Package 'DIDHAD'

January 20, 2025

Title Difference-in-Differences in Heterogeneous Adoption Designs with
Quasi Stayers

Version 1.0.1

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

Description Estimation of Difference-in-Differences (DiD) estimators from de Chaise-
martin and D'Haultfoeuille (2024) <doi:10.2139/ssrn.4284811> in Heterogeneous Adoption De-
signs with no stayers but with quasi stayers.

License MIT + file LICENSE

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

Author Diego Ciccia [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 2024-07-23 15:10:07 UTC

Contents

---

did_had   *Main function of the DIDHAD package*

Description

Estimation of the effect of a treatment on an outcome in a heterogeneous adoption design with no stayers but some quasi stayers (de Chaisemartin and D'Haultfoeuille, 2024).

Usage

```
did_had(
  df,
  outcome,
  group,
  time,
  treatment,
  effects = 1,
  placebo = 0,
  level = 0.05,
  kernel = "uni",
  yatchew = FALSE,
  trends_lin = FALSE,
  dynamic = 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 `lprobust()`. Possible choices are triangular, epanechnikov, uniform and gaussian. By default, the program uses a uniform kernel.

yatchew

(logical) yatchew 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 Section 3 of de Chaisemartin and D'Haultfoeuille (2024) and it is performed for all the dynamic effects and placebos computed by did_had. This option requires the YatchewTest package, which is currently available on CRAN.

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).

graph_off

(logical) by default, did_had() outputs an event-study graph with the effect and placebo estimates and their confidence intervals. When specifying graph_off = TRUE, 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 stayers but some quasi stayers. 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_multiplegt_dyn() package may be used). Therefore, there is variation in treatment intensity, but no variation in treatment timing. HADs without stayers are designs where all groups receive a strictly positive treatment dose at period F: no group remains untreated. 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 and D'Haultfoeuille (2024) which uses so-called "quasi stayers" as the control group. Quasi stayers 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

density of groups' period-two treatment dose needs to be strictly positive at zero, something that can be assessed by plotting a kernel density estimate of that density.

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 stayers. 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-stayers. 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 and D'Haultfoeuille (2024), Calonico, Cattaneo and Farrell (2019), and Calonico, Cattaneo and Farrell (2018).

Interpreting the results from the `yatchew` **option**

Following Theorem 1 and Equation 5 of de Chaisemartin and D'Haultfoeuille (2024), in designs where there are stayers or quasi-stayers, the coefficient from a TWFE regression of Y on D in time periods $F-1$ and $F-1+\ell$ is unbiased for the Average Slope of Treated groups (AST) 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 test statistics are not statistically significant, i.e. the linearity hypothesis cannot be rejected, then one can unbiasedly estimate the $F-1$-to-$F-1+\ell$ AST using a TWFE regression as the one described above.

Contacts

Github repository: chaisemartinPackages/did_had

Mail: chaisemartin.packages@gmail.com

References

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

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.

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 triagular 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,