Package 'getspanel'

May 8, 2026

Title General-to-Specific Modelling of Panel Data

Version 0.2.1

Date 2025-05-12

Description Uses several types of indicator saturation and automated General-to-Specific (GETS) modelling from the 'gets' package and applies it to panel data. This allows the detection of structural breaks in panel data, operationalising a reverse causal approach of causal inference, see Pretis and Schwarz (2022) <doi:10.2139/ssrn.4022745>.

License MIT + file LICENSE

Encoding UTF-8

URL https://github.com/moritzpschwarz/getspanel, https://www.moritzschwarz.org/getspanel/

BugReports https://github.com/moritzpschwarz/getspanel/issues

LazyData true

RoxygenNote 7.3.2

Suggests testthat, knitr, rmarkdown, lfe, prettydoc, plm, fixest, lmtest, sandwich, cowplot

Imports gets, fastDummies, Matrix, ggplot2, stats, mvtnorm

Depends R (>= 3.5.0)

VignetteBuilder knitr, rmarkdown

Config/testthat/edition 3

NeedsCompilation no

Author Felix Pretis [aut], Moritz Schwarz [aut, cre] (ORCID: <https://orcid.org/0000-0003-0340-3780>)

Maintainer Moritz Schwarz <moritz.schwarz@scmo.eu>

Repository CRAN

Date/Publication 2025-05-13 08:00:02 UTC

Contents

`break_uncertainty` *Estimate Breakdate Uncertainty*

Description

Estimate Breakdate Uncertainty

Usage

```
break_uncertainty(x, m = 15, interval = 0.99)
```

Arguments

`x`	An object produced by the isatpanel function
`m`	Maximum range of interval (default is 15 time periods).
`interval`	Approximate level of interval. CI level will be at least > interval. Default 0.99 is a 99% CI, so the time interval will always be the integer that results in at least > 99% coverage.

Value

A data.frame that indicates the uncertainty for each FESIS break. The time interval is given by the estimated date in the 'time' column with a confidence interval of +/- the interval in the tci column.

Examples

```
data(EU_emissions_road)

# Group specification
EU15 <- c("Austria", "Germany", "Denmark", "Spain", "Finland", "Belgium",
          "France", "United Kingdom", "Ireland", "Italy", "Luxembourg",
          "Netherlands", "Greece", "Portugal", "Sweden")

# Prepare sample and data
EU_emissions_road_short <- EU_emissions_road[
EU_emissions_road$country %in% EU15 &
EU_emissions_road$year >= 2000,
]

# Run
result <- isatpanel(
  data = EU_emissions_road_short,
  formula = ltransport.emissions ~ lgdp + I(lgdp^2) + lpop,
  index = c("country", "year"),
  effect = "twoways",
  fesis = TRUE,
  plot = FALSE,
  t.pval = 0.01
)

break_uncertainty(result)
```

---

check.time.subset.vectors

Internal function to check vectors that subset the indicator selection using the time dimension

---

Description

Internal function to check vectors that subset the indicator selection using the time dimension

Usage

```
check.time.subset.vectors(time.vector, vector.name, time, id)
```

Arguments

time.vector	A vector containing the user input in e.g. tis_time or fesis_time
vector.name	The name of argument that the user inputted this vector in. This is just to make error messages more elaborate.
time	The time dimension of isatpanel.
id	The id dimension of isatpanel.

Value

Does not return any value but will throw error if something is not correct.

`EUCO2residential` *CO2 Data for the EU Residential Sector*

Description

CO2 Data for the EU Residential Sector

Usage

```
EUCO2residential
```

Format

A data frame with 1550 rows and 9 variables:

country Country

year Year

lgdp Log Gross Domestic Product

lhdd Log Heating Degree Days

lcdd Log Cooling Degree Days

urban Urban Share

av.rate EU Interest Rate

pop Population

agg.directem Aggregated Direct Emissions

Source

IEA

`EU_emissions_road` *CO2 Data for EU Road Emissions*

Description

CO2 Data for EU Road Emissions

Usage

```
EU_emissions_road
```

Format

A data frame with 1550 rows and 13 variables:

X Index

country Country

year Year

gdp Gross Domestic Product

pop Population

transport.emissions Transport CO2 Emissions

lgdp Log GDP

lpop Log Population

ltransport.emissions Log Transport CO2 Emissions

const Constant

L1.ltransport.emissions Lag 1 Log Transport CO2 Emissions

L1.lgdp Lag 1 Log GDP

L1.lpop Lag 1 Log Population

Source

EDGAR

felmFun *Internal lfe/felm Estimation Method*

Description

Internal lfe/felm Estimation Method

Usage

```
felmFun(y, x, effect, time, id, cluster = "individual", ...)
```

Arguments

y	dependent variable
x	matrix of regressors
effect	Fixed Effect specification
time	Character vector of name of the time variable
id	Character vector of the name of the group variable
cluster	Character vector of the variable(s) to cluster Standard Errors at
...	Further arguments to pass to gets::isat

Value

List to be used by gets::isat

fixestFun *Internal fixest/feols Estimation Method*

Description

Internal fixest/feols Estimation Method

Usage

```
fixestFun(y, x, effect, time, id, cluster = "individual", ...)
```

Arguments

y	dependent variable
x	matrix of regressors
effect	Fixed Effect specification
time	Character vector of name of the time variable
id	Character vector of the name of the group variable
cluster	Character vector of the variable(s) to cluster Standard Errors at
...	Further arguments to pass to gets::isat

Value

List to be used by gets::isat

`get_indicators` *Extract the retained indicators from an* `isatpanel` *object*

Description

Extract the retained indicators from an `isatpanel` object

Usage

```
get_indicators(object, uis_breaks = NULL)
```

Arguments

`object`	An object produced by the isatpanel function.
`uis_breaks`	A string with the names of user-specified indicators.

Value

A list of indicators.

Examples

```
data(EU_emissions_road)

# Group specification
EU15 <- c("Austria", "Germany", "Denmark", "Spain", "Finland", "Belgium",
          "France", "United Kingdom", "Ireland", "Italy", "Luxembourg",
          "Netherlands", "Greece", "Portugal", "Sweden")

# Prepare sample and data
EU_emissions_road_short <- EU_emissions_road[
EU_emissions_road$country %in% EU15 &
EU_emissions_road$year >= 2000,
]

# Run
result <- isatpanel(
  data = EU_emissions_road_short,
  formula = ltransport.emissions ~ lgdp + I(lgdp^2) + lpop,
  index = c("country", "year"),
  effect = "twoways",
  fesis = TRUE,
  plot = FALSE,
  t.pval = 0.01
)
plot(result)
```

```
plot_grid(result)

# print the retained indicators
get_indicators(result)
```

`identify_indicator_timings` *Internal function to identify the timing of selected indicators*

Description

Internal function to identify the timing of selected indicators

Usage

```
identify_indicator_timings(object, uis_breaks = NULL, isat_object = NULL)
```

Arguments

`object`	data.frame
`uis_breaks`	A character vector with the names of the UIS breaks if the `uis` argument was used in isatpanel.
`isat_object`	The object of class `isat` produced by isatpanel.

Value

A list of data.frames

`isatpanel` *Indicator Saturation for Panel Data*

Description

This function is essentially a wrapper function around the `gets::isat()` function from the `gets` package. This function allows the running of various different indicator saturation techniques that can, for example, be used to answer reverse causal questions. Indicator Saturation techniques fully saturate a model with indicators (for example dummy-indicators or step-indicators) and then use an automated block-search algorithm to retain only relevant indicators that improve the model (based on a chosen information criterion).

Usage

```
isatpanel(
  data = NULL,
  formula = NULL,
  index = NULL,
  effect = c("twoways"),
  na.remove = TRUE,
  engine = NULL,
  user.estimator = NULL,
  cluster = "none",
  ar = 0,
  iis = FALSE,
  jiis = FALSE,
  jsis = FALSE,
  fesis = FALSE,
  tis = FALSE,
  csis = FALSE,
  cfesis = FALSE,
  fesis_id = NULL,
  fesis_time = NULL,
  tis_id = NULL,
  tis_time = NULL,
  csis_var = NULL,
  csis_time = NULL,
  cfesis_var = NULL,
  cfesis_id = NULL,
  cfesis_time = NULL,
  uis = NULL,
  t.pval = 0.001,
  plot = TRUE,
  print.searchinfo = TRUE,
  plm_model = "within",
  y = NULL,
  id = NULL,
  time = NULL,
  mxreg = NULL,
  ...
)
```

Arguments

data	The input data.frame object.
formula	Formula argument. The dependent variable will be the left-most element, separated by a ~ symbol from the remaining regressors (e.g. y ~ x + z). Note the intercept will always be removed unless the effect is "none" - this means that if any fixed effects are specified, the intercept will always be removed.
index	Specify the name of the group and time column in the format c("id", "time").

effect	Fixed Effect specification. Possible arguments: "twoways" (Default), "individual", "time", or "none".
na.remove	remove NAs
engine	Estimation function to use. Default is NULL, which uses the default estimation procedure of the gets package. Alternatives are "fixest", "plm", or "felm".
user.estimator	Use a user.estimator
cluster	cluster Standard Errors at this level. Default is "none". Possible values are: "individual", "time", or "twoways".
ar	Autoregressive Term to be included. default is 0.
iis	Logical. Use Impulse Indicator Saturation.
jiis	Logical. Use Joint Impulse Indicator Saturation (Outliers are common across all units). This is essentially just a time fixed effect, but this allows selection of FE.
jsis	Logical. Use Join Step Indicator Saturation (steps are common across all units). Will only be retained if time fixed effects are not included (i.e. effect = 'none' or 'individual'), as they are collinear otherwise.
fesis	Logical. Use Fixed Effect Step Indicator Saturation. Constructed by multiplying a constant (1) with group Fixed Effects. Default is FALSE.
tis	Logical. Use Trend Indicator Saturation. Constructed by fitting a trend for each unit from every observation. Default is FALSE.
csis	Logical. Use Coefficient Step Indicator Saturation. Constructed by Default is FALSE.
cfesis	Logical. Use Coefficient-Fixed Effect Indicator Saturation. Default is FALSE.
fesis_id	The FESIS method can be conducted for all (default) individuals/units (i.e. looking for breaks in individual countries) or just a subset of them. If you want to use a subset, specify the individuals/units for which you want to test the stability of the fixed effect in a character vector.
fesis_time	The FESIS method can be conducted for all (default) time periods (i.e. looking for Fixed Effect Step-shifts at every time period) or just a subset of them. If you want to use a subset, specify the time periods as a numeric vector (for all id's the same like 1:10) or as a list with an equal number of elements as there are id's e.g. list(A = 1:10, B = NULL, C = 5:10).
tis_id	The TIS method can be conducted for all (default) individuals/units (i.e. looking for trends in individual countries) or just a subset of them. If you want to use a subset, specify the individuals/units for which you want to test the trend in a character vector.
tis_time	The TIS method can be conducted for all (default) time periods (i.e. looking for trends at every time period) or just a subset of them. If you want to use a subset, specify the time periods as a numeric vector (for all id's the same like 1:10) or as a list with an equal number of elements as there are id's e.g. list(A = 1:10, B = NULL, C = 5:10).
csis_var	The CSIS method can be conducted for all (default) variables or just a subset of them. If you want to use a subset, please specify the column names of the variable in a character vector.

csis_time	The CSIS method can be conducted for all (default) time periods (i.e. looking for Coefficient Step Shifts across all units at every time period) or just a subset of them. If you want to use a subset, specify the time periods as a numeric vector (e.g. 1:10).'
cfesis_var	The CFESIS method can be conducted for all variables (default) or just a subset of them. If you want to use a subset, please specify the column names of the variable in a character vector.
cfesis_id	The CFESIS method can be conducted for all individuals/units (default) or just a subset of them. If you want to use a subset, please specify the individuals/units to be tested in a character vector.
cfesis_time	The CFESIS method can be conducted for all (default) time periods (i.e. looking for Coefficient Step Shifts per unit at every time period) or just a subset of them. If you want to use a subset, specify the time periods as a numeric vector (for all id's the same like 1:10) or as a list with an equal number of elements as there are id's e.g. list(A = 1:10, B = NULL, C = 5:10).
uis	Matrix or List. This can be used to include a set of UIS (User Specified Indicators). Must be equal to the sample size (so it is recommended to use this only with datasets without NA values. Default is NULL. See the reference by Genaro Sucarrat (2020) below for an explanation of the UIS system.
t.pval	numeric value between 0 and 1. The significance level used for the two-sided regressor significance t-tests
plot	Logical. Should the final object be plotted? Default is TRUE. The output is a combination of plot() and plot_grid() using the cowplot package.
print.searchinfo	logical. If TRUE (default), then detailed information is printed.
plm_model	Type of PLM model (only if engine = "PLM")
y	Deprecated. The dependent variable. Can be used when data, index, and formula are not specified.
id	Deprecated. Can be used when data, index, and formula are not specified. Must be a vector of the grouping variable as a character or factor
time	Deprecated. Can be used when data, index, and formula are not specified. Must be a vector of the time variable as an integer or numeric.
mxreg	Deprecated.The co-variates matrix. Superseded by the formula argument.
...	Further arguments to gets::isat()

Value

A list with class 'isatpanel'.

References

Felix Pretis and Moritz Schwarz (2022). Discovering What Mattered: Answering Reverse Causal Questions by Detecting Unknown Treatment Assignment and Timing as Breaks in Panel Models. January 31, 2022. Available at SSRN: https://ssrn.com/abstract=4022745 or http://dx.doi.org/10.2139/ssrn.4022745

Genaro Sucarrat. User-Specified General-to-Specific and Indicator Saturation Methods, The R Journal (2020) 12:2, pages 388-401. Available at: https://journal.r-project.org/archive/2021/RJ-2021-024/index.html

See Also

gets::isat()

Examples

```
data(EU_emissions_road)

# Group specification
EU15 <- c("Austria", "Germany", "Denmark", "Spain", "Finland", "Belgium",
          "France", "United Kingdom", "Ireland", "Italy", "Luxembourg",
          "Netherlands", "Greece", "Portugal", "Sweden")

# Prepare sample and data
EU_emissions_road_short <- EU_emissions_road[
EU_emissions_road$country %in% EU15 &
EU_emissions_road$year >= 2000,
]

# Run
result <- isatpanel(
  data = EU_emissions_road_short,
  formula = ltransport.emissions ~ lgdp + I(lgdp^2) + lpop,
  index = c("country", "year"),
  effect = "twoways",
  fesis = TRUE,
  plot = FALSE,
  t.pval = 0.01
)
plot(result)
plot_grid(result)

# print the retained indicators
get_indicators(result)
```

---

logLik.plm *Log-Likelihood Function for a plm object*

Description

Log-Likelihood Function for a plm object

Usage

```
## S3 method for class 'plm'
logLik(object, ...)
```

Arguments

`object`	A plm object
`...`	Further Arguments

Value

The Log-Likelihood

`pandata_simulated` *Simulated Panel Data*

Description

Simulated Panel Data

Usage

```
pandata_simulated
```

Format

A data frame with 400 rows and 9 variables:

country A random country

year Year

gdp A simulated Gross Domestic Product

temp A simulated variable standing for temperature

const The constant

country_1 A dummy for country 1

country_2 A dummy for country 2

country_3 A dummy for country 3

country_4 A dummy for country 4

...

Source

https://github.com/moritzpschwarz/getspanel/

plmFun *plm Function to estimate isatpanel*

Description

plm Function to estimate isatpanel

Usage

```
plmFun(y, x, time, id, cluster, effect, model = "pooling", ...)
```

Arguments

y	Dependent Variable
x	matrix or data.frame of regressors
time	Vector of time variable
id	Vector of group variable
cluster	cluster specification
effect	effect specification
model	model specification
...	Further arguments passed to plm

Value

A list to be used by gets::isat

plot.isatpanel *Plotting an isatpanel object*

Description

Plotting an isatpanel object

Usage

```
## S3 method for class 'isatpanel'
plot(
  x,
  max.id.facet = 16,
  facet.scales = "free",
  title = NULL,
  zero_line = FALSE,
  ...
)
```

Arguments

x	An object produced by the isatpanel function
max.id.facet	The resulting plot will be faceted for each individual in the panel. Beyond a certain number, this might result in unreadable figures. Default set at 16.
facet.scales	To be passed to ggplot2::facet_wrap. Default is "free" (i.e. a separate y axis for each panel group/id). Alternatives are: "fixed", "fixed_y", and "fixed_x".
title	Plot title. Must be a character vector.
zero_line	Plot a horizontal line at y = 0. Default is FALSE.
...	Further arguments to be passed to ggplot2.

Value

A ggplot2 plot that plots an 'isatpanel' object and shows observed data, the fitted values, and all identified breaks and impulses.

---

plot_counterfactual *Plot the Counterfactual Path*

Description

Plot the Counterfactual Path

Usage

```
plot_counterfactual(
  x,
  plus_t = 5,
  facet.scales = "free",
  title = NULL,
  zero_line = FALSE,
  regex_exclude_indicators = NULL
)
```

Arguments

x	An object produced by the isatpanel function
plus_t	Number of time periods for the counterfactual to be displayed (default = 5).
facet.scales	To be passed to ggplot2::facet_wrap. Default is "free" (i.e. a separate y axis for each panel group/id). Alternatives are: "fixed", "fixed_y", and "fixed_x".
title	Plot title. Must be a character vector.
zero_line	Plot a horizontal line at y = 0. Default is FALSE.
regex_exclude_indicators	A regex character vector to exclude the inclusion of certain indicators in the plot. Default = NULL. Use with care, experimental.

Value

A ggplot2 plot that plots an 'isatpanel' object and shows the counterfactuals for each break.

Examples

```
data(EU_emissions_road)

# Group specification
EU15 <- c("Austria", "Germany", "Denmark", "Spain", "Finland", "Belgium",
          "France", "United Kingdom", "Ireland", "Italy", "Luxembourg",
          "Netherlands", "Greece", "Portugal", "Sweden")

# Prepare sample and data
EU_emissions_road_short <- EU_emissions_road[
EU_emissions_road$country %in% EU15 &
EU_emissions_road$year >= 2000,
]

# Run
result <- isatpanel(
  data = EU_emissions_road_short,
  formula = ltransport.emissions ~ lgdp + I(lgdp^2) + lpop,
  index = c("country", "year"),
  effect = "twoways",
  fesis = TRUE,
  plot = FALSE,
  t.pval = 0.01
)
plot(result)
plot_grid(result)
plot_counterfactual(result)
```

---

plot_grid *Plotting an isatpanel object*

Description

Plotting an isatpanel object

Usage

```
plot_grid(x, title = NULL, regex_exclude_indicators = NULL, ...)
```

Arguments

x	An object produced by the isatpanel function
title	Plot title. Must be a character vector.

regex_exclude_indicators
: A regex character vector to exclude the inclusion of certain indicators in the plot. Default = NULL. Use with care, experimental.

...
: Further arguments to be passed to ggplot2.

Value

A ggplot2 plot that plots an 'isatpanel' object and shows all indicators as a grid to give a good and quick overview.

Examples

```
data(EU_emissions_road)

# Group specification
EU15 <- c("Austria", "Germany", "Denmark", "Spain", "Finland", "Belgium",
          "France", "United Kingdom", "Ireland", "Italy", "Luxembourg",
          "Netherlands", "Greece", "Portugal", "Sweden")

# Prepare sample and data
EU_emissions_road_short <- EU_emissions_road[
EU_emissions_road$country %in% EU15 &
EU_emissions_road$year >= 2000,
]

# Run
result <- isatpanel(
  data = EU_emissions_road_short,
  formula = ltransport.emissions ~ lgdp + I(lgdp^2) + lpop,
  index = c("country", "year"),
  effect = "twoways",
  fesis = TRUE,
  plot = FALSE,
  t.pval = 0.01
)
plot(result)
plot_grid(result)
```

---

plot_residuals *Plot Residuals from 'isatpanel' against OLS*

Description

Plot Residuals from 'isatpanel' against OLS

Usage

```
plot_residuals(isatpanelobject)
```

Arguments

isatpanelobject
: An output from the 'isatpanel' function

Value

A ggplot2 plot that plots an 'isatpanel' object and shows the residuals over time in comparison to an OLS model.

Examples

```
data(EU_emissions_road)

# Group specification
EU15 <- c("Austria", "Germany", "Denmark", "Spain", "Finland", "Belgium",
          "France", "United Kingdom", "Ireland", "Italy", "Luxembourg",
          "Netherlands", "Greece", "Portugal", "Sweden")

# Prepare sample and data
EU_emissions_road_short <- EU_emissions_road[
EU_emissions_road$country %in% EU15 &
EU_emissions_road$year >= 2000,
]

# Run
result <- isatpanel(
  data = EU_emissions_road_short,
  formula = ltransport.emissions ~ lgdp + I(lgdp^2) + lpop,
  index = c("country", "year"),
  effect = "twoways",
  fesis = TRUE,
  plot = FALSE,
  t.pval = 0.01
)
plot(result)
plot_residuals(result)
```

---

print.isatpanel *Printing isatpanel results*

Description

Printing isatpanel results

Usage

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

Arguments

x	An isatpanel object.
...	Further arguments passed to print

Value

Print output of the 'isatpanel.result' list element of the 'isatpanel' object.

---

robust_isatpanel *Get robust Standard Errors for the isatpanel result*

Description

Get robust Standard Errors for the isatpanel result

Usage

```
robust_isatpanel(
  object,
  robust = TRUE,
  HAC = FALSE,
  lag = NULL,
  type = "HC0",
  cluster = "group"
)
```

Arguments

object	An isatpanel object
robust	Logical (TRUE or FALSE). Should the Standard Errors be robustified for Heterogeneity? This uses plm::vcovHC with the specified type (default is "HC0").
HAC	Should Heteroscedasticity and Autocorrelation Robust Standard Errors be used? This uses plm::vcovNW, which uses the Newey-West estimator.
lag	Maximum Number of Lags to be used with plm::vcovNW using the Newey-West estimator. Cannot be specified when HAC = FALSE. Default is NULL.
type	Character string. Type of Robust procedure e.g. 'HC0' for White SE or 'HC3' for Lang.
cluster	Should an object with clustered S.E. be included? Choose between 'group' or 'time' or FALSE. Uses plm::vcovHC with the cluster argument.

Value

A list with robust estimates

Examples

```
data(EU_emissions_road)

# Group specification
EU15 <- c("Austria", "Germany", "Denmark", "Spain", "Finland", "Belgium",
          "France", "United Kingdom", "Ireland", "Italy", "Luxembourg",
          "Netherlands", "Greece", "Portugal", "Sweden")

# Prepare sample and data
EU_emissions_road_short <- EU_emissions_road[
EU_emissions_road$country %in% EU15 &
EU_emissions_road$year >= 2000,
]

# Run
result <- isatpanel(
  data = EU_emissions_road_short,
  formula = ltransport.emissions ~ lgdp + I(lgdp^2) + lpop,
  index = c("country", "year"),
  effect = "twoways",
  fesis = TRUE,
  plot = FALSE,
  t.pval = 0.01
)
robust_isatpanel(result)
```

Within_plm *Use the within transformation from the plm package*

Description

Use the within transformation from the plm package

Usage

```
Within_plm(df, effect = "twoways")
```

Arguments

df	A data.frame object
effect	The fixed effect specification. Values possible: "twoways" (default), "individual", "time", "nested"

Value

A data.frame object with the transformation complete

Index