Package ‘plm’

January 26, 2010

Version 1.2-3

Date 2010-01-26

Title Linear Models for Panel Data

Author Yves Croissant <yves.croissant@let.ish-lyon.cnrs.fr>, Giovanni Millo <Giovanni_Millo@Generali.com>

Maintainer Yves Croissant <yves.croissant@let.ish-lyon.cnrs.fr>

Depends R (>= 1.8.0), stats, kinship, nlme, Formula (>= 0.2-0), MASS, sandwich

Suggests lmtest, car

Description A set of estimators and tests for panel data.

License GPL (>= 2)

URL http://www.R-project.org

Repository CRAN

Date/Publication 2010-01-26 13:46:20

R topics documented:

`Cigar` *Cigarette Consumption*

Description

a panel of 46 observations from 1963 to 1992

total number of observations : 1380

observation : regional

country : United States

Usage

```
data(Cigar)
```

Format

A data frame containing :

state state abbreviation

year the year

price price per pack of cigarettes

pop population

pop16 population above the age of 16

cpi consumer price index (1983=100)

ndi per capita disposable income

sales cigarette sales in packs per capita

pimin minimum price in adjoining states per pack of cigarettes

Source

Online complements to Baltagi (2001).

http://www.wiley.com/legacy/wileychi/baltagi/.

References

Baltagi, Badi H. (2001) *Econometric Analysis of Panel Data*, 2nd ed., John Wiley and Sons.

Baltagi, B.H. and D. Levin (1992) "Cigarette taxation: raising revenues and reducing consumption", *Structural Changes and Economic Dynamics*, **3**, 321–335.

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**, 117–126.

`Crime` *Crime in North Carolina*

Description

a panel of 90 observations (counties) from 1981 to 1987

total number of observations : 630

observation : regional

country : United States

Usage

```
data(Crime)
```

Format

A data frame containing :

county county identifier

year year from 1981 to 1987

crmrte crimes committed per person

prbarr 'probability' of arrest

prbconv 'probability' of conviction

prbpris 'probability' of prison sentenc

avgsen average sentence, days

polpc police per capita

density people per square mile

taxpc tax revenue per capita

region factor. One of 'other', 'west' or 'central'.

smsa factor. Does the individual reside in a SMSA (standard metropolitan statistical area)?

pctmin percentage minority in 1980

wcon weekly wage in construction

wtuc weekly wage in trns, util, commun

wtrd weekly wage in whole sales and retail trade

wfir weekly wage in finance, insurance and real estate

wser weekly wage in service industry

wmfg weekly wage in manufacturing

wfed weekly wage of federal emplyees

wsta weekly wage of state employees

wloc weekly wage of local governments employees

mix offence mix: face-to-face/other

pctymle percentage of young males

Source

Online complements to Baltagi (2001).

`http://www.wiley.com/legacy/wileychi/baltagi/`, .

See also Journal of Applied Econometrics data archive entry for Baltagi (2006) at

`http://www.econ.queensu.ca/jae/2006-v21.4/baltagi/`.

References

Cornwell, C. and W.N. Trumbull (1994) "Estimating the economic model of crime with panel data", *Review of Economics and Statistics*, **76**, 360–366.

Baltagi, B. H. (2006) "Estimating an economic model of crime using panel data from North Carolina", *Journal of Applied Econometrics*, 21(4), 543–547.

Baltagi, B. H. (2001) *Econometric Analysis of Panel Data*, John Wiley and Sons.

`dynformula` *Dynamic Formula*

Description

A function to easily create a formula with lags and differences

Usage

```
dynformula(formula, lag.form = NULL, diff.form = NULL, log.form = NULL)
```

Arguments

`formula`	a formula,
`lag.form`	a list containing the lag structure of each variable in the formula,
`diff.form`	a vector (or a list) of logical values indicating whether variables should be differenced,
`log.form`	a vector (or a list) of logical values indicating whether variables should be in logarithms.

Details

`lag.form` is a list, `diff.form` and `log.form` are vectors (or lists) that should be of length equal to the total number of variables. Each element of these lists/vectors is:

- either a vector of length 2 (`c(1,4)` means lags 1,2,3 and 4) or a scalar (`3` means lags 0,1,2,3 except for the left–hand side variable for which it is 1,2,3) for `lag.form`.
- logical values for `diff.form` and `log.form`.

It can also be an incomplete named list/vector (for example, to apply the transformation for only some variables) with eventually an unnamed element which then is the default value.

Value

An object of class `c("dynformula","formula")`, which is a formula with four additional attributes: `var`, the names of the variables in the formula, `lag`, `diff`, and `log`, which store the information about lags, differences and logs, respectively.

A `formula` method coerces the `dynformula` object to a standard `formula`.

Author(s)

Yves Croissant

Examples

```
# all variables in log, x1, x2 and x3 laged twice, y laged once and x3 differenced
z <- dynformula(y ~ x1 + x2 + x3, lag.form = list(2, y = 1),
                diff.form = c(x3 = TRUE), log.form = TRUE)
formula(z)
```

`EmplUK` *Employment and Wages in the United Kingdom*

Description

An unbalanced panel of 140 observations from 1976 to 1984

total number of observations : 1031

observation : firms

country : United Kingdom

Usage

```
data(EmplUK)
```

Format

A data frame containing :

firm firm index

year year

sector the sector of activity

emp employment

wage wages

capital capital

output output

Source

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*, vol. **58**(2), 1991, pp.227–297.

ercomp *Estimation of the error components*

Description

This function enables the estimation of the variance components of a panel model.

Usage

```
ercomp(object, ...)
## S3 method for class 'formula':
ercomp(object, data, effect = c("individual", "time", "twoways"),
                        method = c("swar", "walhus", "amemiya", "nerlove"),
                         index = NULL, ...)
## S3 method for class 'plm':
ercomp(object, ...)
## S3 method for class 'ercomp':
print(x, digits= max(3, getOption("digits") - 3), ...)
```

Arguments

`object`	a `formula` or a `plm` object,
`data`	a `data.frame`,
`method`	method of estimation for the variance components, see `lm` for details,
`effect`	the effects introduced in the model, see `lm` for details,
`index`	the indexes,
`x`	a `ercomp` object,
`digits`	digits,
`...`	further arguments.

Value

An object of class `"ercomp"`: a list containing a list called `sigma2` which contains the estimates of the variance components, and `theta` which is the parameters used for the transformation of the variables.

Author(s)

Yves Croissant

References

Amemiya, T. (1971) The estimation of the variances in a variance–components model, *International Economic Review*, **12**, pp.1–13.

Nerlove, M. (1971) Further evidence on the estimation of dynamic economic relations from a time–series of cross–sections, *Econometrica*, **39**, pp.359–382.

Swamy, P.A.V.B. and Arora, S.S. (1972) The exact finite sample properties of the estimators of coefficients in the error components regression models, *Econometrica*, **40**, pp.261–275.

Wallace, T.D. and Hussain, A. (1969) The use of error components models in combining cross section with time series data, *Econometrica*, **37**(1), pp.55–72.

See Also

plm where the estimates of the variance components are used if a random effects model is estimated

Examples

```
data("Produc", package = "plm")
# an example of the formula method
ercomp(log(gsp)~log(pcap)+log(pc)+log(emp)+unemp, data=Produc, method="walhus",effect="time"
# same with the plm method
z <- plm(log(gsp)~log(pcap)+log(pc)+log(emp)+unemp,
         data=Produc, random.method="walhus",
         effect="time",model="random")
ercomp(z)
# a two-ways model
ercomp(log(gsp)~log(pcap)+log(pc)+log(emp)+unemp, data=Produc, method="amemiya",effect="twow
```

---

fixef.plm *Extract the Fixed Effects*

Description

This function extracts the fixed effects from a `plm` object.

Usage

```
## S3 method for class 'plm':
fixef(object, effect = NULL,
                         type = c("level", "dfirst", "dmean"), ...)
## S3 method for class 'fixef':
print(x, digits = max(3, getOption("digits") - 2),
width = getOption("width"),...)
## S3 method for class 'fixef':
summary(object, ...)
## S3 method for class 'summary.fixef':
print(x, digits = max(3, getOption("digits") -
2), width = getOption("width"),...)
```

Arguments

`x,object`	an object of class `"plm"`, an object of class `"fixef"` for the `print` and the `summary` method,
`effect`	one of `"individual"` or `"time"`, only relevant in case of two–ways effects models,
`type`	one of `level`, `dmean` or `dfirst`.
`digits`	digits,
`width`	the maximum length of the lines in the print output,
`...`	further arguments.

Details

The summary method prints the effects in deviation from the overallintercept, the standard error and the t–values.

Value

An object of class `"fixef"`. It is a numeric vector containing the fixed effects with two attributes: `se` which contains the standard errors and `intercept` which is the overall intercept. With the `type` argument, the fixed effects may be returned in levels, as deviations from the overall mean or as deviations from the first value of the index.

Author(s)

Yves Croissant

See Also

`plm`

Examples

```
data("Grunfeld", package = "plm")
gi <- plm(inv ~ value + capital, data = Grunfeld, model = "within")
fixef(gi)
summary(fixef(gi))
# extract time effects in a twoways effects model
gi <- plm(inv ~ value + capital, data = Grunfeld,
          model = "within", effect = "twoways")
fixef(gi,effect = "time")
```

`Gasoline` *Gasoline Consumption*

Description

A panel of 18 observations from 1960 to 1978

total number of observations : 342

observation : country

country : OECD

Usage

```
data(Gasoline)
```

Format

A data frame containing :

country a factor with 18 levels

year the year

lgaspcar logarithm of motor gasoline consumption per car

lincomep logarithm of real per-capita income

lrpmg logarithm of real motor gasoline price

lcarpcap logarithm of the stock of cars per capita

Source

Online complements to Baltagi (2001).

http://www.wiley.com/legacy/wileychi/baltagi/.

References

Baltagi, Badi H. (2001) *Econometric Analysis of Panel Data*, 2nd ed., John Wiley and Sons.

Baltagi, B.H. and J.M. Griffin (1983) "Gasoline demand in the OECD: An application of pooling and testing procedures", *European Economic Review*, **22**(2), 117–137.

`Grunfeld` *Grunfeld's Investment Data*

Description

A panel of 10 observations from 1935 to 1954

total number of observations : 200

observation : production units

country : United States

Usage

```
data(Grunfeld)
```

Format

A data frame containing :

firm observation

year date

inv gross Investment

value value of the firm

capital stock of plant and equipment

Source

Online complements to Baltagi (2001).

`http://www.wiley.com/legacy/wileychi/baltagi/`.

References

Baltagi, Badi H. (2001) *Econometric Analysis of Panel Data*, 2nd ed., John Wiley and Sons.

See Also

For the complete Grunfeld data (11 firms), see `Grunfeld`, in the `AER` package.

Hedonic *Hedonic Prices of Census Tracts in the Boston Area*

Description

A cross-section

number of observations : 506

observation : regional

country : United States

Usage

```
data(Hedonic)
```

Format

A dataframe containing:

mv median value of owner–occupied homes

crim crime rate

zn proportion of 25,000 square feet residential lots

indus proportion of nonretail business acres

chas is the tract bounds the Charles River ?

nox annual average nitrogen oxide concentration in parts per hundred million

rm average number of rooms

age proportion of owner units built prior to 1940

dis weighted distances to five employment centers in the Boston area

rad index of accessibility to radial highways

tax full value property tax rate (\$/\$10,000)

ptratio pupil/teacher ratio

blacks proportion of blacks in the population

lstat proportion of population that is lower status

townid town identifier

Source

Online complements to Baltagi (2001).

http://www.wiley.com/legacy/wileychi/baltagi/.

References

Baltagi, Badi H. (2001): *Econometric Analysis of Panel Data*, 2nd ed., John Wiley and Sons.

Belsley, D.A., E. Kuh and R. E. Welsch (1980): *Regression Diagnostics: Identifying Influencial Data and Sources of Collinearity*, John Wiley, New York.

Harrison, D. and D.L. Rubinfeld (1978): "Hedonic housing prices and the demand for clean air", *Journal of Environmental Economics and Management*, **5**, 81–102.

`LaborSupply` *Wages and Hours Worked*

Description

A panel of 532 observations from 1979 to 1988

number of observations : 5320

Usage

```
data(LaborSupply)
```

Format

A data frame containing :

lnhr log of annual hours worked

lnwg log of hourly wage

kids number of children

age age

disab bad health

id id

year year

Source

Online complements to Ziliak (1997).

Journal of Business Economics and Statistics web site: `http://www.amstat.org/publications/jbes/`.

References

Cameron, A.C. and P.K. Trivedi (2005) *Microeconometrics : methods and applications*, Cambridge, pp. 708–15, 754–6.

Ziliak, J.P. (1997) "Efficient Estimation With Panel Data when Instruments are Predetermined: An Empirical Comparison of Moment-Condition Estimators", *Journal of Business and Economic Statistics*, **419–431**.

Males — *Wages and Education of Young Males*

Description

A panel of 545 observations from 1980 to 1987

total number of observations : 4360

observation : individuals

country : United States

Usage

```
data(Males)
```

Format

A data frame containing :

nr identifier

year year

school years of schooling

exper years of experience (computed as `age-6-school`)

union wage set by collective bargaining ?

ethn a factor with levels `black, hisp, other`

maried maried?

health health problem ?

wage log of hourly wage

industry a factor with 12 levels

occupation a factor with 9 levels

residence a factor with levels `rural area, north east, northern central, south`

Source

Journal of Applied Econometrics data archive

http://www.econ.queensu.ca/jae/1998-v13.2/vella-verbeek/.

References

Vella, F. and M. Verbeek (1998) "Whose Wages Do Unions Raise? A Dynamic Model of Unionism and Wage Rate Determination for Young Men", *Journal of Applied Econometrics*, **13**(2), 163–183.

Verbeek, M. (2004) *A Guide to Modern Econometrics*, John Wiley and Sons, http://www.econ.kuleuven.ac.be/GME, chapter 10.

mtest *Arellano–Bond test of Serial Correlation*

Description

Test of serial correlation for models estimated by GMM

Usage

```
mtest(object, order=1, vcov=NULL)
```

Arguments

object	an object of class "pgmm",
order	the order of the serial correlation (1 or 2),
vcov	a matrix of covariance for the coefficients or a function to compute it.

Details

The Arellano–Bond test is a test of correlation based on the residuals of the estimation. By default, the computation is done with the standard covariance matrix of the coefficients. A robust estimator of this covariance matrix can be supplied with the `vcov` argument.

Value

An object of class "htest".

Author(s)

Yves Croissant

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*, vol. **58**(2), 1991, pp.227–297.

See Also

pgmm

Examples

```
data("EmplUK", package = "plm")
ar <- pgmm(dynformula(log(emp) ~ log(wage) + log(capital) +
            log(output), list(2,1,2,2)), data = EmplUK, effect = "twoways", model = "twostep
mtest(ar, 1)
mtest(ar, 2, vcovHC)
```

pbgtest *Breusch–Godfrey Test for Panel Models*

Description

Test of serial correlation for (the idiosyncratic component of) the errors in panel models.

Usage

```
pbgtest(x,...)
## S3 method for class 'panelmodel':
pbgtest(x, order = NULL, ...)
## S3 method for class 'formula':
pbgtest(x, ..., order = NULL)
```

Arguments

x	an object of class "panelmodel" or of class "formula",
order	an integer indicating the order of serial correlation to be tested for. Defaults to the minimum number of observations over the time dimension,
...	further arguments.

Details

This Lagrange multiplier test uses the auxiliary model on (quasi-)demeaned data taken from a model of class `plm` which may be a `pooling` (the default), `random` or `within` model. It performs a Breusch–Godfrey test (using `bgtest` from package `lmtest`) on the residuals of the (quasi-)demeaned model, which should be serially uncorrelated under the null of no serial correlation in idiosyncratic errors, as illustrated in Wooldridge (2002). The function takes the demeaned data, estimates the model and calls `bgtest`.

Unlike most other tests for serial correlation in panels, this one allows to choose the order of correlation to test for.

Value

An object of class "htest".

Author(s)

Giovanni Millo

References

Breusch, T. (1978) Testing for autocorrelation in dynamic linear models, *Australian Economic Papers*, **17**, pp.334–355.

Godfrey, L. (1978) Testing against general autoregressive and moving average error models when the regressors include lagged dependent variables, *Econometrica*, **46**; pp. 1293–1302.

Wooldridge, J.M. (2002) *Econometric Analysis of Cross-Section and Panel Data*, MIT Press, p. 288.

See Also

`pdwtest` for the analogous panel Durbin–Watson test, `bgtest` for the Breusch–Godfrey test for serial correlation in the linear model. `pbltest`, `pbsytest`, `pwartest` and `pwfdtest` for other serial correlation tests for panel models.

Examples

```
data("Grunfeld", package = "plm")
g <- plm(inv ~ value + capital, data = Grunfeld, model = "random")
pbgtest(g)
pbgtest(g, order = 4)

## formula interface
pbgtest(inv ~ value + capital, data = Grunfeld, model = "random")
```

`pbltest` *Baltagi and Li Serial Dependence Test For Random Effects Models*

Description

Baltagi and Li (1995)'s Lagrange multiplier test for AR(1) or MA(1) idiosyncratic errors in panel models with random effects.

Usage

```
pbltest(x, data, alternative = c("twosided","onesided"), index = NULL, ...)
```

Arguments

`x`	a model formula,
`data`	a `data.frame`,
`alternative`	one of `"twosided"`, `"onesided"`. Selects either $H_A : \rho \neq 0$ or $H_A : \rho 0$ (i.e., the Normal or the Chi-squared version of the test),
`index`	the index of the `data.frame`,
`...`	further arguments.

Details

This is a Lagrange multiplier test for the null of no serial correlation, against the alternative of either an AR(1) or an MA(1) process, in the idiosyncratic component of the error term in a random effects panel model (as the analytical expression of the test turns out to be the same under both alternatives, see Baltagi and Li (1995, 1998)). The `alternative` argument, defaulting to `twosided`, allows testing for positive serial correlation only, if set to `onesided`.

Value

An object of class `"htest"`.

Author(s)

Giovanni Millo

References

Baltagi, B.H. and Li, Q. (1995) Testing AR(1) against MA(1) disturbances in an error component model, *Journal of Econometrics* **68**, p.133-151.

Baltagi, B.H. and Li, Q. (1997) Monte Carlo results on pure and pretest estimators of an error components model with autocorrelated disturbances, *Annales d'economie et de statistique* **48**, p.69-82.

See Also

`pdwtest`, `bgtest`, `pbsytest`, `pwartest` and `pwfdtest` for other serial correlation tests for panel models.

Examples

```
data("Grunfeld", package = "plm")
pbltest(inv ~ value + capital, data = Grunfeld)
```

`pbsytest` *Bera, Sosa-Escudero and Yoon Locally–Robust Lagrange Multiplier Tests for Panel Models*

Description

Test for residual serial correlation (or individual random effects) locally robust vs. individual random effects (serial correlation) for panel models and joint test by Baltagi and Li.

Usage

```
pbsytest(x,...)
## S3 method for class 'panelmodel':
pbsytest(x, test = c("ar","re","j"), ...)
## S3 method for class 'formula':
pbsytest(x, data, ..., test = c("ar","re","j"))
```

Arguments

x	an object of class "formula" or of class panelmodel,
data	a data.frame,
test	a character string indicating which test to perform: first-order serial correlation (ar), random effects (re) or joint test for either of them (j).
...	further arguments.

Details

These Lagrange multiplier tests are robust vs. local misspecification of the alternative hypothesis, i.e. they test the null of serially uncorrelated residuals against AR(1) residuals in a pooling model, allowing for local departures from the assumption of no random effects; or they test the null of no random effects allowing for local departures from the assumption of no serial correlation in residuals. They use only the residuals of the pooled OLS model and correct for local misspecification as outlined in Bera *et al.* (2001).

The joint test is due to Baltagi and Li (1991) and is added for convenience under this same interface.

Value

An object of class "htest".

Author(s)

Giovanni Millo

References

Bera, A.K., Sosa-Escudero, W. and Yoon, M. (2001), Tests for the error component model in the presence of local misspecification, *Journal of Econometrics*, **101**, pp. 1-23.

Baltagi, B. and Li, Q. (1991), A joint test for serial correlation and random individual effects, *Statistics and Probability Letters*, **11**, pp. 277-280.

See Also

plmtest for individual and/or time random effects tests based on a correctly specified model; pbltest, pbgtest and pdwtest for serial correlation tests in random effects models.

Examples

```
## Example in Bera et al.
data(Grunfeld, package = "plm")
## Bera et al. use a subset of the original Grunfeld data,
## so results are slightly different here
## default is AR testing
pbsytest(inv ~ value + capital, data = Grunfeld, index = c("firm","year"))
pbsytest(inv ~ value + capital, data = Grunfeld, test="re", index = c("firm","year"))
pbsytest(inv ~ value + capital, data = Grunfeld, test="j", index = c("firm","year"))
```

`pcdtest` *Tests of cross-section dependence for panel models*

Description

Pesaran's CD or Breusch-Pagan's LM (local or global) tests for cross sectional dependence in panel models

Usage

```
pcdtest(x, ...)
## S3 method for class 'panelmodel':
pcdtest(x, test=c("cd","sclm","lm"), w=NULL, ...)
## S3 method for class 'formula':
pcdtest(x, data, index=NULL, model=NULL, test=c("cd","sclm","lm"), w=NULL, ...)
```

Arguments

`x`	an object of class `formula` or `panelmodel`, describing the model to be tested
`data`	a `data.frame`
`index`	an optional numerical index, in case `data` has to be formatted by `plm.data`
`model`	an optional character string indicating which type of model to estimate; if left to `NULL`, the original heterogeneous specification of Pesaran is used
`test`	the type of test statistic to be returned. One of "cd" for Pesaran's CD statistic, "lm" for Breusch and Pagan's original LM statistic, "sclm" for the scaled version of Breusch and Pagan's LM statistic
`w`	an `n x n matrix` describing proximity between observations, with $w_i j = a$ where a is any number such that `as.logical(a)=TRUE`, if i, j are neighbours, 0 or any number b such that `as.logical(b)=FALSE` elsewhere
`...`	further arguments to be passed on to `plm`, such as e.g. `effect` or `random.method`

Details

These tests are originally meant to use the residuals of separate estimation of one time-series regression for each cross-sectional unit in order to check for cross-sectional dependence. If a different model specification (`within`, `random`, ...) is assumed consistent, one can resort to its residuals for testing (which is common, e.g., when the time dimension's length is insufficient for estimating the heterogeneous model). If the time dimension is insufficient and `model=NULL`, the function defaults to estimation of a `within` model and issues a warning. The main argument of this function may be either a model of class `panelmodel` or a `formula` and `dataframe`; in the second case, unless `model` is set to `NULL`, all usual parameters relative to the estimation of a `plm` model may be passed on. The test is compatible with any consistent `panelmodel` for the data at hand, with any specification of `effect`. E.g., specifying `effect="time"` or `effect="twoways"` allows to test for residual cross-sectional dependence after the introduction of time fixed effects to account for common shocks. A *local* version of either test can be computed supplying a proximity matrix (coercible to `logical`) providing information on whether any pair of observations are neighbours or not. If `w` is supplied, only neighbouring pairs will be used in computing the test; else, `w` will default to `NULL` and all observations will be used. The matrix need not be binary, so commonly used "row-standardized" matrices can be employed as well. `nb` objects from **spdep** must instead be transformed into matrices by `nb2mat` before using.

Value

An object of class `"htest"`.

References

Breusch, T.S. and A.R. Pagan (1980), The Lagrange multiplier test and its applications to model specification in econometrics, *Review of Economic Studies*, 47, pp. 239–253.

Pesaran, H. (2004), General Diagnostic Tests for Cross Section Dependence in Panels, *CESIfo* Working Paper 1229. http://www.cesifo-group.de/DocCIDL/cesifo1_wp1229.pdf

Examples

```
data(Grunfeld, package = "plm")
## test on heterogeneous model (separate time series regressions)
pcdtest(inv ~ value + capital, data=Grunfeld,
        index = c("firm","year"))
## test on two-way fixed effects homogeneous model
pcdtest(inv ~ value + capital, data=Grunfeld, model="within",
        effect="twoways", , index = c("firm","year"))
## test on model object
g <- plm(inv ~ value + capital, data=Grunfeld, index = c("firm","year"))
pcdtest(g)
## scaled LM test
pcdtest(g,test="sclm")
```

`pdata.frame` *data.frame for panel data*

Description

An object of this class is a data.frame with an attribute that describes its time and individual dimensions.

Usage

```
pdata.frame(x, index = NULL, drop.index = FALSE, row.names = TRUE)
## S3 method for class 'pdata.frame':
x[i, j, drop = TRUE]
## S3 method for class 'pdata.frame':
x[[y]]
```

```
## S3 method for class 'pdata.frame':
x$y
## S3 method for class 'pdata.frame':
print(x, ...)
## S3 method for class 'pdata.frame':
as.data.frame(x, row.names = NULL, optional = FALSE, ...)
```

Arguments

`x`	a `data.frame` for the `pdata.frame` function and a `pdata.frame` for the methods,
`i`	see `Extract`,
`j`	see `Extract`,
`y`	one of the columns of the `data.frame`,
`index`	this argument indicates the individual and time indexes. See details,
`drop`	see `Extract`,
`drop.index`	should the indexes be removed from the data.frame ?
`optional`	see `as.data.frame`
`row.names`	should “fancy” row names be computed ?
`...`	further arguments

Details

The `index` argument indicates the dimensions of the panel. It can be:

- a character string which is the name of the individual index variable, in this case a new variable called “time” which contains the time index is added,
- an integer, the number of individuals in case of balanced panel, in this case two new variables “time” and “id” which contain the individual and the time indexes are added,
- a vector of two character strings which contains the names of the individual and of the time indexes.

The `index` attribute is a `data.frame` which contains the individual and the time indexes. The `"[["` and `"$"` extract a series from the `pdata.frame`. The `"index"` attribute is then added to the series and a class attribute `"pseries"` is added. The `"["` method behaves as for `data.frame`, except that the extraction is also applied to the `index` attribute. `as.data.frame` removes the index from the `pdata.frame` and adds it to every series.

Value

a `pdata.frame` object: this is a `data.frame` with an `index` attribute which is a `data.frame` with two variables, the individual and the time indexes.

Author(s)

Yves Croissant

Examples

```
data("Wages", package = "plm")
Wag <- pdata.frame(Wages, 595)

# Gasoline contains two variables which are individual and time indexes
data("Gasoline", package = "plm")
Gas <- pdata.frame(Gasoline, c("country","year"), drop = TRUE)

# Hedonic is an unbalanced panel, townid is the individual index
data("Hedonic", package="plm")
Hed <- pdata.frame(Hedonic, "townid", row.names = FALSE)
```

---

pdim *Check for the Dimensions of the Panel*

Description

This function checks the number of individuals and time observations in the panel and whether it is balanced or not.

Usage

```
pdim(x,...)
## S3 method for class 'data.frame':
pdim(x, index = NULL, ...)
## S3 method for class 'panelmodel':
pdim(x, ...)
## S3 method for class 'pdata.frame':
pdim(x, ...)
```

Arguments

x	a `data.frame`, a `pdata.frame` or a `panelmodel` object,
index	see `pdata.frame`,
...	further arguments.

Details

`pdim` is called by the estimation functions.

Value

An object of class `pdim` containing the following elements:

nT	a list containing `n`, the number of individuals, `T`, the number of time observations, `N` the total number of observations,

`TinT`	a list containing two vectors : `Ti` gives the number of observations for each individuals and `nt` gives the number of individuals observed for each period,
`balanced`	a logical value : `TRUE` for a balanced panel, `FALSE` for an unbalanced panel,
`panel.names`	a list of character vectors : `id.names` contains the names of each individual and `time.names` contains the names of each period.

Author(s)

Yves Croissant

See Also

`pdata.frame`

Examples

```
# There are 595 individuals

data("Wages", package = "plm")
pdim(Wages, 595)

# Gasoline contains two variables which are individual and time indexes
# and are the first two variables
data("Gasoline", package="plm")
pdim(Gasoline)

# Hedonic is an unbalanced panel, townid is the individual index
data("Hedonic", package = "plm")
pdim(Hedonic, "townid")

# An example of the panelmodel method
data("Produc", package = "plm")
z <- plm(log(gsp)~log(pcap)+log(pc)+log(emp)+unemp,data=Produc,
         model="random", subset = gsp > 5000)
pdim(z)
```

`pdwtest` *Durbin–Watson Test for Panel Models*

Description

Test of serial correlation for (the idiosyncratic component of) the errors in panel models.

Usage

```
pdwtest(x, ...)
## S3 method for class 'panelmodel':
pdwtest(x, ...)
## S3 method for class 'formula':
pdwtest(x, data, ...)
```

Arguments

`x`	an object of class `"panelmodel"` or of class `"formula"`,
`data`	a `data.frame`,
`...`	further arguments to be passed on to `dwtest`.

Details

This Durbin–Watson test uses the auxiliary model on (quasi-)demeaned data taken from a model of class `plm` which may be a `pooling` (the default), `random` or `within` model. It performs a dw test (using `dwtest` from package `lmtest`) on the residuals of the (quasi-)demeaned model, which should be serially uncorrelated under the null of no serial correlation in idiosyncratic errors. The function takes the demeaned data, estimates the model and calls `dwtest`.

Value

An object of class `"htest"`.

Author(s)

Giovanni Millo

References

Baltagi, B.H. (2005) *Econometric Analysis of Panel Data*, 3rd. ed., Wiley, p. 98.

Wooldridge, J.M. (2002) *Econometric Analysis of Cross-Section and Panel Data*, MIT Press, p. 288.

See Also

`pbgtest` for the analogous Breusch–Godfrey test, `dwtest` for the Breusch–Godfrey test for serial correlation in the linear model. `pbltest`, `pbsytest`, `pwartest` and `pwfdtest` for other serial correlation tests for panel models.

Examples

```
data("Grunfeld", package = "plm")
g <- plm(inv ~ value + capital, data = Grunfeld, model="random")
pdwtest(g)
pdwtest(g, alternative="two.sided")
## formula interface
pdwtest(inv ~ value + capital, data=Grunfeld, model="random")
```

pFormula *pFormula: An extended Formula interface for panel data*

Description

pFormula is a Formula object, with methods useful for panel data.

Usage

```
pFormula(object)
## S3 method for class 'pFormula':
as.Formula(x, ...)
## S3 method for class 'pFormula':
model.frame(formula, data, ...,
                                 lhs = NULL, rhs = NULL)
## S3 method for class 'pFormula':
model.matrix(object, data,
                                  model = c("pooling","within","Between",
  "between","mean","random","fd"),
                                  effect = c("individual","time","twoways"),
                                  rhs = 1,
                                  theta = NULL, ...)
```

Arguments

`object, formula, x`
: an object of class "pFormula",

`data`
: a data.frame,

`effect`
: the effects introduced in the model, one of "individual", "time" or "twoways",

`model`
: one of "pooling", "within", "between", "random", "fd" and "ht",

`theta`
: the parameter for the transformation if model = "random",

`lhs`
: ,

`rhs`
: ,

`...`
: further arguments.

Details

The lhs and rhs arguments are inherited from Formula. The model.frame method returns a pdata.frame object. The model.matrix unables the transformation specified by the model and effect arguments.

Value

An object of class c("pFormula", "Formula".

Author(s)

Yves Croissant

---

pFtest *F Test for Individual and/or Time Effects*

Description

Test of individual and/or time effects based on the comparison of the `within` and the `pooling` model.

Usage

```
pFtest(x, ...)
## S3 method for class 'plm':
pFtest(x, z, ...)
## S3 method for class 'formula':
pFtest(x, data, ...)
```

Arguments

`x`	an object of class `"plm"` or of class `"formula"`,
`z`	an object of class `"plm"`,
`data`	a `data.frame`,
`...`	further arguments.

Details

For the `plm` method, the argument of this function is two `plm` objects, the first being a within model, the second a pooling model. The effects tested are either individual, time or twoways, depending on the effects introduced in the model.

Value

An object of class `"htest"`.

Author(s)

Yves Croissant

See Also

`plmtest` for Lagrange multipliers tests of individuals and/or time effects.

Examples

```
data("Grunfeld", package="plm")
gi <- plm(inv ~ value + capital, data = Grunfeld, model = "pooling")
gt <- plm(inv ~ value + capital, data = Grunfeld,
          effect = "time", model = "within")
gd <- plm(inv ~ value + capital, data = Grunfeld,
          effect = "twoways", model = "within")
pFtest(gt, gi)
pFtest(gd, gi)
pFtest(inv ~ value + capital, data = Grunfeld, effect = "twoways")
```

`pggls` *General FGLS Estimators*

Description

General FGLS estimators for panel data (balanced or unbalanced)

Usage

```
pggls(formula, data, subset, na.action, effect = c("individual","time"), model = c(
index = NULL, ...)
## S3 method for class 'pggls':
summary(object, ...)
## S3 method for class 'summary.pggls':
print(x,digits = max(3, getOption("digits") -
2), width = getOption("width"),...)
```

Arguments

`formula`	a symbolic description of the model to be estimated,
`object, x`	an object of class `pggls`,
`data`	a `data.frame`,
`subset`	see `lm`,
`na.action`	see `lm`,
`effect`	the effects introduced in the model, one of `"individual"` or `"time"`,
`model`	one of `"within"` or `"random"`,
`index`	the indexes, see `plm.data`,
`digits`	digits,
`width`	the maximum length of the lines in the print output,
`...`	further arguments.

Details

pggls is a function for the estimation of linear panel models by general feasible generalized least squares, either with or without fixed effects. General FGLS is based on a two-step estimation process: first a model is estimated by OLS (random) or fixed effects (within), then its residuals are used to estimate an error covariance matrix for use in a feasible-GLS analysis. This framework allows the error covariance structure inside every group (if effect="individual", else symmetric) of observations to be fully unrestricted and is therefore robust against any type of intragroup heteroskedasticity and serial correlation. Conversely, this structure is assumed identical across groups and thus general FGLS estimation is inefficient under groupwise heteroskedasticity. Note also that this method requires estimation of $T(T+1)/2$ variance parameters, thus efficiency requires N > > T (if effect="individual", else the opposite).

Value

An object of class c("pggls","panelmodel") containing:

- coefficients: the vector of coefficients,
- residuals: the vector of residuals,
- fitted.values: the vector of fitted.values,
- vcov: the covariance matrix of the coefficients,
- df.residual: degrees of freedom of the residuals,
- model: a data.frame containing the variables used for the estimation,
- call: the call,
- sigma: the estimated intragroup (or cross-sectional, if effect="time") covariance of errors,

Author(s)

Giovanni Millo

References

Kiefer, N. M. (1980) Estimation of Fixed Effects Models for Time Series of Cross-Sections with Arbitrary Intertemporal Covariance, *Journal of Econometrics*, **14**, 195–202.

Wooldridge J. M. (2002) *Econometric Analysis of Cross Section and Panel Data*, MIT Press.

Examples

```
data("Produc", package = "plm")
zz <- pggls(log(gsp) ~ log(pcap) + log(pc) + log(emp) + unemp, data = Produc, model = "rando
summary(zz)
```

pgmm *Generalized Method of Moments (GMM) Estimation for Panel Data*

Description

Generalized method of moments estimation for static or dynamic models with panel data.

Usage

```
pgmm(formula, data, effect = c("individual","twoways", "none"),
     model = c("onestep", "twosteps"),
     instruments = NULL, gmm.inst, lag.gmm, lag.form,
     transformation = c("d", "ld"),
     fsm = NULL, ...)
## S3 method for class 'pgmm':
summary(object,robust, ...)
## S3 method for class 'summary.pgmm':
print(x, digits = max(3, getOption("digits") - 2),
    width = getOption("width"),...)
```

Arguments

`formula`	a symbolic description for the model to be estimated. Should be an object of class `"dynformula"` or of class `"formula"`. In the latter case, a `lag.form` argument should be provided,
`object,x`	an object of class `"pgmm"`,
`data`	a `data.frame`,
`effect`	the effects introduced in the model, one of `"individual"` or `"twoways"`,
`model`	one of `"onestep"` or `"twosteps"`,
`instruments`	a one-sided formula containing the normal instruments,
`gmm.inst`	a one-sided formula containing the GMM instruments,
`lag.gmm`	a list containing the lags to use for each GMM instrument, or a list of two named lists (`"d"` for difference and `"l"` for level) in case of "system GMM",
`lag.form`	a list containing the lags to use for each variable of the formula. Only relevant if `formula` is a standard formula and not a `dynformula`,
`transformation`	the kind of transformation to apply to the model: either `"d"` (the default value) for the "difference GMM" model or `"ld"` for the "system GMM",
`fsm`	the matrix for the one step estimator : one of `"I"` (identity matrix) or `"G"` ($= D'D$ where D is the first-difference operator) if `transformation="d"`, one of `"GI"` or `"full"` if `transformation="ld"`,
`digits`	digits,
`width`	the maximum length of the lines in the print output,
`robust`	if `TRUE`, robust inference is performed in the summary,
`...`	further arguments.

Details

pgmm estimates a model for panel data with a generalized method of moments (GMM) estimator. The description of the model to estimate is provided with a dynformula. The GMM instruments are provided by filling the pgmm.inst argument with a one-sided formula. By default, all the variables of the model which are not used as GMM instruments are used as normal instruments with the same lag structure as the one specified in the dynformula, but the user may also specify a one-sided formula to indicate expicitly the variables to use as normal instruments. The lags used for the GMM instruments are indicated with the lag.gmm argument. For each instrument, it is either a vector of length 2 (e.g., c(99,3) indicates to use all the lags available up to t-3), or a scalar (e.g, 2 indicates to use the lags 0, 1 and 2). If a single vector (or scalar) is written, then this lag structure is replicated for all GMM instruments. Otherwise, lag.gmm should be a list of length equal to the number of GMM instruments. transformation indicates how the model should be transformed for the estimation. "d" gives the “difference GMM” model (see Arellano and Bond (1991)), "ld" the “system GMM” model (see Blundell and Bond (1998)).

pgmm is an attempt to adapt GMM estimators available within the DPD library for GAUSS (see Arellano and Bond 1998) and Ox (see Doornik, Arellano and Bond 2006) and with the xtabond2 library for STATA (see Roodman 2009).

Value

An object of class c("pgmm","panelmodel"), which has the following elements:

coefficients	the vector (or the list for fixed effects) of coefficients,
residuals	the vector of residuals,
fitted.values	the vector of fitted.values,
vcov	the covariance matrix of the coefficients,
df.residual	degrees of freedom of the residuals,
model	a list containing the variables used for the estimation for each individual,
W	a list containing the instruments for each individual (two lists in case of “sys–GMM”),
K	a list containing K the number of explanatory variables, Ky the number of lags of the dependent variable and Kt the number of time dummies,
A1	the weighting matrix for the one–step estimator,
A2	the weighting matrix for the two–steps estimator,
call	the call.

It has print, summary and print.summary methods.

Author(s)

Yves Croissant

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*, vol. **58**(2), 1991, pp.227–297.

Arellano, M. and Bond, S. (1998) Dynamic Panel Data Estimation Using DPD98 for GAUSS: A Guide for Users.

Blundell, R. and Bond, S. (1998) Initial Conditions and Moment Restrictions in Dynamic Panel Data Models, *Journal of Econometrics*, vol. **87**, pp.115–143.

Doornik, J., Arellano, M. and Bond, S. (2006) Panel Data Estimation using DPD for Ox. http://www.doornik.com/download/dpd.pdf

Roodman, D. (2009) How to do xtabond2: An Introduction to "difference" and "system" GMM in Stata, *Stata Journal*, vol. **9**(1), pp. 86–136. Working Paper version at http://ideas.repec.org/p/cgd/wpaper/103.html

See Also

`dynformula` for dynamic formulas, `sargan` for Sargan tests and `mtest` for Arellano–Bond's tests of multicollinearity.

Examples

```
data("EmplUK", package = "plm")

## Arellano and Bond (1991), table 4b
z1 <- pgmm(log(emp) ~ log(wage) + log(capital) + log(output),
            lag.form = list(2,1,0,1), data = EmplUK,
            effect = "twoways", model = "twosteps",
            gmm.inst = ~log(emp), lag.gmm = list(c(2,99)))
summary(z1)

## Blundell and Bond (1998) table 4 (cf DPD for OX p.12 col.4)
z2 <- pgmm(dynformula(log(emp) ~ log(wage) + log(capital), list(1,1,1)),
            data = EmplUK, effect = "twoways", model = "onestep",
            gmm.inst = ~log(emp) + log(wage) + log(capital),
            lag.gmm = c(2,99), transformation = "ld")
summary(z2, robust = TRUE)
```

---

pht — *Hausman-Taylor Estimator for Panel Data*

Description

The Hausman-Taylor estimator is an instrumental variable estimator without external instruments.

Usage

```
pht(formula, data, subset, na.action, index = NULL, ...)
## S3 method for class 'pht':
summary(object, ...)
## S3 method for class 'summary.pht':
print(x, digits = max(3, getOption("digits") - 2), width = getOption("width"), ...)
```

Arguments

`formula`	a symbolic description for the model to be estimated,
`object,x`	an object of class "plm",
`data`	a `data.frame`,
`subset`	see `lm`,
`na.action`	see `lm`,
`index`	the indexes,
`digits`	digits,
`width`	the maximum length of the lines in the print output,
`...`	further arguments.

Details

`pht` estimates panels models using the Hausman-Taylor estimator. The model is specified a two-part formula, the second part containing the exogenous variables.

Value

An object of class `c("pht", "plm", "panelmodel")`.

A "`pht`" object contains the same element as `plm`, with a further argument called `varlist` which describes the typology of the variables. It has `summary` and `print.summary` methods.

Author(s)

Yves Croissant

References

Hausman, J.A. and Taylor W.E. (1981) Panel data and unobservable individual effects, *Econometrica*, **49**, pp.1377–1398.

Examples

```
data("Wages", package = "plm")
ht <- plm(lwage~wks+south+smsa+married+exp+I(exp^2)+
          bluecol+ind+union+sex+black+ed |
          sex+black+bluecol+south+smsa+ind,
          data=Wages,model="ht",index=595)
summary(ht)
```

phtest *Hausman Test for Panel Models*

Description

Specification test for panel models.

Usage

```
phtest(x, ...)
## S3 method for class 'panelmodel':
phtest(x, x2, ...)
## S3 method for class 'formula':
phtest(x, data, ..., model = c("within","random"))
```

Arguments

`x`	an object of class "`panelmodel`" or "`formula`",
`x2`	an object of class "`panelmodel`",
`model`	a vector containing the names of two models,
`data`	a `data.frame`,
`...`	further arguments passed to `plm`

Details

The Hausman test is based on the difference of the vectors of coefficients of two different models.

Value

An object of class "`htest`".

Author(s)

Yves Croissant

References

Hausman, J.A. (1978), Specification tests in econometrics, *Econometrica*, **46**, pp.1251–1271.

Examples

```
data("Gasoline", package = "plm")
form <- lgaspcar ~ lincomep + lrpmg + lcarpcap
wi <- plm(form, data = Gasoline, model = "within")
re <- plm(form, data = Gasoline, model = "random")
phtest(wi, re)
phtest(form, data = Gasoline)
```

plm *Panel Data Estimators*

Description

Linear models for panel data estimated using the `lm` function on transformed data.

Usage

```
plm(formula, data, subset, na.action, effect = c("individual","time","twoways"),
    model = c("within","random","ht","between","pooling","fd"),
    random.method = c("swar","walhus","amemiya","nerlove"),
    inst.method = c("bvk","baltagi"), index = NULL, ...)
## S3 method for class 'plm':
summary(object, ...)
## S3 method for class 'summary.plm':
print(x, digits = max(3, getOption("digits") - 2),
    width = getOption("width"), ...)
```

Arguments

`formula`	a symbolic description for the model to be estimated,
`object,x`	an object of class `"plm"`,
`data`	a `data.frame`,
`subset`	see `lm`,
`na.action`	see `lm`,
`effect`	the effects introduced in the model, one of `"individual"`, `"time"` or `"twoways"`,
`model`	one of `"pooling"`, `"within"`, `"between"`, `"random"`, `"fd"` and `"ht"`,
`random.method`	method of estimation for the variance components in the random effects model, one of `"swar"` (the default value), `"amemiya"`, `"walhus"` and `"nerlove"`,
`inst.method`	the instrumental variable transformation: one of `"bvk"` and `"baltagi"`,
`index`	the indexes,
`digits`	digits,
`width`	the maximum length of the lines in the printed output,
`...`	further arguments.

Details

`plm` is a general function for the estimation of linear panel models. It supports the following estimation methods: pooled OLS (`model="pooling"`), fixed effects (`"within"`), random effects (`"random"`), first–differences (`"fd"`) and between (`"between"`). It supports unbalanced panels and two–way effects (although not with all methods).

For random effect models, 4 estimators of the transformation parameter are available : `swar` (Swamy and Arora), `amemiya`, `walhus` (Wallace and Hussain) and `nerlove`.

Instrumental variables estimation is obtained using two-part formulas, the second part indicating the instrumental variables used. This can be a complete list of instrumental variables or an update of the first part. If, for example, the model is `y ~ x1 + x2 + x3`, `x1`, `x2` are endogenous and `z1`, `z2` are external instruments, the model can be estimated with:

- `formula=y~x1+x2+x3 | x3+z1+z2`,
- `formula=y~x1+x2+x3 | .-x1-x2+z1+z2`.

Balestra and Varadharajan–Krishnakumar's or Baltagi's method is used if `inst.method="bvk"` or if `inst.method="baltagi"`.

The Hausman and Taylor estimator is computed if `model="ht"`.

Value

An object of class `c("plm","panelmodel")`.

A `"plm"` object has the following elements :

`coefficients`	the vector of coefficients,
`vcov`	the covariance matrix of the coefficients,
`residuals`	the vector of residuals,
`df.residual`	degrees of freedom of the residuals,
`formula`	an object of class `'pFormula'` descrbing the model,
`model`	a data.frame of class `'pdata.frame'` containing the variables used for the estimation: the response is in first position and the two indexes in the last positions,
`ercomp`	an object of class `'ercomp'` providing the estimation of the components of the errors (for random effect models only),
`call`	the call,

It has `print`, `summary` and `print.summary` methods.

Author(s)

Yves Croissant

References

Amemiya, T. (1971) The estimation of the variances in a variance–components model, *International Economic Review*, **12**, pp.1–13.

Balestra, P. and Varadharajan–Krishnakumar, J. (1987) Full information estimations of a system of simultaneous equations with error components structure, *Econometric Theory*, **3**, pp.223–246.

Baltagi, B.H. (1981) Simultaneous equations with error components, *Journal of Econometrics*, **17**, pp.21–49.

Baltagi, B.H. (2001) *Econometric Analysis of Panel Data*, 2nd ed. John Wiley and Sons, Ltd.

Hausman, J.A. and Taylor W.E. (1981) Panel data and unobservable individual effects, *Econometrica*, **49**, pp.1377–1398.

Nerlove, M. (1971) Further evidence on the estimation of dynamic economic relations from a time–series of cross–sections, *Econometrica*, **39**, pp.359–382.

Swamy, P.A.V.B. and Arora, S.S. (1972) The exact finite sample properties of the estimators of coefficients in the error components regression models, *Econometrica*, **40**, pp.261–275.

Wallace, T.D. and Hussain, A. (1969) The use of error components models in combining cross section with time series data, *Econometrica*, **37**(1), pp.55–72.

Examples

```
data("Produc", package = "plm")
zz <- plm(log(gsp) ~ log(pcap) + log(pc) + log(emp) + unemp, data = Produc, index = c("state
summary(zz)
```

---

plm.data — *Data Frame Special Format for Panel Data*

Description

This function transforms a data frame in a format suitable for using with the estimation functions of `plm`.

Usage

```
x <- plm.data(x, index = NULL)
```

Arguments

`x`	a `data.frame`,
`index`	a vector (of length one or two) indicating the (individual and time) indexes.

Details

`indexes` can be:

- a character string which is the name of the individual index variable, in this case a new variable called "time" containing the time index is added,
- an integer, the number of individuals in the case of balanced panel, in this case two new variables "time" and "id" containing the individual and the time indexes are added,
- a vector of two character strings which contains the names of the individual and of the time indexes.

Value

A `data.frame`.

Author(s)

Yves Croissant

Examples

```
# There are 595 individuals
data("Wages", package = "plm")
Wages <- plm.data(Wages, 595)

# Gasoline contains two variables which are individual and time indexes
# The pdata.frame is called gas
data("Gasoline", package = "plm")
Gasoline <- plm.data(Gasoline, c("country","year"))
summary(Gasoline)

# Hedonic is an unbalanced panel, townid is the individual index
data("Hedonic", package = "plm")
Hedonic <- plm.data(Hedonic, "townid")
```

---

plmtest *Lagrange Multiplier Tests for Panel Models*

Description

Test of individual and/or time effects for panel models.

Usage

```
plmtest(x, ...)
## S3 method for class 'plm':
plmtest(x, effect = c("individual", "time", "twoways"),
                         type = c("honda","bp", "ghm", "kw"), ...)
## S3 method for class 'formula':
plmtest(x, data, ..., effect = c("individual", "time", "twoways"),
                         type = c("honda","bp", "ghm", "kw"))
```

Arguments

`x`	an object of class `"plm"`,
`data`	a `data.frame`,
`effect`	a character string indicating which effects are tested:individual effects (`"individual"`), time effects (`"time"`) or both (`"twoways"`),
`type`	a character string indicating the test to be computed : `"bp"` for Breush-Pagan (1980), `"ghm"` for Gourieroux, Holly and Monfort (1982), `"honda"` for Honda (1985) or `"kw"` for King and Wu (1997),
`...`	further arguments passed to `plm`.

Details

These Lagrange multiplier tests use only the residuals of the pooling model. The main argument of this function may be either a pooling model of class `plm` or an object of class `formula`.

Value

An object of class `"htest"`.

Author(s)

Yves Croissant

References

Breusch, T.S. and Pagan, A.R. (1980) The Lagrange multiplier test and its applications to model specification in econometrics, *Review of Economic Studies*, **47**, pp.239–253.

Gourieroux, C., Holly, A. and Monfort, A. (1982) Likelihood ratio test, Wald test, and Kuhn–Tucker test in linear models with inequality constraints on the regression parameters, *Econometrica*, **50**, pp.63–80.

Honda, Y. (1985) Testing the error components model with non–normal disturbances, *Review of Economic Studies*, **52**, pp.681–690.

King, M.L. and Wu, P.X. (1997) Locally optimal one–sided tests for multiparameter hypotheses, *Econometric Reviews*, **33**, pp.523–529.

See Also

`pFtest` for individual and/or time effects tests based on the within model.

Examples

```
data("Grunfeld", package = "plm")
g <- plm(inv ~ value + capital, data = Grunfeld, model = "pooling")
plmtest(g)
plmtest(g, effect="time")
plmtest(inv ~ value + capital, data = Grunfeld, type = "honda")
plmtest(inv ~ value + capital, data = Grunfeld, type="ghm",effect="twoways")
plmtest(inv ~ value + capital, data = Grunfeld, type="kw",effect="twoways")
```

`pmodel.response` *A function to extract the model.response*

Description

pmodel.response has several methods to conveniently extract the response of several objects.

Usage

```
pmodel.response(object, ...)
## S3 method for class 'data.frame':
pmodel.response(object,
                                     model = c("pooling","within","Between",
                                               "between","mean","random","fd"),
                                    effect = c("individual","time","twoways"),
                                    lhs = NULL,
                                    theta = NULL, ...)
## S3 method for class 'pFormula':
pmodel.response(object, data,
                                   model = c("pooling","within","Between",
                                             "between","mean","random","fd"),
                                   effect = c("individual","time","twoways"),
                                   lhs = NULL,
                                   theta = NULL, ...)
## S3 method for class 'plm':
pmodel.response(object,
                              model = c("pooling","within","Between",
                                        "between","mean","random","fd"),
                              effect = c("individual","time","twoways"),
                              theta = NULL, ...)
```

Arguments

`object`	an object of class `"plm"`,
`data`	a `data.frame`,
`effect`	the effects introduced in the model, one of `"individual"`, `"time"` or `"twoways"`,
`model`	one of `"pooling"`, `"within"`, `"between"`, `"random"`, `"fd"` and `"ht"`,
`theta`	the parameter for the transformation if `model = "random"`,
`lhs`	,
`...`	further arguments.

Details

The model response is extracted (from a `data.frame`, a `pFormula` or a `plm` object, and the transformation specified by `effect` and `model` is applied to it.

Value

A numeric vector.

Author(s)

Yves Croissant

pooltest *Test of Poolability*

Description

A Chow test for the poolability of the data.

Usage

```
pooltest(x, ...)
## S3 method for class 'plm':
pooltest(x, z, ...)
## S3 method for class 'formula':
pooltest(x, data, ...)
```

Arguments

x	an object of class "plm",
z	an object of class "pvcm" obtained with model="within",
data	a data.frame,
...	further arguments passed to plm.

Details

pooltest is an *F* test of stability (or Chow test) for the coefficients of a panel model. The estimated plm object should be a "pooling" model or a "within" model (the default); intercepts are assumed to be identical in the first case and different in the second case.

Value

An object of class "htest".

Author(s)

Yves Croissant

Examples

```
data("Gasoline", package = "plm")
form <- lgaspcar ~ lincomep + lrpmg + lcarpcap
gasw <-  plm(form, data = Gasoline, model = "within")
gasp <-  plm(form, data = Gasoline, model = "pooling")
gasnp <-  pvcm(form, data = Gasoline, model = "within")
pooltest(gasw, gasnp)
pooltest(gasp, gasnp)

pooltest(form, data = Gasoline, effect = "individual", model = "within")
pooltest(form, data = Gasoline, effect = "individual", model = "pooling")
```

`Produc` *US States Production*

Description

A panel of 48 observations from 1970 to 1986

total number of observations : 816

observation : regional

country : United States

Usage

```
data(Produc)
```

Format

A data frame containing :

state the state

year the year

pcap private capital stock

hwy highway and streets

water water and sewer facilities

util other public buildings and structures

pc public capital

gsp gross state products

emp labor input measured by the employement in non–agricultural payrolls

unemp state unemployment rate

Source

Online complements to Baltagi (2001).

http://www.wiley.com/legacy/wileychi/baltagi/.

References

Baltagi, Badi H. (2003) *Econometric analysis of panel data*, 2nd ed. John Wiley and Sons,

Baltagi, B. H. and N. Pinnoi (1995) "Public capital stock and state productivity growth: further evidence", *Empirical Economics*, **20**, 351–359.

Munnell, A.H. (1990) "Why has productivity growth declined? Productivity and public investment", *New England Economic Review*, 3–22.

pseries *panel series*

Description

A class for panel series for which several useful computations are available.

Usage

```
between(x, ...)
Between(x, ...)
Within(x, ...)
## S3 method for class 'pseries':
lag(x, k = 1, ...)
## S3 method for class 'pseries':
diff(x, lag = 1, ...)
## S3 method for class 'pseries':
as.matrix(x, idbyrow = TRUE, ...)
## S3 method for class 'pseries':
between(x, effect = c("individual", "time"), ...)
## S3 method for class 'pseries':
Between(x, effect = c("individual", "time"), ...)
## S3 method for class 'pseries':
Within(x, effect = c("individual", "time"), ...)
## S3 method for class 'pseries':
summary(object, ...)
## S3 method for class 'summary.pseries':
print(x, ...)
```

Arguments

`x, object`	a `pseries` or a `summary.pseries` object,
`effect`	the (individual or time) effect,
`k, lag`	the number of lags for the `lag` and `diff` methods,
`idbyrow`	if TRUE, the lines of the matrix are the individuals,
`...`	further arguments.

Details

A `pseries` is obtained when a series is extracted from a `pdata.frame` object. It consists on the original series with the `index` attribute of the `pdata.frame`. Specific transformations (within and between) are available and special `lag` and `diff` methods are provided.

Value

All these functions return an object of class `pseries`, except `between` which is a numeric vector.

Author(s)

Yves Croissant

Examples

```
# Create first a pdata.frame
data("EmplUK", package = "plm")
Em <- pdata.frame(EmplUK)
# Then extract a serie, which becomes a pseries
z <- Em$output
class(z)
# obtain the matrix representation
as.matrix(z)
# compute the between, within transformation
between(z)
Within(z)
# Between replicate the values for each time observations
Between(z)
# compute the first and third lag, and the difference laged twice
lag(z)
lag(z, 3)
diff(z, 2)
```

---

purtest *Unit root tests for panel data*

Description

purtest implements several testing procedures that have been proposed to test unit root hypotheses with panel data.

Usage

```
purtest(object, data = NULL, index = NULL,
        test= c("levinlin", "ips", "madwu", "hadri"),
        exo = c("none", "intercept", "trend"),
        lags = c("SIC", "AIC", "Hall"), pmax = 10, Hcons = TRUE,
        q = NULL, dfcor = FALSE, fixedT = TRUE, ...)
## S3 method for class 'purtest':
print(x, ...)
## S3 method for class 'purtest':
summary(object, ...)
## S3 method for class 'summary.purtest':
print(x, ...)
```

Arguments

- `object, x` — Either a `'data.frame'` or a matrix containing the time series, a `'pseries'` object, a formula, or the name of a column of a `'data.frame'`, or a `'pdata.frame'` on which the test has to be computed; a`'purtest'` object for the print and summary methods,
- `data` — a `'data.frame'` or a `'pdata.frame'` object,
- `index` — the indexes,
- `test` — the test to be computed: one of `levinlin` for Levin, Lin and Chu (2002), `ips` for Im, Pesaran and Shin (2003), `madwu` for Maddala and Wu (1999), and `hadri` for Hadri (2000),
- `exo` — the exogenous variables to introduce in the augmented Dickey-Fuller regressions: this can be nothing (`'none'`), individual intercepts (`'intercept'`) or individual intercepts and trends (`'trend'`),
- `lags` — the number of lags to be used for the augmented Dickey-Fuller regressions: either an integer (the number of lags for all time series), a vector of integers (one for each time series), or a character string for an automatic computation of the number of lags, based on either the AIC (`'AIC'`), the SIC (`'SIC'`) or on Hall's method (`'Hall'`),
- `pmax` — maximum number of lags,
- `Hcons` — a boolean indicating whether the heteroscedasticity-consistent test of Hadri should be computed,
- `q` — the bandwidth for the estimation of the long-run variance,
- `dfcor` — should the standard deviation of the regressions be computed using a degrees-of-freedom correction?
- `fixedT` — should the different ADF regressions be computed using the same number of observations?,
- `...` — further arguments.

Details

All these tests except `'hadri'` are based on the estimation of augmented Dickey-Fuller regressions for each time series. A statistic is then computed using the t-statistic associated with the lagged variable. The kind of test to be computed can be specified in several ways:

A `formula`/`data` interface (if `data` is a `data.frame`, an additional `index` argument should be specified); the formula should be of the form: `y~0`, `y~1` or `y~trend` for a test with no exogenous variable, with an intercept or with a time trend, respectively.

A `data.frame`, a `matrix`, a `pseries`: in this case, the exogenous variables are specified using the `exo` argument.

Value

An object of class `'purtest'`: a list with the elements `'statistic'` (a `'htest'` object), `'call'`, `'args'`, `'idres'` (containing results from the individual regressions), and `'adjval'` (containing the simulated means and variances needed to compute the statistics).

Author(s)

Yves Croissant

References

Hadri K. (2000). “Testing for Unit Roots in Heterogeneous Panel Data”, *The Econometrics Journal*, 3, pp. 148–161.

Im K.S., Pesaran M.H. and Shin Y. (2003). “Testing for Unit Roots in Heterogeneous Panels”, *Journal of Econometrics*, 115(1), pp. 53–74.

Levin A., Lin C.F. and Chu C.S.J. (2002). “Unit Root Test in Panel Data: Asymptotic and Finite Sample Properties”, *Journal of Econometrics*, 108, pp. 1–24.

Maddala G.S. and Wu S. (1999). “A Comparative Study of Unit Root Tests with Panel Data and a New Simple Test”, *Oxford Bulletin of Economics and Statistics*, 61, Supplement 1, pp. 631–652.

Examples

```
data("Grunfeld", package = "plm")
y <- data.frame(split(Grunfeld$inv, Grunfeld$firm))
purtest(y, pmax = 4, type = "intercept", test = "madwu")
```

`pvar` *Check for Cross-Sectional and Time Variation*

Description

This function checks for each variable of a panel if it varies cross-sectionally and over time.

Usage

```
pvar(x, ...)
## S3 method for class 'data.frame':
pvar(x, index = NULL, ...)
## S3 method for class 'pdata.frame':
pvar(x, ...)
## S3 method for class 'pvar':
print(x, ...)
```

Arguments

`x`	a `data.frame`,
`index`	see `plm.data`,
`...`	further arguments.

Details

`pvar` can be time consuming for “big” panels.

Value

An object of class `pvar` containing the following elements:

`id.var` a logical vector with `TRUE` values if the variable has individual variation, `FALSE` otherwise,

`time.var` a logical vector with `TRUE` values if the variable has time variation, `FALSE` otherwise,

Author(s)

Yves Croissant

Examples

```
#There are 595 individuals

data("Wages",package="plm")
pvar(Wages, 595)

# Gasoline contains two variables which are individual and time indexes
# and are the first two variables
data("Gasoline", package = "plm")
pvar(Gasoline)

# Hedonic is an unbalanced panel, townid is the individual index; # the drop.index argument
data("Hedonic", package = "plm")
pvar(Hedonic,"townid", drop.index = TRUE)

# same using pdata.frame
Hed <- pdata.frame(Hedonic, "townid", drop.index = TRUE)
pvar(Hed)
```

---

pvcm *Variable Coefficients Models for Panel Data*

Description

Estimators for random and fixed effects models with variable coefficients.

Usage

```
pvcm(formula, data, subset, na.action, effect = c("individual","time"),
     model = c("within","random"), index = NULL, ...)
## S3 method for class 'pvcm':
summary(object, ...)
## S3 method for class 'summary.pvcm':
print(x, digits = max(3, getOption("digits") -2),
    width = getOption("width"), ...)
```

Arguments

`formula`	a symbolic description for the model to be estimated,
`object, x`	an object of class `"pvcm"`,
`data`	a `data.frame`,
`subset`	see `lm`,
`na.action`	see `lm`,
`effect`	the effects introduced in the model: one of `"individual"`, `"time"`,
`model`	one of `"within"`, `"random"`,
`index`	the indexes, see `plm.data`,
`digits`	digits,
`width`	the maximum length of the lines in the print output,
`...`	further arguments.

Details

`pvcm` estimates variable coefficients models. Time or individual effects are introduced, respectively, if `effect="time"` or `effect="individual"` (the default value).

Coefficients are assumed to be fixed if `model="within"` and random if `model="random"`. In the first case, a different model is estimated for each individual (or time period). In the second case, the Swamy (1970) model is estimated. It is a generalized least squares model which uses the results of the previous model.

Value

An object of class `c("pvcm","panelmodel")`, which has the following elements :

`coefficients`	the vector (or the list for fixed effects) of coefficients,
`residuals`	the vector of residuals,
`fitted.values`	the vector of fitted.values,
`vcov`	the covariance matrix of the coefficients,
`df.residual`	degrees of freedom of the residuals,
`model`	a `data.frame` containing the variables used for the estimation,
`call`	the call,
`Delta`	the estimation of the covariance matrix of the coefficients (random effect models only),
`std.error`	the standard errors for all the coefficients for each individual (within models only),

`pvcm` objects have `print`, `summary` and `print.summary` methods.

Author(s)

Yves Croissant

References

Swamy, P.A.V.B. (1970). Efficient Inference in a Random Coefficient Regression Model, *Econometrica*, **38**(2), pp.311–323.

Examples

```
data("Produc", package = "plm")
zw <- pvcm(log(gsp) ~ log(pcap) + log(pc) + log(emp) + unemp, data = Produc, model = "withi
zr <- pvcm(log(gsp) ~ log(pcap) + log(pc) + log(emp) + unemp, data = Produc, model = "rando
```

---

pwartest *Wooldridge Test for AR(1) Errors in FE Panel Models*

Description

Test of serial correlation for (the idiosyncratic component of) the errors in fixed-effects panel models.

Usage

```
pwartest(x,...)
## S3 method for class 'panelmodel':
pwartest(x, ...)
## S3 method for class 'formula':
pwartest(x, data, ...)
```

Arguments

`x`	an object of class `formula` or of class `panelmodel`,
`data`	a `data.frame`,
`...`	further arguments to be passed on to `linear.hypothesis` or to `vcovHC`.

Details

As Wooldridge (2003, Sec. 10.5.4) observes, under the null of no serial correlation in the errors, the residuals of a FE model must be negatively serially correlated, with $cor(\hat{u}_{it}, \hat{u}_{is}) = -1/(T-1)$ for each t, s. He suggests basing a test for this null hypothesis on a pooled regression of FE residuals on their first lag: $\hat{u}_{i,t} = \alpha + \delta \hat{u}_{i,t-1} + \eta_{i,t}$. Rejecting the restriction $\delta = -1/(T-1)$ makes us conclude against the original null of no serial correlation.

`pwartest` estimates the `within` model and retrieves residuals, then estimates an AR(1) `pooling` model on them. The test statistic is obtained by applying `linear.hypothesis()` to the latter model to test the above restriction on δ, setting the covariance matrix to `vcovHC` with the option `method="arellano"` to control for serial correlation.

Unlike the `pbgtest` and `pdwtest`, this test does not rely on large-T asymptotics and has therefore good properties in “short” panels. Furthermore, it is robust to general heteroskedasticity.

Value

An object of class "htest".

Author(s)

Giovanni Millo

References

Wooldridge, J.M. (2002) *Econometric Analysis of Cross-Section and Panel Data*, MIT Press, Sec. 10.5.4, page 274.

See Also

pwfdtest, pdwtest, pbgtest, pbltest, pbsytest.

Examples

```
data("EmplUK", package = "plm")
pwartest(log(emp) ~ log(wage) + log(capital), data = EmplUK)
```

---

pwfdtest — *Wooldridge first-difference-based test for AR(1) errors in levels or first-differenced panel models*

Description

First-differencing-based test of serial correlation for (the idiosyncratic component of) the errors in either levels or first-differenced panel models.

Usage

```
pwfdtest(x,...)
## S3 method for class 'panelmodel':
pwfdtest(x, ..., h0 = c("fd","fe"))
## S3 method for class 'formula':
pwfdtest(x, data, ..., h0 = c("fd","fe"))
```

Arguments

x	an object of class formula,
data	a data.frame,
h0	the null hypothesis: one of "fd","fe"
...	further arguments to be passed on to linear.hypothesis or to vcovHC

Details

As Wooldridge (2003, 10.6.3) observes, if the idiosyncratic errors in the model in levels are uncorrelated (which we label hypothesis `"fe"`), then the errors of the model in first differences (FD) must be serially correlated with $cor(\hat{e}_{it}, \hat{e}_{is}) = -0.5$ for each t, s. If on the contrary the levels model's errors are a random walk, then there must be no serial correlation in the FD errors (hypothesis `"fd"`). Both the fixed effects (FE) and the first-differenced (FD) estimators remain consistent under either assumption, but the relative efficiency changes: FE is more efficient under `"fe"`, FD under `"fd"`.

Wooldridge (ibid.) suggests basing a test for either hypothesis on a pooled regression of FD residuals on their first lag: $\hat{e}_{i,t} = \alpha + \delta \hat{e}_{i,t-1} + \eta_{i,t}$. Rejecting the restriction $\delta = -0.5$ makes us conclude against the null of no serial correlation in errors of the levels equation (`"fe"`). The null hypothesis of no serial correlation in differenced errors (`"fd"`) is tested in a similar way, but based on the zero restriction on δ. Rejecting `"fe"` favours the use of the first-differences estimator and the contrary, although it is possible that both be rejected.

`pwartest` estimates the `fd` model and retrieves residuals, then estimates an AR(1) `pooling` model on them. The test statistic is obtained by applying `linear.hypothesis()` to the latter model to test the relevant restriction on δ, setting the covariance matrix to `vcovHC` with the option `method="arellano"` to control for serial correlation.

Unlike the `pbgtest` and `pdwtest`, this test does not rely on large-T asymptotics and has therefore good properties in "short" panels. Furthermore, it is robust to general heteroskedasticity. The `"fe"` version can be used to test for error autocorrelation regardless of whether the maintained specification has fixed or random effects (see Drukker, 2003).

Value

An object of class `"htest"`.

Author(s)

Giovanni Millo

References

Drukker, D.M. (2003) Testing for serial correlation in linear panel-data models, The Stata Journal, 3(2), pp. 168–177.

Wooldridge, J.M. (2003) Econometric analysis of cross-section and panel data, MIT Press, 10.6.3, page 282.

See Also

`pdwtest`, `pbgtest`, `pwartest`

Examples

```
data(EmplUK)
pwfdtest(log(emp) ~ log(wage) + log(capital), data = EmplUK)
pwfdtest(log(emp) ~ log(wage) + log(capital), data = EmplUK, h0 = "fe")
```

pwtest *Wooldridge's Test for Unobserved Effects in Panel Models*

Description

Semi-parametric test for the presence of (individual or time) unobserved effects in panel models.

Usage

```
pwtest(x,...)
## S3 method for class 'panelmodel':
pwtest(x, ...)
## S3 method for class 'formula':
pwtest(x, data, ...)
```

Arguments

`x`	an object of class `"formula"`,
`data`	a `data.frame`,
`...`	further arguments passed to `plm`.

Details

This semi-parametric test checks the null hypothesis of zero correlation between errors of the same group. Therefore, it has power both against individual effects and, more generally, any kind of serial correlation.

The test relies on large-N asymptotics. It is valid under error heteroskedasticity and departures from normality.

The above is valid if `effect="individual"`, which is the most likely usage. If `effect="time"`, symmetrically, the test relies on large-T asymptotics and has power against time effects and, more generally, against cross-sectional correlation.

Value

An object of class `"htest"`.

Author(s)

Giovanni Millo

References

Wooldridge, J.M. (2002) *Econometric Analysis of Cross-Section and Panel Data*, MIT Press, Sec. 10.4.4., page 264.

See Also

pbltest, pbgtest, pdwtest, pbsytest, pwartest, pwfdtest for tests for serial correlation in panel models. plmtest for tests for random effects.

Examples

```
data("Produc", package = "plm")
pwtest(log(gsp) ~ log(pcap) + log(pc) + log(emp) + unemp, data = Produc)
```

---

sargan *Hansen–Sargan Test of Overidentifying Restrictions*

Description

A test of overidentifying restrictions for models estimated by GMM.

Usage

```
sargan(object)
```

Arguments

object	an object of class "pgmm".

Details

The Hansen–Sargan test calculates the quadratic form of the moment restrictions that is minimized while computing the GMM estimator. It follows asymptotically a chi-square distribution with number of degrees of freedom equal to the difference between the number of moment conditions and the number of coefficients.

Value

An object of class "htest".

Author(s)

Yves Croissant

References

Hansen, L.P. (1982), Large Sample Properties of Generalized Methods of Moments Estimators, *Econometrica*, 50, 1029–54.

Sargan, J.D. (1958), The Estimation of Economic Relationships using Instrumental Variables, *Econometrica*, 26, pp.393–415.

See Also

pgmm

Examples

```
data("EmplUK", package = "plm")
ar <- pgmm(dynformula(log(emp) ~ log(wage) + log(capital) +
            log(output), list(2,1,2,2)), data = EmplUK, effect = "twoways", model = "twostep
             lag.gmm = list(c(2,99)))
sargan(ar)
```

Snmesp *Employment and Wages in Spain*

Description

A panel of 738 observations from 1983 to 1990

total number of observations: 5904

observation: firms

country: Spain

Usage

```
data(Snmesp)
```

Format

A data frame containing:

firm firm index

year year

n log of employment

w log of wages

y log of real output

i log of intermediate inputs

k log of real capital stock

f real cash flow

Source

Journal of Business Economics and Statistics data archive:

http://www.amstat.org/publications/jbes/.

References

Alonso–Borrego, C. and Arellano, M. (1999). Symmetrically Normalized Instrumental-Variable Estimation Using Panel Data, *Journal of Business and Economic Statistics*, **17(1)**, 36–49.

SumHes *The Penn World Table, v. 5*

Description

A panel of 125 observations from 1960 to 1985

total number of observations : 3250

observation : country

country : World

Usage

```
data(SumHes)
```

Format

A data frame containing :

year the year

country the country name (factor)

opec OPEC member?

com communist regime?

pop country's population (in thousands)

gdp real GDP per capita (in 1985 US dollars)

sr saving rate (in percent)

Source

Online supplements to Hayashi (2000).

http://fhayashi.fc2web.com/datasets.htm

References

Hayashi, F. (2000) *Econometrics*, Princeton University Press, , chapter 5, 358-363.

Summers, R. and A. Heston (1991) "The Penn World Table (mark 5): An expanded set of international comparisons, 1950–1988", *Quarterly Journal of Economics*, **29**, 229–256.

vcovBK *Beck and Katz Robust Covariance Matrix Estimators*

Description

Unconditional Robust covariance matrix estimators *a la Beck and Katz* for panel models.

Usage

```
## S3 method for class 'plm':
vcovBK(x, type = c("HC0", "HC1", "HC2", "HC3", "HC4"),
                                cluster = c("group","time"),
                                diagonal = FALSE,
                                ...)
```

Arguments

x	an object of class "plm"
type	one of "HC0","HC1","HC2","HC3","HC4",
cluster	one of "group","time"
diagonal	a logical value specifying whether to force nondiagonal elements to zero
...	further arguments.

Details

vcovBK is a function for estimating a robust covariance matrix of parameters for a panel model according to the Beck and Katz (1995) method, a.k.a. Panel Corrected Standard Errors (PCSE), which uses an unconditional estimate of the error covariance across time periods (groups) inside the standard formula for coefficient covariance. Observations may be clustered either by "group" to account for timewise heteroskedasticity and serial correlation or by "time" to account for cross-sectional heteroskedasticity and correlation. It must be borne in mind that the Beck and Katz formula is based on N- (T-) asymptotics and will not be appropriate elsewhere.

The diagonal logical argument can be used, if set to TRUE, to force to zero all nondiagonal elements in the estimated error covariances; this is appropriate if both serial and cross-sectional correlation are assumed out, and yields a timewise- (groupwise-) heteroskedasticity-consistent estimator.

Weighting schemes are analogous to those in vcovHC in package sandwich and are justified theoretically (although in the context of the standard linear model) by MacKinnon and White (1985) and Cribari-Neto (2004) (see Zeileis, 2004).

The main use of vcovBK is to be an argument to other functions, e.g. for Wald-type testing: as vcov to coeftest(), waldtest() and other methods in the lmtest package; and as vcov to linear.hypothesis() in the car package (see the examples). Notice that the vcov argument may be supplied a function (which is the safest) or a matrix (see Zeileis (2004), 4.1-2 and examples below).

Value

An object of class "matrix" containing the estimate of the covariance matrix of coefficients.

Author(s)

Giovanni Millo

References

Beck, N. and Katz, J. (1995) What to do (and not to do) with time-series-cross-section data in comparative politics. *American Political Science Review*, **89(3)**, 634–647.

Greene, W. H. (2003) *Econometric Analysis*, 5th ed. Macmillan Publishing Company, New York, 323.

Examples

```
library(lmtest)
library(car)
data("Produc", package="Ecdat")
zz <- plm(log(gsp)~log(pcap)+log(pc)+log(emp)+unemp, data=Produc, model="random")
## standard coefficient significance test
coeftest(zz)
## robust significance test, cluster by group
## (robust vs. serial correlation), default arguments
coeftest(zz, vcov=vcovBK)
## idem with parameters, pass vcov as a function argument
coeftest(zz, vcov=function(x) vcovBK(x, type="HC1"))
## idem, cluster by time period
## (robust vs. cross-sectional correlation)
coeftest(zz, vcov=function(x) vcovBK(x,
 type="HC1", cluster="time"))
## idem with parameters, pass vcov as a matrix argument
coeftest(zz, vcov=vcovBK(zz, type="HC1"))
## joint restriction test
waldtest(zz, update(zz, .~.-log(emp)-unemp), vcov=vcovBK)
## test of hyp.: 2*log(pc)=log(emp)
linear.hypothesis(zz, "2*log(pc)=log(emp)", vcov=vcovBK)
```

---

vcovHC *Robust Covariance Matrix Estimators*

Description

Robust covariance matrix estimators *a la White* for panel models.

Usage

```
## S3 method for class 'plm':
vcovHC(x,
                              method = c("arellano", "white1", "white2"),
                              type = c("HC0", "HC1", "HC2", "HC3", "HC4"),
                              cluster = c("group","time"),
                              ...)
## S3 method for class 'pgmm':
vcovHC(x,...)
```

Arguments

`x`	an object of class `"plm"` which should be the result of a random effect or a within model or a model of class `"pgmm"`,
`method`	one of `"arellano"`,`"white1"`,`"white2"`,
`type`	one of `"HC0"`,`"HC1"`,`"HC2"`,`"HC3"`,`"HC4"`,
`cluster`	one of `"group"`,`"time"`
`...`	further arguments.

Details

`vcovHC` is a function for estimating a robust covariance matrix of parameters for a fixed effects or random effects panel model according to the White method (White 1980, 1984; Arellano 1987). Observations may be clustered by `"group"` (`"time"`) to account for serial (cross-sectional) correlation.

All types assume no intragroup (serial) correlation between errors and allow for heteroskedasticity across groups (time periods). As for the error covariance matrix of every single group of observations, `"white1"` allows for general heteroskedasticity but no serial (cross-sectional) correlation; `"white2"` is `"white1"` restricted to a common variance inside every group (time period) (see Greene (2003), 13.7.1-2 and Wooldridge (2003), 10.7.2); `"arellano"` (see ibid. and the original ref. Arellano (1987)) allows a fully general structure w.r.t. heteroskedasticity and serial (cross-sectional) correlation.

Weighting schemes are analogous to those in `vcovHC` in package `sandwich` and are justified theoretically (although in the context of the standard linear model) by MacKinnon and White (1985) and Cribari-Neto (2004) (see Zeileis, 2004).

The main use of `vcovHC` is to be an argument to other functions, e.g. for Wald-type testing: as `vcov` to `coeftest()`, `waldtest()` and other methods in the `lmtest` package; and as `vcov` to `linear.hypothesis()` in the `car` package (see the examples). Notice that the `vcov` argument may be supplied a function (which is the safest) or a matrix (see Zeileis (2004), 4.1-2 and examples below).

A special procedure, proposed by Windmeijer (2005) for `pgmm` objects is provided.

Value

An object of class `"matrix"` containing the estimate of the asymptotic covariance matrix of coefficients.

Author(s)

Giovanni Millo \& Yves Croissant

References

Arellano, M. (1987) Computing robust standard errors for within group estimators, *Oxford bulletin of Economics and Statistics*, **49**, 431–434.

Cribari-Neto, F. (2004) Asymptotic inference under heteroskedasticity of unknown form. *Computational Statistics \& Data Analysis* **45**, 215–233.

Greene, W. H. (1993) *Econometric Analysis*, 2nd ed. Macmillan Publishing Company, New York.

MacKinnon, J. G. and White H. (1985) Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties. *Journal of Econometrics* **29**, 305–325.

Weidmeijer, F. (2005) A finite sample correction for the variance of linear efficicent two–step GMM estimators, *Journal of Econometrics*, **126**, pp.25–51.

White H. (1980) *Asymptotic Theory for Econometricians*, Ch. 6, Academic Press, Orlando (FL).

White H. (1984) A heteroskedasticity-consistent covariance matrix and a direct test for heteroskedasticity. *Econometrica* **48**, 817–838.

Wooldridge J. M. (2003) *Econometric Analysis of Cross Section and Panel Data*, MIT Press

Zeileis A. (2004) Econometric Computing with HC and HAC Covariance Matrix Estimators. *Journal of Statistical Software*, **11**(10), 1–17. URL http://http://www.jstatsoft.org/v11/i10/.

Examples

```
library(lmtest)
library(car)
data("Produc", package="Ecdat")
zz <- plm(log(gsp)~log(pcap)+log(pc)+log(emp)+unemp, data=Produc, model="random")
## standard coefficient significance test
coeftest(zz)
## robust significance test, cluster by group
## (robust vs. serial correlation)
coeftest(zz, vcov=vcovHC)
## idem with parameters, pass vcov as a function argument
coeftest(zz, vcov=function(x) vcovHC(x, method="arellano", type="HC1"))
## idem, cluster by time period
## (robust vs. cross-sectional correlation)
coeftest(zz, vcov=function(x) vcovHC(x, method="arellano",
 type="HC1", cluster="group"))
## idem with parameters, pass vcov as a matrix argument
coeftest(zz, vcov=vcovHC(zz, method="arellano", type="HC1"))
## joint restriction test
waldtest(zz, update(zz, .~.-log(emp)-unemp), vcov=vcovHC)
## test of hyp.: 2*log(pc)=log(emp)
linear.hypothesis(zz, "2*log(pc)=log(emp)", vcov=vcovHC)

## Robust inference for GMM models
data("EmplUK", package="plm")
```

```
ar <- pgmm(dynformula(log(emp)~log(wage)+log(capital)+log(output),list(2,1,2,2)),
    data=EmplUK, effect="twoways", model="twosteps",
    gmm.inst=~log(emp), lag.gmm=list(c(2,99)))
rv <- vcovHC(ar)
mtest(ar, order=2, vcov=rv)
```

---

vcovSCC *Driscoll and Kraay (1998) Robust Covariance Matrix Estimator*

Description

Nonparametric robust covariance matrix estimators *a la Driscoll and Kraay* for panel models with cross-sectional *and* serial correlation.

Usage

```
## S3 method for class 'plm':
vcovSCC(x, type = c("HC0", "HC1", "HC2", "HC3", "HC4"),
                                 maxlag=NULL, ...)
```

Arguments

x	an object of class "plm"
type	one of "HC0","HC1","HC2","HC3","HC4",
maxlag	either NULL or a positive integer specifying the maximum lag order before truncation
...	further arguments.

Details

vcovSCC is a function for estimating a robust covariance matrix of parameters for a panel model according to the Driscoll and Kraay (1998) method, which is consistent with cross-sectional and serial correlation in a T-asymptotic setting and irrespective of the N dimension. The use with random effects models is undocumented.

Weighting schemes are analogous to those in vcovHC in package sandwich and are justified theoretically (although in the context of the standard linear model) by MacKinnon and White (1985) and Cribari-Neto (2004) (see Zeileis, 2004).

The main use of vcovSCC is to be an argument to other functions, e.g. for Wald-type testing: as vcov to coeftest(), waldtest() and other methods in the lmtest package; and as vcov to linear.hypothesis() in the car package (see the examples). Notice that the vcov argument may be supplied a function (which is the safest) or a matrix (see Zeileis (2004), 4.1-2 and examples below).

Value

An object of class "matrix" containing the estimate of the covariance matrix of coefficients.

Author(s)

Giovanni Millo, partially ported from Daniel Hoechle's Stata code

References

Driscoll, J.C. and Kraay, A.C. (1998) Consistent Covariance Matrix Estimation with Spatially Dependent Panel Data. *Review of Economics and Statistics* **80**, 549–560.

Hoechle, D. (2007) Robust standard errors for panel regressions with cross-sectional dependence. *Stata Journal*, **7(3)**, 281–312.

Examples

```
library(lmtest)
library(car)
data("Produc", package="Ecdat")
zz <- plm(log(gsp)~log(pcap)+log(pc)+log(emp)+unemp, data=Produc, model="pooling")
## standard coefficient significance test
coeftest(zz)
## SCC robust significance test, default
coeftest(zz, vcov=vcovSCC)
## idem with parameters, pass vcov as a function argument
coeftest(zz, vcov=function(x) vcovSCC(x, type="HC1", maxlag=4))
## joint restriction test
waldtest(zz, update(zz, .~.-log(emp)-unemp), vcov=vcovSCC)
## test of hyp.: 2*log(pc)=log(emp)
linear.hypothesis(zz, "2*log(pc)=log(emp)", vcov=vcovSCC)
```

Wages *Panel Data of Individual Wages*

Description

A panel of 595 observations from 1976 to 1982, taken from the Panel Study of Income Dynamics (PSID).

total number of observations : 4165

observation : individuals

country : United States

Usage

```
data(Wages)
```

Format

A data frame containing:

exp years of full-time work experience.

wks weeks worked.

bluecol blue collar?

ind works in a manufacturing industry?

south resides in the south?

smsa resides in a standard metropolitan statistical area?

married married?

sex a factor with levels `"male", "female"`

union individual's wage set by a union contract?

ed years of education.

black is the individual black?

lwage logarithm of wage.

Source

Online complements to Baltagi (2001).

http://www.wiley.com/legacy/wileychi/baltagi/

References

Baltagi, Badi H. (2003) *Econometric Analysis of Panel Data*, 2nd ed., John Wiley and Sons.

Cornwell, C. and P. Rupert (1988) "Efficient estimation with panel data: an empirical comparison of instrumental variables estimators", *Journal of Applied Econometrics*, **3**, 149–155.

Index