Package ‘DIDmultiplegt’

July 16, 2024

Type Package

Title Estimators DID with Multiple Groups and Periods

Version 2.0.0

Maintainer Diego Ciccia <diego.ciccia@kellogg.northwestern.edu>

Description Estimators of Difference-in-Differences based on de Chaisemartin and D'Haultfoeuille.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R(>= 3.4.0)

Imports dplyr (>= 1.0.0), fixest (>= 0.6.0), plotrix, sampling, stats, parallel, magrittr, stringr, DIDmultiplegtDYN, DIDHAD

Author Diego Ciccia [aut, cre],
Felix Knau [aut],
Mélitine Malezieux [aut],
Doulo Sow [aut],
Shuo Zhang [aut],
Clément de Chaisemartin [aut]

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2024-07-16 21:10:11 UTC

Contents

did_multiplegt 2
did_multiplegt_old 3
wagepan_mgt 7

Index **8**

`did_multiplegt` *Main function*

Description

Library of Estimators in Difference-in-Difference (DID) designs with multiple groups and periods.

Usage

```
did_multiplegt(mode, ...)
```

Arguments

`mode`	("dyn", "had", "old") Estimator selector. The `dyn` mode calls did_multiplegt_dyn, the `had` mode calls did_had and the `old` mode calls did_multiplegt_old.
`...`	Options passed to specified estimator. For more details on allowed options, check out the command-specific documentation: did_multiplegt_dyn, did_had, did_multiplegt_old.

Overview

`did_multiplegt` wraps in a single command all the estimators from de Chaisemartin and D'Haultfoeuille. Depending on the mode argument, this command can be used to call the following estimators.

did_multiplegt_dyn. In `dyn` mode, the command computes the DID event-study estimators introduced in de Chaisemartin and D'Haultfoeuille (2024a). This mode can be used both with a binary and staggered (absorbing) treatment and a non-binary treatment (discrete or continuous) that can increase or decrease multiple times. The estimator is also robust to heterogeneous effects of the current and lagged treatments. Lastly, it can be used with data where the panel st is unblanced or more disaggregated than the group level.

did_had. In `had` mode, the command computes the DID estimator introduced in de Chaisemartin and D'Haultfoeuille (2024b). This mode estimates the effect of a treatment on an outcome in a heterogeneous adoption design (HAD) with no stayers but some quasi stayers.

did_multiplegt_old. In `old` mode, the command computes the DID estimators introduced in de Chaisemartin and D'Haultfoeuille (2020). This mode corresponds to the old version of the did_multiplegt command. Specifically, it can be used to estimate DID_M, i.e. the average across t and d of the treatment effects of groups that have treatment d at $t-1$ and change their treatment at t, using groups that have treatment d at $t-1$ and t as controls. This mode could also be used to compute event-study estimates, but we strongly suggest to use the `dyn` mode, since it is way faster and includes comprehensive estimation and post-estimation support.

References

de Chaisemartin, C and D'Haultfoeuille, X (2020). American Economic Review, vol. 110, no. 9. Two-Way Fixed Effects Estimators with Heterogeneous Treatment Effects.

de Chaisemartin, C and D'Haultfoeuille, X (2024a). Review of Economics and Statistics, 1-45. Difference-in-Differences Estimators of Intertemporal Treatment Effects.

de Chaisemartin, C and D'Haultfoeuille, X (2024b). Two-way Fixed Effects and Differences-in-Differences Estimators in Heterogeneous Adoption Designs.

Vella, F. and Verbeek, M. 1998. Journal of Applied Econometrics 13(2), 163–183. Whose wages do unions raise? a dynamic model of unionism and wage rate determination for young men.

Examples

```
# Test all modes using Vella and Verbeek (1998) data:
data("wagepan_mgt")
wagepan_mgt$X <- runif(n=nrow(wagepan_mgt)) * (wagepan_mgt$year >= 1983)
Y = "lwage"
G = "nr"
T = "year"
D = "union"
X = "X"
did_multiplegt(mode = "old", wagepan_mgt, Y, G, T, D)
did_multiplegt(mode = "dyn", wagepan_mgt, Y, G, T, D, graph_off = TRUE)
did_multiplegt(mode = "had", wagepan_mgt, Y, G, T, X, graph_off = TRUE)
```

`did_multiplegt_old` *did_multiplegt_old*

Description

Estimates the effect of a treatment on an outcome, in sharp DID designs with multiple groups and periods.

Usage

```
did_multiplegt_old(
  df,
  Y,
  G,
  T,
  D,
  controls = c(),
  placebo = 0,
  dynamic = 0,
  threshold_stable_treatment = 0,
  recat_treatment = NULL,
  trends_nonparam = NULL,
  trends_lin = NULL,
  brep = 0,
  cluster = NULL,
  covariance = FALSE,
  average_effect = NULL,
  parallel = FALSE
)
```

Arguments

- `df` — the data frame for input
- `Y` — the name of Y variable
- `G` — the name of group variable
- `T` — the name of time variable
- `D` — the name of treatment variable
- `controls` — the list of names of control variables, empty if not specified
- `placebo` — the number of placebo estimators to be estimated. Placebo estimators compare switchers' and non-switchers' outcome evolution before switchers' treatment changes. Under the parallel trends assumption underlying the DID_M estimator, the placebo estimators should not significantly differ from 0. The number of placebos requested can be at most equal to the number of time periods in the data minus 2.
- `dynamic` — the number of dynamic treatment effects to be estimated. This option should only be used in staggered adoption designs, where each group's treatment is weakly increasing over time, and when treatment is binary. The estimators of dynamic effects are similar to the DID_M estimator, except that they make use of long differences of the outcome (e.g. from $t-1$ to $t+1$) rather than first differences. The number of dynamic effects requested can be at most equal to the number of time periods in the data minus 2.
- `threshold_stable_treatment` — this option may be useful when the treatment is continuous, or takes a large number of values. The DIDM estimator uses as controls groups whose treatment does not change between consecutive time periods. With a continuous treatment, there may not be any pair of consecutive time periods between which the treatment of at least one group remains perfectly stable. For instance, if the treatment is rainfall and one uses a county $\times$ year data set, there is probably not a single county*year whose rainfall is exactly the same as in the same county in the previous year. Then, one needs to specify the `threshold_stable_treatment = #` option, with # a positive real number. For each pair of consecutive time periods, the command will use counties whose rainfall changed in absolute value by less than # as controls. # should be large enough so that there are counties whose rainfall levels change by less than # between two consecutive years, but it should be small enough so that a change in rainfall of # would be unlikely to affect the outcome.
- `recat_treatment` — pools some values of the treatment together when determining the groups whose outcome evolution are compared. This option may be useful when the treatment takes a large number of values, and some very rare in the sample. For instance, assume that treatment D takes the values 0, 1, 2, 3, and 4, but few observations have a treatment equal to 2. Then, there may be a pair of consecutive time periods where one group goes from 2 to 3 units of treatment, but no group has a treatment equal to 2 at both dates. To avoid loosing that observation, one can create a variable `D_recat` that takes the same value when D=1 or 2 (e.g.: `D_recat=(D>=1)+(D>=3)+(D>=4)`), and then specify the `recat_treatment = "D_recat"` option. Then, the command can also use groups with a treatment

equal to 1 at two consecutive dates as controls for groups going from 2 to 3 units of treatment, thus making it more likely that all switchers have a non-empty set of controls.

`trends_nonparam`
: when this option is specified, time fixed effects interacted with varlist are included in the estimation. varlist can only include one categorical variable. For instance, if one works with county $\times$ year data set and one wants to allow for state-specific trends, then one should write `trends_nonparam = "state"`, where state is the state identifier.

`trends_lin`
: when this option is specified, linear time trends interacted with varlist are included in the estimation. varlist can only include one categorical variable. For instance, if one works with a year data set and one wants to allow for village-specific linear trends, one should write `trends_lin = "village"`, where village is the village identifier. The `trends_nonparam= varlist` and `trends_lin = varlist` cannot be specified at the same time.

`brep`
: The number of bootstrap replications to be used in the computation of estimators' standard errors. If the option is specified, `did_multiplegt_old` returns a graph with all the estimated treatment effects and placebos, and their 95 % confidence intervals constructed using a normal approximation. Otherwise, the command does not compute estimators' standard errors. If the option is specified, it plots a graph with all the estimated treatment effects and placebos, and their 95 % confidence intervals constructed using a normal approximation.

`cluster`
: the standard errors of the estimators using a block bootstrap at the varname level. Only one clustering variable is allowed.

`covariance`
: if this option and the `brep = #` option are specified, the command computes the covariances between all the pairs of instantaneous and dynamic effects requested, and between all the pairs of placebos requested. This option can be useful to assess whether some average of the instantaneous and dynamic effects is statistically significant. For instance, assume that you estimate the instantaneous effect, effect_0, and one dynamic effect, effect_1. You would assess whether $2/3$ effect_0+ $1/3$ effect_1, a weighted average of those two effects, is statistically significant. You can specify the covariances option, use the fact that Var($2/3$ effect_0+$1/3$ effect_1)=4/9V(effect_0)+1/9V(effect_1)+4/9cov(effect_0,effect_1) to compute the standard error of $2/3$ effect_0+$1/3$ effect_1, and finally assess if this average effect is significant. This option can also be useful to run an F-test of whether the placebos are all equal to 0, when several placebos are requested.

`average_effect`
: if that option is specified, the command will compute an average of the instantaneous and dynamic effects requested. If `average_effect = "simple"` is specified, the command will compute ple average of the effects and its standard error. If `average_effect = "prop_number_switchers"` is specified, the command will compute an average where each effect receives a weight proportional to the number of switchers the effect to. When average_effect is specified, the covariances option also has to be specified, and the number of dynamic effects requested should be greater than or equal to 1.

`parallel`
: perform bootstrap on multicore if `TRUE`.

Value

`did_multiplegt_old` returns an object class that has the following values effect, effect of the treatment se_effect, standard error of the treatment when bootstraping N_effect, number of samples used placebo_i, estimated placebo effect i periods before switchers switch treatment, for all i in 0, 1, ..., k se_placebo_i, estimated standard error of placebo_i, if the option brep has been specified N_placebo_i, number of observations used in the estimation of placebo_i placebo_i, estimated dynamic effect i periods, for all i in 0, 1, ..., k se_placebo_i, estimated standard error of dynamic_i, if the option brep has been specified N_placebo_i, number of observations used in the estimation of dynamic_i

Overview

did_multiplegt_old estimates the effect of a treatment on an outcome, using group- (e.g. county- or state-) level panel data with multiple groups and periods. Like other recently proposed DID estimation commands (did, didimputation...), did_multiplegt can be used with a binary and staggered (absorbing) treatment. But unlike those other commands, did_multiplegt_old can also be used with a non-binary treatment (discrete or continuous) that can increase or decrease multiple times. The panel of groups may be unbalanced: not all groups have to be observed at every period (see FAQ section for more info on that). The data may also be at a more disaggregated level than the group level (e.g. individual-level wage data to measure the effect of a regional-level minimum-wage on individuals' wages).

It computes the DID_M estimator introduced in Section 4 of Chaisemartin and D'Haultfoeuille (2019), which generalizes the standard DID estimator with two groups, two periods and a binary treatment to situations with many groups,many periods and a potentially non-binary treatment. For each pair of consecutive time periods $t-1$ and t and for each value of the treatment d, the package computes a DID estimator comparing the outcome evolution among the switchers, the groups whose treatment changes from d to some other value between $t-1$ and t, to the same evolution among control groups whose treatment is equal to d both in $t-1$ and t. Then the DID_M estimator is equal to the average of those DIDs across all pairs of consecutive time periods and across all values of the treatment. Under a parallel trends assumption, DID_M is an unbiased and consistent estimator of the average treatment effect among switchers, at the time period when they switch.

The package can also compute placebo estimators that can be used to test the parallel trends assumption.

Finally, in staggered adoption designs where each group's treatment is weakly increasing over time, it can compute estimators of switchers' dynamic treatment effects, one time period or more after they have started receiving the treatment.

WARNING: To estimate event-study/dynamic effects, we strongly recommend using the much faster did_multiplegt_dyn command, available from the CRAN repository. In addition to that, did_multiplegt_dyn offers more options than did_multiplegt_old.

Examples

```
# estimating the effect of union membership on wages
# using the same panel of workers as in Vella and Verbeek (1998)
data("wagepan_mgt")
Y = "lwage"
G = "nr"
T = "year"
```

```
D = "union"
controls = c("hours")

did_multiplegt_old(wagepan_mgt, Y, G, T, D, controls)
```

`wagepan_mgt` *wagepan_mgt*

Description

A subset of data from Vella and Verbeek (1998).

Usage

```
wagepan_mgt
```

Format

‘wagepan_mgt‘ A data frame with 4,360 rows and 5 columns:

lwage Log wage.

nr Worker ID.

year Year

union Union status.

hours Annual Hours worked.

Source

<http://fmwww.bc.edu/ec-p/data/wooldridge/wagepan.des>

Index

∗ **datasets**
    wagepan_mgt, 7

did_had, *2*
did_multiplegt, 2
did_multiplegt_dyn, *2*
did_multiplegt_old, *2*, 3

wagepan_mgt, 7