Package ‘VARtests’

November 2, 2018

Type Package

Title Tests for Error Autocorrelation, ARCH Errors, and Cointegration in Vector Autoregressive Models

Version 2.0.5

Date 2018-11-02

Author Markus Belfrage [aut, cre], Paul Catani [ctb], Niklas Ahlgren [ctb]

Depends R(>= 3.0.2)

LinkingTo Rcpp(>= 0.12.10), RcppArmadillo

Imports methods, Rcpp, sn

Maintainer Markus Belfrage <markus.belfrage@gmail.com>


LazyData yes

NeedsCompilation yes

License GPL (>= 3)

Repository CRAN

Date/Publication 2018-11-02 17:30:03 UTC

R topics documented:

ACtest ........................................... 2
archBootTest .................................. 5
cointBootTest .................................. 9
DataFiles ....................................... 12
**ACtest**

Test for Error Autocorrelation in VAR Models.

**Description**
Performs LM tests for error AC in VAR models. The code is based on Paul Catani’s original R code used in the paper *Wild Bootstrap Tests for Autocorrelation in Vector Autoregressive Models* (Ahlgren and Catani, 2016).

**Usage**

```r
ACtest(fit, h = 4, HCTYPE = c("LM", "HC0", "HC1", "HC2", "HC3"), univariate = FALSE)
```

**Arguments**

- `fit`: an object of class "VARfit" that was returned by the `VARfit` function, or an object of class "verest" from the function "VAR" in the "vars" package.
- `h`: the lag length of the alternative VAR(h) model for the errors.
- `HCTYPE`: a vector containing some or all (default) of "LM", "HC0", "HC1", "HC2" and "HC3". The "LM" type assumes homoscedastic errors, while the "HC" types are different Heteroskedasticity-consistent tests (see 'Details').
- `univariate`: either FALSE, TRUE, or "only". If TRUE, the test will also be performed in its univariate form, if "only", only the univariate version will be used.
- `x`: Object with class attribute 'ACtest'.
- `...`: further arguments passed to or from other methods.

**Details**
To run the wild bootstrap version of the test, please use the output from this function with the function `wildBoot`.

**Mathematical details**
The tests for error AC are based on the least squares (LS) residuals from the $K$-dimensional vector autoregressive (VAR) model with $p$ lags (abstracting from deterministic terms):

\[ y_t = \Pi_1 y_{t-1} + \cdots + \Pi_p y_{t-p} + u_t, \quad E(u_t) = 0, \quad E(u_t u'_t) = \Omega, \quad t = 1, \ldots, N. \]

The LS residuals are...
\[ \tilde{u}_t = y_t - \tilde{\Pi}_1 y_{t-1} - \cdots - \tilde{\Pi}_p y_{t-p}, \]

where \( \tilde{\Pi}_1, \ldots, \tilde{\Pi}_p \) are the LS estimates of the \( K \times K \) parameter matrices \( \Pi_1, \ldots, \Pi_p \).

The LM statistic is computed from the auxiliary model

\[ \tilde{u}_t = \Pi_1 y_{t-1} + \cdots + \Pi_p y_{t-p} + D_1 \tilde{u}_{t-1} + \cdots + D_h \tilde{u}_{t-h} + e_t \]

\[ = (Z'_{t-1} \otimes I_K) \phi + (\tilde{U}'_{t-1} \otimes I_K) \psi + e_t, \]

where \( Z_{t-1} = (y_{t-1}' , \ldots , y_{t-p}' )' \), \( \phi = \text{vec}(\Pi_1 , \ldots , \Pi_p)' \), \( \tilde{U}_{t-1} = (\tilde{u}'_{t-1} , \ldots , \tilde{u}'_{t-h} )' \) and \( \psi = \text{vec}(D_1 , \ldots , D_h)' \). The symbol \( \otimes \) denotes the Kronecker product and the symbol vec denotes the column vectorisation operator. The first \( h \) values of the residuals \( \tilde{u}_t \) are set to zero in the auxiliary model, so that the series length is equal to the series length in the original VAR model.

The LM statistic for error AC of order \( h \) is given by

\[ Q_{LM}(h) = N \hat{\psi}' (\hat{\Sigma}^{\psi \psi})^{-1} \hat{\psi}, \]

where \( \hat{\psi} \) is the LS estimate of \( \psi \) and \( \hat{\Sigma}^{\psi \psi} \) is the block of

\[
\left( N^{-1} \sum_{t=1}^{N} \left[ \begin{array}{c} Z_{t-1} \otimes I_K \\ \tilde{U}_{t-1} \otimes I_K \end{array} \right] \hat{\Sigma}_u^{-1} \left[ \begin{array}{c} Z_{t-1}' \otimes I_K \\ \tilde{U}_{t-1}' \otimes I_K \end{array} \right] \right)^{-1}
\]

corresponding to \( \psi \). Here \( \hat{\Sigma}_u = N^{-1} \sum_{t=1}^{N} \tilde{u}_t \tilde{u}_t' \) is the estimator of the error covariance matrix from the VAR model.

The multivariate heteroskedasticity-consistent covariance matrix estimator (HCCME) for the auxiliary model is given by (Hafner and Herwartz 2009)

\[ V_N^{-1} W_N V_N^{-1} = (\Gamma_N \otimes I_K)^{-1} W_N (\Gamma_N \otimes I_K)^{-1}, \]

where

\[ V_N = \Gamma_N \otimes I_K, \]

\[ \Gamma_N = \frac{1}{N} \sum_{t=1}^{N} \left( \begin{array}{cc} \tilde{U}_{t-1} \\ Z_{t-1} \end{array} \right) \left( \begin{array}{cc} \tilde{U}'_{t-1} \\ Z'_{t-1} \end{array} \right), \]

\[ W_N = \frac{1}{N} \sum_{t=1}^{N} \left( \begin{array}{cc} \tilde{U}_{t-1} \\ Z_{t-1} \end{array} \right) \left( \begin{array}{cc} \tilde{U}'_{t-1} \\ Z'_{t-1} \end{array} \right) \otimes (\tilde{u}_t \tilde{u}_t'). \]

The HCCME-based LM statistics for error AC are obtained from the expression for \( Q_{LM}(h) \) by replacing \( \hat{\Sigma}^{\psi \psi} \) by the block of \( V_N^{-1} W_N V_N^{-1} = (\Gamma_N \otimes I_K)^{-1} W_N (\Gamma_N \otimes I_K)^{-1} \) corresponding to \( \psi \) and with \( \tilde{u}_t \) defined by \( HC_0 \), \( HC_1 \), \( HC_2 \) and \( HC_3 \), respectively.

\( HC_0 \) uses \( \tilde{u}_t \tilde{u}_t'. \)

\( HC_1 \) multiplies the elements of \( \tilde{u}_t \tilde{u}_t' \) by \( N/(N-Kp) \).
AC test

\[ \text{HC}_2 \text{ replaces } \hat{\mathbf{u}}_t \text{ by } \hat{\mathbf{u}}_t/(1 - h_t)^{1/2}, \text{ where } h_t = \mathbf{Z}_t(\mathbf{Z}' \mathbf{Z})^{-1} \mathbf{Z}_t' \text{ is the } t\text{th diagonal element of } \mathbf{Z}(\mathbf{Z}' \mathbf{Z})^{-1} \mathbf{Z}', \text{ and } \mathbf{Z} = (\mathbf{Z}_0, \ldots, \mathbf{Z}_{N-1}). \]

\[ \text{HC}_3 \text{ replaces } \hat{\mathbf{u}}_t \text{ by } \hat{\mathbf{u}}_t/(1 - h_t). \]

See MacKinnon and White (1985) for details.

The recursive-design wild bootstrap (WB) tests for error AC are computed using Algorithm 1 in Ahlgren and Catani (2016). The Fixed-design WB tests for error AC are computed using Algorithm 2 in Ahlgren and Catani (2016).

Value

- a list of class "ACtest".
- fit: the fit argument object.
- inputType: the type of object of fit.
- HCTYPE: a vector of the HCTYPE's used.
- h: the lag length of the alternative VAR(h) model for the errors.
- pValues: a 1 x 5 matrix of the P values of the tests.
- Q: a 1 x 5 matrix of the Q statistics of the tests.
- unipValues: a K x 5 matrix of the P values of the univariate tests.
- uniQ: a K x 5 matrix of the Q statistics of the univariate tests.
- univariate: the 'univariate' argument.
- description: who ran the test and when.
- time: computation time taken to run the test.
- call: how the function ACtest() was called.

References


See Also

- VARfit to estimate a VAR(p), and wildBoot to run the Wild Bootstrap versions of the tests.

Examples

```r
fit <- VARfit(y = VodafoneCDs, p = 3, const = TRUE, trend = FALSE)
test <- ACtest(fit = fit, h = 1, HCTYPE = c("LM", "HC0", "HC1", "HC2", "HC3"))
```
ArchBootTest

Combined LM test for ARCH errors in VAR models.

Description

Performs the bootstrap combined Lagrange multiplier (LM) test for autoregressive conditional heteroskedastic (ARCH) errors in vector autoregressive (VAR) models of Catani and Ahlgren (2016).

The tests of Eklund and Teräsvirta (2007), as well as the Multivariate LM test for ARCH as described for example in Lütkepohl (2006, sect. 16.5), are also included if the arguments ET respectively MARCH are set to TRUE. The bootstrap procedure for those are the same as in Catani and Ahlgren (2016).

Usage

archBootTest(fit, h = 2, B = 499, CA = TRUE, ET = TRUE, MARCH = TRUE, dist = "norm", skT.param = c(0, 1, 0, 5))

## S3 method for class 'archBootTest'
print(x, ...)

Arguments

- **fit**: an object of class "VARfit" that was returned by the VARfit function, or an object of class "verest" from the function VAR in the vars package.
- **h**: the lag length of the alternative VAR(h) model for the errors.
- **B**: the number of bootstrap simulations.
- **CA**: if TRUE, the Catani and Ahlgren (2017) test will run.
- **ET**: if TRUE, the Eklund and Teräsvirta (2007) test will run.
- **MARCH**: if TRUE, the Multivariate LM test for ARCH will run. See e.g. Lütkepohl (2006, sect. 16.5).
- **dist**: the error distribution. Either "norm" for the standard normal distribution, or "skT" for the skew-t distribution. The parameters of the skew-t distribution can be set with the skT.param argument. Can also be a function that returns random draws as an (N-p) x K matrix or a vector of length (N-p) * K.
- **skT.param**: a vector of four parameters for the skew-t distribution in case "skT" was used for the dist argument. The function rmst is used to draw the errors and the parameters are passed as skT.param = c(xi, Omega, alpha, nu)
- **x**: Object with class attribute ‘archBootTest’.
- **...**: further arguments passed to or from other methods.
Details

All tests for ARCH are based on Cholesky-standardised least squares (LS) residuals from the $K$-dimensional vector autoregressive (VAR) model with $p$ lags (abstracting from deterministic terms):

$$y_t = \Pi_1 y_{t-1} + \cdots + \Pi_p y_{t-p} + u_t, \quad E(u_t) = 0, \quad E(u_t u_t') = \Omega, \quad t = 1, \ldots, N.$$  

The LS residuals are

$$\hat{u}_t = y_t - \hat{\Pi}_1 y_{t-1} - \cdots - \hat{\Pi}_p y_{t-p},$$

where $\hat{\Pi}_1, \ldots, \hat{\Pi}_p$ are the LS estimates of the $K \times K$ parameter matrices $\Pi_1, \ldots, \Pi_p$. The multivariate LS residuals are $\hat{U} = (\hat{u}_1, \ldots, \hat{u}_K)$, which is an $N \times K$ matrix. The Cholesky-standardised LS residuals are

$$\tilde{w}_t = (S^{-1}_{\hat{U}})' \hat{u}_t,$$

where $S_{\hat{U}}$ is the Cholesky factor of $N^{-1} \hat{U}' \hat{U}$, i.e. $S_{\hat{U}}$ is the (unique) upper triangular matrix such that

$$\hat{\Omega} = S_{\hat{U}}' S_{\hat{U}}, \quad \hat{\Omega}^{-1} = (N^{-1} \hat{U}' \hat{U})^{-1} = S_{\hat{U}}^{-1} (S_{\hat{U}}^{-1})'.$$

The LM test for ARCH of order $h$ (Engle 1982) in equation $i, i = 1, \ldots, K$, is a test of $H_0: b_1 = \cdots = b_h$ against $H_1: b_j \neq 0$ for at least one $j \in \{1, \ldots, h\}$ in the auxiliary regression

$$\tilde{w}_t^2 = b_0 + b_1 \tilde{w}_{i,t-1}^2 + \cdots + b_h \tilde{w}_{i,t-h}^2 + e_{it}.$$  

The LM statistic has the form

$$LM_i = (N - p) R_i^2,$$

where $R_i^2$ is $R^2$ from the auxiliary regression for equation $i$.

The combined LM statistic (Dufour et al. 2010, Catani and Ahlgren 2016) is given by

$$\tilde{LM} = 1 - \min_{1 \leq i \leq K} (p(LM_i)),$$

where $p(LM_i)$ is the $p$-value of the $LM_i$ statistic, derived from the asymptotic $\chi^2(h)$ distribution. The test is only available as a bootstrap test. The bootstrap $p$-value is simulated using Bootstrap Algorithm 1 of Catani and Ahlgren (2016) if the errors are normal,

$$w_{i1}, \ldots, w_{iT} \sim \mathcal{N}(0, 1),$$

and Bootstrap Algorithm 2 if the errors are skew-$t$ (by setting the function argument $\text{dist} = \text{"skt"}$),

$$w_{i1}, \ldots, w_{iT} \sim \text{skT}(0, 1; \lambda, v),$$

where $\lambda$ is the skewness parameter and $v$ is the degrees-of-freedom parameter of the skew-$t$ distribution. These parameters can be set with the $\text{skt.param}$ argument.

The multivariate LM test for ARCH of order $h$ is a generalisation of the univariate test, and is based on the auxiliary regression.
The null hypothesis is \( H \) for at least one \( j \in \{1, \ldots, h\} \). The multivariate LM statistic has the form
\[
MLM = \frac{1}{2}(N - p)K(K + 1) - (N - p)\text{tr}(\hat{\Omega}_{\text{vech}}\hat{\Omega}^{-1}),
\]
where \( \hat{\Omega}_{\text{vech}} \) is the estimator of the error covariance matrix from the auxiliary regression and \( \hat{\Omega} = N^{-1}\sum_{t=1}^{N} \hat{\mathbf{u}}_t\hat{\mathbf{u}}_t' \) is the estimator of the error covariance matrix from the VAR model (see Lütkepohl 2006, sect. 16.5). The \( MLM \) statistic is asymptotically distributed as \( \chi^2(K^2(K + 1)^2h/4) \). The test is available as an asymptotic test using the asymptotic \( \chi^2(K^2(K + 1)^2h/4) \) distribution to derive the \( p \)-value, and as a bootstrap test. The bootstrap \( p \)-value is simulated using Bootstrap Algorithms 1 and 2 of Catani and Ahlgren (2016). The asymptotic validity of the bootstrap multivariate LM test has not been established.

The Eklund and Teräsvirta (2007) LM test of constant error covariance matrix assumes the alternative hypothesis is a constant conditional correlation autoregressive conditional heteroskedasticity (CCC-ARCH) process of order \( h \): \( \mathbf{H}_t = \mathbf{D}_t\mathbf{P}\mathbf{D}_t \), where \( \mathbf{D}_t = \text{diag}(h_{1t}^{1/2}, \ldots, h_{Kt}^{1/2}) \) is a diagonal matrix of conditional standard deviations of the errors \( \{\mathbf{u}_t\} \) and \( \mathbf{P} = (\rho_{ij}) \), \( i, j = 1, \ldots, K \), is a positive definite matrix of conditional correlations. The conditional variance \( \mathbf{h}_t = (h_{1t}, \ldots, h_{Kt})' \) is assumed to follow a CCC-ARCH(\( h \)) process:
\[
\mathbf{h}_t = \mathbf{a}_0 + \sum_{j=1}^{h} \mathbf{A}_j \mathbf{u}_{t-j}^{(2)},
\]
where \( \mathbf{a}_0 = (a_{01}, \ldots, a_{0K})' \) is a \( K \)-dimensional vector of positive constants, \( \mathbf{A}_1, \ldots, \mathbf{A}_h \) are \( K \times K \) diagonal matrices and \( \mathbf{u}_{t}^{(2)} = (u_{1t}^2, \ldots, u_{Kt}^2)' \).

The null hypothesis is \( H_0 : \text{diag}(\mathbf{A}_1) = \cdots = \text{diag}(\mathbf{A}_h) = 0 \) against \( H_1 : \text{diag}(\mathbf{A}_j) \neq 0 \) for at least one \( j \in \{1, \ldots, h\} \). The LM statistic has the form
\[
LM_{\text{CCC}} = (N - p)\mathbf{s}(\hat{\theta})'\mathbf{I}(\hat{\theta})^{-1}\mathbf{s}(\hat{\theta}),
\]
where \( \mathbf{s}(\hat{\theta}) \) and \( \mathbf{I}(\hat{\theta}) \) are the score vector and information matrix, respectively, estimated under the null hypothesis (see Eklund and Teräsvirta 2007 for details). The asymptotic distribution of the \( LM_{\text{CCC}} \) statistic is \( \chi^2(Kh) \). The test is available as an asymptotic test using the asymptotic \( \chi^2(Kh) \) distribution to derive the \( p \)-value, and as a bootstrap test. The bootstrap \( p \)-value is simulated using Bootstrap Algorithms 1 and 2 of Catani and Ahlgren (2016). The asymptotic validity of the bootstrap \( LM_{\text{CCC}} \) test has not been established.

Value

a list of class "AChTest".

fit the fit argument object.
**archBootTest**

inputType  the type of object of fit.
h  the lag length $h$ of the alternative VAR($h$) model for the errors.
B  the number of bootstrap simulations.
K  the number of series/equations in the fitted VAR model.
CA  the CA input argument.
ET  the ET input argument.
MARCH  the MARCH input argument.
dist  the dist argument object.
standardizedResi  the Cholesky-standardized residuals.

CA_LM  the combined LM statistic of Catani and Ahlgren (2016), computed as $1 - \min(P(\text{CA_LMi}))$.
CA_bootPV  the bootstrap $P$-value of the combined LM test of Catani and Ahlgren (2016).
CA_LMi  the LM statistics of Catani and Ahlgren (2016) for each time series.
CA_LMijStar  an (N-p) x K matrix of the bootstrap LM statistics for each time series (columns) and bootstrap sample (rows), for the Catani and Ahlgren (2016) test.
CA_uniBootPV  a vector of length K with the univariate bootstrap $P$-values for each time series, for the Catani and Ahlgren (2016) test.

ET_bootPV  the bootstrap $P$-value of the Eklund and Teräsvirta (2007) test.
MARCH_LM  the LM statistic of the Multivariate LM test for ARCH. See e.g. Lütkepohl (2006, sect. 16.5).
MARCH_PV  the $P$-value of the MARCH LM test statistic.
MARCH_bootPV  the bootstrap $P$-value of the MARCH test.
MARCH_LMStar  the bootstrap LM test statistics for the MARCH test.
description  who ran the test and when.
time  computation time taken to run the test.
call  how the function `actestHI()` was called.

**References**


cointBootTest

See Also

`VARfit` to estimate a VAR(p).

Examples

```r
fit <- VARfit(y = VodafoneCDS, p = 3, const = TRUE, trend = FALSE)
test <- archBootTest(fit = fit, h = 5, B = 199, CA = TRUE, ET = TRUE, MARCH = TRUE, dist = "norm")
```

Description

This function uses the bootstrap and wild bootstrap to test the cointegration rank of a VAR model. The test is an implementation of Cavaliere, Rahbek & Taylor (2012, 2014), and is used in Ahlgren & Catani (2018).

Usage

```r
cointBootTest(y, r = "sequence", p, model = 1, signif = 0.05, dummies = NULL, B = 999,
boot_type = c("b", "WB"), WB_dist = c("rademacher", "normal", "mammen"))
```

Arguments

- `y`: a T x K matrix containing the time series.
- `r`: either "sequence" or a vector of integers (0 <= r <= K - 1, where K is the number of columns in y). If a vector of integers, r is the cointegration rank being tested. If r = "sequence", the bootstrap sequential algorithm will be used (see 'details').
- `p`: the lag order of the model.
- `model`: either 1 (no deterministic terms), 2 (restricted constant), or 3 (restricted linear trend). See 'details' below.
- `signif`: if r = "sequence" is used, signif sets the significance level of the tests in the sequential algorithm.
- `dummies`: (optional) dummy variables. Must have the same number of rows as y. The models will then be estimated with the dummy variables, but the dummy variables are not used in the bootstrap DGP. In the boot_type = "B" version, the residuals used to draw the bootstrap errors do not include rows corresponding to observations where any of the dummies are equal to 1.
- `B`: the number of bootstrap replications.
boot_type

either "B", "WB", or both. "B" uses the iid bootstrap algorithm, while "WB" uses the wild bootstrap algorithm.

WB_dist

The distribution used for the wild bootstrap. Either "rademacher", "normal", or "mammen".

x

Object with class attribute 'cointBootTest'.

... further arguments passed to or from other methods.

Details

Consider the $K$-dimensional heteroskedastic cointegrated VAR model of Cavaliere, Rahbek and Taylor (2014):

$$
\Delta y_t = \alpha \beta' y_{t-1} + \sum_{i=1}^{p-1} \Gamma_i \Delta y_{t-i} + \alpha \rho' D_t + \phi d_t + \varepsilon_t, \quad t = 1, \ldots, T,
$$

where $\alpha$ and $\beta$ are $(K \times r)$ matrices of rank $r < K$, the number $r$ being the cointegration rank. $D_t$ and $d_t$ differ according to the model argument in the following manner:

- model 1: $D_t = 0$ and $d_t = 0$ (no deterministic terms)
- model 2: $D_t = 1$ and $d_t = 0$ (restricted constant)
- model 3: $D_t = 1$ and $d_t = 1$ (restricted linear trend)

The likelihood ratio (LR) statistic for testing cointegration rank $r$ against $K$ is

$$
Q_{r,T} = -T \sum_{i=r+1}^{K} \log(1 - \hat{\lambda}_i),
$$

where the eigenvalues $\hat{\lambda}_1 > \ldots > \hat{\lambda}_K$ are the $K$ largest solutions to a certain eigenvalue problem (see Johansen 1996).


1. Estimate the model under $H(r)$ using Gaussian PMLE yielding the estimates $\hat{\beta}^{(r)}$, $\hat{\alpha}^{(r)}$, $\hat{\rho}^{(r)}$, $\hat{\Gamma}_1^{(r)}, \ldots, \hat{\Gamma}_{p-1}^{(r)}$, and $\hat{\phi}^{(r)}$, together with the corresponding residuals, $\hat{\varepsilon}_{r,t}$.

2. Check that the equation $|\hat{A}^{(r)}(z)| = 0$, with $\hat{A}^{(r)}(z) := (1-z)I_K - \hat{\alpha}^{(r)}z - \sum_{i=1}^{p-1} \hat{\Gamma}_i^{(r)}(1-z)z^i$, has $K-r$ roots equal to 1 and all other roots outside the unit circle. If so, proceed to step 3.

3. Construct the bootstrap sample recursively from

$$
\Delta y_{r,t}^* = \hat{\alpha}^{(r)}\hat{\beta}^{(r)'} y_{r,t-1}^* + \sum_{i=1}^{p-1} \hat{\Gamma}_i^{(r)} \Delta y_{r,t-i}^* + \hat{\alpha}^{(r)}\hat{\rho}^{(r)'} D_t + \hat{\phi}^{(r)} d_t + \hat{\varepsilon}_{r,t}^*, \quad t = 1, \ldots, T,
$$

initialized at $y_{r,j}^* = y_j, j = 1 - p, \ldots, 0$, and with the $T$ bootstrap errors $\hat{\varepsilon}_{r,t}^*$ generated using the residuals $\hat{\varepsilon}_{r,t}$.

The bootstrap errors are generated depending on the boot_type argument in the following manner:

- boot_type = "B": The i.i.d. bootstrap, such that $\varepsilon_{r,t}^* := \hat{\varepsilon}_{r,\mathcal{U}_t}$, where $\mathcal{U}_t, t = 1, \ldots, T$ is an i.i.d. sequence of discrete uniform distributions on $\{1, 2, \ldots, T\}$.
boot_type = "WB": The wild bootstrap, where for each \( t = 1, \ldots, T,\) \( \tilde{\varepsilon}_{t} := \tilde{\varepsilon}_{r,t} w_{t}, \) where \( w_{t}, t = 1, \ldots, T,\) is an i.i.d. sequence distributed according to the wb_dist argument.

4. Using the bootstrap sample, \( \{y_{r,t}\}, \) and denoting by \( \hat{\lambda}_{1}^{*} > \ldots > \hat{\lambda}_{K}^{*} \) the ordered solutions to the bootstrap analogue of the eigenvalue problem, compute the bootstrap LR statistic \( Q_{r,T}^{*} := -\sum_{i=r+1}^{K} \log(1 - \hat{\lambda}_{i}^{*}) \). Define the corresponding \( p \)-value as \( p_{r,T}^{*} := 1 - G_{r,T}^{*}(Q_{r,T}^{*}) \), \( G_{r,T}^{*}(\cdot) \) denoting the conditional (on the original data) cdf of \( Q_{r,T}^{*} \).

5. The bootstrap test of \( H(r) \) against \( H(K) \) at level \( \eta \) rejects \( H(r) \) if \( p_{r,T}^{*} \leq \eta, \) \( r = 0, \ldots, K - 1 \).

If \( r = \) "sequence", the algorithm is repeated for each null hypothesis \( H(r), r = 0, \ldots, K - 1, \) and the first null hypothesis with a \( p_{r,T}^{*} > \eta \) is selected as the cointegration rank. If \( p_{r,T}^{*} \leq \eta, r = 0, \ldots, K - 1, \) the rank selected is \( \hat{r} = K. \)

**Value**

a list of class "cointBootTest".

- **eigen_val** the eigenvalues.
- **eigen_vec** the eigenvectors.
- **alpha** a matrix with the estimated alpha parameters for the model with \( r = K \) (for other values of \( r \), the alpha parameters are the first \( r \) columns of this matrix).
- **beta** a matrix with the estimated beta parameters for the model with \( r = K \) (for other values of \( r \), the beta parameters are the first \( r \) columns of this matrix).
- **gamma** a list of matrices with the estimated gamma parameters. Each parameter matrix corresponds to the model estimated under the null hypothesis in \( r (0:(K-1) \) if \( r = \) "sequence"), in the same order.
- **rho** a matrix with the estimated rho parameters for the model with \( r = K \) (for other values of \( r \), the rho parameters are the first \( r \) columns of this matrix).
- **phi** a list of matrices with the estimated phi parameters. Each parameter matrix corresponds to the model estimated under the null hypothesis in \( r (0:(K-1) \) if \( r = \) "sequence"), in the same order.
- **dummy_coefs** a list of matrices with the estimated dummy parameters. Each parameter matrix corresponds to the model estimated under the null hypothesis in \( r (0:(K-1) \) if \( r = \) "sequence"), in the same order.
- **residuals** a list of residual matrices, one for each model estimated under the null hypothesis in \( r (0:(K-1) \) if \( r = \) "sequence"), in that order.
- **Q** a vector with the Q test statistics. If \( r = \) "sequence", then the first element is for the null hypothesis \( r = 0, \) and the last is for \( r = K - 1. \) Otherwise, the order corresponds to the \( r \) argument.
- **B.Q** a matrix of the iid bootstrap Q statistics. Each column represent the null hypothesis in the order of \( r (0:(K-1) \) if \( r = \) "sequence").
- **WB.Q** a matrix of the wild bootstrap Q statistics. Each column represent the null hypothesis in the order of \( r (0:(K-1) \) if \( r = \) "sequence").
- **B.r** the selected cointegration rank from the iid bootstrap test, if \( r = \) "sequence" were used.
the selected cointegration rank from the wild bootstrap test, if \( r = "\text{sequence}" \) were used.

- `B.pv`: a vector with the bootstrap P.values, in the order of \( r (0:(K-1) \text{ if } r = "\text{sequence}" \).

- `WB.pv`: a vector with the wild bootstrap P.values, in the order of \( r (0:(K-1) \text{ if } r = "\text{sequence}" \).

- `B.errors`: the number of times the bootstrap simulations had to be resimulated due to errors.

- `WB.errors`: the number of times the wild bootstrap simulations had to be resimulated due to errors.

- `companion_eigen`: a list of matrices with the eigenvalues of the companion matrix. The inverse of the eigenvalues are the roots in step 2 of the bootstrap algorithm (see the .pdf version of this help file).

**References**


**Examples**

```r
## Not run:

test <- cointBootTest(y = VodafoneCDS, r = "sequence", p = 2, model = 3, signif = 0.05,
                        dummies = NULL, B = 999, boot_type = c("B", "WB"), WB_dist = "rademacher")
test

## End(Not run)
```

**DataFiles**

*Multiple Time Series Data Set*

**Description**

The data matrix VodafoneCDS is used for the examples. It consists of 804 daily observations, from 1 January 2009 to 31 January 2012, of Vodafone’s Credit Default Swap prices (the ’CDS’ column) and the credit spread on its bond over the risk-free rate (the ’SWSP’ column). For more information, see Ahlgren and Catani (2016), who used the same data set.
References

---

**VARfit**

*(VAR) (Vector Autoregression)* Model Fitting.

**Description**
Estimates a VAR(p) model from data. The function’s purpose is to be used before the `actest` or `archboottest` functions, and doesn’t have the same full functionality as for example the function `VAR` in the `vars` package.

**Usage**

```r
VARfit(y, p = 1, const = TRUE, trend = FALSE, exogen = NULL, univariate = FALSE)
```

**Arguments**
- `y`: an N x K matrix containing the time series to be modeled.
- `p`: the lag length of the VAR(p) process.
- `const`: if TRUE (default) the model will include a constant.
- `trend`: if TRUE the model will include a trend.
- `exogen`: a matrix or vector of exogenous variable(s). Must have the same number of rows as `y` does.
- `univariate`: if TRUE univariate (i.e. an AR(p) model for each timeseries) estimates will also be printed.

**Value**
a list of class "VARfit" with the following slots:
- `y`: the data matrix.
- `p`: the lag length p.
- `N`: the number of observations/rows in the time series 'y'.
- `K`: the number of variables in the time series 'y'.
- `const`: TRUE if the model included a constant.
- `trend`: TRUE if the model included a trend.
- `exogen`: the exogen variable(s).
- `Z`: the \([N - p] \times [Kp + numberOf(const, trend, dummy)]\) design matrix, where the lags of the y matrix are appended.
- `call`: how the function `VARfit()` was called.
- `coef`: a matrix of the estimated parameters.
resid the \([N - p] \times K\) matrix of residuals.
uniCoef a matrix of the estimated parameter for the univariate case (an AR(p) for each column in 'y').
uniResid the \([N - p] \times K\) matrix of residuals from the univariate (AR(p)) cases.
univariate the 'univariate' argument.
NnonLagVar the number of non-endogenous variables (i.e. the number of 'exogen' variables + const + trend).
description who fitted the model and when.
time computation time taken for the estimation.

See Also

ACTest to test for error autocorrelations, and archBootTest to test for ARCH errors.

Examples

```r
fit <- VARfit(y = VodafoneCDS, p = 3, const = TRUE, trend = FALSE)
```

Description

residuals.VARfit() returns the residuals and coef.VARfit() returns the coefficients of a fitted VAR model of class 'VARfit', while print.VARfit() prints the parameters of the estimated model.

Usage

```r
## S3 method for class 'VARfit'
coef(object, ...)
## S3 method for class 'VARfit'
print(x, ...)
## S3 method for class 'VARfit'
residuals(object, ...)
```

Arguments

- `x, object` the fitted VAR model of class 'VARfit' (as returned by the function `VARfit`).
- `...` additional arguments.
**VARsim**

*Simulates vector autoregressive (VAR) series*

**Description**

This function simulates VAR(p) series. For quick use, the function can use the estimated model returned by **VARfit** or **VAR** as the DGP (Data Generating Process), if passed to the **fittedModel** argument.

**Usage**

```
VARsim(N = 200, K = 2, p = 1, const = TRUE, trend = FALSE, exogen = NULL,
        coef = NULL, dist = "normal", Ystart = NULL, errors = NULL, fittedModel = NULL)
```

**Arguments**

- **N**
  - The length of the series.
- **K**
  - The number of series/equations.
- **p**
  - The lag length of the VAR(p).
- **const**
  - if TRUE, a constant will be used.
- **trend**
  - if TRUE, a trend will be used.
- **exogen**
  - a matrix of exogenous variables. Should either have \( N \) or \( (N - p) \) number of rows. If the latter, the last row will be matched with the last row of the simulated series \( y \).
- **coef**
  - a matrix of coefficients. E.g. a 2 dimensional VAR(2) with a constant, a trend and one exogenous variable must be entered in the following order (the same as returned by **VARfit**):
    ```
    \[
    \begin{bmatrix}
    y1 & y2 \\
    \text{const} & \text{const} \\
    \text{trend} & \text{trend} \\
    \text{exo1} & \text{exo1} \\
    y1[-1] & y1[-1] \\
    y2[-1] & y2[-1] \\
    y1[-2] & y1[-2] \\
    y2[-2] & y2[-2]
    \end{bmatrix}
    \]
    ```
- **dist**
  - the distribution of the error terms (currently only "normal" for i.i.d. standard normal.)
- **Ystart**
  - a \( p \times K \) matrix of the start values. These will be present in the simulated series. If NULL, the start values will be set to zero.
- **errors**
  - (optional) a matrix of error terms. If supplied, **VARsim()** will use those instead of sampling the errors. Must have \( K \) columns and either \( N \) or \( (N - p) \) number of rows. If the latter, the last row will be matched with the last row of the simulated series \( y \).
fittedModel (optional) either an object of class "vars", as returned by \texttt{VAR}, or an object of class "VARfit", as returned by \texttt{VARfit}. If used, the estimated model of the fittedModel will be used as a DGP (Data Generating Process). Unless the arguments \texttt{N}, \texttt{exogen} and/or \texttt{Ystart} were supplied, they too will be taken from the fittedModel object.

Value

an \( N \times K \) matrix of the simulated \( \text{VAR}(p) \).

Examples

# Simulates from a fitted DGP:
fit <- \texttt{VARfit(y = VodafoneCDS, p = 1, const = TRUE)}
y <- \texttt{VARsim(fittedModel = fit)}
VARfit(y = y, p = 1, const = TRUE)

# Simulates from user given parameters.
# Includes an exogenous variable:
VARcoef <- \texttt{matrix(c(1, 2, 1, 0.5, 0.1, 0.3, 0.2, 0.5), ncol = 2, byrow = TRUE)}
exo <- \texttt{matrix(rnorm(500))}
y <- \texttt{VARsim(N = 500, K = 2, p = 1, const = TRUE, trend = FALSE, exogen = exo, coef = VARcoef, dist = "normal")}
VARfit(y = y, p = 1, const = TRUE, exogen = exo)

## Not run:

## End(Not run)
Usage

wildBoot(test, WBtype = c("recursive", "fixed"), B = 199,
        WBdist = c("rademacher", "normal", "mammen"),
        HCTYPE = c("LM", "HC0", "HC1", "HC2", "HC3"), univariate = FALSE)
# S3 method for class 'wildBoot'
print(x, ...)

Arguments

test An object of class ACtest.
WBtype set this to "recursive" for the recursive version of the test, "fixed". for the fixed version, or c("recursive", "fixed") to run both versions.
B the number of bootstrap simulations (default is 199).
WBdist the distribution used for the wild bootstrap error terms. Either "rademacher", "normal" or "mammen".
HCTYPE a vector of the HCTYPE’s used for the WB test. The types must have been used in the test object.
univariate either FALSE, TRUE, or "only". If TRUE, the test will also be performed univariately, if "only", only the univariate version will be used. For TRUE and "only", the univariate version must have been used in the test object.
x Object with class attribute ‘wildBoot’.
... further arguments passed to or from other methods.

Value

da list of class "wildBoot".

test the test argument object.
WBtype the WBtype input.
B the number of bootstrap simulations.
WBdist the distribution used for the wild bootstrap error terms.
HCTYPE the HCTYPE’s used for the WB test.
description who ran the function and when.
time computation time taken to run the WB test.
call how the function wildboot() was called.
numberOfErrors number of times the bootstrap simulations gave an error (can happen if one of the matrices becomes computational singular). If this happens, the particular bootstrap sample will be resimulated.
numberOfNA number of times the test from the bootstrap simulations returned NA. If this happens, the particular bootstrap sample will be resimulated.
WBr.Q a B x 5 matrix of the recursive wild bootstrap (if used) Q statistics. Each row is the Q statistics from a bootstrap sample.
WBr.pv the P. values from the recursive design wild bootstrap (if used).
wildBoot

\texttt{Wbf.Q} a B x 5 matrix of the \textbf{fixed} wild bootstrap (if used) Q statistics. Each row is the Q statistics from a bootstrap sample.

\texttt{Wbf.pv} the P. values from the \textbf{fixed} design wild bootstrap (if used).

\texttt{unilist} a list of the corresponding above items (\texttt{Wbf.Q}, \texttt{Wbf.pv}, \texttt{Wbf.pv}) for each respective univariate test. The elements of the list is named "uni1WBr.Q", "uni2WBr.pv", etc.

References

See Also
\texttt{actest} for the test that is required to be run before the \texttt{wildBoot} function.

Examples

```r
## Not run:
fit <- VARfit(y = VodafoneCDS, p = 3, const = TRUE, trend = FALSE)
test <- ACTest(fit = fit, h = 1, HCTYPE = c("LM", "HC0", "HC1", "HC2", "HC3"))
w <- wildBoot(test = test, WBTYPE = "recursive", B = 199,
                WBDIST = "rademacher", HCTYPE = c("LM", "HC3"))

## End(Not run)
```
Index

ACtest, 2, 13, 14, 16–18
archBootTest, 5, 13, 14

coef.VARfit (VARfit-methods), 14
cointBootTest, 9

DataFiles, 12

print.ACtest (ACtest), 2
print.archBootTest (archBootTest), 5
print.cointBootTest (cointBootTest), 9
print.VARfit (VARfit-methods), 14
print.wildBoot (wildBoot), 16

residuals.VARfit (VARfit-methods), 14
rmst, 5

VAR, 5, 13, 15, 16
VARfit, 2, 4, 5, 9, 13, 14–16
VARfit-methods, 14
VARsim, 15
VodafoneCDS (DataFiles), 12

wildBoot, 2, 4, 16