Package 'nardl'

May 7, 2018

Type Package

Title Nonlinear Cointegrating Autoregressive Distributed Lag Model

Version 0.1.5

Author Taha Zaghdoudi

Maintainer Taha Zaghdoudi <zedtaha@gmail.com>

Description Computes the nonlinear cointegrating autoregressive distributed lag model with p lags of the dependent variables and q lags of independent variables proposed by (Shin, Yu & Greenwood-Nimmo, 2014 <doi:10.1007/978-1-4899-8008-3_9>).

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Imports stats, strucchange, tseries, Formula, gtools

Suggests testthat

BugReports https://github.com/zedtaha/nardl/issues

URL https://github.com/zedtaha/nardl

NeedsCompilation no

Repository CRAN

Date/Publication 2018-05-07 17:03:14 UTC

R topics documented:

`Nardl-package` *Nonlinear Cointegrating Autoregressive Distributed Lag Model*

Description

Computes the nonlinear cointegrating autoregressive distributed lag model with p lags of the dependent variable and q lags of independent variables proposed by (Shin, Yu & Greenwood-Nimmo, 2014 <doi:10.1007/978-1-4899-8008-3_9>).

Details

Package:	Nardl
Type:	Package
Version:	0.1.5
Date:	2018-05-07
License:	GPL-3

In this package, we apply the ordinary least squares method to estimate the cointegrating nonlinear ARDL (NARDL) model in which short and long-run nonlinearities are introduced via positive and negative partial sum decompositions of the explanatory variables.Besides, we provide the CUSUM, CUSUMSQ model stability tests, model selection via aic, bic and rsqaured criteria and the dynamic multipliers plot.

Author(s)

Taha Zaghdoudi

Taha Zaghdoudi <zedtaha@gmail.com>

References

Shin, Y., Yu, B., Greenwood-Nimmo, M. (2011): Modelling asymmetric cointegration and dynamic multipliers in a nonlinear ARDL framework. *Working paper* http://ssrn.com/abstract=1807745

`ArchTest` *ARCH test*

Description

Computes the Lagrange multiplier test for conditional heteroscedasticity of Engle (1982), as described by Tsay (2005, pp. 101-102).

Usage

```
ArchTest(x, lags = 12, demean = FALSE)
```

Arguments

x	numeric vector
lags	positive integer number of lags
demean	logical: If TRUE, remove the mean before computing the test statistic.

Examples

```
reg<-nardl(food~inf,fod,ic="aic",maxlags = TRUE,graph = TRUE,case=3)
x<-reg$selresidu
nlag<-reg$np
ArchTest(x,lags=nlag)
```

---

bp2 *LM test for serial correlation*

Description

LM test for serial correlation

Usage

```
bp2(object, nlags, fill = NULL, type = c("F", "Chi2"))
```

Arguments

object	fitted lm model
nlags	positive integer number of lags
fill	starting values for the lagged residuals in the auxiliary regression. By default 0.
type	Fisher or Chisquare statistics

Examples

```
reg<-nardl(food~inf,fod,ic="aic",maxlags = TRUE,graph = TRUE,case=3)
lm2<-bp2(reg$fit,reg$np,fill=0,type="F")
```

cumsq *Function cumsq*

Description

Function cumsq

Usage

```
cumsq(e, k, n)
```

Arguments

e	is the recursive errors
k	is the estimated coefficients length
n	is the recursive errors length

Examples

```
reg<-nardl(food~inf,fod,ic="aic",maxlags = TRUE,graph = TRUE,case=3)
e<-reg$rece
k<-reg$k
n<-reg$n
cumsq(e=e,k=k,n=n)
```

cusum *Function cusum*

Description

Function cusum

Usage

```
cusum(e, k, n)
```

Arguments

e	is the recursive errors
k	is the estimated coefficients length
n	is the recursive errors length

Examples

```
reg<-nardl(food~inf,fod,ic="aic",maxlags = TRUE,graph = TRUE,case=3)
e<-reg$rece
k<-reg$k
n<-reg$n
cusum(e=e,k=k,n=n)
```

---

fod *Indian yearly data of inflation rate and percentage food import to total import*

Description

The data frame fod contains the following variables:

- food: percentage food import to total import
- inf: inflation rate
- year: the year

Usage

```
data(fod)
```

Format

A data frame with 54 rows and 2 variables

---

nardl *Nonlinear ARDL function*

Description

Nonlinear ARDL function

Usage

```
nardl(formula, data, p = NULL, q = NULL, ic = c("aic", "bic", "ll", "R2"),
  maxlags = TRUE, graph = FALSE, case = 3)
```

Arguments

formula	food~inf or food~inflI(inf^2)
data	the dataframe
p	lags of dependent variable
q	lags of independent variables
ic	: c("aic","bic","ll","R2") criteria model selection
maxlags	if TRUE auto lags selection
graph	TRUE to show stability tests plot
case	case number 3 for (unrestricted intercert, no trend) and 5 (unrestricted intercept, unrestricted trend), 1 2 and 4 not supported

Examples

```
##########################################
# Fit the nonlinear cointegrating autoregressive distributed lag model
##########################################
# Load data
data(fod)
##########################################
#example 1: nardl with fixed p and q lags
##########################################
reg<-nardl(food~inf,p=4,q=4,fod,ic="aic",maxlags = FALSE,graph = FALSE,case=3)
summary(reg)

##########################################
# example 2:auto selected lags (maxlags=TRUE)
##########################################
reg<-nardl(food~inf,fod,ic="aic",maxlags = TRUE,graph = FALSE,case=3)
summary(reg)

##########################################
# example 3: Cusum and CusumQ plot (graph=TRUE)
##########################################
reg<-nardl(food~inf,fod,ic="aic",maxlags = TRUE,graph = TRUE,case=3)
```

---

plotmplier *Dynamic multiplier plot*

Description

Dynamic multiplier plot

Usage

```
plotmplier(model, np, k, h)
```

Arguments

model	the fitted model
np	the selected number of lags
k	number of decomposed independent variables
h	is the horizon over which multipliers will be computed

Examples

```
############################
# Dynamic multipliers plot
############################
# Load data
data(fod)
reg<-nardl(food~inf,p=4,q=4,fod,ic="aic",maxlags = FALSE,graph = TRUE,case=3)
plotmplier(reg,reg$np,1,10)
```

---

pssbounds *pssbounds*

Description

display the necessary critical values to conduct the Pesaran, Shin and Smith 2001 bounds test for cointegration. See http://andyphilips.github.io/pssbounds/.

Usage

```
pssbounds(obs, fstat, tstat = NULL, case, k)
```

Arguments

obs	number of observations
fstat	value of the F-statistic
tstat	value of the t-statistic
case	case number
k	number of regressors appearing in lag levels

Details

pssbounds is a module to display the necessary critical values to conduct the Pesaran, Shin and Smith (2001) bounds test for cointegration. Critical values using the F-test are the default; users can also include the critical values of the t-test with the tstat parameter.

As discussed in Philips (2016), the upper and lower bounds of the cointegration test are non-standard, and depend on the number of observations, the number of regressors appearing in levels, and the restrictions (if any) placed on the intercept and trend. Asymptotic critical values are

provided by Pesaran, Shin, and Smith (2001), and small-sample critical values by Narayan (2005). The following five cases are possible: I (no intercept, no trend), II (restricted intercept, no trend), III (unrestricted intercept, no trend), IV (unrestricted intercept, restricted trend), V (unrestricted intercept, unrestricted trend). See Pesaran, Shin and Smith (2001) for more details; Case III is the most common.

More details are available at http://andyphilips.github.io/pssbounds/.

Value

None

Author(s)

Soren Jordan, <sorenjordanpols@gmail.com>, sorenjordan.com

Andrew Q Philips, <aphilips@pols.tamu.edu>, people.tamu.edu/~aphilips/

References

If you use pssbounds, please cite:

Jordan, Soren and Andrew Q. Philips. "pss: Perform bounds test for cointegration and perform dynamic simulations."

and

Philips, Andrew Q. "Have your cake and eat it too? Cointegration and dynamic inference from autoregressive distributed lag models" Working Paper.

Narayan, Paresh Kumar. 2005. "The Saving and Investment Nexus for China: Evidence from Cointegration Tests." Applied Economics 37(17):1979-1990.

Pesaran, M Hashem, Yongcheol Shin and Richard J Smith. 2001. "Bounds testing approaches to the analysis of level relationships." Journal of Applied Econometrics 16(3):289-326.

Examples

```
reg<-nardl(food~inf,fod,ic="aic",maxlags = TRUE,graph = TRUE,case=3)
pssbounds(case=reg$case,fstat=reg$fstat,obs=reg$obs,k=reg$k)
# F-stat concludes I(1) and cointegrating, t-stat concludes I(0).
```

`summary.nardl` *Summary of a nardl model*

Description

`summary` method for a `nardl` model.

Usage

```
## S3 method for class 'nardl'
summary(object, ...)
```

Arguments

`object`	is the object of the function
`...`	not used

Value

an object of the S3 class `summary.nardl` with the following components:

Index