, **se**, **qq**, **x**, **pvalue** and **loglhd** up to and including lag **maxlag** will be correct.

7 Accuracy

The computations are believed to be stable.

8 Further Comments

The time taken is roughly proportional to nmk.

For each order of autoregressive model that has been estimated, g13dp returns the maximum of the log-likelihood function. An alternative means of choosing the order of a vector AR process is to choose the order for which Akaike's information criterion is smallest. That is, choose the value of l for which $-2 \times \textbf{loglhd}(l) + 2lk^2$ is smallest. You should be warned that this does not always lead to the same choice of l as indicated by the sample partial autoregression matrices and the likelihood ratio statistics.

9 Example

```
k = int32(2);
z = [-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];
m = int32(10);
[maxlag, parlag, se, qq, x, pvalue, loglhd, ifail] = g13dp(k, z, m)
```

```
maxlag =
          10
```

```
parlag =
(:,:,1) =
    0.7568    0.0617
    0.0608    0.5703
(:,:,2) =
   -0.1614   -0.1348
   -0.0925   -0.0646
(:,:,3) =
    0.2373    0.0444
    0.0474   -0.2477
(:,:,4) =
   -0.0976    0.1517
    0.4018   -0.1940
(:,:,5) =
    0.2566   -0.0264
    0.3997   -0.0213
(:,:,6) =
   -0.0754    0.1125
    0.1963   -0.1056
(:,:,7) =
   -0.0542    0.0970
    0.5745   -0.0800
(:,:,8) =
    0.1474    0.0411
    0.9158   -0.2422
(:,:,9) =
   -0.0388    0.0993
   -0.4996    0.1725
(:,:,10) =
    0.1887    0.1310
   -0.1826   -0.0398
se =
(:,:,1) =
    0.0915    0.0919
    0.1292    0.1297
(:,:,2) =
    0.1448    0.1088
    0.2132    0.1602
(:,:,3) =
    0.1278    0.0950
    0.2219    0.1650
(:,:,4) =
    0.1341    0.0987
    0.2277    0.1675
(:,:,5) =
    0.1405    0.1064
    0.2416    0.1830
(:,:,6) =
    0.1557    0.1111
    0.2693    0.1921
(:,:,7) =
    0.1658    0.1209
    0.2672    0.1948
(:,:,8) =
    0.1883    0.1276
    0.2460    0.1667
(:,:,9) =
    0.2511    0.1401
    0.3237    0.1806
(:,:,10) =
    0.2750    0.1571
    0.3715    0.2122
qq =
(:,:,1) =
    2.7313    0.6060
    0.6060    5.4403
(:,:,2) =
    2.5303    0.4819
    0.4819    5.4859
(:,:,3) =
```

```
     1.7553     0.5144
     0.5144     5.2914
(:,:,4) =
     1.6610     0.7644
     0.7644     4.7856
(:,:,5) =
     1.5040     0.5895
     0.5895     4.4473
(:,:,6) =
     1.4804     0.6681
     0.6681     4.4254
(:,:,7) =
     1.4781     0.7478
     0.7478     3.8376
(:,:,8) =
     1.4150     0.5246
     0.5246     2.4155
(:,:,9) =
     1.3217     0.4804
     0.4804     2.1964
(:,:,10) =
     1.2062     0.5765
     0.5765     2.2005
x =
   49.8836
    3.3467
   13.9619
    7.0706
    5.1838
    2.0826
    5.0744
   10.9907
    3.9365
    3.1748
pvalue =
    0.0000
    0.5016
    0.0074
    0.1322
    0.2690
    0.7206
    0.2797
    0.0267
    0.4147
    0.5290
loglhd =
 -196.2102
 -190.6563
 -177.2009
 -168.7939
 -161.7405
 -157.1822
 -149.8045
 -136.4157
 -129.8435
 -123.8453
ifail =
           0
```