The function `mcov` computes estimates of the lag 1 moving cross-covariance matrix of non-stationary (and stationary) time series. Notice that the following library is needed to be installed before using the `mcov` function:

```r
library(roll)
```

Usage

```r
mcov(x,w,l)
```

Arguments

- `x`: a T-by-m data matrix, where the rows are "T" time points, and the columns are "m" variables
- `w`: window width (i.e. window size) at which the `mcov` is calculated
- `l`: the lag at which the `mcov` is calculated

Value

- `mcov`: a symmetric m-by-m matrix, which is the lag 1 moving cross-covariance matrix of the data matrix `x`

Note

Choosing the window width (i.e. `w`) is vital to enhance the results of the `mcov` to extract accurate information from the data. The size of `w` depends on the degree of stationarity of the data. Small window sizes are suitable for data that exhibit strong non-stationarity. For stationary data, a window of size "w=T-2" is used.

Author(s)

Fayed Alshammri

References


Examples

```r
##The data matrix X is a non-stationary time series with m=6 and T=1500.
## Generate x_t
m=6;T=1500
# Generate x_t
X=mat.or.vec(m,T)
a1=arima.sim(list(order=c(1,1,1),ar=0.75,ma=0.9),n=T+1,sd=1)
for(i in 1:2) X[i,]=a1[i+1:T]
a2=arima.sim(list(order=c(1,1,1),ar=0.6,ma=-1.4),n=T+1,sd=1)
for(i in 3:4) X[i,]=a2[(i-1):(T+i-2)]
a3=arima.sim(list(order=c(1,1,1),ar=-0.7,ma=-2.3),n=T+1,sd=1)
for(i in 5:6) X[i,]=a3[(i-3):(T+i-4)]
X=t(X)
X=ts(X)
##calculate the lag 1 moving cross-covariance of x, with w=100 and l=2.
myresult=mcov(X,100,2)
myresult
```

[Package `mcov` version 0.1.0 Index]