Package ‘pdR’
November 18, 2018

Type Package
Title Threshold Model and Unit Root Tests in Cross-section and Time Series Data
Version 1.6
Date 2018-11-18
Author Ho Tsung-wu
Maintainer Ho Tsung-wu <tsungwu@ntnu.edu.tw>
License GPL (>= 2)
LazyLoad yes
Depends plm, tcltk
Imports boot, car, coefplot, lmtest, sandwich, papeR
NeedsCompilation no
Repository CRAN
Date/Publication 2018-11-18 10:20:03 UTC

R topics documented:

pdR-package ................................................................. 2
bank_income ................................................................. 3
cigaretts ................................................................. 4
contts ................................................................. 5
crime ................................................................. 6
dur_john ................................................................. 7
hegy.reg ................................................................. 8
HEGY.test ................................................................. 8
iClick.plm1way ................................................................. 10
iClick.plm2way ................................................................. 11
Panel Data Regression: Threshold Model and Unit Root Tests

**Description**

Functions for analysis of panel data, including the panel threshold model of Hansen (1999, JE), panel unit root test of Chang (2002, JE) based upon instruments generating functions (IGF), and panel seasonal unit root test based upon Hylleberg et al. (1990, JE).

**Details**

This version offers formatted output. This package designs a specification function `ptm()` to estimate the panel threshold model of Hansen (1999). The key feature of `ptm()` is to generalize Hansen’s original code to allow multiple (more-than-one) regime-dependent right-hand-side independent variables; Dr. Hansen’s original code admits only 1 regime-dependent right-hand-side independent variable. This version also includes panel unit root tests based on the instrument generating functions (IGF), proposed by Chang (2002, J. of Econometrics), and the panel version of Hylleberg et al. (1990) seasonal unit root test, proposed by Otero, et al. (2005, 2007).

Package: pdR
Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.
Maintainer: Ho Tsung-wu <tsungwu@ntnu.edu.tw>

References


bank_income

Panel data of bank, 2001Q1~2010Q1

Description

A quarterly panel data frame with 1000 observations on the following 7 variables, unbalanced panel data ranges from 2001Q1~2010Q1.

Usage

data("bank_income")

Format

ID a numeric vector
Qtr a numeric vector
preTax_Income a numeric vector
shortRatio a numeric vector
longRatio a numeric vector
Current_ratio a numeric vector
LoanDeposit_ratio a numeric vector
**Examples**

```r
data(bank_income)
```

---

### cigaretts

**Cigaretts consumption of US states**

---

**Description**

Cigaretts consumption of US states

**Usage**

```r
data(cigaretts)
```

**Format**

A data frame of 48 US states’ cigaretts consumption

- **State**: State abbreviation, N
- **Year**: Year, t
- **Y_SALES**: Cigarette sales in packs per capita, deflated by population
- **X1_PRICE**: P=Real price per pack of cigarettes, deflated by 1983 CPI.
- **X2_PIMIN**: Real minimum price in adjoining states per pack of cigarettes, deflated by CPI
- **X3_NDI**: Per capita disposable income

**References**


**Examples**

```r
data(cigaretts)
head(cigaretts)
```
Function for extracting components from a lm object

Description

Extract the standard error and t-stat of the a-th parameter estimate of a lm object

Usage

contts(lm, a)

Arguments

lm       lm object
a        The a-th parameter estimate of a linear model regression

Value

se.coef  The standard error of the selected coefficient
t.stat   The t-stat of the selected coefficient

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle.

References


Examples

x=rnorm(100)
y=1+0.2*x+rnorm(100)
LMout=lm(y~x)
contts(LMout, 1)

#$se.coef
#[1] 0.1081023

#$t.stat
#(Intercept)  
# 10.60401
**Annual crime dataset of US counties**

**Description**

Annual crime dataset of US counties

**Usage**

data(crime)

**Format**

A data frame of US counties

- **county**: counties index, N
- **year**: Year, t
- **crmrte**: crime rate (crime/population)
- **prbarr**: probability of arrest (arrests/offenses)
- **prbconv**: probability of conviction, given arrest
- **prbpris**: probability of a prison, given conviction
- **avgsen**: sanction severity (average prison sentence in days)
- **polpc**: ability of police force to detect crime (# of police per capita)
- **density**: population density (POP/area)
- **taxpc**: Taxpayment per capita
- **region**: region index of county
- **smsa**: =1 if SAMA, POP>50000; =0 else
- **pctmin**: See Baltagi(2006) for details
- **wcon**: See Baltagi(2006) for details
- **wtuc**: See Baltagi(2006) for details
- **wtrd**: See Baltagi(2006) for details
- **wfr**: See Baltagi(2006) for details
- **wser**: See Baltagi(2006) for details
- **wmfg**: See Baltagi(2006) for details
- **wfed**: See Baltagi(2006) for details
- **wsta**: See Baltagi(2006) for details
- **wloc**: See Baltagi(2006) for details
- **mix**: See Baltagi(2006) for details
- **pctymle**: See Baltagi(2006) for details
References


The cross-country growth data in Durlauf and Johnson(1995)

Description

The Durlauf-Johnson data manipulated by Hansen(2000), excluding missing variables and oil states.

Usage

data(dur_john)

Format

A data frame with 19 countries

gdpGrowth  Economic growth measured by GDP of 1960 and 1985
logGDP60  log Per capita GDP in 1960
Inv_GDP  Average ratio of investment (including Government Investment) to GDP from 1960 to 1985
popGrowth  Average growth rate of working-age population 1960 to 1985
School  Average fraction of working-age population enrolled in secondary school from 1960 to 1985
GDP60  Per capita GDP in 1960
Literacy  fraction of the population over 15 years old that is able to read and write in 1960

Details


Examples

data(dur_john)
head(dur_john)
**hegy.reg**

*Generate the HEGY regressors.*

**Description**

This function generates the level regressors in HEGY regression, without differenced lag terms.

**Usage**

```r
hegy.reg(wts)
```

**Arguments**

- `wts` Univariate time series, with a possibly seasonal stochastic trend

**Details**

This function automatically identifies the frequency of time series data, and generate necessary level components as described in Eq.(3.7) of Hylleberg et al (1990).

**Author(s)**

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle

**References**


**Examples**

```r
data(inf.Q)
y = inf.Q[,1]
hegy.reg(y)
```

---

**HEGY.test**

*Seasonal unit root test based on Hylleberg et al. (1990)*

**Description**

The function performs seasonal unit root test based on Eq.(3.6) of Hylleberg et al. (1990), univariate time series.

**Usage**

```r
HEGY.test(wts, itsd, regvar = 0, selectlags = list(mode = "signf", Pmax = NULL))
```
Arguments

- **wts**: Univariate time series
- **itsd**: Options for \( c(i,t,sd) \)
  - \( i=1 \), intercept; \( =0 \) no intercept
  - \( t=1 \), trend; \( =0 \) no deterministic trend
  - \( sd=1 \), season dummy \( 1:(s-1);=0 \) no.

- **regvar**: Additional regressors

- **selectlags**: Selection of lags
  - Mode. Criteria for selection, having three options: "signf", "bic", "aic".
  - \( P_{max} \), maximum number of lags.

Details

Mode for selectlags has three options, AIC and BIC use R built-in functions for linear model and their meanings are popular and straightforward. They include only lags that meet specific criterion, others are dropped from regressors. That is, lag orders of your model may not be a regular sequence. See also selPsignf() and selPabic().

Value

- **stats**: Tests statistics for HEGY regression coefficients.
- **hegycoefs**: HEGY regression coefficients.
- **lagsorder**: Lags order. "aic" or "bic" returns a scalar; "signf" returns a sequence of numbers.
- **lagcoefs**: Coefficients of lag terms.
- **regvarcoefs**: Coefficient(s) of additional regressor(s).

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle

References


Examples

```r
data(inf.Q)
y<-inf.Q[,1]
hegy.out<-HEGY.test(wts=y, itsd=c(1,0,c(1:3)), regvar=0, selectlags=list(mode="aic", Pmax=12))
hegy.out$stats # HEGY test statistics
names(hegy.out) # HEGY objects, which can be called by using $, see below.
hegy.out$hegycoefs
hegy.out$regvarcoefs
```
iClick.plm1way

iClick GUI for one-way panel data analysis, based on package plm.

Description

This function generates analysis of panel data by iClick.plm.

Usage

iClick.plm1way(dep,indep,Formula,data,bootrep=99,ENDOG,IV,inst.method)

Arguments

- dep: Column number of dependent variable; e.g., dep=data[,2]. Default is NULL
- indep: Column number of Independent variables; e.g., indep=data[,c(3,5,8)]. Default is NULL
- Formula: Equation input by explicit formula; e.g., y=x1+x2+x3. Default is NULL
- data: A panel data class declared by plm.
- bootrep: Replication number of bootstrapping for fixed effect, the default number is 99 to avoid unnecessary computation.
- ENDOG: For 2SLS, declare endogeneous variables here; otherwise, keep it as default by NULL.
- IV: For 2SLS, declare IV variables here; otherwise, keep it as default by NULL.
- inst.method: For 2SLS, select estimation method.

Value

GUI output button.

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.

See Also

Package plm.

Examples

#library("pdR")
#data("bank_income")
data1.plm=plm.data(bank_income,index="ID");
#head(data1.plm,2)
iClick.plm1way(dep=4,indep=c(5,7,8),data=data1.plm)
iClick.plm2way

iClick GUI for two-way panel data analysis, based on package plm.

Description
This function generates analysis of panel data by iClick.plm. Declare either dep and indep or Formula.

Usage
iClick.plm2way(dep, indep, Formula, data, bootrep=99, ENDOG, IV, inst.method)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dep</td>
<td>Column number of dependent variable; e.g., dep=data[,2]. Default is NULL</td>
</tr>
<tr>
<td>indep</td>
<td>Column number of Independent variables; e.g., indep=data[,c(3,5,8)]. Default is NULL</td>
</tr>
<tr>
<td>Formula</td>
<td>Equation input by explicit formula; e.g., y=x1+x2+x3. Default is NULL</td>
</tr>
<tr>
<td>data</td>
<td>A panel data class declared by plm.</td>
</tr>
<tr>
<td>bootrep</td>
<td>Replication number of bootstrapping for fixed effect, the default number is 99 to avoid unnecessary computation.</td>
</tr>
<tr>
<td>ENDOG</td>
<td>For 2SLS, declare endogeneous variables here; otherwise, keep it as default by NULL.</td>
</tr>
<tr>
<td>IV</td>
<td>For 2SLS, declare IV variables here; otherwise, keep it as default by NULL.</td>
</tr>
<tr>
<td>inst.method</td>
<td>For 2SLS, select estimation method. Details see package plm.</td>
</tr>
</tbody>
</table>
**Value**

GUI output button.

**Author(s)**

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.

**See Also**

Package plm.

**Examples**

```r
# unmark to run
#library("pdR")
data("productivity")
data2.plm=plm.data(productivity,index="state")
head(data2.plm)
formula2="log(y_gsp)~log(x1_hwy)+log(x2_water)"
#iClick.plm2way(Formula=formula2,data=data2.plm)

#data("crime")
data3.plm= plm.data(crime, index = c("county"))
head(data3.plm)
formula3="log(crmrte)-log(prbarr)+log(polpc)+log(prbconv)+
log(avgxsen)+log(wcon)+log(wtuc)+
log(wtrd)+log(wfir)+log(wser)+log(wmfg)+log(wfed)
+lollg(wsta)+log(wloc)+log(pctymle)+log(pctmin)+smsa+region"
endoc="(log(prbarr)","log(polpc)"
iv=c("log(taxpc)","log(mix)"
#iClick.plm3way(Formula=formula3,data=data3.plm,ENDOG=endo,IV=iv)
```

---

**IGF**

*Unit root test based on Change(2002)*

**Description**

This function estimates the unit root regression based on instrument generating function of Change(2002) and returns useful outputs.

**Usage**

`IGF(y, maxp, ic, spec)`
Arguments

\( y \)  
A univariate time series data

\( \text{maxp} \)  
the max number of lags

\( \text{ic} \)  
Information criteria, either "AIC" or "BIC"

\( \text{spec} \)  
regression model specification.
  =0, no intercept and trend.
  =1, intercept only.
  =2, intercept and trend.

Details


Value

\( \text{tstat.IGF} \)  
IGF unit root test

\( \text{beta} \)  
regression coefficients. The first one is the AR(1) coefficient of unit root, and the last one is the intercept or trend

\( \text{sdev} \)  
The IGF standard error for unit root coefficient

\( \text{cv} \)  
The scalar C in IGF equation

\( p \)  
The optimal number of lag

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.

References


Examples

data(inf19)
y <- inf19[,1]
IGF(y,maxp=35,ic="BIC",spec=2)$tstat.IGF
inf19

Monthly inflation time series of 19 countries

Description


Usage

data(inf19)

Format

A data frame with 19 countries

AUSTRIA inflation of Austria
BELGIUM inflation of Belgium
CANADA inflation of Canada
DENMARK inflation of Denmark
FINLAND inflation of Finland
FRANCE inflation of France
GREECE inflation of Greece
ICELAND inflation of Iceland
ITALY inflation of Italy
JAPAN inflation of Japan
LUXEMBOURG inflation of Luxembourg
NETHERLANDS inflation of Netherlands
NORWAY inflation of Norway
PORTUGAL inflation of Portugal
SPAIN inflation of Spain
SWEDEN inflation of Sweden
SWITZERLAND inflation of Switzerland
UK inflation of UK
USA inflation of USA

Details

Monthly CIP, seasonaly differenced of log CPI of 19 countries

Examples

data(inf19)
head(inf19)
Description


Usage

data(inf_M)

Format

A data frame with 20 countries

AUSTRALIA inflation of Australia
AUSTRIA inflation of Austria
BELGIUM inflation of Belgium
CANADA inflation of Canada
DENMARK inflation of Denmark
FINLAND inflation of Finland
FRANCE inflation of France
GREECE inflation of Greece
ICELAND inflation of Iceland
ITALY inflation of Italy
JAPAN inflation of Japan
LUXEMBOURG inflation of Luxembourg
NETHERLANDS inflation of Netherlands
NORWAY inflation of Norway
PORTUGAL inflation of Portugal
SPAIN inflation of Spain
SWEDEN inflation of Sweden
SWITZERLAND inflation of Switzerland
UK inflation of UK
USA inflation of USA

Details

Monthly CIP, seasonaly differenced of log CPI of 20 countries

Examples

data(inf_M)
head(inf_M)
Inf_Q

Quarterly inflation time series of 20 countries

Description
Quarterly inflation time series of 19 countries, 1971Q1–2014Q4

Usage
data(Inf_Q)

Format
A data frame with 19 countries
AUSTRALIA inflation of Austria
AUSTRIA inflation of Austria
BELGIUM inflation of Belgium
CANADA inflation of Canada
DENMARK inflation of Denmark
FINLAND inflation of Finland
FRANCE inflation of France
Greece inflation of Greece
ICELAND inflation of Iceland
ITALY inflation of Italy
JAPAN inflation of Japan
LUXEMBOURG inflation of Luxembourg
NETHERLANDS inflation of Netherlands
NORWAY inflation of Norway
PORTUGAL inflation of Portugal
SPAIN inflation of Spain
SWEDEN inflation of Sweden
SWITZERLAND inflation of Switzerland
UK inflation of UK
USA inflation of USA

Details
Quarterly CIP, seasonal difference of log CPI of 20 countries

Examples
data(Inf_Q)
head(Inf_Q)
interpolpval

Extracting critical value and p-value from Table 1 of Hylleberg et. al (1990)

Description

Hylleberg et. al (1990, pp. 226-227) offer simulated critical values for seasonal unitr to test. interpolpval() is an internal call and should not be used independently.

Usage

interpolpval(code, stat, N, swarn = TRUE)

Arguments

code
Type of HEGY model, this will be automatically identified.

stat
Empirical test statistics.

N
Sample size calculating stat above.

swarn
Logical. Whether the warning message for negative p-value will be returned? The default is TRUE.

Value

table
Table for critical value and p-value.

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modifed from Javier Lopez-de-Lacalle

References


invest

Investment data of 565 listed companies, 1973-1987

Description

Investment data of 565 listed companies, 1973-1987, from Hansen’s example

Usage

data(invest)
**ipsHEGY**

**IPS-HEGY seasonal unit root test in panel data, Otero et al.(2007).**

**Description**

This function performs panel data-based HEGY seasonal unit root test, the asymptotics is based upon Otero et al.(2007).

**Usage**

`ipsHEGY(data, itsd, Sel, pmax, CIPS = TRUE)`

**Arguments**

- `data` Panel data, T by N
- `itsd` Options for c(i,t,sd).
  - i=1, intercept;=0 no intercept.
  - t=1, trend;=0 no deterministic trend.
  - sd=1, season dummy 1:(s-1);=0 no.
- `Sel` Selection of lags, having three options: "signf","bic","aic".
- `pmax` Maximum number of lags for searching optimal criteria.
- `CIPS` Logical. If TRUE, using Pesaran(2007) to account for cross-section correlation. The default is TRUE.
Details

Mode for selectlags has three options, AIC and BIC use R built-in functions for linear model and their meanings are popular and straightforward. “signf” includes only statistically significant lags, and statistically insignificant lags are dropped from regressors. That is, once you select this option, lags of your model may not be continuous.

The critical values for panel HEGY are standard normal for individual t-ratios, however, you need to perform simulation for the critical values of F joint test, at pdR 1.3. To this end, you are encouraged to work this out for yourself: using arima.sim() to sample seasonal time series with unit root (1-order difference) and obtain their statistics under the null using ipsHEGY(), then it is straightforward to obtain critical values.

Otero et al. (2007) provide critical values for quarterly frequency.

Value

<table>
<thead>
<tr>
<th>P_HEGY</th>
<th>Panel HEGY statistics.</th>
</tr>
</thead>
<tbody>
<tr>
<td>U_HEGY</td>
<td>Individual HEGY statistics of N units.</td>
</tr>
</tbody>
</table>

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.

References


Examples

data(inf.Q)
dataz<-inf.Q
itsd<-c(1,0,c(1:3))
#Seasonal dummy only takes quarters 1:3,
#because of the presence of common intercept.
Sel<-c("bic", "aic", "bic", "signf".
pmax<-12

OUT<-ipsHEGY(dataz,itsd,Sel,pmax,CIPS=FALSE)
OUT$P_HEGY
OUT$U_HEGY

# Simulation of critical values
**Description**

Determine the optimal number of lags for dynamic regression

**Usage**

```r
lagSelect(y, maxp, ic)
```

**Arguments**

- `y`: A univariate time series data
- `maxp`: the max number of lags
- `ic`: Information criteria, either "AIC" or "BIC"

**Details**

Information criteria "AIC" and "BIC" use the R built-in functions.

**Value**

It returns an integer, indicating the optimal lags

**Author(s)**

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.

**Examples**

```r
#library(pdR)
data(infl9)
y<-infl9[,1]
#lagSelect(y,maxp=25,ic="BIC")
```
lookupCVtable

Function for looking up tabulated critical values and associated p-values of HEGY test.

Description
Function for looking up tabulated critical values and associated p-values, Hylleberg et al. (1990, Table 1a and Table 1b).

Usage
lookupCVtable(code)

Arguments
code Type of HEGY model, this will be automatically identified.

Value
table Table for critical value and p-value.

Author(s)
Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle

References

model
Estimate specified panel threshold model

Description
This function is the main function estimating threshold regression for function ptm().

Usage
model(r, trim, rep, it, qql, cf, xt, ct, thresh, tt, qn1, n, qn, cc, yt, ty, k)
Arguments

- \( r \): vector of threshold estimate(s).
- \( \text{trim} \): value of trimmed percentage.
- \( \text{rep} \): number bootstrap repetition.
- \( \text{it} \): number of regime during computation, used in a for loop.
- \( \text{qq1} \): defined parameter.
- \( \text{cf} \): special declaration, e.g. \( \text{lag()} \).
- \( \text{xt} \): regime independent variables.
- \( \text{ct} \): trace of regime dependent variables.
- \( \text{thresh} \): threshold variable.
- \( \text{tt} \): length of time period.
- \( \text{qn1} \): as defined by \( \text{nrow(qq1)} \).
- \( \text{n} \): number of cross-section units.
- \( \text{qn} \): number of quantiles to examine.
- \( \text{cc} \): as defined by \( 2\times \log(1-\sqrt{\text{conf}_\text{lev}}) \).
- \( \text{yt} \): vectorized dependent variable.
- \( \text{ty} \): trace of \( \text{yt} \).
- \( k \): number of regime-independent independent variables.

Note

Original code offered by Dr. B. E. Hansen (http://www.ssc.wisc.edu/~bhansen/).

References


\[ \text{pIGF} \]

Panel unit root test of Chang(2002)

Description

Compute the panel unit root test statistic of Chang(2002).

Usage

\[ \text{pIGF(datamat, maxp, ic, spec)} \]
productivity

Arguments
- `datamat` T by N panel data. T is the time length, N is the number of cross-section units.
- `maxp` the max number of lags
- `ic` Information criteria, either "AIC" or "BIC"
- `spec` model specification.
  - =0, no intercept and trend.
  - =1, intercept only.
  - =2, intercept and trend.

Details
This function estimates the panel unit root test based on univariate instrument generating function of (Chang,2002).

Value
- `panel.tstat` panel IGF test statistics
- `pvalue` P-value of the panel.tstat

Author(s)
Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.

References

Examples
```r
data(infl919)
datam <- infl919
plGF(datam,maxp=25,ic="BIC",spec=2)
```

productivity

Description
Gross state production data

Usage
```r
data(productivity)
```
Format

A data frame with US production

- **state** US state index, 1-48
- **year** Year index
- **y_gsp** Gross state product
- **x1_hwy** Expenditure of public utility- highway construction
- **x2_water** Expenditure of public utility- water
- **x3_other** Expenditure of others
- **x4_private** Private consumption of each state
- **x5_emp** Employment rate of each state
- **x6_unemp** Unemployment rate of each state

Examples

```r
data(productivity)
head(productivity)
```

---

**ptm**

*Threshold specification of panel data*

Description

A generalized specification for estimating panel threshold model.

Usage

```r
ptm(dep, ind1, ind2, d, bootn, trimn, qn, conf_lev, t, n)
```

Arguments

- **dep** Dependent variable
- **ind1** Independent variables: regime dependent
- **ind2** Independent variables: regime independent
- **d** Threshold variable
- **bootn** Vector of bootstrap repetition
- **trimn** Vector of trimmed percentage
- **qn** Number of quantiles to examine
- **conf_lev** Confidence level
- **t** Length of time period
- **n** Number of cross-section units
**Details**

This code fits only balanced panel data. It generalizes the simple code of Dr. Hansen (http://www.ssc.wisc.edu/~bhansen/), allowing multiple (more-than-one) regime-dependent (ind1) variables. We generalize the original code to better fit general need of threshold modeling in panel data. bootn and trimn are vector of 3 by 1, indicating numbers of three corresponding regimes. This version corrects a slight error incurred by argument max_lag, which is used by Hansen to arrange investment data via lags. In this package, users manipulate data to fit personal research to ptm(), hence this argument is omitted lest should degree of freedom will suffer a loss of N.

**Author(s)**

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, College of Management, National Taiwan Normal University.

**References**


**Examples**

```r
# library(pdR)
data(invest)
dat <- invest[1:1500,]  # subsetting the first 1500 obs., for simplicity
t <- 15  #Length of time period
nt <- nrow(dat)
n <- nt/t  # number of cross-section units

#dep <- as.matrix(dat[,1])  # investment/assets
#th1 <- as.matrix(dat[,2])  # Tobin's Q
#th2 <- as.matrix(dat[,3])  # cash-flow/assets
#ind1 <- cbind(th1, th2)  # regime-dep covariates
#d <- as.matrix(dat[,4])  # Threshold variable
#ind2 <- cbind((th1^2), (th1^3), (th1*d))  # regime-indep covariates:
#bootn <- c(100, 200, 300)  # bootstrapping replications for each threshold estimation
#trimn <- c(0.05, 0.05, 0.05)  # trimmed percentage for each threshold estimation

# qn <- 400
# conf_lev <- 0.95

# Output = ptm(dep, ind1, ind2, d, bootn, trimn, qn, conf_lev, t, n)
# Output[[1]]  # Formatted output of 1st threshold, 2 regimes
# Output[[2]]  # Formatted output of 2nd threshold, 3 regimes
# Output[[3]]  # Formatted output of 3rd threshold, 4 regimes

# In the output, the Regime-dependent Coefficients matrix
# is, from top to bottom, regime-wise.
```
ret

Returns a data.frame of sequential lag matrix.

Description

ret() is similar to embed(), but returns a data.frame specified with colnames, not matrix.

Usage

ret(wts, k)

Arguments

wts  Univariate time series.
k  k-1 lagged terms.

Details

ret() is similar to embed(), but returns a data.frame with colnames, not matrix. Moreover, unlike embed(), ret() fills lagged cells with NA, instead of trimming them.

Value

A T by k dataframe returns. If you need 2 lags, you have to specify k=3, then it returns a dataframe with T by 3 dataframe, the first column is lag0.

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle

References


Examples

data(inf_Q)
y=inf_Q[,2]
ret(y,3)
r_est

A subroutine for model()

Description

This function is a subroutine for model(), estimation procedure.

Usage

r_est(y, r, trim, tt, qq1, qn1, qn, n, cf, xt, ct, thresh)

Arguments

y  vector of dependent variable.
r  number of regime.
trim  value of trimmed percentage.
tt  length of time period.
qq1  parameter defined by as.matrix(unique(thresh)[floor(sq*nrow(as.matrix(sort(unique(thresh)))))]).
qn1  as defined by nrow(qq1).
qn  number of quantiles to examine.
n  parameter of cross-section units.
cf  special declaration, e.g. lag().
xt  regime independent variables.
ct  trace of regime dependent variables.
thresh  threshold variable.

References


Original code from Dr. Hansen (http://www.ssc.wisc.edu/~bhansen/).
SeasComponent

Generate a data matrix of seasonal components

Description

Generate a data matrix of seasonal components, having two pattern cycles.

Usage

SeasComponent(wts, type)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>wts</td>
<td>A univariate time series with monthly or quarterly frequency.</td>
</tr>
<tr>
<td>type</td>
<td>Types of patterns of seasonal cycle.</td>
</tr>
<tr>
<td></td>
<td>=&quot;dummyCycle&quot;, generating dummy variables for the pattern of seasonal cycle, Barsky &amp; Miron (1989)</td>
</tr>
</tbody>
</table>

Details

This function generates data matrix for controlling the pattern of seasonal cycles. type="dummyCycle" generates DUMMY variables with season frequency. type="trgCycle" generates trigonometric pattern.

Value

A dataframe returns. Number of columns is determined by frequency.

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle

References


Examples

data(inf.Q)
y=inf.Q[,2]
SeasComponent(y,type="dummyCycle")
SeasComponent(y,type="trgCycle")
Description

Lagged coefficient estimates are kept if they meet the inequality condition of AIC or BIC.

Usage

```r
selPabic(lmobj, type, Pmax = NULL)
```

Arguments

- `lmobj` Object of `lm()`
- `type` Take the value of "aic" or "bic".
- `Pmax` The maximum number of lag orders.

Details

This is an internal function used for `HEGY.test()`. Beginning with `pamx`, the lag order will be drop if its inclusion worsens the minimum condition. Hence, they may not be a regular sequence. For example, for `pmax=10`, the selected lags may look like `(1,4,5,8,9)`, rather than `1,2,3,...10`.

Value

This function returns the lag orders.

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle

References


Examples

```r
data(inf.Q)
y=inf.Q[,1]
hegy.out<-HEGY.test(wts=y, itsd=c(1,0,c(1:3)),regvar=0, selectlags=list(mode="aic", Pmax=12))
hegy.out$lagsorder
hegy.out$lagcoefs
```
Selection of lags.

Description

Lagged coefficient estimates are kept if they are statistically significant

Usage

```
selPsignf(lmdet, cvref = 1.65, Pmax = NULL)
```

Arguments

- `lmdet`: Object of `lm()`
- `cvref`: Reference of critical values, the default is 1.65.
- `Pmax`: The maximum number of lag orders.

Details

This is an internal function used for `HEGY.test()`. Beginning with `pamx`, the lag order will be kept if it is statistically significant. Hence, the lag orders may not be a regular sequence. For example, for `pmax=10`, the selected lags may look like (1,4,5,8,9), rather than 1,2,3,...10.

Value

This function returns the lag orders.

Author(s)

Ho Tsung-wu <tsungwu@ntnu.edu.tw>, modified from Javier Lopez-de-Lacalle

References


Examples

```
data(inf.Q)
y=inf.Q[,1]
hegy.out<-HEGY.test(wts=y, itsd=c(1,0,c(1:3)), regvar=0, selectlags=list(mode="signf", Pmax=12))
hegy.out$lagsorder
hegy.out$lagcoefs
```
**SMPLSplit_est**

*Estimation of sub-sampled data*

---

**Description**

A function for estimating the subsampled data.

**Usage**

```r
SMPLSplit_est(dat, dep, indep, th, plot)
```

**Arguments**

- `dat`: The data in either data.frame or matrix
- `dep`: The number of column of dependent variable
- `indep`: The number of columns of regime dependent independent variables:
- `th`: The threshold variable
- `plot`: =1, plot; =0, do not plot

**Details**

This code estimates the parameters of sub-sampled data. It generalizes the simple code of Dr. Hansen, allowing Heteroskedastic Errors (White Corrected).

**Note**

Original code offered by Dr. B. E. Hansen (http://www.ssc.wisc.edu/~bhansen/).

**References**


---

**SMPLSplit_example**

*Example code for sample splitting*

---

**Description**

A sample code for learning sample splitting.

**Usage**

```r
SMPLSplit_example(data, dep, indep, th1, th2, trim_per, rep, plot)
```
Arguments

- **data**: The data in either data.frame or matrix
- **dep**: The number of column of dependent variable
- **indep**: The number of columns of regime dependent independent variables:
- **th1**: The first threshold variable
- **th2**: The second threshold variable
- **trim_per**: Trimmed percentage
- **rep**: Number of bootstrap repetition
- **plot**: =1, plot; =0, do not plot

Details

This code is the learning example for learning Hansen’s econometric sample splitting. I detailed the description of each threshold stage.

Note

Original code offered by Dr. B. E. Hansen (http://www.ssc.wisc.edu/~bhansen/).

References


Examples

```r
## Not run, because of bootstrap replication takes time. Users may unmark # and run.
data("dur_john")
# rep <- 500
# trim_per <- 0.15
# dep <- 1
# indep <- c(2,3,4,5)
# th1 <- 6
# th2 <- 7
# OUT=SMPLSplit_example(data=dur_john,dep,indep,th1,th2,trim_per,rep,plot=1)
# OUT$TEST
# OUT$Hypothesis
# OUT$Threshold
# stat=matrix(as.numeric(OUT$TEST),byrow = TRUE,8,2)
# colnames(stat)=c("F-Stat","P-value")
# rownames(stat)=OUT$Hypothesis
# stat
```
SMPLSplit_het

Testing for sample splitting

Description

A function for testing sample split given subsampled data.

Usage

SMPLSplit_het(data, dep, indep, th, trim_per, rep, plot)

Arguments

data The data in either data.frame or matrix
dep The number of columns of dependent variable
indep The number of columns of regime dependent independent variables:
th The threshold variable
trim_per Trimmed percentage
rep Number of bootstrap repetition
plot =1, plot; =0, do not plot

Details

This code tests for the presence of threshold. It generalizes the simple code of Dr. Hansen, allowing Heteroskedastic Errors (White Corrected).

Note

Original code offered by Dr. B. E. Hansen (http://www.ssc.wisc.edu/~bhansen/).

References

### sse_calc

#### a subroutine of model()

**Description**

SSE calculation

**Usage**

`sse_calc(y, x)`

**Arguments**

- `y`: matrix of independent variables.
- `x`: vector of dependent variable.

**References**


Original code from Dr. Hansen (http://www.ssc.wisc.edu/~bhansen/).

### tbar

**Compute the resursive mean**

**Description**

Compute the recursive mean of each series

**Usage**

`tbar(x)`

**Arguments**

- `x`: A univariate time series data

**Details**

This function computes the recursive mean

**Author(s)**

Ho Tsung-wu <tsungwu@ntnu.edu.tw>
Examples

data(inf19)
y <- inf19[,1]
tbar(y)

thr_sse  a subroutine calculating SSE

Description

This function is a sub-routine for model(), calculating SSE of each threshold regression.

Usage

thr_sse(y, q, r, cf, xt, ct, thresh, tt, n)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>parameter.</td>
</tr>
<tr>
<td>q</td>
<td>q1 in model().</td>
</tr>
<tr>
<td>r</td>
<td>parameter.</td>
</tr>
<tr>
<td>cf</td>
<td>as defined in model().</td>
</tr>
<tr>
<td>xt</td>
<td>as defined in model().</td>
</tr>
<tr>
<td>ct</td>
<td>as defined in model().</td>
</tr>
<tr>
<td>thresh</td>
<td>as defined in model().</td>
</tr>
<tr>
<td>tt</td>
<td>as defined in model().</td>
</tr>
<tr>
<td>n</td>
<td>as defined in model().</td>
</tr>
</tbody>
</table>

References


Original code from Dr. Hansen (http://www.ssc.wisc.edu/~bhansen/).
tr \hspace{1em} A sub-routine calculating trace

**Description**

Estimation of trace.

**Usage**

\[ \text{tr}(y, \tt, n) \]

**Arguments**

This function is a sub-routine for model(), calculating trace of matrix data vector.

- \( y \) \hspace{1em} time period length.
- \( n \) \hspace{1em} number of cross-section units.

**References**


Original code from Dr. Hansen (http://www.ssc.wisc.edu/~bhansen/).
Index

*Topic **Seasonal unit root test**
  HEGY.test, 8

*Topic **instrument generating functions**
  IGF, 12

*Topic **unit root**
  IGF, 12

bank_income, 3

cigaretts, 4
contts, 5
crime, 6

dur_john, 7

hegy.reg, 8
HEGY.test, 8

iClick.plm1way, 10
iClick.plm2way, 11
IGF, 12
inf19, 14
inf_M, 15
inf_Q, 16
interpolpval, 17
invest, 17
ipsHEGY, 18

lagSelect, 20
lookupCVtable, 21

model, 21

pdR (pdR-package), 2
pdR-package, 2
pIGF, 22
productivity, 23
ptm, 24

r_est, 27

ret, 26

SeasComponent, 28
selPabic, 29
selPsignf, 30
SMPLSplit_est, 31
SMPLSplit_example, 31
SMPLSplit_het, 33
sse_calc, 34
tbar, 34
thr_sse, 35
tr, 36