

Package ‘DIDHAD’

August 18, 2025

Title Difference-in-Differences in Heterogeneous Adoption Designs with Quasi Untreated Groups

Version 2.0.0

Maintainer Diego Ciccía <diego.ciccía@kellogg.northwestern.edu>

Description Estimation of Difference-in-Differences (DiD) estimators from de Chaisemartin et al. (2025) <doi:10.48550/arXiv.2405.04465> in Heterogeneous Adoption Designs with Quasi Untreated Groups.

License MIT + file LICENSE

Imports YatchewTest (>= 1.1.0), nprobust, ggplot2, plm, rnames, stats, haven, rlang, dplyr

Author Diego Ciccía [aut, cre],
Felix Knau [aut],
Doulo Sow [aut],
Clément de Chaisemartin [aut],
Xavier D'Haultfoeuille [aut]

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2025-08-18 10:40:14 UTC

Contents

did_had	2
Index	6

did_had

Main function of the DIDHAD package

Description

Estimates the effect of a treatment on an outcome in a heterogeneous adoption design with no untreated, but some quasi-untreated groups (see de Chaisemartin et. al. (2025)).

Usage

```
did_had(
  df,
  outcome,
  group,
  time,
  treatment,
  effects = 1,
  placebo = 0,
  level = 0.05,
  kernel = "epa",
  bw_method = "mse-dpi",
  trends_lin = FALSE,
  dynamic = FALSE,
  yatchew = FALSE,
  graph_off = FALSE
)
```

Arguments

df	(data.frame) A data.frame object
outcome	(character) Outcome variable
group	(character) Group Variable
time	(character) Time variable
treatment	(character) Treatment variable
effects	(positive numeric) allows you to specify the number of effects did_had() tries to estimate. Effect ℓ is the treatment's effect at period $F - 1 + \ell$, namely ℓ periods after adoption. By default, the command estimates only 1 effect and in case you specified more effects than your data allows to estimate the number of effects is automatically adjusted to the maximum.
placebo	(nonnegative numeric) allows you to specify the number of placebo estimates did_had() tries to compute. Those placebos are constructed symmetrically to the estimators of the actual effects, except that the outcome evolution from $F - 1$ to $F - 1 + \ell$ in the actual estimator is replaced by the outcome evolution from $F - 1$ to $F - 1 - \ell$ in the placebo.

level	(positive numeric) allows you to specify (1-the level) of the confidence intervals shown by the command. By default this level is set to 0.05, thus yielding 95% level confidence intervals.
kernel	(character in "tri", "epa", "uni" or "gau") allows you to specify the kernel function used by <code>lprobust()</code> . Possible choices are triangular, epanechnikov, uniform and gaussian. By default, the program uses a epanechnikov kernel.
bw_method	(character in "mse-dpi", "mse-rot", "imse-dpi", "imse-rot", "ce-dpi", "ce-rot") allows you to specify the bandwidth selection procedure used by <code>lprobust</code> . By default, the program uses <code>mse-dpi</code> . For more details please consult the <code>lprobust</code> helpfile.
trends_lin	(logical) when this option is specified, the command allows for group-specific linear trends. This is done by using groups' outcome evolution from period $F-2$ to $F-1$ as an estimator of each group-specific linear trend, and then subtracting this trend from groups' actual outcome evolutions. Note: due to the fitting of the linear trend in periods $F-2$ to $F-1$, the number of feasible placebo estimates is reduced by 1 with this option.
dynamic	(logical) when this option is specified, effect ℓ is scaled by groups' average total treatment dose received from period F to $F-1+\ell$. Without this option, effect ℓ is scaled by groups' average treatment dose at period $F-1+\ell$. The latter normalization is appropriate if one assumes that groups' outcome at $F-1+\ell$ is only affected by their current treatment (static model). On the other hand, the former normalization is appropriate if one assumes that groups' outcome at $F-1+\ell$ can be affected by their current and past treatments (dynamic model).
yatchew	(logical) <code>yatchew</code> yields the result from a non-parametric test that the conditional expectation of the $F-1$ to $F-1+\ell$ outcome evolution given the treatment at $F-1+\ell$ is linear (Yatchew, 1997). This test is implemented using the heteroskedasticity-robust test statistic proposed in Appendix E of de Chaisemartin et al. (2025) and it is performed for all the dynamic effects and placebos computed by <code>did_had()</code> . For placebos, the null being tested is that groups' $F-1$ to $F-1-\ell$ outcome evolution is mean independent of their $F-1+\ell$ treatment, a non-parametric version of a standard pre-trends test where the $F-1$ to $F-1-\ell$ outcome evolution is regressed on groups' $F-1+\ell$ treatment. This option requires the <code>YatchewTest</code> package, which is currently available on CRAN.
graph_off	(logical) by default, <code>did_had()</code> outputs an event-study graph with the effect and placebo estimates and their confidence intervals. When specifying <code>graph_off = TRUE</code> , the graph is suppressed.

Value

An list object of `did_had` class. The object contains the estimation results, as well as the selected arguments of the function and a `ggplot` graph with the event study estimates.

Overview

`did_had()` estimates the effect of a treatment on an outcome in a heterogeneous adoption design (HAD) with no untreated, but some quasi-untreated groups. HADs are designs where all groups

are untreated in the first period, and then some groups receive a strictly positive treatment dose at a period F , which has to be the same for all treated groups (with variation in treatment timing, the `did_multipligt_dyn` package may be used). Therefore, there is variation in treatment intensity, but no variation in treatment timing.

HADs without untreated groups are designs where all groups receive a strictly positive treatment dose at period F . Then, one cannot use untreated units to recover the counterfactual outcome evolution that treated groups would have experienced from before to after F , without treatment. To circumvent this, `did_had()` implements the estimator from de Chaisemartin et. al. (2025) which uses so-called "quasi-untreated groups" as the control group. Quasi-untreated groups are groups that receive a "small enough" treatment dose at F to be regarded as "as good as untreated". Therefore, `did_had()` can only be used if there are groups with a treatment dose "close to zero". Formally, the command checks the presence of quasi untreated groups via the test proposed in section 3.3 of de Chaisemartin et al. (2025). This test is automatically performed once for each event-study effect and results are reported in the output table. If the results are the same for each event-study effect, this indicates that the treatment changes only once.

The command makes use of the `lprobust` command by Calonico, Cattaneo and Farrell (2019) to determine an optimal bandwidth, i.e. a treatment dose below which groups can be considered as quasi-untreated. To estimate the treatment's effect, the command starts by computing the difference between the change in outcome of all groups and the intercept in a local linear regression of the outcome change on the treatment dose among quasi-untreated groups. Then, that difference is scaled by groups' average treatment dose at period two. Standard errors and confidence intervals are also computed leveraging `lprobust`. We recommend that users of `did_had()` cite de Chaisemartin et. al. (2025), Calonico, Cattaneo and Farrell (2019), and Calonico, Cattaneo and Farrell (2018).

Interpreting the results from the `yatchew` option

Following Theorem 5 of de Chaisemartin et. al. (2025), in designs where there are untreated or quasi-untreated groups, under a parallel trends assumption the treatment coefficient from a regression of groups' $F - 1$ to $F - 1 + \ell$ outcome evolution on their the treatment at $F - 1 + \ell$ is unbiased for the WAS if and only if the conditional expectation of the outcome evolution from $F - 1$ to $F - 1 + \ell$ given the treatment at $F - 1 + \ell$ is linear. As a result, if the linearity hypothesis cannot be rejected, then one can unbiasedly estimate the WAS at period $F - 1 + \ell$ using the simple OLS regression described above, rather than resorting to the non-parametric estimator computed by `did_had()`.

Contacts

Github repository: [chaisemartinPackages/did_had](https://github.com/chaisemartin/did_had)

Mail: chaisemartin.packages@gmail.com

References

- de Chaisemartin, C., Ciccia, D., D'Haultfoeuille, X. and Knau, F. (2025). [Difference-in-Differences Estimators When No Unit Remains Untreated](#).
- Calonico, S., M. D. Cattaneo, and M. H. Farrell. (2019). [nprobust: Nonparametric Kernel-Based Estimation and Robust Bias-Corrected Inference](#).
- Calonico, S., M. D. Cattaneo, and M. H. Farrell. (2018). [On the Effect of Bias Estimation on Coverage Accuracy in Nonparametric Inference](#).

Yatchew, A. (1997). An elementary estimator of the partial linear model, [doi:10.1016/S0165-1765\(97\)002188](https://doi.org/10.1016/S0165-1765(97)002188).

Examples

```
# The sample data for this example can be downloaded by running:
repo <- "https://raw.githubusercontent.com/chaisemartinPackages/did_had/"
data <- haven::read_dta(paste0(repo, "main/tutorial_data.dta"))

# Estimating the effects over five periods and placebos for four pre-treatment periods,
# suppressing the graph and with a triangular kernel:

summary(did_had(df = data,
  outcome = "y",
  group = "g",
  time = "t",
  treatment = "d",
  effects = 5,
  placebo = 4,
  kernel = "tri",
  graph_off = TRUE))
```

Index

did_had, [2](#)