Package 'did'

June 13, 2026

Title Treatment Effects with Multiple Periods and Groups

Version 2.5.0

URL https://bcallaway11.github.io/did/,
https://github.com/bcallaway11/did/

Description The standard Difference-in-Differences (DID) setup involves two periods and two groups -- a treated group and untreated group. Many applications of DID methods involve more than two periods and have individuals that are treated at different points in time. This package contains tools for computing average treatment effect parameters in Difference in Differences setups with more than two periods and with variation in treatment timing using the methods developed in Callaway and Sant'Anna (2021) <doi:10.1016/j.jeconom.2020.12.001>. The main parameters are group-time average treatment effects which are the average treatment effect for a particular group at a particular time. These can be aggregated into a fewer number of treatment effect parameters, and the package deals with the cases where there is selective treatment timing, dynamic treatment effects, calendar time effects, or combinations of these. There are also functions for testing the Difference in Differences assumption, and plotting group-time average treatment effects.

Depends R (>= 4.1.0)

License GPL-3

Encoding UTF-8

LazyData true

Imports BMisc (>= 1.4.4), Matrix, pbapply, ggplot2, DRDID (>= 1.3.0), generics, methods, tidyr, fastglm, data.table (>= 1.15.4), dreamerr (>= 1.4.0)

VignetteBuilder knitr

Suggests rmarkdown, plm, here, knitr, covr, backports, broom, testthat (>= 3.0.0), withr, remotes, callr

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Brantly Callaway [aut, cre], Pedro H. C. Sant'Anna [aut]

Maintainer Brantly Callaway <brantly.callaway@uga.edu>

Repository CRAN

Date/Publication 2026-06-13 15:50:02 UTC

Contents

aggte *Aggregate Group-Time Average Treatment Effects*

Description

A function to take group-time average treatment effects and aggregate them into a smaller number of parameters. There are several possible aggregations including "simple", "dynamic", "group", and "calendar."

Usage

```
aggte(
  MP,
  type = "group",
  balance_e = NULL,
  min_e = -Inf,
  max_e = Inf,
  na.rm = FALSE,
  bstrap = NULL,
  biters = NULL,
  cband = NULL,
  alp = NULL,
  clustervars = NULL
)
```

Arguments

MP	an MP object (i.e., the results of the att_gt() method)
type	Which type of aggregated treatment effect parameter to compute. One option is "simple" (this just computes a weighted average of all group-time average treatment effects with weights proportional to group size). Other options are "dynamic" (this computes average effects across different lengths of exposure to the treatment and is similar to an "event study"; here the overall effect averages the effect of the treatment across all positive lengths of exposure); "group" (this is the default option and computes average treatment effects across different groups; here the overall effect averages the effect across different groups); and "calendar" (this computes average treatment effects across different time periods; here the overall effect averages the effect across each time period).
balance_e	If set (and if one computes dynamic effects), it balances the sample with respect to event time. For example, if balance_e=2, aggte will drop groups that are not exposed to treatment for at least three periods. (the initial period when e=0 as well as the next two periods when e=1 and the e=2). This ensures that the composition of groups does not change when event time changes.
min_e	For event studies, this is the smallest event time to compute dynamic effects for. By default, min_e = -Inf so that effects at all lengths of exposure are computed.

max_e	For event studies, this is the largest event time to compute dynamic effects for. By default, max_e = Inf so that effects at all lengths of exposure are computed.
na.rm	Logical value if we are to remove missing values from analyses. Default is FALSE.
bstrap	Boolean for whether or not to compute standard errors using the multiplier bootstrap. Default is value set in the MP object. If bstrap=FALSE, then analytical standard errors are reported; these are cluster-robust when a cluster variable was supplied to att_gt.
biters	The number of bootstrap iterations to use. The default is the value set in the MP object, and this is only applicable if bstrap=TRUE.
cband	Boolean for whether or not to compute a uniform confidence band that covers all of the group-time average treatment effects with fixed probability 1-alp. In order to compute uniform confidence bands, bstrap must also be set to TRUE. The default is the value set in the MP object
alp	the significance level, default is value set in the MP object.
clustervars	A vector of variables to cluster on. The default is the cluster variables set in the MP object (i.e., the ones passed to att_gt()). Clustering at the aggregation stage reuses the per-unit cluster information stored by att_gt(), so aggte can only cluster on idname and/or the variable that was passed to att_gt() via its clustervars argument. If a requested cluster variable cannot be honored – because clustervars was not supplied to att_gt(), or because it names a different variable than the one att_gt() clustered on – aggte issues a warning and reports standard errors that do NOT account for clustering. To obtain clustered standard errors for the aggregated parameters, re-run att_gt() with the desired clustervars.

Value

An AGGTEobj object that holds the results from the aggregation

Examples

Initial ATT(g,t) estimates from att_gt()

```
data(mpdta)
set.seed(09152024)
out <- att_gt(yname="lemp",
              tname="year",
              idname="countyreal",
              gname="first.treat",
              xformla=NULL,
              data=mpdta)
```

You can aggregate the ATT(g,t) in many ways.

Overall ATT:

```
aggte(out, type = "simple")
#>
#> Call:
#> aggte(MP = out, type = "simple")
#>
#> Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
#>
#>
#>    ATT    Std. Error     [ 95%  Conf. Int.]
#>  -0.04        0.0119    -0.0633     -0.0166 *
#>
#>
#> ---
#> Signif. codes: `*' confidence band does not cover 0
#>
#> Control Group:  Never Treated,  Anticipation Periods:  0
#> Estimation Method:  Doubly Robust
```

Dynamic ATT (Event-Study):

```
aggte(out, type = "dynamic")
#>
#> Call:
#> aggte(MP = out, type = "dynamic")
#>
#> Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
#>
#>
#> Overall summary of ATT's based on event-study/dynamic aggregation:
#>      ATT    Std. Error     [ 95%  Conf. Int.]
#>  -0.0772        0.0208     -0.118     -0.0365 *
#>
#>
#> Dynamic Effects:
#>  Event time Estimate Std. Error [95% Simult.  Conf. Band]
#>          -3   0.0305     0.0154       -0.0101      0.0711
#>          -2  -0.0006     0.0129       -0.0346      0.0334
#>          -1  -0.0245     0.0145       -0.0628      0.0139
#>           0  -0.0199     0.0119       -0.0513      0.0114
#>           1  -0.0510     0.0160       -0.0934     -0.0085 *
#>           2  -0.1373     0.0393       -0.2412     -0.0333 *
#>           3  -0.1008     0.0356       -0.1949     -0.0067 *
#> ---
#> Signif. codes: `*' confidence band does not cover 0
#>
#> Control Group:  Never Treated,  Anticipation Periods:  0
#> Estimation Method:  Doubly Robust
```

ATT for each group:

```
aggte(out, type = "group")
#>
#> Call:
#> aggte(MP = out, type = "group")
#>
#> Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
#>
#>
#> Overall summary of ATT's based on group/cohort aggregation:
#>     ATT    Std. Error     [ 95%  Conf. Int.]
#>  -0.031        0.0126    -0.0558     -0.0063 *
#>
#>
#> Group Effects:
#>  Group Estimate Std. Error [95% Simult.  Conf. Band]
#>   2004  -0.0797     0.0300       -0.1445     -0.0150 *
#>   2006  -0.0229     0.0163       -0.0582      0.0123
#>   2007  -0.0261     0.0167       -0.0622      0.0101
#> ---
#> Signif. codes: `*' confidence band does not cover 0
#>
#> Control Group:  Never Treated,  Anticipation Periods:  0
#> Estimation Method:  Doubly Robust
```

ATT for each calendar year:

```
aggte(out, type = "calendar")
#>
#> Call:
#> aggte(MP = out, type = "calendar")
#>
#> Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
#>
#>
#> Overall summary of ATT's based on calendar time aggregation:
#>      ATT    Std. Error     [ 95%  Conf. Int.]
#>  -0.0417        0.0175    -0.0759     -0.0075 *
#>
#>
#> Time Effects:
#>  Time Estimate Std. Error [95% Simult.  Conf. Band]
#>  2004  -0.0105     0.0242       -0.0685      0.0475
#>  2005  -0.0704     0.0319       -0.1467      0.0059
#>  2006  -0.0488     0.0206       -0.0980      0.0004
#>  2007  -0.0371     0.0133       -0.0689     -0.0052 *
#> ---
#> Signif. codes: `*' confidence band does not cover 0
#>
#> Control Group:  Never Treated,  Anticipation Periods:  0
```

```
#> Estimation Method:  Doubly Robust
```

AGGTEobj *AGGTEobj*

Description

Objects of this class hold results on aggregated group-time average treatment effects

An object for holding aggregated treatment effect parameters.

Usage

```
AGGTEobj(
  overall.att = NULL,
  overall.se = NULL,
  type = "simple",
  egt = NULL,
  att.egt = NULL,
  se.egt = NULL,
  crit.val.egt = NULL,
  inf.function = NULL,
  min_e = NULL,
  max_e = NULL,
  balance_e = NULL,
  call = NULL,
  DIDparams = NULL
)
```

Arguments

overall.att	The estimated overall ATT
overall.se	Standard error for overall ATT
type	Which type of aggregated treatment effect parameter to compute. One option is "simple" (this just computes a weighted average of all group-time average treatment effects with weights proportional to group size). Other options are "dynamic" (this computes average effects across different lengths of exposure to the treatment and is similar to an "event study"; here the overall effect averages the effect of the treatment across all positive lengths of exposure); "group" (this is the default option and computes average treatment effects across different groups; here the overall effect averages the effect across different groups); and "calendar" (this computes average treatment effects across different time periods; here the overall effect averages the effect across each time period).
egt	Holds the length of exposure (for dynamic effects), the group (for selective treatment timing), or the time period (for calendar time effects)
att.egt	The ATT specific to egt

se.egt	The standard error specific to egt
crit.val.egt	A critical value for computing uniform confidence bands for dynamic effects, selective treatment timing, or time period effects.
inf.function	The influence function of the chosen aggregated parameters
min_e	For event studies, this is the smallest event time to compute dynamic effects for. By default, min_e = -Inf so that effects at all lengths of exposure are computed.
max_e	For event studies, this is the largest event time to compute dynamic effects for. By default, max_e = Inf so that effects at all lengths of exposure are computed.
balance_e	If set (and if one computes dynamic effects), it balances the sample with respect to event time. For example, if balance_e=2, aggte will drop groups that are not exposed to treatment for at least three periods. (the initial period when e=0 as well as the next two periods when e=1 and the e=2). This ensures that the composition of groups does not change when event time changes.
call	The function call to aggte
DIDparams	A DIDparams object

Value

an AGGTEobj

---

att_gt *Group-Time Average Treatment Effects*

Description

att_gt computes average treatment effects in DID setups where there are more than two periods of data and allowing for treatment to occur at different points in time and allowing for treatment effect heterogeneity and dynamics. See Callaway and Sant'Anna (2021) for a detailed description.

Usage

```
att_gt(
  yname,
  tname,
  idname = NULL,
  gname,
  xformla = NULL,
  data,
  panel = TRUE,
  allow_unbalanced_panel = FALSE,
  control_group = c("nevertreated", "notyettreated"),
  anticipation = 0,
  weightsname = NULL,
  fix_weights = NULL,
  alp = 0.05,
```

```
  bstrap = TRUE,
  cband = TRUE,
  biters = 1000,
  clustervars = NULL,
  est_method = "dr",
  base_period = "varying",
  faster_mode = TRUE,
  print_details = FALSE,
  pl = FALSE,
  cores = 1,
  compute_inffunc = TRUE,
  ...
)
```

Arguments

yname	The name of the outcome variable
tname	The name of the column containing the time periods
idname	The individual (cross-sectional unit) id name
gname	The name of the variable in data that contains the first period when a particular observation is treated. This should be a positive number for all observations in treated groups. It defines which "group" a unit belongs to. It should be 0 for units in the untreated group.
xformla	A formula for the covariates to include in the model. It should be of the form ~ X1 + X2. Default is NULL which is equivalent to xformla=~1. This is used to create a matrix of covariates which is then passed to the 2x2 DID estimator chosen in est_method. For time-varying covariates: (1) With balanced panel data, in each 2x2 comparison, the covariates are taken to be the value of the covariates in the earlier time period, and all of the underlying computations involve changes in Y as a function of those values of covariates. (2) With repeated cross sections data and unbalanced panel data, the covariates are taken from each time period and computations involve Y_post conditional on X_post minus Y_pre conditional on X_pre. A byproduct of this is that, with balanced panel data and in the presence of time-varying covariates, it is possible to get different numerical results according to whether or not allow_unbalanced_panel=TRUE or FALSE.
data	The name of the data.frame that contains the data
panel	Whether or not the data is a panel dataset. The panel dataset should be provided in long format – that is, where each row corresponds to a unit observed at a particular point in time. The default is TRUE. When using a panel dataset, the variable idname must be set. When panel=FALSE, the data is treated as repeated cross sections.
allow_unbalanced_panel	Whether or not function should "balance" the panel with respect to time and id. The default value is FALSE which means that att_gt() will drop all units where data is not observed in all periods. The advantage of this is that the computations are faster (sometimes substantially).

`control_group` Which units to use as the control group. The default is "nevertreated" which sets the control group to be the group of units that never participate in the treatment. This group does not change across groups or time periods. The other option is to set `group="notyettreated"`. In this case, the control group is set to the group of units that have not yet participated in the treatment in that time period. This includes all never treated units, but it includes additional units that eventually participate in the treatment, but have not participated yet.

`anticipation` The number of time periods before participating in the treatment where units can anticipate participating in the treatment and therefore it can affect their untreated potential outcomes

`weightsname` The name of the column containing the sampling weights. If not set, all observations have same weight. When weights are time-invariant (constant within each unit across periods), all `fix_weights` options produce identical results and no special handling is needed.

When weights vary across time (e.g., time-varying population sizes), the default behavior differs by panel type:

Balanced panel Each 2x2 DiD comparison uses the weight from the earlier of the two time periods involved. For post-treatment cells, this is the base period (g-1). For pre-treatment cells with `base_period="varying"`, this is the pre-treatment period itself. The panel DRDID estimators are used.

Repeated cross sections and unbalanced panels Both periods' per-observation weights are passed directly to the RC DRDID estimators, so each observation carries its own period-specific weight.

Use the `fix_weights` argument to override the default behavior.

`fix_weights` Controls how time-varying sampling weights are resolved. Only relevant when weights vary across time; with time-invariant weights, all options produce identical results. Options:

`NULL` **(default)** For balanced panel: uses the weight from the earlier of the two time periods in each 2x2 comparison. For post-treatment cells, this is the base period (g-1). For pre-treatment cells, this depends on the `base_period` setting. For RC/unbalanced panel: uses per-observation weights from both periods.

`"varying"` Uses per-observation, period-specific weights for all panel types. For balanced panel data, this switches to the repeated cross-section DRDID estimators so that pre-period and post-period observations each carry their own weight. Covariates are held fixed at their pre-period values (same as the default panel estimator). This is the most flexible option for weights but sacrifices the efficiency of the panel estimator. For RC/unbalanced panel, this is identical to the default. Not supported with custom `est_method` functions.

`"base_period"` Fixes weights at the base period (g-1) for all (g,t) cells within a group, for both pre-treatment and post-treatment comparisons. Ensures all cells within a group use the same weights. For unbalanced panels, units not observed in the base period are dropped with a warning. Not supported for repeated cross sections (`panel = FALSE`).

"first_period" Fixes weights at the first time period in the dataset for all (g,t) cells. For unbalanced panels, units not observed in the first period are dropped with a warning. Not supported for repeated cross sections (panel = FALSE).

alp — the significance level, default is 0.05

bstrap — Boolean for whether or not to compute standard errors using the multiplier bootstrap. Default is TRUE (in addition, cband is also by default TRUE indicating that uniform confidence bands will be returned). If bstrap=FALSE, analytical standard errors are reported; these are cluster-robust when clustervars is supplied.

cband — Boolean for whether or not to compute a uniform confidence band that covers all of the group-time average treatment effects with fixed probability 1-alp. In order to compute uniform confidence bands, bstrap must also be set to TRUE. The default is TRUE.

biters — The number of bootstrap iterations to use. The default is 1000, and this is only applicable if bstrap=TRUE.

clustervars — A vector of variables names to cluster on. At most, there can be two variables (otherwise will throw an error) and one of these must be the same as idname which allows for clustering at the individual level. Clustered standard errors are available with the multiplier bootstrap (bstrap=TRUE) or analytically (bstrap=FALSE).

est_method — the method to compute group-time average treatment effects. The default is "dr" which uses the doubly robust approach in the DRDID package. Other built-in methods include "ipw" for inverse probability weighting and "reg" for first step regression estimators. The user can also pass their own function for estimating group time average treatment effects. The required signature depends on the data structure:

Panel data (panel=TRUE): f(y1, y0, D, covariates, i.weights, inffunc, ...) where y1 is an n x 1 vector of post-treatment outcomes, y0 is an n x 1 vector of pre-treatment outcomes, D is a binary vector indicating treatment group membership, covariates is an n x k matrix, i.weights is a vector of sampling weights, and inffunc is a logical requesting influence-function computation.

Repeated cross sections / unbalanced panel (panel=FALSE): f(y, post, D, covariates, i.weights, inffunc, ...) where y is the outcome vector (length n), post is a binary indicator for the post-treatment period, D is a binary treatment indicator, covariates is an n x k matrix, i.weights is a vector of sampling weights, and inffunc is a logical.

In both cases the function should return a list that includes ATT (the estimated group-time average treatment effect) and att.inf.func (an n x 1 influence function — one entry per observation passed into the estimator). The function can return other things as well, but these are the only two that are required. With no covariates (xformla = NULL), the built-in methods ("dr", "ipw", "reg") all reduce to the unconditional difference-in-differences estimator, so the choice among them is irrelevant; a custom est_method function is still called (with an intercept-only covariates matrix) and determines the estimates.

base_period — Whether to use a "varying" base period or a "universal" base period. Either choice results in the same post-treatment estimates of ATT(g,t)'s. In pre-treatment

periods, using a varying base period amounts to computing a pseudo-ATT in each treatment period by comparing the change in outcomes for a particular group relative to its comparison group in the pre-treatment periods (i.e., in pre-treatment periods this setting computes changes from period t-1 to period t, but repeatedly changes the value of t)

A universal base period fixes the base period to always be (g-anticipation-1). This does not compute pseudo-ATT(g,t)'s in pre-treatment periods, but rather reports average changes in outcomes from period t to (g-anticipation-1) for a particular group relative to its comparison group. This is analogous to what is often reported in event study regressions.

Using a varying base period results in an estimate of ATT(g,t) being reported in the period immediately before treatment. Using a universal base period normalizes the estimate in the period right before treatment (or earlier when the user allows for anticipation) to be equal to 0, but one extra estimate in an earlier period.

`faster_mode` This option enables a faster version of `did`, optimizing computation time for large datasets by improving data management within the package. The default is set to `TRUE`. Both modes produce identical results up to numerical precision; while the difference is minimal for small datasets, the speedup is substantial for large ones.

`print_details` Whether or not to show details/progress of computations. Default is `FALSE`.

`pl` Whether or not to use parallel processing

`cores` The number of cores to use for parallel processing

`compute_inffunc`

Whether or not to compute the influence functions. The default is `TRUE`. The influence functions are required for standard errors, uniform confidence bands, the parallel-trends pre-test, and for aggregating the group-time effects with `aggte()`. Set `compute_inffunc = FALSE` to obtain the group-time ATT *point estimates only*: this is faster and uses substantially less memory (no $n \times k$ influence-function matrix is formed or bootstrapped), but the returned object has no standard errors / pre-test and cannot be passed to `aggte()`. The point estimates are identical to those from a full run. When `compute_inffunc = FALSE`, `bstrap` and `cband` are set to `FALSE` automatically.

`...` Additional arguments to be passed to a custom `est_method` function. These are ignored when using built-in estimation methods (`"dr"`, `"ipw"`, `"reg"`).

Value

an `MP` object containing all the results for group-time average treatment effects

The returned `inffunc` matrix collects the estimated influence functions, with one column per ATT(g,t) and one row per cross-sectional unit (one row per observation with repeated cross sections). Its rownames hold the unit ids (`idname`; an internal observation index for repeated cross sections) and are the authoritative link between rows and units: the row ORDER is mode-specific (`faster_mode = FALSE` sorts units by id, while `faster_mode = TRUE` uses an internal (period, cohort, id) ordering), so external consumers of the influence functions must align rows by rowname, never by position.

Options

When `faster_mode = FALSE`, setting `options(did.disable_precompute = TRUE)` disables the one-time positional precompute and assembles every 2x2 comparison from the long data per cell, as in earlier versions. The covariate design matrix is still built once over the full sample under both settings, so results are identical either way; the option exists only as a debugging escape hatch.

Examples:

Basic `att_gt()` call:

```
# Example data
data(mpdta)
set.seed(09152024)
out1 <- att_gt(yname="lemp",
               tname="year",
               idname="countyreal",
               gname="first.treat",
               xformla=NULL,
               data=mpdta)
summary(out1)
#>
#> Call:
#> att_gt(yname = "lemp", tname = "year", idname = "countyreal",
#>     gname = "first.treat", xformla = NULL, data = mpdta)
#>
#> Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
#>
#> Group-Time Average Treatment Effects:
#>  Group Time ATT(g,t) Std. Error [95% Simult.  Conf. Band]
#>   2004 2004  -0.0105     0.0258       -0.0809      0.0599
#>   2004 2005  -0.0704     0.0341       -0.1635      0.0227
#>   2004 2006  -0.1373     0.0384       -0.2423     -0.0322 *
#>   2004 2007  -0.1008     0.0354       -0.1976     -0.0040 *
#>   2006 2004   0.0065     0.0235       -0.0578      0.0708
#>   2006 2005  -0.0028     0.0192       -0.0554      0.0499
#>   2006 2006  -0.0046     0.0184       -0.0548      0.0456
#>   2006 2007  -0.0412     0.0207       -0.0977      0.0153
#>   2007 2004   0.0305     0.0161       -0.0135      0.0746
#>   2007 2005  -0.0027     0.0157       -0.0456      0.0401
#>   2007 2006  -0.0311     0.0184       -0.0815      0.0193
#>   2007 2007  -0.0261     0.0176       -0.0741      0.0220
#> ---
#> Signif. codes: `*' confidence band does not cover 0
#>
#> P-value for pre-test of parallel trends assumption:  0.16812
#> Control Group:  Never Treated,  Anticipation Periods:  0
#> Estimation Method:  Doubly Robust
```

Using covariates:

```
out2 <- att_gt(yname="lemp",
               tname="year",
               idname="countyreal",
               gname="first.treat",
               xformla=~lpop,
               data=mpdta)
summary(out2)
#>
#> Call:
#> att_gt(yname = "lemp", tname = "year", idname = "countyreal",
#>     gname = "first.treat", xformla = ~lpop, data = mpdta)
#>
#> Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
#>
#> Group-Time Average Treatment Effects:
#>  Group Time ATT(g,t) Std. Error [95% Simult.  Conf. Band]
#>   2004 2004  -0.0145     0.0222       -0.0737      0.0446
#>   2004 2005  -0.0764     0.0303       -0.1570      0.0041
#>   2004 2006  -0.1404     0.0382       -0.2421     -0.0388 *
#>   2004 2007  -0.1069     0.0358       -0.2021     -0.0117 *
#>   2006 2004  -0.0005     0.0231       -0.0618      0.0609
#>   2006 2005  -0.0062     0.0188       -0.0561      0.0437
#>   2006 2006   0.0010     0.0204       -0.0534      0.0553
#>   2006 2007  -0.0413     0.0210       -0.0971      0.0145
#>   2007 2004   0.0267     0.0140       -0.0104      0.0639
#>   2007 2005  -0.0046     0.0170       -0.0498      0.0407
#>   2007 2006  -0.0284     0.0187       -0.0782      0.0213
#>   2007 2007  -0.0288     0.0161       -0.0715      0.0140
#> ---
#> Signif. codes: `*' confidence band does not cover 0
#>
#> P-value for pre-test of parallel trends assumption:  0.23267
#> Control Group:  Never Treated,  Anticipation Periods:  0
#> Estimation Method:  Doubly Robust
```

Specify comparison units:

```
out3 <- att_gt(yname="lemp",
               tname="year",
               idname="countyreal",
               gname="first.treat",
               xformla=~lpop,
               control_group = "notyettreated",
               data=mpdta)
summary(out3)
#>
#> Call:
#> att_gt(yname = "lemp", tname = "year", idname = "countyreal",
#>    gname = "first.treat", xformla = ~lpop, data = mpdta, control_group = "notyettreated")
```

```
#>
#> Reference: Callaway, Brantly and Pedro H.C. Sant'Anna.  "Difference-in-Differences with Multiple Tim
#>
#> Group-Time Average Treatment Effects:
#>  Group Time ATT(g,t) Std. Error [95% Simult.  Conf. Band]
#>   2004 2004  -0.0212     0.0217       -0.0788      0.0365
#>   2004 2005  -0.0816     0.0324       -0.1676      0.0044
#>   2004 2006  -0.1382     0.0368       -0.2361     -0.0403 *
#>   2004 2007  -0.1069     0.0344       -0.1984     -0.0154 *
#>   2006 2004  -0.0075     0.0233       -0.0693      0.0544
#>   2006 2005  -0.0046     0.0184       -0.0533      0.0442
#>   2006 2006   0.0087     0.0182       -0.0397      0.0570
#>   2006 2007  -0.0413     0.0205       -0.0956      0.0130
#>   2007 2004   0.0269     0.0136       -0.0091      0.0630
#>   2007 2005  -0.0042     0.0153       -0.0448      0.0364
#>   2007 2006  -0.0284     0.0191       -0.0792      0.0223
#>   2007 2007  -0.0288     0.0167       -0.0732      0.0157
#> ---
#> Signif. codes: `*' confidence band does not cover 0
#>
#> P-value for pre-test of parallel trends assumption:  0.23326
#> Control Group:  Not Yet Treated,  Anticipation Periods:  0
#> Estimation Method:  Doubly Robust
```

References

Callaway, Brantly and Pedro H.C. Sant'Anna. \"Difference-in-Differences with Multiple Time Periods.\" Journal of Econometrics, Vol. 225, No. 2, pp. 200-230, 2021. doi:10.1016/j.jeconom.2020.12.001, https://arxiv.org/abs/1803.09015

`build_sim_dataset` *build_sim_dataset*

Description

A function for building simulated data

Usage

```
build_sim_dataset(sp_list, panel = TRUE)
```

Arguments

`sp_list`	A list of simulation parameters. See `reset.sim` to generate some default values for parameters
`panel`	whether to construct panel data (the default) or repeated cross sections data

Value

a data.frame with the following columns

- G observations group
- X value of covariate
- id observation's id
- cluster observation's cluster (by construction there is no within-cluster correlation)
- period time period for current observation
- Y outcome
- treat whether or not this unit is ever treated

`conditional_did_pretest` *Pre-Test of Conditional Parallel Trends Assumption*

Description

An integrated moments test for the conditional parallel trends assumption holding in all pre-treatment time periods for all groups

Usage

```
conditional_did_pretest(
  yname,
  tname,
  idname = NULL,
  gname,
  xformla = NULL,
  data,
  panel = TRUE,
  allow_unbalanced_panel = FALSE,
  control_group = c("nevertreated", "notyettreated"),
  weightsname = NULL,
  alp = 0.05,
  bstrap = TRUE,
  cband = TRUE,
  biters = 1000,
  clustervars = NULL,
  est_method = "ipw",
  print_details = FALSE,
  pl = FALSE,
  cores = 1
)
```

Arguments

yname	The name of the outcome variable
tname	The name of the column containing the time periods
idname	The individual (cross-sectional unit) id name
gname	The name of the variable in data that contains the first period when a particular observation is treated. This should be a positive number for all observations in treated groups. It defines which "group" a unit belongs to. It should be 0 for units in the untreated group.
xformla	A formula for the covariates to include in the model. It should be of the form ~ X1 + X2. Default is NULL which is equivalent to xformla=~1. This is used to create a matrix of covariates which is then passed to the 2x2 DID estimator chosen in est_method. For time-varying covariates: (1) With balanced panel data, in each 2x2 comparison, the covariates are taken to be the value of the covariates in the earlier time period, and all of the underlying computations involve changes in Y as a function of those values of covariates. (2) With repeated cross sections data and unbalanced panel data, the covariates are taken from each time period and computations involve Y_post conditional on X_post minus Y_pre conditional on X_pre. A byproduct of this is that, with balanced panel data and in the presence of time-varying covariates, it is possible to get different numerical results according to whether or not allow_unbalanced_panel=TRUE or FALSE.
data	The name of the data.frame that contains the data
panel	Whether or not the data is a panel dataset. The panel dataset should be provided in long format – that is, where each row corresponds to a unit observed at a particular point in time. The default is TRUE. When using a panel dataset, the variable idname must be set. When panel=FALSE, the data is treated as repeated cross sections.
allow_unbalanced_panel	Whether or not function should "balance" the panel with respect to time and id. The default value is FALSE which means that att_gt() will drop all units where data is not observed in all periods. The advantage of this is that the computations are faster (sometimes substantially).
control_group	Which units to use as the control group. The default is "nevertreated" which sets the control group to be the group of units that never participate in the treatment. This group does not change across groups or time periods. The other option is to set group="notyettreated". In this case, the control group is set to the group of units that have not yet participated in the treatment in that time period. This includes all never treated units, but it includes additional units that eventually participate in the treatment, but have not participated yet.
weightsname	The name of the column containing the sampling weights. If not set, all observations have same weight.
alp	the significance level, default is 0.05
bstrap	Not used by the pre-test. Critical values and p-values for the Cramer von Mises and Kolmogorov-Smirnov test statistics are always computed with the multiplier bootstrap (using biters iterations), regardless of this argument, and no standard errors are reported.

cband	Boolean for whether or not to compute a uniform confidence band that covers all of the group-time average treatment effects with fixed probability 1-alp. In order to compute uniform confidence bands, bstrap must also be set to TRUE. The default is TRUE.
biters	The number of multiplier bootstrap iterations used to simulate the limiting distribution of the test statistics. The default is 1000.
clustervars	A vector of variables names to cluster on (the multiplier bootstrap then draws cluster-level multipliers). At most, there can be two variables (otherwise will throw an error) and one of these must be the same as idname which allows for clustering at the individual level.
est_method	the method to compute group-time average treatment effects. The default for conditional_did_pretest is "ipw" for inverse probability weighting. Other built-in methods include "dr" which uses the doubly robust approach in the DRDID package and "reg" for first step regression estimators.
print_details	Whether or not to show details/progress of computations. Default is FALSE.
pl	Whether or not to use parallel processing
cores	The number of cores to use for parallel processing. This parallelizes Step 1 (computing the test statistic); Step 2's multiplier bootstrap is vectorized and runs in a single process.

Value

an MP.TEST object

References

Callaway, Brantly and Sant'Anna, Pedro H. C. "Difference-in-Differences with Multiple Time Periods and an Application on the Minimum Wage and Employment." Working Paper https://arxiv.org/abs/1803.09015v2 (2018).

Examples

```
## Not run:
data(mpdta)
pre.test <- conditional_did_pretest(yname="lemp",
                                    tname="year",
                                    idname="countyreal",
                                    gname="first.treat",
                                    xformla=~lpop,
                                    data=mpdta)
summary(pre.test)

## End(Not run)
```

DIDparams *DIDparams*

Description

Object to hold did parameters that are passed across functions

Usage

```
DIDparams(
  yname,
  tname,
  idname = NULL,
  gname,
  xformla = NULL,
  data,
  control_group,
  anticipation = 0,
  weightsname = NULL,
  fix_weights = NULL,
  alp = 0.05,
  bstrap = TRUE,
  biters = 1000,
  clustervars = NULL,
  cband = TRUE,
  print_details = TRUE,
  faster_mode = FALSE,
  pl = FALSE,
  cores = 1,
  est_method = "dr",
  base_period = "varying",
  panel = TRUE,
  true_repeated_cross_sections,
  n = NULL,
  nG = NULL,
  nT = NULL,
  tlist = NULL,
  glist = NULL,
  call = NULL
)
```

Arguments

yname	The name of the outcome variable
tname	The name of the column containing the time periods
idname	The individual (cross-sectional unit) id name

`gname` The name of the variable in `data` that contains the first period when a particular observation is treated. This should be a positive number for all observations in treated groups. It defines which "group" a unit belongs to. It should be 0 for units in the untreated group.

`xformla` A formula for the covariates to include in the model. It should be of the form `~ X1 + X2`. Default is NULL which is equivalent to `xformla=~1`. This is used to create a matrix of covariates which is then passed to the 2x2 DID estimator chosen in `est_method`.

For time-varying covariates: (1) With balanced panel data, in each 2x2 comparison, the covariates are taken to be the value of the covariates in the earlier time period, and all of the underlying computations involve changes in Y as a function of those values of covariates. (2) With repeated cross sections data and unbalanced panel data, the covariates are taken from each time period and computations involve Y_post conditional on X_post minus Y_pre conditional on X_pre. A byproduct of this is that, with balanced panel data and in the presence of time-varying covariates, it is possible to get different numerical results according to whether or not `allow_unbalanced_panel=TRUE` or `FALSE`.

`data` The name of the data.frame that contains the data

`control_group` Which units to use as the control group. The default is "nevertreated" which sets the control group to be the group of units that never participate in the treatment. This group does not change across groups or time periods. The other option is to set `group="notyettreated"`. In this case, the control group is set to the group of units that have not yet participated in the treatment in that time period. This includes all never treated units, but it includes additional units that eventually participate in the treatment, but have not participated yet.

`anticipation` The number of time periods before participating in the treatment where units can anticipate participating in the treatment and therefore it can affect their untreated potential outcomes

`weightsname` The name of the column containing the sampling weights. If not set, all observations have same weight. When weights are time-invariant (constant within each unit across periods), all `fix_weights` options produce identical results and no special handling is needed.

When weights vary across time (e.g., time-varying population sizes), the default behavior differs by panel type:

Balanced panel Each 2x2 DiD comparison uses the weight from the earlier of the two time periods involved. For post-treatment cells, this is the base period (g-1). For pre-treatment cells with `base_period="varying"`, this is the pre-treatment period itself. The panel DRDID estimators are used.

Repeated cross sections and unbalanced panels Both periods' per-observation weights are passed directly to the RC DRDID estimators, so each observation carries its own period-specific weight.

Use the `fix_weights` argument to override the default behavior.

`fix_weights` Controls how time-varying sampling weights are resolved. Only relevant when weights vary across time; with time-invariant weights, all options produce identical results. Options:

NULL **(default)** For balanced panel: uses the weight from the earlier of the two time periods in each 2x2 comparison. For post-treatment cells, this is the base period (g-1). For pre-treatment cells, this depends on the base_period setting. For RC/unbalanced panel: uses per-observation weights from both periods.

"varying" Uses per-observation, period-specific weights for all panel types. For balanced panel data, this switches to the repeated cross-section DRDID estimators so that pre-period and post-period observations each carry their own weight. Covariates are held fixed at their pre-period values (same as the default panel estimator). This is the most flexible option for weights but sacrifices the efficiency of the panel estimator. For RC/unbalanced panel, this is identical to the default. Not supported with custom est_method functions.

"base_period" Fixes weights at the base period (g-1) for all (g,t) cells within a group, for both pre-treatment and post-treatment comparisons. Ensures all cells within a group use the same weights. For unbalanced panels, units not observed in the base period are dropped with a warning. Not supported for repeated cross sections (panel = FALSE).

"first_period" Fixes weights at the first time period in the dataset for all (g,t) cells. For unbalanced panels, units not observed in the first period are dropped with a warning. Not supported for repeated cross sections (panel = FALSE).

alp — the significance level, default is 0.05

bstrap — Boolean for whether or not to compute standard errors using the multiplier bootstrap. Default is TRUE (in addition, cband is also by default TRUE indicating that uniform confidence bands will be returned). If bstrap=FALSE, analytical standard errors are reported; these are cluster-robust when clustervars is supplied.

biters — The number of bootstrap iterations to use. The default is 1000, and this is only applicable if bstrap=TRUE.

clustervars — A vector of variables names to cluster on. At most, there can be two variables (otherwise will throw an error) and one of these must be the same as idname which allows for clustering at the individual level. Clustered standard errors are available with the multiplier bootstrap (bstrap=TRUE) or analytically (bstrap=FALSE).

cband — Boolean for whether or not to compute a uniform confidence band that covers all of the group-time average treatment effects with fixed probability 1-alp. In order to compute uniform confidence bands, bstrap must also be set to TRUE. The default is TRUE.

print_details — Whether or not to show details/progress of computations. Default is FALSE.

faster_mode — This option enables a faster version of did, optimizing computation time for large datasets by improving data management within the package. The default is set to TRUE. Both modes produce identical results up to numerical precision; while the difference is minimal for small datasets, the speedup is substantial for large ones.

pl — Whether or not to use parallel processing

cores — The number of cores to use for parallel processing

est_method
: the method to compute group-time average treatment effects. The default is "dr" which uses the doubly robust approach in the DRDID package. Other built-in methods include "ipw" for inverse probability weighting and "reg" for first step regression estimators. The user can also pass their own function for estimating group time average treatment effects. The required signature depends on the data structure:

  Panel data (panel=TRUE): `f(y1, y0, D, covariates, i.weights, inffunc, ...)` where `y1` is an `n x 1` vector of post-treatment outcomes, `y0` is an `n x 1` vector of pre-treatment outcomes, `D` is a binary vector indicating treatment group membership, `covariates` is an `n x k` matrix, `i.weights` is a vector of sampling weights, and `inffunc` is a logical requesting influence-function computation.

  Repeated cross sections / unbalanced panel (panel=FALSE): `f(y, post, D, covariates, i.weights, inffunc, ...)` where `y` is the outcome vector (length `n`), `post` is a binary indicator for the post-treatment period, `D` is a binary treatment indicator, `covariates` is an `n x k` matrix, `i.weights` is a vector of sampling weights, and `inffunc` is a logical.

  In both cases the function should return a list that includes `ATT` (the estimated group-time average treatment effect) and `att.inf.func` (an `n x 1` influence function — one entry per observation passed into the estimator). The function can return other things as well, but these are the only two that are required. With no covariates (`xformla = NULL`), the built-in methods ("dr", "ipw", "reg") all reduce to the unconditional difference-in-differences estimator, so the choice among them is irrelevant; a custom `est_method` function is still called (with an intercept-only `covariates` matrix) and determines the estimates.

base_period
: Whether to use a "varying" base period or a "universal" base period. Either choice results in the same post-treatment estimates of ATT(g,t)'s. In pre-treatment periods, using a varying base period amounts to computing a pseudo-ATT in each treatment period by comparing the change in outcomes for a particular group relative to its comparison group in the pre-treatment periods (i.e., in pre-treatment periods this setting computes changes from period t-1 to period t, but repeatedly changes the value of t)

  A universal base period fixes the base period to always be (g-anticipation-1). This does not compute pseudo-ATT(g,t)'s in pre-treatment periods, but rather reports average changes in outcomes from period t to (g-anticipation-1) for a particular group relative to its comparison group. This is analogous to what is often reported in event study regressions.

  Using a varying base period results in an estimate of ATT(g,t) being reported in the period immediately before treatment. Using a universal base period normalizes the estimate in the period right before treatment (or earlier when the user allows for anticipation) to be equal to 0, but one extra estimate in an earlier period.

panel
: Whether or not the data is a panel dataset. The panel dataset should be provided in long format – that is, where each row corresponds to a unit observed at a particular point in time. The default is TRUE. When using a panel dataset, the variable `idname` must be set. When `panel=FALSE`, the data is treated as repeated cross sections.

true_repeated_cross_sections
: Whether or not the data really is repeated cross sections. (We include this be-

	cause unbalanced panel code runs through the repeated cross sections code)
n	The number of observations. This is equal to the number of units (which may be different from the number of rows in a panel dataset).
nG	The number of groups
nT	The number of time periods
tlist	a vector containing each time period
glist	a vector containing each group
call	Function call to att_gt

---

ggdid *Plot* did *objects using* ggplot2

Description

Function to plot objects from the did package

Usage

```
ggdid(object, ...)
```

Arguments

object	either a MP object or AGGTEobj object. See help(ggdid.MP) and help(ggdid.AGGTEobj).
...	other arguments

Examples

```
## Not run:
data(mpdta)
out <- att_gt(yname = "lemp",
              gname = "first.treat",
              idname = "countyreal",
              tname = "year",
              xformla = ~1,
              data = mpdta)

# plot all group-time average treatment effects
ggdid(out)

# plot event study aggregation
es <- aggte(out, type = "dynamic")
ggdid(es)

## End(Not run)
```

ggdid.AGGTEobj *Plot* AGGTEobj *objects*

Description

A function to plot AGGTEobj objects

Usage

```
## S3 method for class 'AGGTEobj'
ggdid(
  object,
  ylim = NULL,
  xlab = NULL,
  ylab = NULL,
  title = "",
  xgap = 1,
  legend = TRUE,
  ref_line = 0,
  theming = TRUE,
  ...
)
```

Arguments

object	an AGGTEobj object, as returned by aggte.
ylim	optional y limits for the plot; setting here makes the y limits the same across different plots
xlab	optional x-axis label
ylab	optional y-axis label
title	optional plot title
xgap	optional gap between the labels on the x-axis. For example, xgap=3 indicates that the labels should show up for every third value on the x-axis. The default is 1.
legend	Whether or not to include a legend (which will indicate color of pre- and post-treatment estimates). Default is TRUE.
ref_line	A reference line at this value, usually to compare confidence intervals to 0. Set to NULL to omit.
theming	Set to FALSE to skip all theming so you can do it yourself.
...	other arguments

ggdid.MP *Plot* MP *objects using* ggplot2

Description

A function to plot MP objects

Usage

```
## S3 method for class 'MP'
ggdid(
  object,
  ylim = NULL,
  xlab = NULL,
  ylab = NULL,
  title = "Group",
  xgap = 1,
  ncol = 1,
  legend = TRUE,
  group = NULL,
  ref_line = 0,
  theming = TRUE,
  grtitle = "Group",
  ...
)
```

Arguments

object	an MP object, as returned by att_gt.
ylim	optional y limits for the plot; setting here makes the y limits the same across different plots
xlab	optional x-axis label
ylab	optional y-axis label
title	optional plot title
xgap	optional gap between the labels on the x-axis. For example, xgap=3 indicates that the labels should show up for every third value on the x-axis. The default is 1.
ncol	The number of columns to include in the resulting plot. The default is 1.
legend	Whether or not to include a legend (which will indicate color of pre- and post-treatment estimates). Default is TRUE.
group	Vector for which groups to include in the plots of ATT(g,t). Default is NULL, and, in this case, plots for all groups will be included (ggdid.MP only).
ref_line	A reference line at this value, usually to compare confidence intervals to 0. Set to NULL to omit.

theming	Set to FALSE to skip all theming so you can do it yourself.
grtitle	Title to append before each group name (ggdid.MP only).
...	other arguments

---

glance.AGGTEobj *glance model characteristics from AGGTEobj objects*

Description

glance model characteristics from AGGTEobj objects

Usage

```
## S3 method for class 'AGGTEobj'
glance(x, ...)
```

Arguments

x	a model of class AGGTEobj produced by the aggte() function
...	other arguments passed to methods

---

glance.MP *glance model characteristics from MP objects*

Description

glance model characteristics from MP objects

Usage

```
## S3 method for class 'MP'
glance(x, ...)
```

Arguments

x	a model of class MP produced by the att_gt() function
...	other arguments passed to methods

indicator *indicator*

Description

indicator weighting function

Usage

```
indicator(X, u)
```

Arguments

X	matrix of X's from the data
u	a particular value to compare X's to

Value

numeric vector

Examples

```
data(mpdta)
dta <- subset(mpdta, year==2007)
X <- model.matrix(~lpop, data=dta)
X <- indicator(X, X[1,])
```

mboot *Multiplier Bootstrap*

Description

A function to take an influence function and use the multiplier bootstrap to compute standard errors and critical values for uniform confidence bands.

Usage

```
mboot(inf.func, DIDparams, pl = FALSE, cores = 1, return_V = TRUE)
```

Arguments

inf.func	an influence function
DIDparams	DIDparams object
pl	whether or not to use parallel processing in the multiplier bootstrap, default=FALSE
cores	the number of cores to use with parallel processing, default=1
return_V	whether to compute and return the bootstrap variance matrix V. Default is TRUE. Internal callers that only consume bres, se, or crit.val set this to FALSE to skip the computation (it is the only O(biters x k^2) step in the function).

Value

list with elements

bres	results from each bootstrap iteration
V	variance matrix (NULL when return_V = FALSE)
se	standard errors
crit.val	a critical value for computing uniform confidence bands

---

MP *MP*

Description

Multi-period objects that hold results for group-time average treatment effects

Usage

```
MP(
  group,
  t,
  att,
  V_analytical,
  se,
  c,
  inffunc,
  n = NULL,
  W = NULL,
  Wpval = NULL,
  aggte = NULL,
  alp = 0.05,
  DIDparams = NULL
)
```

Arguments

group	which group (defined by period first treated) an group-time average treatment effect is for
t	which time period a group-time average treatment effect is for
att	the group-average treatment effect for group group and time period t
V_analytical	Analytical estimator for the asymptotic variance-covariance matrix for group-time average treatment effects
se	standard errors for group-time average treatment effects. If bootstrap is set to TRUE, this provides bootstrap-based se.
c	simultaneous critical value if one is obtaining simultaneous confidence bands. Otherwise it reports the critical value based on pointwise normal approximation.
inffunc	the influence function for estimating group-time average treatment effects: one column per ATT(g,t) and one row per cross-sectional unit (one row per observation with repeated cross sections). The rownames hold the unit ids and are the authoritative link between rows and units – the row order differs across faster_mode = TRUE (internal (period, cohort, id) ordering) and faster_mode = FALSE (id-sorted), so align rows by rowname, never by position.
n	the number of unique cross-sectional units (unique values of idname)
W	the Wald statistic for pre-testing the common trends assumption
Wpval	the p-value of the Wald statistic for pre-testing the common trends assumption
aggte	an aggregate treatment effects object
alp	the significance level, default is 0.05
DIDparams	a DIDparams object. A way to optionally return the parameters of the call to att_gt() or conditional_did_pretest().

Value

MP object

MP.TEST *MP.TEST*

Description

An object that holds results from computing pre-test of the conditional parallel trends assumption

Usage

```
MP.TEST(
  CvM = NULL,
  CvMb = NULL,
  CvMcval = NULL,
  CvMpval = NULL,
```

```
  KS = NULL,
  KSb = NULL,
  KScval = NULL,
  KSpval = NULL,
  clustervars = NULL,
  xformla = NULL
)
```

Arguments

CvM	Cramer von Mises test statistic
CvMb	a vector of bootstrapped Cramer von Mises test statistics
CvMcval	CvM critical value
CvMpval	p-value for CvM test
KS	Kolmogorov-Smirnov test statistic
KSb	a vector of bootstrapped KS test statistics
KScval	KS critical value
KSpval	p-value for KS test
clustervars	vector of which variables were clustered on for the test
xformla	formla for the X variables used in the test

---

mpdta *County Teen Employment Dataset*

Description

A dataset containing (the log of) teen employment in 500 counties in the U.S. from 2003 to 2007. This is a subset of the dataset used in Callaway and Sant'Anna (2021). See that paper for additional descriptions.

Usage

```
mpdta
```

Format

A data frame with 2500 rows and 6 variables:

year the year of the observation

countyreal a unique identifier for a particular county

lpop the log of 1000s of population for the county

lemp the log of teen employment in the county

first.treat the year that the state where the county is located raised its minimum wage, it is set equal to 0 for counties that have minimum wages equal to the federal minimum wage over the entire period.

treat whether or not a particular county is ever treated during the sample period. It is time-invariant within each county and equal to 1 exactly when first.treat is positive; it is not a year-specific treatment indicator.

Source

Callaway and Sant'Anna (2021)

`nobs.AGGTEobj` *Number of unique cross-sectional units in an AGGTEobj object*

Description

Number of unique cross-sectional units in an AGGTEobj object

Usage

```
## S3 method for class 'AGGTEobj'
nobs(object, ...)
```

Arguments

`object`	a model of class AGGTEobj produced by the `aggte()` function
`...`	additional arguments (ignored)

Value

Integer. The number of unique cross-sectional units in the data.

`nobs.MP` *Number of unique cross-sectional units in an MP object*

Description

Number of unique cross-sectional units in an MP object

Usage

```
## S3 method for class 'MP'
nobs(object, ...)
```

Arguments

object	a model of class MP produced by the att_gt() function
...	additional arguments (ignored)

Value

Integer. The number of unique cross-sectional units in the data.

---

pre_process_did *Process* did *Function Arguments*

Description

Function to process arguments passed to the main methods in the did package as well as conducting some tests to make sure data is in proper format / try to throw helpful error messages.

Usage

```
pre_process_did(
  yname,
  tname,
  idname,
  gname,
  xformla = NULL,
  data,
  panel = TRUE,
  allow_unbalanced_panel,
  control_group = c("nevertreated", "notyettreated"),
  anticipation = 0,
  weightsname = NULL,
  fix_weights = NULL,
  alp = 0.05,
  bstrap = FALSE,
  cband = FALSE,
  biters = 1000,
  clustervars = NULL,
  est_method = "dr",
  base_period = "varying",
  print_details = TRUE,
  faster_mode = FALSE,
  pl = FALSE,
  cores = 1,
  call = NULL
)
```

Arguments

yname	The name of the outcome variable
tname	The name of the column containing the time periods
idname	The individual (cross-sectional unit) id name
gname	The name of the variable in data that contains the first period when a particular observation is treated. This should be a positive number for all observations in treated groups. It defines which "group" a unit belongs to. It should be 0 for units in the untreated group.
xformla	A formula for the covariates to include in the model. It should be of the form ~ X1 + X2. Default is NULL which is equivalent to xformla=~1. This is used to create a matrix of covariates which is then passed to the 2x2 DID estimator chosen in est_method. For time-varying covariates: (1) With balanced panel data, in each 2x2 comparison, the covariates are taken to be the value of the covariates in the earlier time period, and all of the underlying computations involve changes in Y as a function of those values of covariates. (2) With repeated cross sections data and unbalanced panel data, the covariates are taken from each time period and computations involve Y_post conditional on X_post minus Y_pre conditional on X_pre. A byproduct of this is that, with balanced panel data and in the presence of time-varying covariates, it is possible to get different numerical results according to whether or not allow_unbalanced_panel=TRUE or FALSE.
data	The name of the data.frame that contains the data
panel	Whether or not the data is a panel dataset. The panel dataset should be provided in long format – that is, where each row corresponds to a unit observed at a particular point in time. The default is TRUE. When using a panel dataset, the variable idname must be set. When panel=FALSE, the data is treated as repeated cross sections.
allow_unbalanced_panel	Whether or not function should "balance" the panel with respect to time and id. The default value is FALSE which means that att_gt() will drop all units where data is not observed in all periods. The advantage of this is that the computations are faster (sometimes substantially).
control_group	Which units to use as the control group. The default is "nevertreated" which sets the control group to be the group of units that never participate in the treatment. This group does not change across groups or time periods. The other option is to set group="notyettreated". In this case, the control group is set to the group of units that have not yet participated in the treatment in that time period. This includes all never treated units, but it includes additional units that eventually participate in the treatment, but have not participated yet.
anticipation	The number of time periods before participating in the treatment where units can anticipate participating in the treatment and therefore it can affect their untreated potential outcomes
weightsname	The name of the column containing the sampling weights. If not set, all observations have same weight. When weights are time-invariant (constant within each unit across periods), all fix_weights options produce identical results and no special handling is needed.

When weights vary across time (e.g., time-varying population sizes), the default behavior differs by panel type:

Balanced panel Each 2x2 DiD comparison uses the weight from the earlier of the two time periods involved. For post-treatment cells, this is the base period (g-1). For pre-treatment cells with `base_period="varying"`, this is the pre-treatment period itself. The panel DRDID estimators are used.

Repeated cross sections and unbalanced panels Both periods' per-observation weights are passed directly to the RC DRDID estimators, so each observation carries its own period-specific weight.

Use the `fix_weights` argument to override the default behavior.

`fix_weights` Controls how time-varying sampling weights are resolved. Only relevant when weights vary across time; with time-invariant weights, all options produce identical results. Options:

`NULL` **(default)** For balanced panel: uses the weight from the earlier of the two time periods in each 2x2 comparison. For post-treatment cells, this is the base period (g-1). For pre-treatment cells, this depends on the `base_period` setting. For RC/unbalanced panel: uses per-observation weights from both periods.

`"varying"` Uses per-observation, period-specific weights for all panel types. For balanced panel data, this switches to the repeated cross-section DRDID estimators so that pre-period and post-period observations each carry their own weight. Covariates are held fixed at their pre-period values (same as the default panel estimator). This is the most flexible option for weights but sacrifices the efficiency of the panel estimator. For RC/unbalanced panel, this is identical to the default. Not supported with custom `est_method` functions.

`"base_period"` Fixes weights at the base period (g-1) for all (g,t) cells within a group, for both pre-treatment and post-treatment comparisons. Ensures all cells within a group use the same weights. For unbalanced panels, units not observed in the base period are dropped with a warning. Not supported for repeated cross sections (`panel = FALSE`).

`"first_period"` Fixes weights at the first time period in the dataset for all (g,t) cells. For unbalanced panels, units not observed in the first period are dropped with a warning. Not supported for repeated cross sections (`panel = FALSE`).

`alp` the significance level, default is 0.05

`bstrap` Boolean for whether or not to compute standard errors using the multiplier bootstrap. Default is `TRUE` (in addition, cband is also by default `TRUE` indicating that uniform confidence bands will be returned). If `bstrap=FALSE`, analytical standard errors are reported; these are cluster-robust when `clustervars` is supplied.

`cband` Boolean for whether or not to compute a uniform confidence band that covers all of the group-time average treatment effects with fixed probability `1-alp`. In order to compute uniform confidence bands, `bstrap` must also be set to `TRUE`. The default is `TRUE`.

`biters` The number of bootstrap iterations to use. The default is 1000, and this is only applicable if `bstrap=TRUE`.

clustervars — A vector of variables names to cluster on. At most, there can be two variables (otherwise will throw an error) and one of these must be the same as idname which allows for clustering at the individual level. Clustered standard errors are available with the multiplier bootstrap (bstrap=TRUE) or analytically (bstrap=FALSE).

est_method — the method to compute group-time average treatment effects. The default is "dr" which uses the doubly robust approach in the DRDID package. Other built-in methods include "ipw" for inverse probability weighting and "reg" for first step regression estimators. The user can also pass their own function for estimating group time average treatment effects. The required signature depends on the data structure:

Panel data (panel=TRUE): f(y1, y0, D, covariates, i.weights, inffunc, ...) where y1 is an n x 1 vector of post-treatment outcomes, y0 is an n x 1 vector of pre-treatment outcomes, D is a binary vector indicating treatment group membership, covariates is an n x k matrix, i.weights is a vector of sampling weights, and inffunc is a logical requesting influence-function computation.

Repeated cross sections / unbalanced panel (panel=FALSE): f(y, post, D, covariates, i.weights, inffunc, ...) where y is the outcome vector (length n), post is a binary indicator for the post-treatment period, D is a binary treatment indicator, covariates is an n x k matrix, i.weights is a vector of sampling weights, and inffunc is a logical.

In both cases the function should return a list that includes ATT (the estimated group-time average treatment effect) and att.inf.func (an n x 1 influence function — one entry per observation passed into the estimator). The function can return other things as well, but these are the only two that are required. With no covariates (xformla = NULL), the built-in methods ("dr", "ipw", "reg") all reduce to the unconditional difference-in-differences estimator, so the choice among them is irrelevant; a custom est_method function is still called (with an intercept-only covariates matrix) and determines the estimates.

base_period — Whether to use a "varying" base period or a "universal" base period. Either choice results in the same post-treatment estimates of ATT(g,t)'s. In pre-treatment periods, using a varying base period amounts to computing a pseudo-ATT in each treatment period by comparing the change in outcomes for a particular group relative to its comparison group in the pre-treatment periods (i.e., in pre-treatment periods this setting computes changes from period t-1 to period t, but repeatedly changes the value of t)

A universal base period fixes the base period to always be (g-anticipation-1). This does not compute pseudo-ATT(g,t)'s in pre-treatment periods, but rather reports average changes in outcomes from period t to (g-anticipation-1) for a particular group relative to its comparison group. This is analogous to what is often reported in event study regressions.

Using a varying base period results in an estimate of ATT(g,t) being reported in the period immediately before treatment. Using a universal base period normalizes the estimate in the period right before treatment (or earlier when the user allows for anticipation) to be equal to 0, but one extra estimate in an earlier period.

print_details — Whether or not to show details/progress of computations. Default is FALSE.

faster_mode	This option enables a faster version of did, optimizing computation time for large datasets by improving data management within the package. The default is set to TRUE. Both modes produce identical results up to numerical precision; while the difference is minimal for small datasets, the speedup is substantial for large ones.
pl	Whether or not to use parallel processing
cores	The number of cores to use for parallel processing
call	Function call to att_gt

Value

a DIDparams object

---

pre_process_did2 *Process* did *Function Arguments*

Description

Function to process arguments passed to the main methods in the did package as well as conducting some tests to make sure data is in proper format / try to throw helpful error messages.

Usage

```
pre_process_did2(
  yname,
  tname,
  idname,
  gname,
  xformla = NULL,
  data,
  panel = TRUE,
  allow_unbalanced_panel,
  control_group = c("nevertreated", "notyettreated"),
  anticipation = 0,
  weightsname = NULL,
  fix_weights = NULL,
  alp = 0.05,
  bstrap = FALSE,
  cband = FALSE,
  biters = 1000,
  clustervars = NULL,
  est_method = "dr",
  base_period = "varying",
  print_details = TRUE,
  faster_mode = FALSE,
  pl = FALSE,
```

```
  cores = 1,
  call = NULL
)
```

Arguments

yname	The name of the outcome variable
tname	The name of the column containing the time periods
idname	The individual (cross-sectional unit) id name
gname	The name of the variable in data that contains the first period when a particular observation is treated. This should be a positive number for all observations in treated groups. It defines which "group" a unit belongs to. It should be 0 for units in the untreated group.
xformla	A formula for the covariates to include in the model. It should be of the form ~ X1 + X2. Default is NULL which is equivalent to xformla=~1. This is used to create a matrix of covariates which is then passed to the 2x2 DID estimator chosen in est_method. For time-varying covariates: (1) With balanced panel data, in each 2x2 comparison, the covariates are taken to be the value of the covariates in the earlier time period, and all of the underlying computations involve changes in Y as a function of those values of covariates. (2) With repeated cross sections data and unbalanced panel data, the covariates are taken from each time period and computations involve Y_post conditional on X_post minus Y_pre conditional on X_pre. A byproduct of this is that, with balanced panel data and in the presence of time-varying covariates, it is possible to get different numerical results according to whether or not allow_unbalanced_panel=TRUE or FALSE.
data	The name of the data.frame that contains the data
panel	Whether or not the data is a panel dataset. The panel dataset should be provided in long format – that is, where each row corresponds to a unit observed at a particular point in time. The default is TRUE. When using a panel dataset, the variable idname must be set. When panel=FALSE, the data is treated as repeated cross sections.
allow_unbalanced_panel	Whether or not function should "balance" the panel with respect to time and id. The default value is FALSE which means that att_gt() will drop all units where data is not observed in all periods. The advantage of this is that the computations are faster (sometimes substantially).
control_group	Which units to use as the control group. The default is "nevertreated" which sets the control group to be the group of units that never participate in the treatment. This group does not change across groups or time periods. The other option is to set group="notyettreated". In this case, the control group is set to the group of units that have not yet participated in the treatment in that time period. This includes all never treated units, but it includes additional units that eventually participate in the treatment, but have not participated yet.
anticipation	The number of time periods before participating in the treatment where units can anticipate participating in the treatment and therefore it can affect their untreated potential outcomes

`weightsname` The name of the column containing the sampling weights. If not set, all observations have same weight. When weights are time-invariant (constant within each unit across periods), all `fix_weights` options produce identical results and no special handling is needed.

When weights vary across time (e.g., time-varying population sizes), the default behavior differs by panel type:

Balanced panel Each 2x2 DiD comparison uses the weight from the earlier of the two time periods involved. For post-treatment cells, this is the base period (g-1). For pre-treatment cells with `base_period="varying"`, this is the pre-treatment period itself. The panel DRDID estimators are used.

Repeated cross sections and unbalanced panels Both periods' per-observation weights are passed directly to the RC DRDID estimators, so each observation carries its own period-specific weight.

Use the `fix_weights` argument to override the default behavior.

`fix_weights` Controls how time-varying sampling weights are resolved. Only relevant when weights vary across time; with time-invariant weights, all options produce identical results. Options:

`NULL` **(default)** For balanced panel: uses the weight from the earlier of the two time periods in each 2x2 comparison. For post-treatment cells, this is the base period (g-1). For pre-treatment cells, this depends on the `base_period` setting. For RC/unbalanced panel: uses per-observation weights from both periods.

`"varying"` Uses per-observation, period-specific weights for all panel types. For balanced panel data, this switches to the repeated cross-section DRDID estimators so that pre-period and post-period observations each carry their own weight. Covariates are held fixed at their pre-period values (same as the default panel estimator). This is the most flexible option for weights but sacrifices the efficiency of the panel estimator. For RC/unbalanced panel, this is identical to the default. Not supported with custom `est_method` functions.

`"base_period"` Fixes weights at the base period (g-1) for all (g,t) cells within a group, for both pre-treatment and post-treatment comparisons. Ensures all cells within a group use the same weights. For unbalanced panels, units not observed in the base period are dropped with a warning. Not supported for repeated cross sections (`panel = FALSE`).

`"first_period"` Fixes weights at the first time period in the dataset for all (g,t) cells. For unbalanced panels, units not observed in the first period are dropped with a warning. Not supported for repeated cross sections (`panel = FALSE`).

`alp` the significance level, default is 0.05

`bstrap` Boolean for whether or not to compute standard errors using the multiplier bootstrap. Default is `TRUE` (in addition, cband is also by default `TRUE` indicating that uniform confidence bands will be returned). If `bstrap=FALSE`, analytical standard errors are reported; these are cluster-robust when `clustervars` is supplied.

`cband` Boolean for whether or not to compute a uniform confidence band that covers all of the group-time average treatment effects with fixed probability `1-alp`. In

order to compute uniform confidence bands, bstrap must also be set to TRUE. The default is TRUE.

biters The number of bootstrap iterations to use. The default is 1000, and this is only applicable if bstrap=TRUE.

clustervars A vector of variables names to cluster on. At most, there can be two variables (otherwise will throw an error) and one of these must be the same as idname which allows for clustering at the individual level. Clustered standard errors are available with the multiplier bootstrap (bstrap=TRUE) or analytically (bstrap=FALSE).

est_method the method to compute group-time average treatment effects. The default is "dr" which uses the doubly robust approach in the DRDID package. Other built-in methods include "ipw" for inverse probability weighting and "reg" for first step regression estimators. The user can also pass their own function for estimating group time average treatment effects. The required signature depends on the data structure:

Panel data (panel=TRUE): f(y1, y0, D, covariates, i.weights, inffunc, ...) where y1 is an n x 1 vector of post-treatment outcomes, y0 is an n x 1 vector of pre-treatment outcomes, D is a binary vector indicating treatment group membership, covariates is an n x k matrix, i.weights is a vector of sampling weights, and inffunc is a logical requesting influence-function computation.

Repeated cross sections / unbalanced panel (panel=FALSE): f(y, post, D, covariates, i.weights, inffunc, ...) where y is the outcome vector (length n), post is a binary indicator for the post-treatment period, D is a binary treatment indicator, covariates is an n x k matrix, i.weights is a vector of sampling weights, and inffunc is a logical.

In both cases the function should return a list that includes ATT (the estimated group-time average treatment effect) and att.inf.func (an n x 1 influence function — one entry per observation passed into the estimator). The function can return other things as well, but these are the only two that are required. With no covariates (xformla = NULL), the built-in methods ("dr", "ipw", "reg") all reduce to the unconditional difference-in-differences estimator, so the choice among them is irrelevant; a custom est_method function is still called (with an intercept-only covariates matrix) and determines the estimates.

base_period Whether to use a "varying" base period or a "universal" base period. Either choice results in the same post-treatment estimates of ATT(g,t)'s. In pre-treatment periods, using a varying base period amounts to computing a pseudo-ATT in each treatment period by comparing the change in outcomes for a particular group relative to its comparison group in the pre-treatment periods (i.e., in pre-treatment periods this setting computes changes from period t-1 to period t, but repeatedly changes the value of t)

A universal base period fixes the base period to always be (g-anticipation-1). This does not compute pseudo-ATT(g,t)'s in pre-treatment periods, but rather reports average changes in outcomes from period t to (g-anticipation-1) for a particular group relative to its comparison group. This is analogous to what is often reported in event study regressions.

Using a varying base period results in an estimate of ATT(g,t) being reported in the period immediately before treatment. Using a universal base period nor-

malizes the estimate in the period right before treatment (or earlier when the user allows for anticipation) to be equal to 0, but one extra estimate in an earlier period.

`print_details` Whether or not to show details/progress of computations. Default is `FALSE`.

`faster_mode` This option enables a faster version of `did`, optimizing computation time for large datasets by improving data management within the package. The default is set to `TRUE`. Both modes produce identical results up to numerical precision; while the difference is minimal for small datasets, the speedup is substantial for large ones.

`pl` Whether or not to use parallel processing

`cores` The number of cores to use for parallel processing

`call` Function call to att_gt

Value

a `DIDparams` object

`print.AGGTEobj` *print.AGGTEobj*

Description

prints value of a `AGGTEobj` object

Usage

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

Arguments

`x` a `AGGTEobj` object

`...` extra arguments

print.MP *print.MP*

Description

prints value of a MP object

Usage

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

Arguments

x	a MP object
...	extra arguments

process_attgt *Process Results from* compute.att_gt()

Description

Process Results from compute.att_gt()

Usage

```
process_attgt(attgt.list)
```

Arguments

attgt.list	list of results from compute.att_gt()

Value

list with elements:

group	which group a set of results belongs to
tt	which time period a set of results belongs to
att	the group time average treatment effect

reset.sim *reset.sim*

Description

a function to create a "reasonable" set of parameters to create simulated panel data that obeys a parallel trends assumption. In particular, it provides parameters where the the effect of participating in the treatment is equal to one in all post-treatment time periods.

After calling this function, the user can change particular values of the parameters in order to generate dynamics, heterogeneous effects across groups, etc.

Usage

```
reset.sim(time.periods = 4, n = 5000, ipw = TRUE, reg = TRUE)
```

Arguments

time.periods	The number of time periods to include
n	The total number of observations
ipw	If TRUE, sets parameters so that DGP is compatible with recovering ATT(g,t)'s using IPW (i.e., where logit that just includes a linear term in X works). If FALSE, sets parameters that will be incompatible with IPW. Either way, these parameters can be specified by the user if so desired.
reg	If TRUE, sets parameters so that DGP is compatible with recovering ATT(g,t)'s using regressions on untreated untreated potential outcomes. If FALSE, sets parameters that will be incompatible with using regressions (i.e., regressions that include only linear term in X). Either way, these parameters can be specified by the user if so desired.

Value

list of simulation parameters

sim *sim*

Description

An internal function that builds simulated data, computes ATT(g,t)'s and some aggregations. It is useful for testing the inference procedures in the did function.

Usage

```
sim(
  sp_list,
  ret = NULL,
  bstrap = TRUE,
  cband = TRUE,
  control_group = "nevertreated",
  xformla = ~X,
  est_method = "dr",
  clustervars = NULL,
  panel = TRUE
)
```

Arguments

sp_list	A list of simulation parameters. See `reset.sim` to generate some default values for parameters
ret	which type of results to return. The options are `Wpval` (returns 1 if the p-value from a Wald test that all pre-treatment ATT(g,t)'s are equal is less than .05), `cband` (returns 1 if a uniform confidence band covers 0 for groups and times), `simple` (returns 1 if, using the simple treatment effect aggregation results in rejecting that this aggregated treatment effect parameter is equal to 0), `dynamic` (returns 1 if the uniform confidence band from the dynamic treatment effect aggregation covers 0 in all pre- and post-treatment periods). The default value is NULL, and in this case the function will just return the results from the call to `att_gt`.
bstrap	whether or not to use the bootstrap to conduct inference (default is TRUE)
cband	whether or not to compute uniform confidence bands in the call to `att_gt` (the default is TRUE)
control_group	Whether to use the "nevertreated" comparison group (the default) or the "notyet-treated" as the comparison group
xformla	Formula for covariates in `att_gt` (default is `~X`)
est_method	Which estimation method to use in `att_gt` (default is "dr")
clustervars	Any additional variables which should be clustered on
panel	whether to simulate panel data (the default) or otherwise repeated cross sections data

Value

When `ret=NULL`, returns the results of the call to `att_gt`, otherwise returns 1 if the specified test rejects or 0 if not.

summary.AGGTEobj *Summary Aggregate Treatment Effect Parameter Objects*

Description

A function to summarize aggregated treatment effect parameters.

Usage

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

Arguments

object	an AGGTEobj object
...	other arguments

summary.MP *summary.MP*

Description

prints a summary of a MP object

Usage

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

Arguments

object	an MP object
...	extra arguments

summary.MP.TEST *summary.MP.TEST*

Description

print a summary of test results

Usage

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

Arguments

object	an MP.TEST object
...	other variables

test.mboot *Multiplier Bootstrap for Conditional Moment Test*

Description

A slightly modified multiplier bootstrap procedure for the pre-test of the conditional parallel trends assumption

Usage

```
test.mboot(inf.func, DIDparams, cores = 1)
```

Arguments

inf.func	an influence function
DIDparams	DIDparams object
cores	Unused; retained for backward compatibility. The multiplier bootstrap is computed with vectorized matrix operations in a single process, so this argument has no effect here. In conditional_did_pretest, cores parallelizes only Step 1 (computing the test statistic).

Value

list

bres	CvM test statistics for each bootstrap iteration
crit.val	critical value for CvM test statistic

`tidy.AGGTEobj` *Tidy an AGGTEobj into a data frame*

Description

Returns a tidy data frame of aggregated treatment effect estimates from an `aggte()` result.

Usage

```
## S3 method for class 'AGGTEobj'
tidy(x, ...)
```

Arguments

`x`	a model of class AGGTEobj produced by the `aggte()` function
`...`	Additional arguments to tidying method.

Details

The key distinction between `conf.low`/`conf.high` and `point.conf.low`/`point.conf.high` is that the former accounts for multiple testing across all estimates (simultaneous coverage), while the latter provides marginal (per-estimate) coverage only. Use the simultaneous bands when you want to make joint inferences across all event times or groups.

Value

A data frame whose columns depend on `type`:

type the aggregation type: `"simple"`, `"dynamic"`, `"group"`, or `"calendar"`

term label for each estimate

estimate point estimate

std.error standard error

statistic t-statistic (`estimate / std.error`)

p.value two-sided pointwise p-value (`2 * (1 - pnorm(abs(statistic)))`). Marginal per-estimate; does **not** account for multiple testing across event times or groups.

conf.low, conf.high simultaneous confidence band limits. When `bstrap=TRUE` and `cband=TRUE` these use the bootstrap uniform critical value (`crit.val.egt`); otherwise they equal the pointwise intervals. For `type="simple"` and the overall average row of `type="group"`, a single scalar is returned so simultaneous and pointwise coincide.

point.conf.low, point.conf.high pointwise confidence interval limits always using `qnorm(1 - alp/2)`.

`tidy.MP` *Tidy an MP object into a data frame*

Description

Returns a tidy data frame of group-time average treatment effect estimates from an `att_gt()` result.

Usage

```
## S3 method for class 'MP'
tidy(x, ...)
```

Arguments

`x`	a model of class MP produced by the `att_gt()` function
`...`	Additional arguments to tidying method.

Value

A data frame with one row per ATT(g,t) estimate and columns:

term ATT(g,t) label

group the treatment cohort g

time the time period t

estimate the ATT(g,t) point estimate

std.error standard error

statistic t-statistic (`estimate / std.error`)

p.value two-sided pointwise p-value (`2 * (1 - pnorm(abs(statistic)))`). Marginal per-estimate; does **not** account for multiple testing across ATT(g,t) cells.

conf.low, conf.high simultaneous confidence band limits, using the bootstrap uniform critical value when `bstrap=TRUE` and `cband=TRUE`, otherwise pointwise

point.conf.low, point.conf.high pointwise confidence interval limits using `qnorm(1 - alp/2)`, always

`trimmer` *trimmer*

Description

A utility function to find observations that appear to violate support conditions. This function is not called anywhere in the code, but it is just useful for debugging some common issues that users run into.

Usage

```
trimmer(
  g,
  tname,
  idname,
  gname,
  xformla,
  data,
  control_group = "notyettreated",
  threshold = 0.999
)
```

Arguments

`g`	is a particular group (below I pass in 2009)
`tname`	The name of the column containing the time periods
`idname`	The individual (cross-sectional unit) id name
`gname`	The name of the variable in `data` that contains the first period when a particular observation is treated. This should be a positive number for all observations in treated groups. It defines which "group" a unit belongs to. It should be 0 for units in the untreated group.
`xformla`	A formula for the covariates to include in the model. It should be of the form `~ X1 + X2`. Default is NULL which is equivalent to `xformla=~1`. This is used to create a matrix of covariates which is then passed to the 2x2 DID estimator chosen in `est_method`. For time-varying covariates: (1) With balanced panel data, in each 2x2 comparison, the covariates are taken to be the value of the covariates in the earlier time period, and all of the underlying computations involve changes in Y as a function of those values of covariates. (2) With repeated cross sections data and unbalanced panel data, the covariates are taken from each time period and computations involve Y_post conditional on X_post minus Y_pre conditional on X_pre. A byproduct of this is that, with balanced panel data and in the presence of time-varying covariates, it is possible to get different numerical results according to whether or not `allow_unbalanced_panel=TRUE` or `FALSE`.
`data`	The name of the data.frame that contains the data

`control_group` Which units to use as the control group. The default is "nevertreated" which sets the control group to be the group of units that never participate in the treatment. This group does not change across groups or time periods. The other option is to set `group="notyettreated"`. In this case, the control group is set to the group of units that have not yet participated in the treatment in that time period. This includes all never treated units, but it includes additional units that eventually participate in the treatment, but have not participated yet.

`threshold` the cutoff for which observations are flagged as likely violators of the support condition.

Value

list of ids of observations that likely violate support conditions

Index