Package ‘uroot’

June 24, 2019

Version 2.0-10
Date 2019-06-18
Title Unit Root Tests for Seasonal Time Series
Depends R (>= 3.0.0), stats
SystemRequirements nvcc (release >= 7.1) (NVIDIA Cuda Compiler driver)
Encoding UTF-8
Description Seasonal unit roots and seasonal stability tests.
P-values based on response surface regressions are available for both tests.
P-values based on bootstrap are available for seasonal unit root tests.
A parallel implementation of the bootstrap method requires a CUDA capable GPU
with compute capability >= 3.0, otherwise a debugging version fully coded in R is used.
NeedsCompilation yes
LazyData yes
LazyDataCompression xz
License GPL-2
Author Javier López-de-Lacalle [aut],
Georgi N. Boshnakov [cre]
Maintainer Georgi N. Boshnakov <georgi.boshnakov@manchester.ac.uk>
Repository CRAN
Date/Publication 2019-06-24 18:20:03 UTC

R topics documented:

uroot-package .......................................................... 2
bgt.data ............................................................ 3
ch.data ............................................................ 3
ch.rs.pvalue ........................................................ 4
ch.test ............................................................. 5
hegy.boot.pval ...................................................... 7
hegy.rs.pvalue ...................................................... 9
hegy.test .......................................................... 10
seasonal.dummies ................................................... 14
uroot.raw.pvalue .................................................. 15
Index

uroot-package  Unit Root Tests for Seasonal Time Series

Description

Canova and Hansen (CH) test for seasonal stability and Hylleberg, Engle, Granger and Yoo (HEGY) test for seasonal unit roots.

Details

Version >= 2.0.0 is a revival of the former package uroot. Some of the functions provided in the original versions have been coded from the scratch.

The current version provides the Canova and Hansen (CH) test for seasonal stability and the Hylleberg, Engle, Granger and Yoo (HEGY) test for seasonal unit roots.

New features:
The original functions have been enhanced with the following new features: 1) the tests are now applicable to series of any seasonal periodicity (not only quarterly and monthly data), 2) p-values based on response surface regressions are available, 3) bootstrapped p-values are available for the HEGY test statistics.

Old features not currently supported:
Some of the utilities available in the initial versions of the package are not available now: graphics for seasonal series, graphical user interface, bootstrap versions of the statistics, ADF test and KPSS test. Some of these utilities are available in other packages and some will probably be incorporated in future versions of this package.

System requirements:
Windows systems: GPU parallelization of the bootstrap is not currently operational on windows systems.
Unix systems: GPU parallelization of the bootstrap requires a CUDA capable GPU with compute capability >= 3.0.

Author(s)

Javier López-de-Lacalle <javlacalle@yahoo.es>.

References


---

**bgt.data**

*BGT-data Sample Data Set*

**Description**


**Usage**

`bgt.data`

**Format**

A list containing time series objects.

**References**


---

**ch.data**

*CH-data Sample Data Set*

**Description**

U.S. post World War II macroeconomic time series. The data set is described and employed in Canova and Hansen (JBES, 1995, Section 5.1).

**Usage**

`ch.data`

**Format**

A list containing time series objects.
Source

http://www.ssc.wisc.edu/~bhansen/progs/jbes_95.html

---

**ch.rs.pvalue**

*P-values for the CH test statistic*

---

**Description**

Compute p-values for the Canova and Hansen (CH) test statistic based on the response surface regressions approach.

**Usage**

```
ch.rs.pvalue(x, type, lag1, S, n, nobsreg, VMdf)
```

**Arguments**

- `x`: a numeric. The value of the CH statistic.
- `type`: a string specifying the formulation of the test, "dummy" for seasonal dummies or "trigonometric" for seasonal cycles.
- `lag1`: logical indicating whether a first order lag of was included in the regression model.
- `S`: numeric, the periodicity of the data.
- `n`: numeric, the number of observations.
- `nobsreg`: an integer indicating the number of points employed in the response surface regression.
- `VMdf`: numeric, the degrees of freedom of the Von Mises distribution.

**Details**

Ported from Gretl code provided by Díaz-Emparanza and Moral (2013).

For type="dummy", the p-value for the joint test statistic is not available.

This function is mainly intended to be used internally by `ch.test`.

**Value**

A numeric giving the calculated p-value.

**References**

See Also

ch.test.

Ch.test

Canova and Hansen Test for Seasonal Stability

Description

Canova and Hansen (CH) test statistic for the null hypothesis of a stable seasonal pattern.

Usage

ch.test(x, type = c("dummy", "trigonometric"), lag1 = FALSE, NW.order = NULL, sid = NULL, xreg = NULL, pvalue = c("RS", "raw"), rs.nobsreg = 13)

Arguments

x a univariate seasonal time series.

type a character string specifying the formulation of the test, "dummy" for seasonal dummies or "trigonometric" for seasonal cycles.

lag1 logical, if TRUE a first order lag of the time series x is included in the regression model. The default is FALSE.

NW.order an integer, the lag truncation parameter to be used in the Newey and West covariance matrix.

sid an optional numeric vector, the target seasonal dummies or cycles to be tested. By default all the individual and joint test statistics are returned.

xreg an optional vector or matrix of external regressors with the same length or number of rows as the length of the input time series x.

pvalue a character specifying the method employed to compute p-values: "RS", the default, interpolation based on response surface regressions; "raw", interpolation in the tabulated values provided in the reference paper for the Von Mises distribution.

rs.nobsreg an integer indicating the number of points employed in the response surface regression (only for pvalue = "RS").

Details

The seasons or seasonal cycles to be tested can be chosen through an indicator variable defined in the argument sid. By default, all the t-statistics related to each individual dummy or cycle and the joint F-statistic are returned.

If type = "dummy", the index of the target seasons can be specified in sid. For example, in a quarterly series: sid=2 returns the test statistic to the stability of the second quarter; sid=c(1,3)
returns the joint test statistic for the first and third quarters; \texttt{sid=c(1,2,3,4)} returns the joint test statistic for the null of seasonal stability at all seasons.

If \texttt{type = "trigonometric"}, the indicator vector \texttt{sid} must be of length \texttt{floor(frequency(x)/2)} and will consist of ones and zeros. Each element in \texttt{sid} is related to each seasonal cycle according to the same order in which the seasonal frequencies, \( w_j \), are defined: \( w_j = 2\pi j/S, \ j = 1, \ldots, Sh \), where \( S \) is the periodicity and \( Sh \) is \texttt{floor(frequency(x)/2)}. For example, in a monthly series: \texttt{sid=c(0,0,0,0,1)} returns the test statistic to the stability of the cycle with frequency \( w_6 = \pi \); \texttt{sid=c(1,0,0,0,1)} returns the joint test statistic for cycles related to frequencies \( w_1 = \pi/6 \) and \( w_6 = \pi \); \texttt{sid=c(1,1,1,1,1)} returns the joint test statistic for the stability of all seasonal cycles.

The following keywords are also admitted: \texttt{sid="all"}, computes all the test statistic related to each individual season or cycle as well as the joint test statistic for all seasons or cycles; \texttt{sid="joint"} computes the joint test statistic for all seasons or cycles.

\textbf{Value}

A list of class \texttt{"CHtest"} with components:

- \texttt{statistics} the value of the test statistics.
- \texttt{pvalues} the p-values for each test statistics.
- \texttt{method} a character string describing the type of test.
- \texttt{data.name} a character string giving the name of the data.
- \texttt{type} the value of the input argument type.
- \texttt{fitted.model} the fitted regression model.
- \texttt{NW.order} the value of the input argument \texttt{codeNW.order}.
- \texttt{isNullxreg} logical, auxiliary element for \texttt{print} method. Were external regressors defined in the argument \texttt{xreg}?
- \texttt{type.pvalue} character, the value of the input argument \texttt{codepvalue}.
- \texttt{pvlabels} a vector of characters containing a label related to each p-values. Auxiliary element for \texttt{print} method.

The \texttt{method print} displays the test statistics and p-values; \texttt{summary} shows the same output and includes the fitted regression model.

\textbf{Note}

When \texttt{type = "dummy"}, the p-value for the joint test statistic based on response surface regressions is not available. If \texttt{pvalue = "RS"}, the p-value reported for the joint test statistic in the trigonometric version is based on the tables given in the reference paper, Canova and Hansen (1995).

When \texttt{sid} is a numeric (selected combination of dummies or cycles), the reported p-values are based on interpolation in tables; if \texttt{pvalue = "RS"}, it is changed to \texttt{"raw"} and a warning is given.
References


See Also

- ch.rs.pvalue seasonal.cycles, seasonal.dummies, uroot.raw.pvalue.

Examples

```r
library(uroot)
# example for the series "hours" with the same options
# employed in Canova and Hansen (1995)
data("ch-data")
hours <- diff(log(ch.data$hours))
res1 <- ch.test(x = hours, type = "dummy", lag1 = TRUE, NW.order = 4)
res1
# the auxiliary regression is stored in the element "fitted.model"
summary(res1$fitted)

## Not run:
# this requires tables not included in the current version of the package
# see note in main documentation file, uroot-package
res2 <- ch.test(x = hours, type = "trigonometric", lag1 = TRUE, NW.order = 4)
res2
summary(res2$fitted)
## End(Not run)
```

---

**hegy.boot.pval**  
*Bootstrapped P-Values for the HEGY Test Statistics*

**Description**

Compute p-values for the HEGY test statistics by means of bootstrap.

**Usage**

```r
hegy.boot.pval(x, model0, stats0,  
deterministic = c(1,0,0), lag.method = c("fixed", "AIC", "BIC"), maxlag = 0,  
byseason = FALSE, nb = 500, u = NULL, debug.tid = -1)
```
Arguments

- **x**: a univariate seasonal time series.
- **model0**: the fitted model returned by `hegy.test` for the original data.
- **stats0**: the statistics returned by `hegy.test` for the original data.
- **deterministic**: a vector of length three containing zeros or ones to indicate, respectively, whether a constant, a trend or seasonal dummies are included in the regression equation of the test.
- **lag.method**: a character specifying the lag order selection method.
- **maxlag**: the maximum lag order to be considered by `lag.method`.
- **byseason**: logical, should the residuals be resampled by season? If TRUE, the residuals are split by the season they belong to and resampled accordingly; otherwise, the entire series of residuals is resampled regardless of the season they belong to.
- **nb**: the number of bootstrap replicates.
- **u**: optional matrix of integers giving the indices of the resampled residuals. Intended for debugging.
- **debug.tid**: numeric, if positive, the bootstrap replicate of the data generated at iteration `debug.tid` is returned (intended for debugging).

Details

See `hegy.test` for further details about the arguments that have the same name in both functions (deterministic, `lag.method`, `maxlag`).

Bootstrapped p-values follow the approach described in Burridge and Robert Taylor (2004), except that here, the residuals are resampled regardless of the season they belong to.

Value

A numeric vector containing the p-values of the test statistics. The vector is named following the same convention as `statistics` and `pvalues` returned by `hegy.test`.

If the number of bootstrap replicates is `nb = 1`, the resampled series is returned (relevant for inspection of how the resampled series look like and for debugging).

References


See Also

`hegy.test`. 
**Examples**

```r
## Not run:
hegy.test(x, deterministic = c(1,1,1), lag.method = "fixed", maxlag = 1,
pvalue = "bootstrap")
## alternatively, full R non-parallel version
hegy.boot.pval(x, res$fit, res$stat, deterministic = c(1,1,1),
    lag.method = "fixed", maxlag = 1, nb = 1000)
## End(Not run)
```

---

**hegy.rs.pvalue**

*P-values based on response surface regressions for the HEGY test statistics*

**Description**

Compute p-values for the Hylleberg, Engle, Granger and Yoo (HEGY) test statistic by interpolation in precompiled response surfaces.

**Usage**

```r
hegy.rs.pvalue(x, type = c("zero", "pi", "pair", "seasall", "all"),
deterministic = c(1, 0, 0), lag.method = c("fixed", "AIC", "BIC"),
lag.order, s, n, nobsreg)
```

**Arguments**

- **x**
  - a numeric, the value of the HEGY statistic.
- **type**
  - a character, the type of test statistic, the regressor(s) to which the statistic is related.
- **deterministic**
  - a vector of length three containing zeros and ones to indicate, respectively, whether a constant, a trend or seasonal dummies should be included in the regression equation of the test.
- **lag.method**
  - a character specifying the lag order selection criterion.
- **lag.order**
  - a numeric, the lag order employed in the auxiliary regression.
- **s**
  - numeric, the periodicity of the data.
- **n**
  - numeric, the number of observations.
- **nobsreg**
  - an integer indicating the number of points employed in the response surface regression.
Details

Ported from Gretl code provided by Díaz-Emparanza (2014).

The original source includes tables with coefficients for the calculation of p-values when the HQC lag order selection criterion is used. These tables are not included here.

The case with no deterministic terms (deterministic = c(0, 0, 0)) is not considered.

This function is mainly intended to be used internally by `hegy.test`.

Value

A numeric, the p-value.

References


See Also

`hegy.test`.

---

**hgy.test**

*Hylleberg, Engle, Granger and Yoo Test for Seasonal Unit Roots*

Description

Hylleberg, Engle, Granger and Yoo (HEGY) test statistics for the null hypothesis seasonal unit roots.

Usage

```r
ehegy.test(x, deterministic = c(1, 0, 0),
         lag.method = c("fixed", "AIC", "BIC", "AICc"), maxlag = 0,
         pvalue = c("RS", "bootstrap", "raw"), rs.nobsreg = 15,
         boot.args = list(seed = 123, lag.method = lag.method[1], maxlag = maxlag,
                           byseason = FALSE, nb = 1000, BTdim = c(100, 10), debug.tid = -1))
```

`hegy.regressors(x)`
Arguments

Arguments

- **x**: a univariate seasonal time series.
- **deterministic**: a vector of length three containing zeros or ones to indicate, respectively, whether a constant, a trend or seasonal dummies are included in the regression equation of the test.
- **lag.method**: a character specifying the lag order selection method.
- **maxlag**: the maximum lag order to be considered by lag.method.
- **pvalue**: a character specifying the method employed to compute p-values: "RS", the default, interpolation based on response surface regressions; "bootstrap", bootstrap; "raw" interpolation in the tables provided in the reference papers.
- **rs.nobsreg**: an integer indicating the number of points employed in the response surface regression (only for pvalue = "RS").
- **boot.args**: a list containing the parameters employed in the bootstrap. See details below.

Details

The regression equation employed to obtain the tests statistics may include the following deterministic terms: a constant, a linear trend, seasonal dummies. These terms are selected by setting to 1 the corresponding element in the vector deterministic: deterministic = c(0,0,0) no deterministic components, deterministic = c(1,0,0) includes a constant, deterministic = c(1,0,1) a constant and seasonal dummies, deterministic = c(1,1,0) a constant and a trend and deterministic = c(1,1,1) includes the three components.

The regression equation may include lags of the dependent variable. When lag.method = "fixed", the lag order is fixed to maxlag; otherwise, maxlag is the maximum number of lags considered in a lag selection procedure that minimises the lag.method criterion, which can be AIC or BIC or corrected AIC, AICC, obtained as $AIC + \frac{2k(k+1)}{n-k-1}$, where $k$ is the number of parameters and $n$ is the number of available observations in the model.

Response surface based p-values, pvalue="RS", is not available with option lag.method = "AICC".

P-values based on response surface regressions follow the method described in MacKinnon (1994), Harvey+vanDijk (2006) and Díaz-Emparanza (2014). Gretl code provided by Díaz-Emparanza (2014) has been ported to function hegy.rs.pvalue. Note: the case deterministic = c(0,0,0) is not considered; NAs are returned if p-values based on response surfaces are selected for that case.

Bootstrapped p-values follow the approach described in Burridge and Robert Taylor (2004). The following arguments can be defined in boot.args:

- **seed**: a numeric, the seed of the random generator employed for resampling the residuals.
- **lag.method**: a character, the lag order selection method. This is the same as lag.method employed in the original regression, except that here, the AIC, BIC and AICC are obtained upon the residual sums of squares rather than the likelihood value.
- **maxlag**: a numeric, maximum number of lags. Same behaviour as maxlag explained above for the original regression.
- **byseason**: logical, should the residuals be resampled by season? If TRUE, the residuals belonging to the same season are resampled (at each time $t$, the residuals belonging to the corresponding season at time $t$ are resampled); otherwise, the entire series of residuals is resampled regardless of the season they belong to.
• `nb`: a numeric, the number of bootstrap replicates.

• `Bdim`: a vector of length two containing the number of blocks and the number of threads per block to be run on parallel on the GPU. The product of these two elements must be equal or greater than `nb`.

• `debug_tid`: an integer employed for debugging. Currently ignored.

By default `boot.args$lag.method` and `boot.args$maxlags` are set equal to the same options employed for the original data in arguments `lag.method` and `maxlag`; if the default options are desired, these values need not be explicitly defined in the list `boot.args`.

The standard definition of the AIC, BIC and AICc criteria is used for the original series. For the bootstrapped series, these criteria are defined upon the residual sum squares of the model:

\[
\text{AIC} = n \log \left( \frac{\text{RSS}}{n} \right) + 2k
\]

\[
\text{BIC} = n \log \left( \frac{\text{RSS}}{n} \right) + k \log(n)
\]

\[
\text{AICc} = \text{AIC} + \frac{2k(k + 1)}{n - k - 1}
\]

where `RSS` is the residual sum of squares, `k` is the number of parameters in the model and `n` is the number of available observations.

Given a maximum lag, `maxlag`, the first `maxlag` observations are removed before fitting the models to be compared by the selected criterion. In this way, all the models contain the same number of observations. Once the lag order is selected, the statistics are obtained using the entire sample (except those observations that are missed due to the eventually chosen lags).

The HEGY regressors were originally proposed in Hylleberg et al. (1990) for quarterly data. They are generalized to monthly series in Beaulieu and Miron (1993) and to weekly data in Cáceres (1996). Franses and Hobijn (1997) show tabulated values for bimonthly and biannual data as well as quarterly and monthly data. `hegy_regressors` follows the expressions given in Smith et al. (2009), which define the regressors for a general periodicity of the time series.

**Value**

`hegy.test` returns a list of class "HEGYtest" with the following components:

- `statistics`: the value of the test statistics.
- `pvalues`: the p-values for each test statistic.
- `method`: a character string describing the type of test.
- `data.name`: a character string giving the name of the data.
- `fitted.model`: the fitted regression model.
- `lag.method`: a character, the lag order selection criterion.
lag.order    a numeric, the number of lags included in the regression.

strdet    a character, auxiliary element for print describing the deterministic elements that were selected.

type.pvalue    a character, the value of the input argument codepvalue.

bootstrap    a list, parameter options employed in the bootstrap (if pvalue = "bootstrap").

boot.chosen.lags    a vector, the lag orders chosen for each bootstrap replicate (if pvalue = "bootstrap" and boot.args$lag.method != "fixed").

type.pvalue    a character, the value of the input argument codepvalue.

bootstrap    a list, parameter options employed in the bootstrap (if pvalue = "bootstrap").

boot.chosen.lags    a vector, the lag orders chosen for each bootstrap replicate (if pvalue = "bootstrap" and boot.args$lag.method != "fixed").

type.pvalue    a character, the value of the input argument codepvalue.

bootstrap    a list, parameter options employed in the bootstrap (if pvalue = "bootstrap").

boot.chosen.lags    a vector, the lag orders chosen for each bootstrap replicate (if pvalue = "bootstrap" and boot.args$lag.method != "fixed").

bootstrap    a list, parameter options employed in the bootstrap (if pvalue = "bootstrap").

boot.chosen.lags    a vector, the lag orders chosen for each bootstrap replicate (if pvalue = "bootstrap" and boot.args$lag.method != "fixed").

pvalabels    a vector of characters containing a label related to each p-values. Auxiliary element for print method.

The method print displays the test statistics and p-values; summary shows the same output and includes the fitted regression model; residuals returns the residuals from the regression model fitted to the original data.

hegy.regressors returns a matrix containing the HEGY regressors which are used to test null of unit root at different frequencies.

References


### See Also

`hegy.rs.pvalue`, `uroot.raw.pvalue`.

### Examples

```r
x <- bgt.data[["LCONSEXPCO"]]
hegy.test(x, deterministic = c(1,1,1), lag.method = "fixed", maxlag = 1)

## Not run:
# this requires CUDA capable GPU
hegy.test(x, deterministic = c(1,1,1), lag.method = "fixed", maxlag = 1,
  pvalue = "bootstrap")
## End(Not run)
```

---

### seasonal.dummies  
*Seasonal Dummies and Seasonal Cycles*

#### Description

Generate variables of seasonal dummies and seasonal cycles.

#### Usage

```r
seasonal.dummies(x)  
seasonal.cycles(x)
```

#### Arguments

- **x**  
a univariate seasonal time series.

#### Value

A multivariate time series containing the dummies or cycles by columns.

#### Examples

```r
# In terms of model fitting
# both sets of variables are equivalent
x <- diff(log(AirPassengers))
sd <- seasonal.dummies(x)
fit1 <- lm(x ~ sd[,1])
summary(fit1)
sc <- seasonal.cycles(x)
fit2 <- lm(x ~ sc)
summary(fit1)
all.equal(fitted(fit1), fitted(fit2))
```
uroot.raw.pvalue  

Original Tables of Critical Values

Description

Compute p-values by interpolation in the tables of critical values provided in the original references given below.

Usage

```r
uroot.raw.pvalue(x, type = c("CH", "HEGY"), v, n, ctd, S, Ftpi)
```

Arguments

- `x` a numeric. The value of the CH statistic.
- `type` a character specifying the type of test statistic.
- `v` numeric, the degrees of freedom of the Von Mises distribution. Only for `type="CH"`.
- `n` numeric, the number of observations.
- `ctd` a character indicating the deterministic elements that were included in the HEGY regression. This argument is defined as `paste(deterministic, collapse = "")`, where deterministic is the argument of that name that was passed to `hegy.test`. Only for `type="ADF"` or `type="HEGY"`.
- `S` numeric, the periodicity of the data.
- `Ftpi` a character indicating whether the type of statistic: "zero", t-test for the zero frequency; "pi", t-test for the frequency \( \pi \); "pair", F-test for the pairs of complex conjugates frequencies. Only for `type="ADF"` or `type="HEGY"`.

Details

This function is used internally by `ch.test` and `hegy.test`.

Value

A numeric giving the calculated p-value.

References


See Also

ch.test, hegy.test.
Index

*Topic datasets
  bgt.data, 3
  ch.data, 3

*Topic package
  uroot-package, 2

*Topic ts
  ch.rs.pvalue, 4
  ch.test, 5
  hegy.boot.pval, 7
  hegy.rs.pvalue, 9
  hegy.test, 10
  seasonal.dummies, 14
  uroot-package, 2
  uroot.raw.pvalue, 15

AIC, 11

bgt.data, 3
BIC, 11

ch.data, 3
ch.print (ch.test), 5
ch.rs.pvalue, 4, 7
ch.summary (ch.test), 5
ch.test, 4, 5, 5, 15, 16

hegy.boot.pval, 7
hegy.print (hegy.test), 10
hegy.regressors (hegy.test), 10
hegy.residuals (hegy.test), 10
hegy.rs.pvalue, 9, 11, 14
hegy.summary (hegy.test), 10
hegy.test, 8, 10, 10, 15, 16

seasonal.cycles, 7
seasonal.cycles (seasonal.dummies), 14
seasonal.dummies, 7, 14

uroot-package, 2
uroot.raw.pvalue, 7, 14, 15