Package ‘multibreakeR’

May 24, 2023

Type Package
Title Tests for a Structural Change in Multivariate Time Series
Version 0.1.0
Description Flexible implementation of a structural change point detection algorithm for multivariate time series.
   It authorizes inclusion of trends, exogenous variables, and break test on the intercept or on the full vector autoregression system.
License GPL
Language en-US
Imports dplyr, ggplot2, reshape2, rlang (>= 0.4.11), scales, stats
Suggests knitr, rmarkdown, roxygen2, testthat (>= 3.0.0)
VignetteBuilder knitr
Config/testthat/edition 3
Encoding UTF-8
LazyData true
RoxygenNote 7.2.3
URL https://github.com/loicym/multibreakeR
NeedsCompilation no
Author Loic Marechal [cre, aut]
Maintainer Loic Marechal <loic.marechal@unil.ch>
Depends R (>= 3.5.0)
Repository CRAN
Date/Publication 2023-05-24 08:20:08 UTC

R topics documented:

AicBic ................................................................. 2
Beta ................................................................. 3

1
Description

Compute the AIC and BIC criteria for lags from 1 to q.max

Usage

AicBic(mat.y, q.max, mat.x = NULL, trend = FALSE, intercept = TRUE)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mat.y</td>
<td>A matrix object of time series</td>
</tr>
<tr>
<td>q.max</td>
<td>The maximum lag considered</td>
</tr>
<tr>
<td>mat.x</td>
<td>An optional matrix of covariates</td>
</tr>
<tr>
<td>trend</td>
<td>If a trend is considered (default to false)</td>
</tr>
<tr>
<td>intercept</td>
<td>If the test is on the intercept (default to true)</td>
</tr>
</tbody>
</table>

Value

A data frame object that contains all AIC (first row) and BIC (second row) for all the q.max lags tested.

Examples

data(example_data)
aic.bic <- AicBic(mat.y = example_data,
                   q.max = 2,
                   trend = FALSE,
                   intercept = TRUE)
### Description

Compute the matrix of parameters and the covariance matrix of errors in OLS, FGLS, or IGLS mode.

#### Usage

\[
\text{Beta(mat.z, mat.y.ex, n.eq, p, est.mode, iter)}
\]

#### Arguments

- **mat.z**: A matrix object of time series, regressor matrix
- **mat.y.ex**: A matrix object of time series, regressor matrix
- **n.eq**: number of equations in the VAR
- **p**: number of observations
- **est.mode**: estimation mode: "OLS", "FGLS", or "IGLS"
- **iter**: If "IGLS" is used, how many iterations before stopping

#### Value

A list with the matrix of beta parameters as first element and the covariance matrix of error as second element.

### ConfidenceInterval

Compute the confidence interval in time unit.

#### Usage

\[
\text{ConfidenceInterval(mat.g, mat.s, mat.sigma, mat.r, mat.beta, cv, p)}
\]

#### Arguments

- **mat.g**: A matrix object of time series
- **mat.s**: A selection matrix
- **mat.sigma**: The covariance matrix
- **mat.r**: The selection vector of parameters
- **mat.beta**: The matrix of parameters
- **cv**: A vector of critical values
- **p**: The length of the vector
The difference in time unit around the break

ConformableMatrix

Description

Compute the list of matrices with correct dimensions to pass later in the computation

Usage

ConformableMatrix(mat.y, q, mat.x = NULL, trend = FALSE, intercept = TRUE)

Arguments

mat.y The matrix object of time series
q The chosen lag
mat.x The matrix of optional covariates
trend Whether we add a trend. Default = FALSE
intercept Whether the break test is on the intercept only. Default = TRUE

Value

A list of conformed matrices

Examples

data(example_data)
conf.matrix <- ConformableMatrix(mat.y = example_data, q = 2)

Example MultibreakeR simulated data

Description

Data generated with the function Simul() of the multibreakeR package with 100 time series observations (n = 100), five time series (p = 5), a break intensity of 1 (intensity = 1), and a break occurring at 35% of the sample (when.break = 0.35). These are also the default arguments of the Simul() function.

Usage

data(example_data)
**Fstat**

**Format**

A matrix object

**Source**

https://github.com/loicym/multibreakeR

**References**

MultibreakeR generated data

**Examples**

```r
list.breaks <- Main(mat.y = Simul(p = 2, when.break = 0.5),
mat.x = NULL,
trend = FALSE,
intercept = TRUE,
ci = c(0.9, 0.95, 0.99),
est.mode = "OLS",
iter = 3,
aic.bic.mode = "AIC",
q.max = 2,
trim = 0.4,
pos.break = FALSE)
```

---

**Fstat**

**Description**

Compute the f-statistic for the break test

**Usage**

```r
Fstat(mat.r, mat.beta, mat.z, p, mat.sigma)
```

**Arguments**

- `mat.r` The selection matrix for the parameters
- `mat.beta` The matrix of parameters
- `mat.z` The matrix of original and "breaking" time series
- `p` The number of observations
- `mat.sigma` The covariance matrix

**Value**

The f-statistic scalar
### Lags

**Description**

Compute the lags for the mat.y time series matrix

**Usage**

\[
\text{Lags(mat.y, q)}
\]

**Arguments**

- **mat.y**: The matrix of time series
- **q**: The lag chosen

**Value**

A list of original (dependent) and lagged (independent) time series matrix

**Examples**

```r
data(example_data)
list.lags <- Lags(mat.y = example_data, q = 2)
```

### Main

**Description**

Entry point for the whole computation of the algorithm of Bai, Lumsdaine, and Stock (1998)

**Usage**

```r
Main(
    mat.y, mat.x = NULL,
    trend = FALSE, intercept = TRUE,
    ci = c(0.9, 0.95, 0.99),
    est.mode = "OLS",
    iter = 3,
    aic.bic.mode = "AIC",
    q.max = 2,
    trim = 0.15,
    pos.break = FALSE
)
```
Arguments

mat.y The matrix object of time series
mat.x The matrix of optional covariates
trend Whether we add a trend. Default = FALSE
intercept Whether the break test is on the intercept only. Default = TRUE
cli A vector of confidence intervals. Default = c(0.9, 0.95, 0.99)
est.mode Estimation mode. Can be "OLS", "FGLS", or "IGLS"
iter Maximum number of iterations in the "IGLS" mode. Default to 3
aic.bic.mode Can be "AIC" or "BIC" depending on the criterion chosen for the lag selection
q.max Maximum lag tested for the AIC or BIC criterion
trim Percentage for the trim value for the starting and ending window over which the algorithm is not tested. Default to 15%
pos.break Whether we want to select the maximum positive break only and discard the negative ones. Default to FALSE

Value

A list of the vector of f-statistics, the maximum f-statistic retained, the confidence interval, the critical values, the break date, the original matrix of time series tested, the matrix with breaking and not breaking covariates, the index of the break in the time series, the size of the break (mean.shift), the optimal "AIC" or "BIC", a ggplot object (g1), and the trimmed dates.

Examples

```r
data(example_data)
list.results <- Main(mat.y = example_data, q = 2)
```

Description

Generate a ggplot2 object to depict the break and the time series tested

Usage

```r
PlotStats(my.dates, my.vars, f.stat, mat.ci = mat.ci, mat.y)
```

Arguments

my.dates A vector of dates
my.vars The variables tested
f.stat The f-statistics
mat.ci The matrix of confidence intervals
mat.y The original time series
**Value**

A ggplot2 object

---

**Sigma**

**Description**

#compute the covariance matrix of errors as in Bai, Lumsdaine, and Stock (1998)

**Usage**

Sigma(mat.z, mat.y.ex, mat.beta, n.eq)

**Arguments**

- **mat.z**: A matrix of breaking and non breaking time series
- **mat.y.ex**: A vectorized matrix of time series
- **mat.beta**: The matrix of parameters
- **n.eq**: The number of equations in the VAR system

**Value**

The covariance matrix of errors

---

**Simul**

**Description**

#Simulate data to test the functions

**Usage**

Simul(n = 100, p = 5, intensity = 1, when.break = 0.5)

**Arguments**

- **n**: The number of time series observations
- **p**: The number of time series
- **intensity**: The intensity of the break
- **when.break**: When should the break be simulated (as a percentage of the time series sample)
Value
A matrix of time series with a common break

Examples
```r
data(example_data)
simul.data <- Simul(n = 100, p = 5, intensity = 1, when.break = 0.5)
```

Description
Computes the critical values for a vector of confidence intervals proposed (ci)

Usage
```r
Vdistr(ci)
```

Arguments
- `ci` A vector of confidence intervals

Value
A vector of critical values

Examples
```r
vect.cv <- Vdistr(ci = c(0.9, 0.95, 0.99))
```
Index

* datasets
  example_data, 4

AicBic, 2
Beta, 3
ConfidenceInterval, 3
ConformableMatrix, 4
example_data, 4
Fstat, 5
Lags, 6
Main, 6
PlotStats, 7
Sigma, 8
Simul, 8
Vdistr, 9