NAG Toolbox

nag_tsa_inhom_ma (g13mg)

1 Purpose

nag_tsa_inhom_ma (g13mg) provides a moving average, moving norm, moving variance and moving standard deviation operator for an inhomogeneous time series.

2 Syntax

```
[ma, p, pn, wma, rcomm, ifail] = nag_tsa_inhom_ma(ma, t, tau, m1, m2, sinit,
inter, ftype, p, 'nb', nb, 'pn', pn, 'rcomm', rcomm)
```

```
[ma, p, pn, wma, rcomm, ifail] = g13mg(ma, t, tau, m1, m2, sinit, inter, ftype,
p, 'nb', nb, 'pn', pn, 'rcomm', rcomm)
```

3 Description

nag_tsa_inhom_ma (g13mg) provides a number of operators for an inhomogeneous time series. The time series is represented by two vectors of length n; a vector of times, t; and a vector of values, z. Each element of the time series is therefore composed of the pair of scalar values (t_i, z_i), for $i = 1, 2, \ldots, n$. Time t can be measured in any arbitrary units, as long as all elements of t use the same units.

The main operator available, the moving average (MA), with parameter τ is defined as

$$\mathrm{MA}[\tau, m_1, m_2; y](t_i) = \frac{1}{m_2 - m_1 + 1} \sum_{j=m_1}^{m_2} \mathrm{EMA}[\tilde{\tau}, j; y](t_i) \quad (1)$$

where $\tilde{\tau} = \frac{2\tau}{m_2+m_1}$, m_1 and m_2 are user-supplied integers controlling the amount of lag and smoothing respectively, with $m_2 \geq m_1$ and $\mathrm{EMA}[\cdot]$ is the iterated exponential moving average operator.

The iterated exponential moving average, $\mathrm{EMA}[\tilde{\tau}, m; y](t_i)$, is defined using the recursive formula:

$$\mathrm{EMA}[\tilde{\tau}, m; y](t_i) = \mathrm{EMA}[\tilde{\tau}; \mathrm{EMA}[\tilde{\tau}, m-1; y](t_i)](t_i)$$

with

$$\mathrm{EMA}[\tilde{\tau}, 1; y](t_i) = \mathrm{EMA}[\tilde{\tau}; y](t_i)$$

and

$$\mathrm{EMA}[\tilde{\tau}; y](t_i) = \mu \mathrm{EMA}[\tilde{\tau}; y](t_{i-1}) + (\nu - \mu) y_{i-1} + (1 - \nu) y_i$$

where

$$\mu = e^{-\alpha} \quad \text{and} \quad \alpha = \frac{t_i - t_{i-1}}{\tilde{\tau}}.$$

The value of ν depends on the method of interpolation chosen and the relationship between y and the input series z depends on the transformation function chosen. nag_tsa_inhom_ma (g13mg) gives the option of three interpolation methods:

1. Previous point: $\nu = 1$.
2. Linear: $\nu = (1 - \mu)/\alpha$.
3. Next point: $\nu = \mu$.

and three transformation functions:

1. Identity: $y_i = z_i^{[p]}$.
2. Absolute value: $y_i = |z_i|^p$.
3. Absolute difference: $y_i = |z_i - \text{MA}[\tau, m_1, m_2; z](t_i)|^p$.

where the notation $[p]$ is used to denote the integer nearest to p. In addition, if either the absolute value or absolute difference transformation are used then the resulting moving average can be scaled by p^{-1}.

The various parameter options allow a number of different operators to be applied by nag_tsa_inhom_ma (g13mg), a few of which are:

(i) **Moving Average (MA)**, as defined in (1) (obtained by setting **ftype** $= 1$ and **p** $= 1$).

(ii) **Moving Norm (MNorm)**, defined as

$$\text{MNorm}(\tau, m, p; z) = \text{MA}[\tau, 1, m; |z|^p]^{1/p}$$

(obtained by setting **ftype** $= 4$, **m1** $= 1$ and **m2** $= m$).

(iii) **Moving Variance (MVar)**, defined as

$$\text{MVar}(\tau, m, p; z) = \text{MA}[\tau, 1, m; |z - \text{MA}[\tau, 1, m; z]|^p]$$

(obtained by setting **ftype** $= 3$, **m1** $= 1$ and **m2** $= m$).

(iv) **Moving Standard Deviation (MSD)**, defined as

$$\text{MSD}(\tau, m, p; z) = \text{MA}[\tau, 1, m; |z - \text{MA}[\tau, 1, m; z]|^p]^{1/p}$$

(obtained by setting **ftype** $= 5$, **m1** $= 1$ and **m2** $= m$).

For large datasets or where all the data is not available at the same time, z and t can be split into arbitrary sized blocks and nag_tsa_inhom_ma (g13mg) called multiple times.

4 References

Dacorogna M M, Gencay R, MÏller U, Olsen R B and Pictet O V (2001) *An Introduction to High-frequency Finance* Academic Press

Zumbach G O and MÏller U A (2001) Operators on inhomogeneous time series *International Journal of Theoretical and Applied Finance* **4(1)** 147–178

5 Parameters

5.1 Compulsory Input Parameters

1: **ma**(**nb**) – REAL (KIND=nag_wp) array

z_i, the current block of observations, for $i = k + 1, \ldots, k + b$, where k is the number of observations processed so far, i.e., the value supplied in **pn** on entry.

2: **t**(**nb**) – REAL (KIND=nag_wp) array

t_i, the times for the current block of observations, for $i = k + 1, \ldots, k + b$, where k is the number of observations processed so far, i.e., the value supplied in **pn** on entry.

If $t_i \leq t_{i-1}$, **ifail** $= 31$ will be returned, but nag_tsa_inhom_ma (g13mg) will continue as if t was strictly increasing by using the absolute value. The lagged difference, $t_i - t_{i-1}$ must be sufficiently small that $e^{-\alpha}$, $\alpha = (t_i - t_{i-1})/\tilde{\tau}$ can be calculated without overflowing, for all i.

3: **tau** – REAL (KIND=nag_wp)

τ, the parameter controlling the rate of decay. τ must be sufficiently large that $e^{-\alpha}$, $\alpha = (t_i - t_{i-1})/\tilde{\tau}$ can be calculated without overflowing, for all i, where $\tilde{\tau} = \frac{2\tau}{m_2 + m_1}$.

Constraint: **tau** > 0.0.

4: **m1** – INTEGER

m_1, the iteration of the EMA operator at which the sum is started.

Constraint: $\mathbf{m1} \geq 1$.

5: **m2** – INTEGER

m_2, the iteration of the EMA operator at which the sum is ended.

Constraint: $\mathbf{m2} \geq \mathbf{m1}$.

6: **sinit**(:) – REAL (KIND=nag_wp) array

The dimension of the array **sinit** must be at least $2 \times \mathbf{m2} + 3$ if $\mathbf{ftype} = 3$ or 5, and at least $\mathbf{m2} + 2$ otherwise

If $\mathbf{pn} = 0$, the values used to start the iterative process, with

$\mathbf{sinit}(1) = t_0$,

$\mathbf{sinit}(2) = y_0$,

$\mathbf{sinit}(j+2) = \mathrm{EMA}[\tau, j; y](t_0)$, for $i = 1, 2, \ldots, \mathbf{m2}$.

In addition, if $\mathbf{ftype} = 3$ or 5 then

$\mathbf{sinit}(\mathbf{m2} + 3) = z_0$,

$\mathbf{sinit}(\mathbf{m2} + j + 2) = \mathrm{EMA}[\tau, j; z](t_0)$, for $j = 1, 2, \ldots, \mathbf{m2}$.

i.e., initial values based on the original data z as opposed to the transformed data y.

If $\mathbf{pn} \neq 0$, **sinit** is not referenced.

Constraint: if $\mathbf{ftype} \neq 1$, $\mathbf{sinit}(j) \geq 0$, for $j = 2, 3, \ldots, \mathbf{m2} + 2$.

7: **inter**(**2**) – INTEGER array

The type of interpolation used with **inter**(1) indicating the interpolation method to use when calculating $\mathrm{EMA}[\tau, 1; z]$ and **inter**(2) the interpolation method to use when calculating $\mathrm{EMA}[\tau, j; z]$, $j > 1$.

Three types of interpolation are possible:

$\mathbf{inter}(i) = 1$
Previous point, with $\nu = 1$.

$\mathbf{inter}(i) = 2$
Linear, with $\nu = (1 - \mu)/\alpha$.

$\mathbf{inter}(i) = 3$
Next point, $\nu = \mu$.

Zumbach and MÏller (2001) recommend that linear interpolation is used in second and subsequent iterations, i.e., $\mathbf{inter}(2) = 2$, irrespective of the interpolation method used at the first iteration, i.e., the value of **inter**(1).

Constraint: $\mathbf{inter}(i) = 1$, 2 or 3, for $i = 1, 2$.

8: **ftype** – INTEGER

The function type used to define the relationship between y and z when calculating $\mathrm{EMA}[\tau, 1; y]$. Three functions are provided:

$\mathbf{ftype} = 1$
The identity function, with $y_i = z_i^{[p]}$.

$\mathbf{ftype} = 2$ or 4
The absolute value, with $y_i = |z_i|^p$.

ftype $= 3$ or 5
The absolute difference, with $y_i = |z_i - \mathrm{MA}[\tau, m; y](t_i)|^p$.

If **ftype** $= 4$ or 5 then the resulting vector of averages is scaled by p^{-1} as described in **ma**.

Constraint: **ftype** $= 1$, 2, 3, 4 or 5.

9: **p** – REAL (KIND=nag_wp)

p, the power used in the transformation function.

Constraint: $\mathbf{p} \neq 0$.

5.2 Optional Input Parameters

1: **nb** – INTEGER

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

b, the number of observations in the current block of data. At each call the size of the block of data supplied in **ma** and **t** can vary; therefore **nb** can change between calls to nag_tsa_inhom_ma (g13mg).

Constraint: $\mathbf{nb} \geq 0$.

2: **pn** – INTEGER

Default: 0

k, the number of observations processed so far. On the first call to nag_tsa_inhom_ma (g13mg), or when starting to summarise a new dataset, **pn** must be set to 0. On subsequent calls it must be the same value as returned by the last call to nag_tsa_inhom_ma (g13mg).

Constraint: $\mathbf{pn} \geq 0$.

3: **rcomm**($lrcomm$) – REAL (KIND=nag_wp) array

Communication array, used to store information between calls to nag_tsa_inhom_ma (g13mg). On the first call to nag_tsa_inhom_ma (g13mg), or if all the data is provided in one go, **rcomm** need not be provided.

5.3 Output Parameters

1: **ma**(**nb**) – REAL (KIND=nag_wp) array

The moving average:

if **ftype** $= 4$ or 5
$\mathbf{ma}(i) = \{\mathrm{MA}[\tau, m_1, m_2; y](t_i)\}^{1/p}$,

otherwise
$\mathbf{ma}(i) = \mathrm{MA}[\tau, m_1, m_2; y](t_i)$.

2: **p** – REAL (KIND=nag_wp)

If **ftype** $= 1$, then $[p]$, the actual power used in the transformation function is returned, otherwise **p** is unchanged.

3: **pn** – INTEGER

Default: 0

$k + b$, the updated number of observations processed so far.

4: **wma**(**nb**) – REAL (KIND=nag_wp) array

Either the moving average or exponential moving average, depending on the value of **ftype**.

if **ftype** $= 3$ or 5
  wma$(i) = \mathrm{MA}[\tau; y](t_i)$

otherwise
  wma$(i) = \mathrm{EMA}[\tilde{\tau}; y](t_i)$.

5: **rcomm**$(lrcomm)$ – REAL (KIND=nag_wp) array

Communication array, used to store information between calls to nag_tsa_inhom_ma (g13mg).

6: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 11$

Constraint: **nb** ≥ 0.

ifail $= 31$ (*warning*)

Constraint: **t** should be strictly increasing.

ifail $= 32$

Constraint: **t**$(i) \neq$ **t**$(i-1)$ if linear interpolation is being used.

ifail $= 41$

Constraint: **tau** > 0.0.

ifail $= 42$

Constraint: if **pn** > 0 then **tau** must be unchanged since previous call.

ifail $= 51$

Constraint: **m1** ≥ 1.

ifail $= 52$

Constraint: if **pn** > 0 then **m1** must be unchanged since previous call.

ifail $= 61$

Constraint: **m2** $\geq$ **m1**.

ifail $= 62$

Constraint: if **pn** > 0 then **m2** must be unchanged since previous call.

ifail $= 71$

Constraint: if **ftype** $\neq 1$, **sinit**$(j) \geq 0.0$, for $j = 2, 3, \ldots,$ **m2** $+ 2$.

ifail $= 81$

Constraint: **inter**$(1) = 1$, 2 or 3.

ifail $= 82$

Constraint: **inter**$(2) = 1$, 2 or 3.

ifail $= 83$

Constraint: if **pn** $\neq 0$, **inter** must be unchanged since the last call.

ifail $= 91$

Constraint: **ftype** $= 1$, 2, 3, 4 or 5.

ifail $= 92$

Constraint: if **pn** $\neq 0$, **ftype** must be unchanged since the previous call.

ifail $= 101$

Constraint: absolute value of **p** must be representable as an integer.

ifail $= 102$

Constraint: if **ftype** $\neq 1$, **p** $\neq 0.0$. If **ftype** $= 1$, the nearest integer to **p** must not be 0.

ifail $= 103$

Constraint: if **ftype** $= 1$, 2 or 4 and **ma**$(i) = 0$ for any i then **p** > 0.0.

ifail $= 104$

Constraint: if **p** < 0.0, **ma**(i) − **wma**$(i) \neq 0.0$, for any i.

ifail $= 105$

Constraint: if **pn** > 0 then **p** must be unchanged since previous call.

ifail $= 111$

Constraint: **pn** ≥ 0.

ifail $= 112$

Constraint: if **pn** > 0 then **pn** must be unchanged since previous call.

ifail $= 131$

rcomm has been corrupted between calls.

ifail $= 141$

Constraint: if **pn** $= 0$, $lrcomm = 0$ or $lrcomm \geq 2$**m2** $+ 20$.

ifail $= 142$

Constraint: if **pn** $\neq 0$, $lrcomm \geq 2$**m2** $+ 20$.

ifail $= 301$ (*warning*)

Truncation occurred to avoid overflow, check for extreme values in **t**, **ma** or for **tau**. Results are returned using the truncated values.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

Not applicable.

8 Further Comments

Approximately $4m_2$ real elements are internally allocated by nag_tsa_inhom_ma (g13mg). If **ftype** $= 3$ or 5 then a further **nb** real elements are also allocated.

The more data you supply to nag_tsa_inhom_ma (g13mg) in one call, i.e., the larger **nb** is, the more efficient the function will be, particularly if the function is being run using more than one thread.

Checks are made during the calculation of α and y_i to avoid overflow. If a potential overflow is detected the offending value is replaced with a large positive or negative value, as appropriate, and the calculations performed based on the replacement values. In such cases **ifail** $= 301$ is returned. This should not occur in standard usage and will only occur if extreme values of **ma**, **t** or **tau** are supplied.

9 Example

The example reads in a simulated time series, (t, z) and calculates the moving average. The data is supplied in three blocks of differing sizes.

9.1 Program Text

```
function g13mg_example

fprintf('g13mg example results\n\n');

m1 = nag_int(1);
m2 = nag_int(2);
ftype = nag_int(1);
p = 1;
inter = [nag_int(3); 2];
tau = 2;
sinit = zeros(8, 1);
nb = [5, 10, 15];
rcomm = zeros(20+2*m2, 1);
t = cell(3, 1);
z = cell(3, 1);
t{1}  = [ 7.5;  8.2; 18.1; 22.8; 25.8];
ma{1} = [ 0.6;  0.6;  0.8;  0.1;  0.2];
t{2}  = [26.8; 31.1; 38.4; 45.9; 48.2; 48.9; 57.9; 58.5; 63.9; 65.2];
ma{2} = [ 0.2;  0.5;  0.7;  0.1;  0.4;  0.7;  0.8;  0.3;  0.2;  0.5];
t{3}  = [66.6; 67.4; 69.3; 69.9; 73.0; 75.6; 77.0; 84.7; 86.8; 88.0; ...
         88.5; 91.0; 93.0; 93.7; 94.0];
ma{3} = [ 0.2;  0.3;  0.8;  0.6;  0.1;  0.7;  0.9;  0.6;  0.3;  0.1;  ...
          0.1;  0.4;  1.0;  1.0;  0.1];

fprintf('             Time           MA\n');

% Loop over each block of data.
for i = 1:numel(nb)
  if i == 1
    % Initialise the moving average operator for this block of data
    [ma{i}, p, pn, wma, rcomm, ifail] = ...
    g13mg( ...
           ma{i}, t{i}, tau, m1, m2, sinit, inter, ftype, p, 'rcomm', rcomm);
  else
```

```
    % Update the moving average operator for this block of data
    [ma{i}, p, pn, wma, rcomm, ifail] = ...
    g13mg( ...
           ma{i}, t{i}, tau, m1, m2, sinit, inter, ftype, p, ...
           'pn', pn, 'rcomm', rcomm);
  end

  % Display the results for this block of data
  for j=1:nb(i)
    fprintf('%3d    %10.1f    %10.3f\n', pn-nb(i)+j, t{i}(j), ma{i}(j));
  end
  fprintf('\n');
end
```

9.2 Program Results

```
     g13mg example results

             Time          MA
  1           7.5         0.545
  2           8.2         0.567
  3          18.1         0.786
  4          22.8         0.214
  5          25.8         0.187

  6          26.8         0.192
  7          31.1         0.444
  8          38.4         0.680
  9          45.9         0.155
 10          48.2         0.298
 11          48.9         0.406
 12          57.9         0.777
 13          58.5         0.677
 14          63.9         0.258
 15          65.2         0.351

 16          66.6         0.291
 17          67.4         0.289
 18          69.3         0.572
 19          69.9         0.593
 20          73.0         0.244
 21          75.6         0.532
 22          77.0         0.715
 23          84.7         0.618
 24          86.8         0.426
 25          88.0         0.284
 26          88.5         0.240
 27          91.0         0.332
 28          93.0         0.723
 29          93.7         0.814
 30          94.0         0.744
```