Package ‘wavemulcor’

February 14, 2011

Type Package

Title Wavelet routine for multiple correlation

Version 1.2

Date 2011-02-13

Author Javier Fernandez-Macho (UPV/EHU)

Maintainer Javier Fernandez-Macho <javier.fernandezmacho@ehu.es>

Depends waveslim

Description Wavelet routines that calculate single sets of wavelet multiple correlations and cross-correlations out of n variables (either 1D time series, 2D images or 3D arrays). They can later be plotted in single graphs, as an alternative to trying to make sense out of several sets of wavelet correlations or wavelet cross-correlations. The code is based on the calculation, at each wavelet scale, of the square root of the coefficient of determination in a linear combination of variables for which such coefficient of determination is a maximum. The code provided here is based on the wave.correlation routine in Brandon Whitcher's waveslim R package Version: 1.6.4, which in turn is based on wavelet methodology developed in Percival and Walden (2000); Gencay,Selcuk and Whitcher (2001) and others.

License GPL (>= 2)

LazyLoad yes

R topics documented:

`wavemulcor-package` *Wavelet routine for multiple correlation*

Description

Produces an estimate of the multiscale multiple correlation (as defined below) along with approximate confidence intervals.

Details

Package:	wavemulcor
Type:	Package
Version:	1.2
Date:	2011-02-13
License:	GPL (>= 2)
LazyLoad:	yes

The *wavemulcor* package contains two routines, *wave.multiple.correlation* and *wave.multiple.cross.correlation*, that calculate single sets of, respectively, wavelet multiple correlations and wavelet multiple cross-correlations out of n variables. They can later be plotted in single graphs, as an alternative to trying to make sense out of $n(n-1)/2$ sets of wavelet correlations or $n(n-1)/2 \times J$ sets of wavelet cross-correlations. The code is based on the calculation, at each wavelet scale, of the square root of the coefficient of determination in a linear combination of variables for which such coefficient of determination is a maximum.

Note

Dependencies: waveslim

Author(s)

Javier Fernández-Macho (UPV/EHU)
Maintainer: Javier Fernández-Macho <javier.fernandezmacho@ehu.es>

References

Fernández-Macho, Javier (2011) The wavelet multiple correlation, (mimeo).
Gencay, R., F. Selcuk and B. Whitcher (2001) *An Introduction to Wavelets and Other Filtering Methods in Finance and Economics*, Academic Press.
Whitcher, B. (2010) *'waveslim' R Package.*

`wave.multiple.correlation` *Wavelet routine for multiple correlation*

Description

Produces an estimate of the multiscale multiple correlation (as defined below) along with approximate confidence intervals.

Usage

```
wave.multiple.correlation(xx, N, p = 0.975, ymaxr=NULL)
```

Arguments

`xx`	A list of n (multiscaled) time series, usually the outcomes of dwt or modwt, *i.e.* xx <- list(v1.modwt.bw, v2.modwt.bw, v3.modwt.bw)
`N`	length of the time series
`p`	one minus the two-sided p-value for the confidence interval, *i.e.* the cdf value.
`ymaxr`	index number of the variable whose correlation is calculated against a linear combination of the rest, otherwise at each wavelet level wmc chooses the one maximizing the multiple correlation.

Details

The routine calculates one single set of wavelet multiple correlations out of n variables that can be plotted in a single graph, as an alternative to trying to make sense out of $n(n-1)/2$ sets of wavelet correlations. The code is based on the calculation, at each wavelet scale, of the square root of the coefficient of determination in the linear combination of variables for which such coefficient of determination is a maximum. The code provided here is based on the wave.correlation routine in Brandon Whitcher's *waveslim* R package Version: 1.6.4, which in turn is based on wavelet methodology developed in Percival and Walden (2000); Gencay, Selcuk and Whitcher (2001) and others.

Value

List of two elements:
xy.mulcor: matrix with as many rows as levels in the wavelet transform object. The first column provides the point estimate for the wavelet multiple correlation, followed by the lower and upper bounds from the confidence interval.
YmaxR: numeric vector giving, at each wavelet level, the index number of the variable whose correlation is calculated against a linear combination of the rest. By default, *wmc* chooses at each wavelet level the variable maximizing the multiple correlation.

Note

Needs *waveslim* package to calculate *dwt* or *modwt* coefficients as inputs to the routine (also for data in the example).

Author(s)

Javier Fernández-Macho, Dpt. of Econometrics and Statistics, & Instituto de Economía Pública, University of the Basque Country, Agirre Lehendakari etorb. 83, E48015 BILBAO, Spain. (email: javier.fernandezmacho@ehu.es).

References

Fernández-Macho, Javier (2011) The wavelet multiple correlation, (mimeo).

Examples

```
## Based on data from Figure 7.8 in Gencay, Selcuk and Whitcher (2001)
## plus one random series.

library(wavemulcor)
data(exchange)
returns <- diff(log(as.matrix(exchange)))
returns <- ts(returns, start=1970, freq=12)
wf <- "d4"
J <- 6

demusd.modwt <- modwt(returns[,"DEM.USD"], wf, J)
demusd.modwt.bw <- brick.wall(demusd.modwt, wf)
jpyusd.modwt <- modwt(returns[,"JPY.USD"], wf, J)
jpyusd.modwt.bw <- brick.wall(jpyusd.modwt, wf)
rand.modwt <- modwt(rnorm(length(returns[,"DEM.USD"])), wf, J)
rand.modwt.bw <- brick.wall(rand.modwt, wf)

xx <- list(demusd.modwt.bw, jpyusd.modwt.bw, rand.modwt.bw)

Lst <- wave.multiple.correlation(xx, N = length(xx[[1]][[1]]))
returns.modwt.cor <- Lst$xy.mulcor[1:J,]
YmaxR <- Lst$YmaxR

exchange.names <- c("DEM.USD", "JPY.USD", "RAND")

##Producing plot

par(mfrow=c(1,1), las=0, mar=c(5,4,4,2)+.1)
matplot(2^(0:(J-1)), returns.modwt.cor[-(J+1),], type="b",
  log="x", pch="*LU", xaxt="n", lty=1, col=c(1,4,4),
  xlab="Wavelet Scale", ylab="Wavelet Multiple Correlation")
axis(side=1, at=2^(0:7))
abline(h=0)
text(2^(0:7), min(returns.modwt.cor[-(J+1),])-0.03,
  labels=exchange.names[YmaxR], adj=0.5, cex=.5)
```

`wave.multiple.cross.correlation`
Wavelet routine for multiple cross-correlation

Description

Produces an estimate of the multiscale multiple cross-correlation (as defined below).

Usage

```
wave.multiple.cross.correlation(xx, lag.max = NULL, ymaxr = NULL)
```

Arguments

`xx`	A list of n (multiscaled) time series, usually the outcomes of dwt or modwt, *i.e.* xx <- list(v1.modwt.bw, v2.modwt.bw, v3.modwt.bw)

`lag.max`	maximum lag. If not set, it defaults to half the square root of the length of the original series.
`ymaxr`	index number of the variable whose correlation is calculated against a linear combination of the rest, otherwise at each wavelet level wmc chooses the one maximizing the multiple correlation.

Details

The routine calculates one single set of wavelet multiple cross-correlations out of n variables that can be plotted as one single set of graphs (one per wavelet level), as an alternative to trying to make sense out of $n(n-1)/2 \times J$ sets of wavelet cross-correlations. The code is based on the calculation, at each wavelet scale, of the square root of the coefficient of determination in a linear combination of variables that includes a lagged variable for which such coefficient of determination is a maximum.

Value

List of two elements:
xy.mulcor: matrix with as many rows as levels in the wavelet transform object. The columns provide the point estimates for the wavelet multiple cross-correlations at different lags.
YmaxR: numeric vector giving, at each wavelet level, the index number of the variable whose correlation is calculated against a linear combination of the rest. By default, *wmcc* chooses at each wavelet level the variable maximizing the multiple correlation.

Note

Needs *waveslim* package to calculate *dwt* or *modwt* coefficients as inputs to the routine (also for data in the example).

Author(s)

Javier Fernández-Macho, Dpt. of Econometrics and Statistics, & Instituto de Economía Pública, University of the Basque Country, Agirre Lehendakari etorb. 83, E48015 BILBAO, Spain. (email: javier.fernandezmacho@ehu.es).

References

Fernández-Macho, Javier (2011) The wavelet multiple correlation, (mimeo).

Examples

```
## Based on data from Figure 7.9 in Gencay, Selcuk and Whitcher (2001)
## plus one random series.

library(wavemulcor)
data(exchange)
returns <- diff(log(exchange))
returns <- ts(returns, start=1970, freq=12)
wf <- "d4"
J <- 6
lmax <- 36
n <- dim(returns)[1]

demusd.modwt <- modwt(returns[,"DEM.USD"], wf, J)
demusd.modwt.bw <- brick.wall(demusd.modwt, wf)
```

```
jpyusd.modwt <- modwt(returns[,"JPY.USD"], wf, J)
jpyusd.modwt.bw <- brick.wall(jpyusd.modwt, wf)
rand.modwt <- modwt(rnorm(length(returns[,"DEM.USD"])), wf, J)
rand.modwt.bw <- brick.wall(rand.modwt, wf)

##xx <- list(demusd.modwt.bw, jpyusd.modwt.bw)
xx <- list(demusd.modwt.bw, jpyusd.modwt.bw, rand.modwt.bw)

Lst <- wave.multiple.cross.correlation(xx, lmax)
returns.cross.cor <- as.matrix(Lst$xy.mulcor[1:J,])
YmaxR <- Lst$YmaxR

exchange.names <- c("DEM.USD", "JPY.USD", "RAND")
rownames(returns.cross.cor)<-rownames(returns.cross.cor,
  do.NULL = FALSE, prefix = "Level ")
lags <- length(-lmax:lmax)

lower.ci <- tanh(atanh(returns.cross.cor) - qnorm(0.975) /
sqrt(matrix(trunc(n/2^(1:J)), nrow=J, ncol=lags)- 3))
upper.ci <- tanh(atanh(returns.cross.cor) + qnorm(0.975) /
sqrt(matrix(trunc(n/2^(1:J)), nrow=J, ncol=lags)- 3))

par(mfrow=c(3,2), las=1, pty="m", mar=c(2,3,1,0)+.1, oma=c(1.2,1.2,0,0))
for(i in J:1) {
matplot((1:(2*lmax+1)),returns.cross.cor[i,], type="l", lty=1, ylim=c(-1,1),
  xaxt="n", xlab="", ylab="", main=rownames(returns.cross.cor)[[i]][1])
if(i<3) {axis(side=1, at=seq(1, 2*lmax+1, by=12),
  labels=seq(-lmax, lmax, by=12))}
#axis(side=2, at=c(-.2, 0, .5, 1))
lines(lower.ci[i,], lty=1, col=2) ##Add Connected Line Segments to a Plot
lines(upper.ci[i,], lty=1, col=2)
abline(h=0,v=lmax+1)              ##Add Straight horiz and vert Lines to a Plot
text(1,1, labels=exchange.names[YmaxR[i]], adj=0.25, cex=.8)
}
par(las=0)
mtext('Lag (months)', side=1, outer=TRUE, adj=0.5)
mtext('Wavelet Multiple Cross-Correlation', side=2, outer=TRUE, adj=0.5)
```

Index