Package ‘panelvar’

November 25, 2024

Type Package

Title Panel Vector Autoregression

Version 0.5.6

Description We extend two general methods of moment estimators to panel vector autoregression models (PVAR) with p lags of endogenous variables, predetermined and strictly exogenous variables. This general PVAR model contains the first difference GMM estimator by Holtz-Eakin et al. (1988) <doi:10.2307/1913103>, Arellano and Bond (1991) <doi:10.2307/2297968> and the system GMM estimator by Blundell and Bond (1998) <doi:10.1016/S0304-4076(98)00009-8>. We also provide specification tests (Hansen overidentification test, lag selection criterion and stability test of the PVAR polynomial) and classical structural analysis for PVAR models such as orthogonal and generalized impulse response functions, bootstrapped confidence intervals for impulse response analysis and forecast error variance decompositions.

License GPL (>= 2)

LazyData TRUE

Depends R (>= 3.5)

Imports knitr, MASS, Matrix (>= 1.2-11), progress, matrixcalc, texreg, ggplot2, reshape2

Suggests rmarkdown

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Michael Sigmund [aut], Robert Ferstl [aut, cre]

Maintainer Robert Ferstl <robert.ferstl@ur.de>

Repository CRAN

Date/Publication 2024-11-25 14:30:02 UTC

Contents

`abdata` *Employment UK data*

Description

This data set contains labor demand data from a panel of firms in the United Kingdom. The panel is unlanced.

Usage

```
abdata
```

Format

The variables are:

c1 Record ID

ind Firm index

year Year

emp Employment

wage Wage

cap Capital

indoutpt Industrial output

n, w, k, ys Logs of variables

rec Record number

yearm1 Lagged year

id ID

nL1, nL2, wL1, kL1, kL2, ysL1, ysL2 Lags of log variables

yr1976 - yr1984 Time dummies

Source

https://www.stata-press.com/data/r13/abdata.dta

References

Arellano, M. and Bond, S. (1991) "Some tests of specification for panel data: Monte Carlo evidence and an application to employment equations", *The Review of Economic Studies*, **58**(2), 227-297, doi:10.2307/2297968

Andrews_Lu_MMSC *Andrews Lu MMSC Criteria based on Hansen-J-Statistic*

Description

...

Usage

```
Andrews_Lu_MMSC(model, HQ_criterion = 2.1)

## S3 method for class 'pvargmm'
Andrews_Lu_MMSC(model, HQ_criterion = 2.1)
```

Arguments

`model`	A PVAR model
`HQ_criterion`	Hannan Quinn criterion

Value

BIC, AIC and HQIC

References

Andrews, D., Lu, B. (2001) Consistent Model and Momement Selection Procedures for GMM Estimation with Application to Dynamic Panel Data Models, *Journal of Econometrics*, **101**(1), 123–164, doi:10.1016/S03044076(00)000774

Examples

```
data("ex3_abdata")
Andrews_Lu_MMSC(ex3_abdata)
```

bootstrap_irf *Empirical estimation of PVAR Impulse Response Confidence Bands*

Description

Uses blockwise sampling of individuals (bootstrapping).

Usage

```
bootstrap_irf(
  model,
  typeof_irf,
  n.ahead,
  nof_Nstar_draws,
  confidence.band,
  mc.cores
)

## S3 method for class 'pvargmm'
bootstrap_irf(
  model,
  typeof_irf = c("OIRF", "GIRF"),
  n.ahead,
  nof_Nstar_draws,
  confidence.band = 0.95,
  mc.cores = getOption("mc.cores", 2L)
)

## S3 method for class 'pvarfeols'
bootstrap_irf(
  model,
  typeof_irf = c("OIRF", "GIRF"),
  n.ahead,
  nof_Nstar_draws,
  confidence.band = 0.95,
  mc.cores = getOption("mc.cores", 2L)
)
```

Arguments

`model`	A PVAR model
`typeof_irf`	`"OIRF"` or `GIRF`
`n.ahead`	n ahead steps
`nof_Nstar_draws`	Number of draws
`confidence.band`	Confidence band
`mc.cores`	Number of cores to use

Examples

```
## Not run:
data("ex1_dahlberg_data")
ex1_dahlberg_data_bs <-  bootstrap_irf(ex1_dahlberg_data, typeof_irf = c("GIRF"),
                                       n.ahead = 8,
                                       nof_Nstar_draws = 500,
```

```
                                              confidence.band = 0.95,
                                              mc.cores = 100)


## End(Not run)
data("ex1_dahlberg_data")
ex1_dahlberg_data_girf <-  girf(ex1_dahlberg_data, n.ahead = 8, ma_approx_steps= 8)
data("ex1_dahlberg_data_bs")
plot(ex1_dahlberg_data_girf, ex1_dahlberg_data_bs)
```

`Cigar` *Cigar data*

Description

This panel data set consists of 46 U.S. States over the period 1963-1992.

Usage

```
Cigar
```

Format

The variables are:

state State abbreviation

year Year

price Price per pack of cigarettes

pop Population

pop16 Population above the age of 16.

cpi Consumer price index with (1983=100

ndi Per capita disposable income

sales Cigarette sales in packs per capita

pimin Minimum price in adjoining states per pack of cigarettes

All variables all also available as logs.

Source

https://www.wiley.com/legacy/wileychi/baltagi/supp/Cigar.txt

References

Baltagi, B.H. and D. Levin (1992) "Cigarette taxation: raising revenues and reducing consumption", *Structural Change and Economic Dynamics*, **3**(2), 321-335, doi:10.1016/0954349X(92)900104.

Baltagi, B.H., J.M. Griffin and W. Xiong (2000) "To pool or not to pool: homogeneous versus heterogeneous estimators applied to cigarette demand", *Review of Economics and Statistics*, **82**(1), 117-126, doi:10.1162/003465300558551.

Baltagi, B.H. (2013) "Econometric analysis of panel data", 5th edition, *John Wiley and Sons* Cigar

---

`coef.pvarfeols` *Extract PVARFEOLS(p) Model Coefficients*

Description

Extract PVARFEOLS(p) Model Coefficients

Usage

```
## S3 method for class 'pvarfeols'
coef(object, ...)
```

Arguments

`object`	object
`...`	further arguments

`coef.pvargmm` *Extract PVAR(p) Model Coefficients*

Description

Extract PVAR(p) Model Coefficients

Usage

```
## S3 method for class 'pvargmm'
coef(object, ...)
```

Arguments

`object`	object
`...`	further arguments

Examples

```
data("ex1_dahlberg_data")
coef(ex1_dahlberg_data)
```

coef.pvarhk *Extract PVARHK(p) Model Coefficients*

Description

Extract PVARHK(p) Model Coefficients

Usage

```
## S3 method for class 'pvarhk'
coef(object, ...)
```

Arguments

`object`	object
`...`	further arguments

Dahlberg *Swedish municipalities data*

Description

The panel data set consists of 265 Swedish municipalities and covers 9 years (1979-1987).

Usage

```
Dahlberg
```

Format

The variables are:

id ID number for municipality

year Year

expenditures Total expenditures

revenues Total own-source revenues

grants Intergovernmental grants received by the municipality

Total expenditures contains both capital and current expenditures.

Expenditures, revenues, and grants are expressed in million SEK. The series are deflated and in per capita form. The implicit deflator is a municipality-specific price index obtained by dividing total local consumption expenditures at current prices by total local consumption expenditures at fixed (1985) prices.

The data are gathered by Statistics Sweden and obtained from Financial Accounts for the Municipalities (Kommunernas Finanser).

Source

http://qed.econ.queensu.ca/jae/2000-v15.4/dahlberg-johansson/

References

M. Dahlberg and E. Johansson (2000) "An examination of the dynamic behavior of local governments using GMM bootstrapping methods", *Journal of Applied Econometrics*, **15**(4), 401-416, https://www.jstor.org/stable/2678589.

`ex1_dahlberg_data` *Dahlberg results example 1*

Description

Dahlberg results example 1

Usage

```
ex1_dahlberg_data
```

Format

An object of class `pvargmm` of length 35.

`ex1_dahlberg_data_bs` *Dahlberg bootstrap results example 1*

Description

Dahlberg bootstrap results example 1

Usage

```
ex1_dahlberg_data_bs
```

Format

An object of class `list` of length 4.

`ex2_nlswork2_data_bs` *NLS Work 2 bootstrap results example 2*

Description

NLS Work 2 bootstrap results example 2

Usage

```
ex2_nlswork2_data_bs
```

Format

An object of class `list` of length 4.

`ex3_abdata` *Example results for Employment UK data*

Description

Example results for Employment UK data

Usage

```
ex3_abdata
```

Format

An object of class `pvargmm` of length 36.

`extract` *Extract Coefficients and GOF Measures from a Statistical Object*

Description

Extract Coefficients and GOF Measures from a Statistical Object

Usage

```
extract(model, ...)

## S3 method for class 'pvargmm'
extract(model, ...)

## S3 method for class 'pvarfeols'
extract(model, ...)

## S3 method for class 'pvarhk'
extract(model, ...)
```

Arguments

model	Model
...	Further arguments passed to or from other methods

Examples

```
data("ex1_dahlberg_data")
extract(ex1_dahlberg_data)
```

---

fevd_orthogonal *Forcast Error Variance Decomposition for PVAR*

Description

Computes the forecast error variance decomposition of a PVAR(p) model.

Usage

```
fevd_orthogonal(model, n.ahead = 10)

## S3 method for class 'pvargmm'
fevd_orthogonal(model, n.ahead = 10)

## S3 method for class 'pvarfeols'
fevd_orthogonal(model, n.ahead = 10)
```

Arguments

model	A PVAR model
n.ahead	Number of steps

Details

The estimation is based on orthogonalised impulse response functions.

Value

A list with forecast error variances as matrices for each variable.

Note

A `plot` method will be provided in future versions.

References

Pfaff, B. (2008) VAR, SVAR and SVEC Models: Implementation Within R Package vars, *Journal of Statistical Software* **27**(4) https://www.jstatsoft.org/v27/i04/

See Also

`pvargmm` for model estimaion

`oirf` for orthogonal impulse response function

Examples

```
data("ex1_dahlberg_data")
fevd_orthogonal(ex1_dahlberg_data, n.ahead = 8)
```

---

fixedeffects *Extracting Fixed Effects*

Description

Extracting Fixed Effects

Usage

```
fixedeffects(model, ...)

## S3 method for class 'pvargmm'
fixedeffects(model, Only_Non_NA_rows = TRUE, ...)
```

Arguments

`model`	Model
`...`	Further arguments passed to or from other methods
`Only_Non_NA_rows`	Filter NA rows

Examples

```
data("ex1_dahlberg_data")
fixedeffects(ex1_dahlberg_data)
```

girf *Generalized Impulse Response Function*

Description

Generalized Impulse Response Function

Usage

```
girf(model, n.ahead, ma_approx_steps)

## S3 method for class 'pvargmm'
girf(model, n.ahead, ma_approx_steps)
```

Arguments

`model`	A PVAR model
`n.ahead`	Any stable AR() model has an infinite MA representation. Hence any shock can be simulated infinitely into the future. For each forecast step t you need an additional MA term.
`ma_approx_steps`	MA approximation steps

Examples

```
data("ex1_dahlberg_data")
girf(ex1_dahlberg_data, n.ahead = 8, ma_approx_steps= 8)
```

hansen_j_test *Sargan-Hansen-J-Test for Overidentification*

Description

Sargan-Hansen-J-Test for Overidentification

Usage

```
hansen_j_test(model, ...)

## S3 method for class 'pvargmm'
hansen_j_test(model, ...)
```

Arguments

`model`	A PVAR model
`...`	Further arguments passed to or from other methods

Examples

```
data("ex1_dahlberg_data")
hansen_j_test(ex1_dahlberg_data)
```

---

knit_print.pvarfeols *Knit Print Method for pvarfeols*

Description

Knit Print Method for pvarfeols

Usage

```
## S3 method for class 'pvarfeols'
knit_print(x, ...)
```

Arguments

x	object
...	further arguments

---

knit_print.pvargmm *Knit Print Method for pvargmm*

Description

Knit Print Method for pvargmm

Usage

```
## S3 method for class 'pvargmm'
knit_print(x, ...)
```

Arguments

x	object
...	further arguments

`knit_print.pvarhk` *Knit Print Method for pvarhk*

Description

Knit Print Method for pvarhk

Usage

```
## S3 method for class 'pvarhk'
knit_print(x, ...)
```

Arguments

`x`	object
`...`	further arguments

`knit_print.summary.pvarfeols` *Knit Print summary Method*

Description

Knit Print summary Method

Usage

```
## S3 method for class 'summary.pvarfeols'
knit_print(x, ...)
```

Arguments

`x`	object
`...`	further arguments

knit_print.summary.pvargmm

Knit Print summary Method

Description

Knit Print summary Method

Usage

```
## S3 method for class 'summary.pvargmm'
knit_print(x, ...)
```

Arguments

x	object
...	further arguments

knit_print.summary.pvarhk

Knit Print summary Method

Description

Knit Print summary Method

Usage

```
## S3 method for class 'summary.pvarhk'
knit_print(x, ...)
```

Arguments

x	object
...	further arguments

`nlswork2` *NLS Work 2 data*

Description

NLS Work 2 data

Usage

```
nlswork2
```

Format

An object of class `data.frame` with 16094 rows and 21 columns.

`oirf` *Orthogonal Impulse Response Function*

Description

Orthogonal Impulse Response Function

Usage

```
oirf(model, n.ahead)
```

Arguments

`model`	A PVAR model
`n.ahead`	Any stable AR() model has an infinite MA representation. Hence any shock can be simulated infinitely into the future. For each forecast step t you need an addtional MA term.

Examples

```
data("ex1_dahlberg_data")
oirf(ex1_dahlberg_data, n.ahead = 8)
```

plot.pvarstability *S3 plot method for pvarstability object, returns a* `ggplot` *object*

Description

S3 plot method for pvarstability object, returns a `ggplot` object

Usage

```
## S3 method for class 'pvarstability'
plot(x, ...)
```

Arguments

`x`	object
`...`	further arguments

print.pvarfeols *S3 Print Method for pvarfeols*

Description

S3 Print Method for pvarfeols

Usage

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

Arguments

`x`	object
`...`	further arguments

print.pvargmm — *S3 Print Method for pvargamm*

Description

S3 Print Method for pvargamm

Usage

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

Arguments

`x`	object
`...`	further arguments

print.pvarhk — *S3 Print Method for pvarhk*

Description

S3 Print Method for pvarhk

Usage

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

Arguments

`x`	object
`...`	further arguments

print.pvarstability — *S3 print method for pvarstability object*

Description

S3 print method for pvarstability object

Usage

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

Arguments

x	object
...	further arguments

print.summary.pvarfeols — *S3 Print Method for summary.pvarfeols*

Description

S3 Print Method for summary.pvarfeols

Usage

```
## S3 method for class 'summary.pvarfeols'
print(x, ...)
```

Arguments

x	object
...	further arguments

print.summary.pvargmm *S3 Print Method for summary.pvargmm*

Description

S3 Print Method for summary.pvargmm

Usage

```
## S3 method for class 'summary.pvargmm'
print(x, ...)
```

Arguments

`x`	object
`...`	further arguments

print.summary.pvarhk *S3 Print Method for summary.pvarhk*

Description

S3 Print Method for summary.pvarhk

Usage

```
## S3 method for class 'summary.pvarhk'
print(x, ...)
```

Arguments

`x`	object
`...`	further arguments

pvalue — *P-value S3 Method*

Description

P-value S3 Method

Usage

```
pvalue(object, ...)

## S3 method for class 'pvargmm'
pvalue(object, ...)

## S3 method for class 'pvarfeols'
pvalue(object, ...)

## S3 method for class 'pvarhk'
pvalue(object, ...)
```

Arguments

`object`	Object
`...`	Further arguments

Examples

```
data("ex1_dahlberg_data")
pvalue(ex1_dahlberg_data)
```

pvarfeols — *Fixed Effects Estimator for PVAR Model*

Description

This function estimates a stationary PVAR with fixed effects.

Usage

```
pvarfeols(
  dependent_vars,
  lags,
  exog_vars,
  transformation = c("demean"),
  data,
  panel_identifier = c(1, 2)
)
```

Arguments

`dependent_vars` Dependent variables

`lags` Number of lags of dependent variables

`exog_vars` Exogenous variables

`transformation` Demeaning "demean"

`data` Data set

`panel_identifier`
Vector of panel identifiers

Examples

```
data(Cigar)
ex1_feols <-
pvarfeols(dependent_vars = c("log_sales", "log_price"),
          lags = 1,
          exog_vars = c("cpi"),
          transformation = "demean",
          data = Cigar,
          panel_identifier= c("state", "year"))

summary(ex1_feols)
```

`pvargmm` *GMM Estimation of Panel VAR Models*

Description

Estimates a panel vector autoregressive (PVAR) model with fixed effects.

Usage

```
pvargmm(
  dependent_vars,
  lags,
  predet_vars,
  exog_vars,
  transformation = "fd",
  data,
  panel_identifier = c(1, 2),
  steps,
  system_instruments = FALSE,
  system_constant = TRUE,
  pca_instruments = FALSE,
  pca_eigenvalue = 1,
  max_instr_dependent_vars,
  max_instr_predet_vars,
```

```
  min_instr_dependent_vars = 2L,
  min_instr_predet_vars = 1L,
  collapse = FALSE,
  tol = 1e-09,
  progressbar = TRUE
)
```

Arguments

`dependent_vars` Dependent variables

`lags` Number of lags of dependent variables

`predet_vars` Predetermined variables

`exog_vars` Exogenous variables

`transformation` First-difference "`fd`" or forward orthogonal deviations "`fod`"

`data` Data set

`panel_identifier` Vector of panel identifiers

`steps` "`onestep`", "`twostep`" or "`mstep`" estimation

`system_instruments` System GMM estimator

`system_constant` Constant only available with the System GMM estimator in each equation

`pca_instruments` Apply PCA to instruments matrix

`pca_eigenvalue` Cut-off eigenvalue for PCA analysis

`max_instr_dependent_vars` Maximum number of instruments for dependent variables

`max_instr_predet_vars` Maximum number of instruments for predetermined variables

`min_instr_dependent_vars` Minimum number of instruments for dependent variables

`min_instr_predet_vars` Minimum number of instruments for predetermined variables

`collapse` Use collapse option

`tol` relative tolerance to detect zero singular values in "`ginv`"

`progressbar` show progress bar

Details

The first vector autoregressive panel model (PVAR) was introduced by Holtz-Eakin et al. (1988). Binder et al. (2005) extend their equation-by-equation estimator for a PVAR model with only endogenous variables that are lagged by one period. We further improve this model in Sigmund and Ferstl (2021) to allow for p lags of m endogenous variables, k predetermined variables and n strictly exogenous variables.

Therefore, we consider the following stationary PVAR with fixed effects.

$$\mathbf{y}_{i,t} = \mu_i + \sum_{l=1}^{p} \mathbf{A}_l \mathbf{y}_{i,t-l} + \mathbf{B}\mathbf{x}_{i,t} + \mathbf{C}\mathbf{s}_{i,t} + \epsilon_{i,t}$$

$\mathbf{I}_m$ denotes an $m \times m$ identity matrix. Let $\mathbf{y}_{i,t} \in \mathsf{R}^m$ be an $m \times 1$ vector of endogenous variables for the ith cross-sectional unit at time t. Let $\mathbf{y}_{i,t-l} \in \mathsf{R}^m$ be an $m \times 1$ vector of lagged endogenous variables. Let $\mathbf{x}_{i,t} \in \mathsf{R}^k$ be an $k \times 1$ vector of predetermined variables that are potentially correlated with past errors. Let $\mathbf{s}_{i,t} \in \mathsf{R}^n$ be an $n \times 1$ vector of strictly exogenous variables that neither depend on ϵ_t nor on ϵ_{t-s} for $s = 1, \ldots, T$. The idiosyncratic error vector $\epsilon_{i,t} \in \mathsf{R}^m$ is assumed to be well-behaved and independent from both the regressors $\mathbf{x}_{i,t}$ and $\mathbf{s}_{i,t}$ and the individual error component μ_i. Stationarity requires that all unit roots of the PVAR model fall inside the unit circle, which therefore places some constraints on the fixed effect μ_i. The cross section i and the time section t are defined as follows: $i = 1, 2, ..., N$ and $t = 1, 2, ..., T$. In this specification we assume parameter homogeneity for $\mathbf{A}_l(m \times m)$, $\mathbf{B}(m \times k)$ and $\mathbf{C}(m \times n)$ for all i.

A PVAR model is hence a combination of a single equation dynamic panel model (DPM) and a vector autoregressive model (VAR).

First difference and system GMM estimators for single equation dynamic panel data models have been implemented in the STATA package `xtabond2` by Roodman (2009) and some of the features are also available in the R package **plm**.

For more technical details on the estimation, please refer to our paper Sigmund and Ferstl (2021).

There we define the first difference moment conditions (see Holtz-Eakin et al., 1988; Arellano and Bond, 1991), formalize the ideas to reduce the number of moment conditions by linear transformations of the instrument matrix and define the one- and two-step GMM estimator. Furthermore, we setup the system moment conditions as defined in Blundell and Bond (1998) and present the extended GMM estimator. In addition to the GMM-estimators we contribute to the literature by providing specification tests (Hansen overidentification test, lag selection criterion and stability test of the PVAR polynomial) and classical structural analysis for PVAR models such as orthogonal and generalized impulse response functions, bootstrapped confidence intervals for impulse response analysis and forecast error variance decompositions. Finally, we implement the first difference and the forward orthogonal transformation to remove the fixed effects.

Value

A `pvargmm` object containing the estimation results.

References

Arellano, M., Bond, S. (1991) Some Tests of Specification for Panel Sata: Monte Carlo Evidence and an Application to Employment Equations *The Review of Economic Studies*, **58**(2), 277–297, doi:10.2307/2297968

Binder M., Hsiao C., Pesaran M.H. (2005) Estimation and Inference in Short Panel Vector Autoregressions with Unit Roots and Cointegration *Econometric Theory*, **21**(4), 795–837, doi:10.1017/S0266466605050413

Blundell R., Bond S. (1998). Initial Conditions and Moment Restrictions in Dynamic Panel Data Models *Journal of Econometrics*, **87**(1), 115–143, doi:10.1016/S03044076(98)000098

Holtz-Eakin D., Newey W., Rosen H.S. (1988) Estimating Vector Autoregressions with Panel Data, *Econometrica*, **56**(6), 1371–1395, doi:10.2307/1913103

Roodman, D. (2009) How to Do xtabond2: An Introduction to Difference and System GMM in Stata *The Stata Journal*, **9**(1), 86–136, https://www.stata-journal.com/article.html?article=st0159

Sigmund, M., Ferstl, R. (2021) Panel Vector Autoregression in R with the Package panelvar *The Quarterly Review of Economics and Finance* doi:10.1016/j.qref.2019.01.001

See Also

`stability` for stability tests

`oirf` and `girf` for orthogonal and generalized impulse response functions (including bootstrapped confidence intervals)

`coef.pvargmm`, `se`, `pvalue`, `fixedeffects` for extrator functions for the most important results

`fevd_orthogonal` for forecast error variance decomposition

Examples

```
## Not run:
library(panelvar)
data(abdata)
ex3_abdata <-pvargmm(
 dependent_vars = c("emp"),
 lags = 4,
 predet_vars = c("wage"),
 exog_vars = c("cap"),
 transformation = "fd",
 data = abdata,
 panel_identifier = c("id", "year"),
 steps = c("twostep"),
 system_instruments = TRUE,
 max_instr_dependent_vars = 99,
 max_instr_predet_vars = 99,
 min_instr_dependent_vars = 2L,
 min_instr_predet_vars = 1L,
 collapse = FALSE
)

## End(Not run)
data("ex3_abdata")
summary(ex3_abdata)

data("Dahlberg")
## Not run:
ex1_dahlberg_data <- pvargmm(dependent_vars = c("expenditures", "revenues", "grants"),
                             lags = 1,
                             transformation = "fod",
                             data = Dahlberg,
                             panel_identifier=c("id", "year"),
                             steps = c("twostep"),
```

```
                                system_instruments = FALSE,
                                max_instr_dependent_vars = 99,
                                max_instr_predet_vars = 99,
                                min_instr_dependent_vars = 2L,
                                min_instr_predet_vars = 1L,
                                collapse = FALSE
)

## End(Not run)
data("ex1_dahlberg_data")
summary(ex1_dahlberg_data)
```

---

pvarhk *Hahn Kuehrsteiner Estimator for PVAR Model*

Description

This function estimates a stationary PVAR with fixed effects.

Usage

```
pvarhk(
  dependent_vars,
  exog_vars,
  transformation = c("demean"),
  data,
  panel_identifier = c(1, 2)
)
```

Arguments

dependent_vars	Dependent variables
exog_vars	Exogenous variables
transformation	Demeaning "demean"
data	Data set
panel_identifier	Vector of panel identifiers

References

Hahn J., Kuehrsteiner G. (2002) Asymptotically Unbiased Inference for a Dynamic Panel Model with Fixed Effects When Both n and T Are Large, *Econometrica*, **70**(4), 1639–1657

Examples

```
data(Dahlberg)
ex1_hk <-
pvarhk(dependent_vars = c("expenditures", "revenues", "grants"),
        transformation = "demean",
        data = Dahlberg,
        panel_identifier= c("id", "year"))

summary(ex1_hk)
```

---

residuals_level *Extracting Level Residuals*

Description

Extracting Level Residuals

Usage

```
residuals_level(model, ...)

## S3 method for class 'pvargmm'
residuals_level(model, ...)
```

Arguments

model	Model
...	Further arguments passed to or from other methods

Examples

```
data("ex1_dahlberg_data")
residuals_level(ex1_dahlberg_data)
```

---

se *Standard Error S3 Method*

Description

Standard Error S3 Method

Usage

```
se(object, ...)

## S3 method for class 'pvargmm'
se(object, ...)

## S3 method for class 'pvarfeols'
se(object, ...)

## S3 method for class 'pvarhk'
se(object, ...)
```

Arguments

`object`	Object
`...`	Further arguments

Examples

```
data("ex1_dahlberg_data")
se(ex1_dahlberg_data)
```

---

stability *Stability of PVAR(p) model*

Description

Stability of PVAR(p) model

Usage

```
stability(model, ...)

## S3 method for class 'pvargmm'
stability(model, ...)

## S3 method for class 'pvarfeols'
stability(model, ...)
```

Arguments

`model`	PVAR model
`...`	Further arguments

Value

A `pvarstability` object containing eigenvalue stability conditions

Examples

```
data("ex1_dahlberg_data")
stability_info <- stability(ex1_dahlberg_data)
print(stability_info)
plot(stability_info)
```

---

summary.pvarfeols *S3 Summary Method for pvarfeols*

Description

S3 Summary Method for pvarfeols

Usage

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

Arguments

`object`	object
`...`	further arguments

---

summary.pvargmm *S3 Summary Method for pvargmm*

Description

S3 Summary Method for pvargmm

Usage

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

Arguments

`object`	object
`...`	further arguments

`summary.pvarhk` *S3 Summary Method for pvarhk*

Description

S3 Summary Method for pvarhk

Usage

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

Arguments

`object`	object
`...`	further arguments

Index