Package ‘quantilogram’

October 13, 2022

Title  Cross-Quantilogram
Version  2.2.1
Maintainer  Tatsushi Oka <oka.econ@gmail.com>

Description  Estimation and inference methods for the cross-quantilogram. The cross-quantilogram is a measure of nonlinear dependence between two variables, based on either unconditional or conditional quantile functions. The cross-quantilogram can be considered as an extension of the correlogram, which is a correlation function over multiple lag periods and mainly focuses on linear dependency. One can use the cross-quantilogram to detect the presence of directional predictability from one time series to another. This package provides a statistical inference method based on the stationary bootstrap. See Linton and Whang (2007) <doi:10.1016/j.jeconom.2007.01.004> for univariate time series analysis and Han, Linton, Oka and Whang (2016) <doi:10.1016/j.jeconom.2016.03.001> for multivariate time series analysis.

License  GPL (>= 3)
Encoding  UTF-8
RoxygenNote  7.1.2
Imports  np, quantreg, SparseM, stats
Depends  R (>= 2.10)
LazyData  true
NeedsCompilation  no
Author  Tatsushi Oka [aut, cre], Heejon Han [ctb], Oliver Linton [ctb], Yoon-Jae Whang [ctb]
Repository  CRAN
Date/Publication  2022-03-05 15:20:02 UTC
### Description

The correlation statistics for a given lag order.

### Usage

```r
corr.lag(matH, k)
```

### Arguments

- **matH**: The matrix with the column size of 2.
- **k**: The lag order (integer).

### Details

The function obtains the simple correlation statistics. The values in the first column of input matrix is interacted with the k-lagged values in the second column.
**corr.lag.partial**

**Value**

Correlation

**Author(s)**

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

---

**corr.lag.partial**  
*Partial Cross-correlation function*

**Description**

A function used to obtain partial cross-correlation function for a given lag order.

**Usage**

`corr.lag.partial(matH, k)`

**Arguments**

- `matH`: A matrix with multiple columns (more than 3 columns)
- `k`: The lag order (integer)

**Details**

This function obtains the partial cross-correlation and the simple correlation. To obtain the partial cross-correlation, this function uses the first column of the input matrix and k-lagged values of the rest of the matrix.

**Value**

Partial cross-correlation at k lags and the correlation statistics at k lags.

**Author(s)**

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang
Description

Returns the cross-quantilogram

Usage

crossq(DATA, vecA, k)

Arguments

- **DATA**: An input matrix
- **vecA**: A pair of two probability values at which sample quantiles are estimated
- **k**: A lag order (integer)

Details

This function obtains the cross-quantilogram at the k lag order.

Value

Cross-Quantilogram

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


Examples

```r
## data source
data("sys.risk")

## data: 2 variables
D = sys.risk[,c("Market", "JPM")]

# probability levels for the 2 variables
vecA = c(0.1, 0.5)

## cross-quantilogram with the lag of 5
crossq.max(D, vecA, 5)
```
Description
The cross-quantilograms from 1 to a given lag order.

Usage
crossq.max(DATA, vecA, Kmax)

Arguments
- DATA: An input matrix
- vecA: A pair of two probability values at which sample quantiles are estimated
- Kmax: The maximum lag order (integer)

Details
This function calculates the partial cross-quantilograms up to the lag order users specify.

Value
A vector of cross-quantilogram

Author(s)
Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References

Examples
```r
## data source
data("sys.risk")

## data: 2 variables
D = sys.risk[,c("Market", "JPM")]

# probability levels for the 2 variables
vecA = c(0.1, 0.5)

## cross-quantilogram with lags between 1 and 5
crossq.max(D, vecA, 5)
```
**crossq.max.partial**  
*Partial Cross-Quantilogram upto a given lag order*

**Description**  
The partial cross-quantilograms from 1 to a given lag order.

**Usage**  
```r
crossq.max.partial(DATA, vecA, Kmax)
```

**Arguments**  
- **DATA**  
  An input matrix
- **vecA**  
  A vector of probability values at which sample quantiles are estimated
- **Kmax**  
  The maximum lag order (integer)

**Details**  
The function calculates the partial cross-quantilograms up to the lag order users specify.

**Value**  
A vector of cross-quantilogram and a vector of partial cross-quantilograms

**Author(s)**  
Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

**References**  

**Examples**  
```r
## data source
data("sys.risk")

## data with 3 variables
D = sys.risk[,c("Market", "JPM", "VIX")]

## probability levels for the 3 variables
vecA = c(0.1, 0.1, 0.1)

## partial cross-quantilogram with lags from 1 to 5
crossq.max.partial(D, vecA, 5)
```
**crossq.partial**

**Partial Cross-Quantilogram**

**Description**

Returns the partial cross-quantilogram

**Usage**

```r
crossq.partial(DATA, vecA, k)
```

**Arguments**

- **DATA**: A matrix
- **vecA**: A vector of probability values at which sample quantiles are estimated
- **k**: The lag order

**Details**

This function obtains the partial cross-quantilogram and the cross-quantilogram. To obtain the partial cross-correlation given an input matrix, this function interacts the values of the first column and the k-lagged values of the rest of the matrix.

**Value**

The partial cross-quantilogram and the cross-quantilogram

**Author(s)**

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

**References**


**Examples**

```r
## data source
data("sys.risk")

## data with 3 variables
d = sys.risk[,c("Market", "JPM", "VIX")]

## probability levels for the 3 variables
vecA = c(0.1, 0.1, 0.1)
```
## partial cross-quantilogram with the lag of 5
crossq.max.partial(D, vecA, 5)

crossq.partial.sb  Stationary Bootstrap for the Partial Cross-Quantilogram

**Description**

Returns critical values for the partial cross-quantilogram, based on the stationary bootstrap.

**Usage**

crossq.partial.sb(DATA, vecA, k, gamma, Bsize, sigLev)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA</td>
<td>The original data matrix</td>
</tr>
<tr>
<td>vecA</td>
<td>A pair of two probability values at which sample quantiles are estimated</td>
</tr>
<tr>
<td>k</td>
<td>A lag order</td>
</tr>
<tr>
<td>gamma</td>
<td>A parameter for the stationary bootstrap</td>
</tr>
<tr>
<td>Bsize</td>
<td>The number of repetition of bootstrap</td>
</tr>
<tr>
<td>sigLev</td>
<td>The statistical significance level</td>
</tr>
</tbody>
</table>

**Details**

This function generates critical values for the partial cross-quantilogram, using the stationary bootstrap in Politis and Romano (1994).

**Value**

The bootstrap critical values

**Author(s)**

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

**References**

Description

Returns critical values for the partial cross-quantilogram, based on the stationary bootstrap with the choice of the stationary-bootstrap parameter.

Usage

crossq.partial.sb.opt(DATA, vecA, k, Bsize, sigLev)

Arguments

- **DATA**: The original data matrix
- **vecA**: A pair of two probability values at which sample quantiles are estimated
- **k**: A lag order
- **Bsize**: The number of repetition of bootstrap
- **sigLev**: The statistical significance level

Details

This function generates critical values for the partial cross-quantilogram, using the stationary bootstrap in Politis and Romano (1994).

Value

The bootstrap critical values

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


Description

Returns critical values for the cross-quantilogram, based on the stationary bootstrap.

Usage

crossq.sb(DATA, vecA, k, gamma, Bsize, sigLev)

Arguments

- **DATA**: The original data matrix
- **vecA**: A pair of two probability values at which sample quantiles are estimated
- **k**: A lag order
- **gamma**: A parameter for the stationary bootstrap
- **Bsize**: The number of repetition of bootstrap
- **sigLev**: The statistical significance level

Details

This function generates critical values for for the cross-quantilogram, using the stationary bootstrap in Politis and Romano (1994).

Value

The bootstrap critical values

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


Examples

```r
data("sys.risk") ## data source
D = sys.risk[,c("Market", "JPM")]] ## data: 2 variables

# probability levels for the 2 variables
vecA = c(0.1, 0.5)

## setup for stationary bootstrap
gamma = 1/10 ## bootstrap parameter depending on data
Bsize = 5 ## small size, 5, for test
sigLev = 0.05 ## significance level

## cross-quantilogram with the lag of 5
crossq.sb(D, vecA, 5, gamma, Bsize, sigLev)
```

---

crossq.sb.opt  

**Stationary Bootstrap for the Cross-Quantilogram with the choice of the stationary-bootstrap parameter**

Description

Returns critical values for the cross-quantilogram, based on the stationary bootstrap with the choice of the stationary-bootstrap parameter.

Usage

```r
crossq.sb.opt(DATA, vecA, k, Bsize, sigLev)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA</td>
<td>The original data matrix</td>
</tr>
<tr>
<td>vecA</td>
<td>A pair of two probability values at which sample quantiles are estimated</td>
</tr>
<tr>
<td>k</td>
<td>A lag order</td>
</tr>
<tr>
<td>Bsize</td>
<td>The number of repetition of bootstrap</td>
</tr>
<tr>
<td>sigLev</td>
<td>The statistical significance level</td>
</tr>
</tbody>
</table>

Details

This function generates critical values for the cross-quantilogram, using the stationary bootstrap in Politis and Romano (1994). To choose parameter for the stationary bootstrap, this function first obtains the optimal value for each time serie using the result provided by Politis and White (2004) and Patton, Politis and White (2004) (The R-package, "np", written by Hayfield and Racine is used). Next, the average of the obtained values is used as the parameter value.

Value

The bootstrap critical values
Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


Examples

```r
## data source
data("sys.risk")

## data: 2 variables
D = sys.risk[,c("Market", "JPM")]

# probability levels for the 2 variables
vecA = c(0.1, 0.5)

## setup for stationary bootstrap
Bsize = 5 ## small size 5 for test
sigLev = 0.05 ## significance level

## cross-quantilogram with the lag of 5
crossq.sb.opt(D, vecA, 5, Bsize, sigLev)
```

crossqreg Cross-Quantilogram

Description

Returns the cross-quantilogram

Usage

crossqreg(DATA1, DATA2, vecA, k)
crossqreg

Arguments

  DATA1  An input matrix (T x p1)
  DATA2  An input matrix (T x p2)
  vecA   A pair of two probability values at which sample quantiles are estimated
  k      A lag order (integer)

Details

This function obtains the cross-quantilogram at the k lag order.

Value

Cross-Quantilogram

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


Examples

```r
## data source
data(sys.risk)

## sample size
T = nrow(sys.risk)

## matrix for quantile regressions
## - 1st column: dependent variables
## - the rest: regressors or predictors
D1 = cbind(sys.risk[2:T,"Market"], sys.risk[1:(T-1),"Market"])
D2 = cbind(sys.risk[2:T,"JPM"], sys.risk[1:(T-1),"JPM"])

## probability levels
vecA = c(0.1, 0.2)

## cross-quantilogram with the lag of 5, after quantile regression
crossqreg(D1, D2, vecA, 5)
```
crossqreg.max

Cross-Quantilogram up to a Given Lag Order

Description

The cross-quantilograms from 0 to a given lag order.

Usage

crossqreg.max(DATA1, DATA2, vecA, Kmax)

Arguments

DATA1 An input matrix (T x p1)
DATA2 An input matrix (T x p2)
vecA A pair of two probability values at which sample quantiles are estimated
Kmax The maximum lag order (integer)

Details

This function calculates the partial cross-quantilograms up to the lag order users specify.

Value

A vector of cross-quantilogram

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References

crossqreg.max.partial  Partial Cross-Quantilogram upto a given lag order

Description
The partial cross-quantilograms from 1 to a given lag order.

Usage
crossqreg.max.partial(DATA1, DATA2, vecA, Kmax)

Arguments
DATA1  An input matrix (T x p1)
DATA2  An input matrix (T x p2)
vecA   A vector of probability values at which sample quantiles are estimated
Kmax   The maximum lag order (integer)

Details
This function calculates the partial cross-quantilograms up to the lag order users specify.

Value
A vector of cross-quantilogram and a vector of partial cross-quantilograms

Author(s)
Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References
crossqreg.partial  

Partial Cross-Quantilogram

Description

Returns the partial cross-quantilogram

Usage

crossqreg.partial(DATA1, DATA2, vecA, k)

Arguments

DATA1  
An input matrix (T x p1)

DATA2  
An input matrix (T x p2)

vecA  
A vector of probability values at which sample quantiles are estimated

k  
The lag order

Details

This function obtains the partial cross-quantilogram and the cross-quantilogram. To obtain the partial cross-correlation given an input matrix, this function interacts the values of the first column and the k-lagged values of the rest of the matrix.

Value

The partial cross-quantilogram and the cross-quantilogram

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References

Description

Returns critical values for the cross-quantilogram, based on the stationary bootstrap.

Usage

crossqreg.sb(DATA1, DATA2, vecA, k, gamma, Bsize, sigLev)

Arguments

DATA1  The original data matrix (T x p1)
DATA2  The original data matrix (T x p2)
vecA   A pair of two probability values at which sample quantiles are estimated
k      A lag order
gamma  A parameter for the stationary bootstrap
Bsize  The number of repetition of bootstrap
sigLev The statistical significance level

Details

This function generates critical values for the cross-quantilogram, using the stationary bootstrap in Politis and Romano (1994).

Value

The bootstrap critical values

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


Examples

data(sys.risk)

## sample size
T = nrow(sys.risk)

## matrix for quantile regressions
## - 1st column: dependent variables
## - the rest: regressors or predictors
D1 = cbind(sys.risk[2:T,"Market"], sys.risk[1:(T-1),"Market"])
D2 = cbind(sys.risk[2:T,"JPM"], sys.risk[1:(T-1),"JPM"])

## probability levels
vecA = c(0.1, 0.2)

## setup for stationary bootstrap
gamma = 1/10 ## bootstrap parameter depending on data
Bsize = 5 ## small size 10 for test
sigLev = 0.05 ## significance level

## cross-quantilogram with the lag of 5, after quantile regression
crossqreg.sb(D1, D2, vecA, 5, gamma, Bsize, sigLev)

---

q.hit Quantile Hit

Description

Returns the matrix of quantile-hits

Usage

q.hit(DATA, vecA)

Arguments

DATA A matrix that has time-series observations in its columns
vecA A vector of probability values at which sample quantiles are estimated

Details

This function generates the quantile hits given a vector of probability values. The quantile hits are obtained for each column of an input matrix.

Value

A matrix of quantile-hits
qreg.hit

Author(s)
Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

Description
Returns the matrix of quantile-hits

Usage
qreg.hit(DATA1, DATA2, vecA)

Arguments
DATA1  An input matrix (T x p1+1) with the first column of the dependent variable and the rest of columns with regressors
DATA2  An input matrix (T x p2+1) with the first column of the dependent variable and the rest of columns with regressors
vecA   A vector of probability values at which sample quantiles are estimated

Details
This function generates the quantile hits based on quantile regression, given a vector of probability values. The quantile regressions are estimated for each matrix of data and a pair of quantile hits are produced.

Value
A matrix of quantile-hits

Author(s)
Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References
Description

The Box-Pierce and Ljung-Box type Q-statistics

Usage

Qstat(vecTest, Tsize)

Arguments

vecTest       A vector of test statistics ordered with respect the number of lags
Tsize         A original sample size

Details

This function returns Box-Pierce and Ljung-Box type Q-statistics

Value

the Box-Pierce and Ljung-Box statistics

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


**Qstat.reg.sb**

**Stationary Bootstrap for Q statistics**

**Description**

Stationary Bootstrap procedure to generate critical values for both Box-Pierce and Ljung-Box type Q-statistics.

**Usage**

\[ \text{Qstat.reg.sb}(\text{DATA1}, \text{DATA2}, \text{vecA}, \text{Psize}, \text{gamma}, \text{Bsize}, \text{sigLev}) \]

**Arguments**

- **DATA1**: The original data set (1)
- **DATA2**: The original data set (2)
- **vecA**: A pair of two probability values at which sample quantiles are estimated
- **Psize**: The maximum number of lags
- **gamma**: A parameter for the stationary bootstrap
- **Bsize**: The number of repetition of bootstrap
- **sigLev**: The statistical significance level

**Details**

This function returns critical values for both Box-Pierce and Ljung-Box type Q-statistics through the stationary bootstrap proposed by Politis and Romano (1994).

**Value**

The bootstrap critical values

**Author(s)**

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

**References**


Examples

data(sys.risk)

## sample size
T = nrow(sys.risk)

## matrix for quantile regressions
## - 1st column: dependent variables
## - the rest: regressors or predictors
D1 = cbind(sys.risk[2:T,"Market"], sys.risk[1:(T-1),"Market")
D2 = cbind(sys.risk[2:T,"JPM"], sys.risk[1:(T-1),"JPM")

## probability levels
vecA = c(0.1, 0.2)

## setup for stationary bootstrap
gamma = 1/10 ## bootstrap parameter depending on data
Bsize = 5 ## small size, 5, for test
sigLev = 0.05 ## significance level

## Q statistics with lags from 1 to 5, after quantile regression
Qstat.reg.sb(D1, D2, vecA, 5, gamma, Bsize, sigLev)

Qstat.sb

Stationary Bootstrap for Q statistics

Description

Stationary Bootstrap procedure to generate critical values for both Box-Pierce and Ljung-Box type Q-statistics

Usage

Qstat.sb(DATA, vecA, Psize, gamma, Bsize, sigLev)

Arguments

DATA The original data
vecA A pair of two probability values at which sample quantiles are estimated
Psize The maximum number of lags
gamma A parameter for the stationary bootstrap
Bsize The number of repetition of bootstrap
sigLev The statistical significance level
Details

This function returns critical values for both Box-Pierce and Ljung-Box type Q-statistics through the stationary bootstrap proposed by Politis and Romano (1994).

Value

The bootstrap critical values

Author(s)

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

References


Examples

data("sys.risk") ## data source
D = sys.risk[,c("Market", "JPM")]] ## data: 2 variables

# probability levels for the 2 variables
vecA = c(0.1, 0.5)

## setup for stationary bootstrap
gamma = 1/10 ## bootstrap parameter depending on data
Bsize = 5 ## small size, 5, for test
sigLev = 0.05 ## significance level

## Q statistics with lags from 1 to 5
Qstat.sb(D, vecA, 5, gamma, Bsize, sigLev)
**Arguments**

- **DATA**: The original data
- **vecA**: A pair of two probability values at which sample quantiles are estimated
- **Psize**: The maximum number of lags
- **Bsize**: The number of repetition of bootstrap
- **sigLev**: The statistical significance level

**Details**

This function returns critical values for both Box-Pierce and Ljung-Box type Q-statistics through the stationary bootstrap proposed by Politis and Romano (1994). To choose parameter for the stationary bootstrap, this function first obtains the optimal value for each time serie using the result provided by Politis and White (2004) and Patton, Politis and White (2004) (The R-package, “np”, written by Hayfield and Racine is used). Next, the average of the obtained values is used as the parameter value.

**Value**

The bootstrap critical values

**Author(s)**

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

**References**


**Examples**

```r
data("sys.risk") ## data source
D = sys.risk[,c("Market", "JPM")]) ## data: 2 variables

# probability levels for the 2 variables
vecA = c(0.1, 0.5)

## setup for stationary bootstrap
Bsize = 5  ## small size, 5, for test
sigLev = 0.05 ## significance level
```
sb.index

## Q statistics with lags from 1 to 5
Qstat.sb.opt(D, vecA, 5, Bsize, sigLev)

---

### sb.index

**Stationary Bootstrap Index**

**Description**

A subfunction for the stationary bootstrap

**Usage**

```
sb.index(Nsize, gamma)
```

**Arguments**

- `Nsize` The size of the stationary bootstrap resample
- `gamma` A parameter for the stationary bootstrap.

**Details**

This function resamples blocks of indices with random block lengths. This code follows the MATLAB file of the Oxford MFE Toolbox written by Kevin Sheppard.

**Value**

A vector of indices for the stationary bootstrap

**Author(s)**

Heejoon Han, Oliver Linton, Tatsushi Oka and Yoon-Jae Whang

**References**

The Oxford MFE toolbox (http://www.kevinsheppard.com/wiki/MFE_Toolbox) by Kevin Sheppard
```
26

Description
The dataset contains monthly excess stock returns and stock variance, which are included in the
data set analyzed in Goyal and Welch (2008). Stock returns are measured by the S&P 500 index
and include dividends. A treasury-bill rate is subtracted from stock returns to give excess stock
returns. The stock variance is a volatility estimate based on daily squared returns and is treated as
an estimate of equity risk in the literature. The sample period is from February 1885 to December
2005 with sample size 1,451.

- Date: Year-Month-Day
- Return: excess stock returns
- Variance: stock variance

Usage
data(stock)

Format
A data object with two variables

References
Han, H., Linton, O., Oka, T., and Whang, Y. J. (2016). "The cross-quantilogram: Measuring quantile
dependence and testing directional predictability between time series." Journal of Econometrics,
193(1), 251-270.

Welch, Ivo, and Amit Goyal. "A comprehensive look at the empirical performance of equity premi-

sys.risk

Description
The data set contains the daily CRSP market value weighted index returns, which are used as the
market index returns in Brownless and Engle (2012), and also includes daily stock returns on JP
Morgan Chase (JPM), Goldman Sachs (GS) and American International Group (AIG). The sample
period is from 2 Jan. 2001 to 30 Dec. 2011 with sample size 2,767.

Usage
data(sys.risk)
```
**Format**

A data object with five variables

**Details**

- date: The time index (day)
- Market: The daily CRSP market value weighted index returns
- JPM: stock returns on JP Morgan Chase (JPM)
- GS: stock returns on Goldman Sachs (GS)
- AIG: stock returns on American International Group (AIG)

**References**


Index

* datasets
  stock, 26
  sys.risk, 26

corr.lag, 2
corr.lag.partial, 3
crossq, 4
crossq.max, 5
crossq.max.partial, 6
crossq.partial, 7
crossq.partial.sb, 8
crossq.partial.sb.opt, 9
crossq.sb, 10
crossq.sb.opt, 11
crossqreg, 12
crossqreg.max, 14
crossqreg.max.partial, 15
crossqreg.partial, 16
crossqreg.sb, 17

q.hit, 18
qreg.hit, 19
Qstat, 20
Qstat.reg.sb, 21
Qstat.sb, 22
Qstat.sb.opt, 23

sb.index, 25
stock, 26
sys.risk, 26