Package ‘did2s’

September 27, 2021

Title Two-Stage Difference-in-Differences Following Gardner (2021)

Version 0.4.0

Description Estimates Two-way Fixed Effects difference-in-differences/event-study models using the approach proposed by Gardner (2021). To avoid the problems caused by OLS estimation of the Two-way Fixed Effects model, this function first estimates the fixed effects and covariates using untreated observations and then in a second stage, estimates the treatment effects.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Depends R (>= 2.10), fixest (>= 0.9.0)

Imports dplyr, magrittr, ggplot2, glue, stats, stringr, rlang, tibble, tidyr, rsample, purrr, cli, Matrix, Rcpp, did, broom, methods, staggered, didimputation, gt

URL http://kylebutts.com/did2s/

Suggests rmarkdown, knitr, haven, testthat (>= 3.0.0)

VignetteBuilder knitr

LinkingTo RcppArmadillo, Rcpp

Config/testthat/edition 3

NeedsCompilation yes

Author Kyle Butts [aut, cre] (<https://orcid.org/0000-0002-9048-8059>), John Gardner [aut] (<https://orcid.org/0000-0002-4028-6862>), Grant McDermott [ctb] (<https://orcid.org/0000-0001-7883-8573>)

Maintainer Kyle Butts <kyle.butts@colorado.edu>

Repository CRAN

Date/Publication 2021-09-27 08:50:07 UTC
R topics documented:

- castle .................................................. 2
- df_het .................................................. 2
- df_hom .................................................. 3
- did2s .................................................... 4
- event_study ............................................. 8
- gen_data ............................................... 9

Index 11

---

castle  

Data from Cheng and Hoekstra (2013)

Description

State-wide panel data from 2000-2010 that has information on castle-doctrine, the so-called "stand-your-ground" laws that were implemented by 20 states.

Usage

castle

Format

A data frame with 550 rows and 5 variables:

- sid  state id, unit of observation
- year  time in panel data
- l_homicide  log of the number of homicides per capita
- effyear  year that castle doctrine is passed
- post  0/1 variable for when castle doctrine is active
- time_til  time relative to castle doctrine being passed into law

---

df_het

Simulated data with two treatment groups and heterogenous effects  
Generated using the following call:  
did2s::gen_data(panel = c(1990, 2020), g1 = 2000, g2 = 2010, g3 = 0, te1 = 2, te2 = 1,  
tef = c(0.05, 0.15, 0.15), te_m = 0)

Description

Simulated data with two treatment groups and heterogenous effects
Generated using the following call:  
did2s::gen_data(panel = c(1990, 2020), g1 = 2000, g2 = 2010, g3 = 0, te1 = 2, te2 = 1,  
tef = c(0.05, 0.15, 0.15), te_m = 0)
**df_hom**

**Usage**

`df_het`

**Format**

A data frame with 31000 rows and 15 variables:

- `unit` individual in panel data
- `year` time in panel data
- `g` the year that treatment starts
- `dep_var` outcome variable
- `treat` T/F variable for when treatment is on
- `rel_year` year relative to treatment start. Inf = never treated.
- `rel_year_binned` year relative to treatment start, but <=-6 and =>6 are binned.
- `unit_fe` Unit FE
- `year_fe` Year FE
- `error` Random error component
- `te` Static treatment effect = te
- `te_dynamic` Dynamic treatment effect = te_m
- `state` State that unit is in
- `group` String name for group

---

Simulated data with two treatment groups and homogenous effects

*Generated using the following call: did2s::gen_data(panel = c(1990, 2020), g1 = 2000, g2 = 2010, g3 = 0, te1 = 2, te2 = 2, te3 = 0, te_m1 = 0, te_m2 = 0, te_m3 = 0)*

---

**Description**

Simulated data with two treatment groups and homogenous effects

Generated using the following call: `did2s::gen_data(panel = c(1990, 2020), g1 = 2000, g2 = 2010, g3 = 0, te1 = 2, te2 = 2, te3 = 0, te_m1 = 0, te_m2 = 0, te_m3 = 0)`

**Usage**

`df_hom`
Format

A data frame with 31000 rows and 15 variables:

- **unit**: individual in panel data
- **year**: time in panel data
- **g**: the year that treatment starts
- **dep_var**: outcome variable
- **treat**: T/F variable for when treatment is on
- **rel_year**: year relative to treatment start. Inf = never treated.
- **rel_year_binned**: year relative to treatment start, but <=-6 and >=6 are binned.
- **unit_fe**: Unit FE
- **year_fe**: Year FE
- **error**: Random error component
- **te**: Static treatment effect = te
- **te_dynamic**: Dynamic treatment effect = te_m
- **group**: String name for group
- **state**: State that unit is in
- **weight**: Weight from runif()

---

**did2s**

*Calculate two-stage difference-in-differences following Gardner (2021)*

---

**Description**

Calculate two-stage difference-in-differences following Gardner (2021)

**Usage**

did2s(
  data,
  yname,
  first_stage,
  second_stage,
  treatment,
  cluster_var,
  weights = NULL,
  bootstrap = FALSE,
  n_bootstraps = 250,
  return_bootstrap = FALSE,
  verbose = TRUE
)
Arguments

data: The dataframe containing all the variables
yname: Outcome variable
first_stage: Fixed effects and other covariates you want to residualize with in first stage. Formula following \texttt{fixest::feols}. Fixed effects specified after ";\". 
second_stage: Second stage, these should be the treatment indicator(s) (e.g., treatment variable or event-study leads/lags). Formula following \texttt{fixest::feols}. Use \texttt{i()} for factor variables, see \texttt{fixest::i}.
treatment: A variable that = 1 if treated, = 0 otherwise
cluster_var: What variable to cluster standard errors. This can be IDs or a higher aggregate level (state for example)
weights: Optional. Variable name for regression weights.
bootstrap: Optional. Should standard errors be calculated using bootstrap? Default is \texttt{FALSE}.
n_bootstraps: Optional. How many bootstraps to run. Default is 250.
return_bootstrap: Optional. Logical. Will return each bootstrap second-stage estimate to allow for manual use, e.g., percentile standard errors and empirical confidence intervals.
verbose: Optional. Logical. Should information about the two-stage procedure be printed back to the user? Default is \texttt{TRUE}.

Value

\texttt{fixest} object with adjusted standard errors (either by formula or by bootstrap). All the methods from \texttt{fixest} package will work, including \texttt{fixest::esttable} and \texttt{fixest::coefplot}.

Examples

Load example dataset which has two treatment groups and homogeneous treatment effects

# Load Example Dataset
data("df_hom")

\textbf{Static TWFE:}

You can run a static TWFE fixed effect model for a simple treatment indicator

\begin{verbatim}
static <- did2s(df_hom, 
yname = "dep_var", treatment = "treat", cluster_var = "state", 
first_stage = \~ 0 | unit + year, 
second_stage = \~ i(treat, ref=FALSE))
\end{verbatim}

\texttt{\#> Running Two-stage Difference-in-Differences}

\texttt{\#> \cdot first stage formula \~ 0 | unit + year\}

\texttt{\#> \cdot second stage formula \~ i(treat, ref = FALSE)\}

\texttt{\#> \cdot The indicator variable that denotes when treatment is on is \textquotesingle{treat} \}

\texttt{\#> \cdot Standard errors will be clustered by \textquotesingle{state}\}
Event Study:

Or you can use relative-treatment indicators to estimate an event study estimate.

```r
es <- did2s(df_hom,
    yname = "dep_var", treatment = "treat", cluster_var = "state",
    first_stage = ~ 0 | unit + year,
    second_stage = ~ i(rel_year, ref = c(-1, Inf)))
```

```
Running Two-stage Difference-in-Differences

• first stage formula `~ 0 | unit + year`
• second stage formula `~ i(rel_year, ref = c(-1, Inf))`
• The indicator variable that denotes when treatment is on is `treat`
• Standard errors will be clustered by `state`
```

```r
fixest::esttable(es)

<table>
<thead>
<tr>
<th>rel_year</th>
<th>Coef.</th>
<th>Std. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>-20</td>
<td>-0.0335</td>
<td>(0.0697)</td>
</tr>
<tr>
<td>-19</td>
<td>0.0581</td>
<td>(0.0588)</td>
</tr>
<tr>
<td>-18</td>
<td>0.0348</td>
<td>(0.0578)</td>
</tr>
<tr>
<td>-17</td>
<td>0.0236</td>
<td>(0.0670)</td>
</tr>
<tr>
<td>-16</td>
<td>0.0115</td>
<td>(0.0542)</td>
</tr>
<tr>
<td>-15</td>
<td>-0.0148</td>
<td>(0.0769)</td>
</tr>
<tr>
<td>-14</td>
<td>0.1150</td>
<td>(0.0613)</td>
</tr>
<tr>
<td>-13</td>
<td>-0.0108</td>
<td>(0.0720)</td>
</tr>
<tr>
<td>-12</td>
<td>-0.0727</td>
<td>(0.0635)</td>
</tr>
<tr>
<td>-11</td>
<td>0.0666</td>
<td>(0.0559)</td>
</tr>
<tr>
<td>-10</td>
<td>0.0396</td>
<td>(0.0382)</td>
</tr>
<tr>
<td>-9</td>
<td>-0.0109</td>
<td>(0.0379)</td>
</tr>
<tr>
<td>-8</td>
<td>0.0105</td>
<td>(0.0388)</td>
</tr>
<tr>
<td>-7</td>
<td>-0.0001</td>
<td>(0.0445)</td>
</tr>
<tr>
<td>-6</td>
<td>-0.0829*</td>
<td>(0.0388)</td>
</tr>
<tr>
<td>-5</td>
<td>0.0189</td>
<td>(0.0429)</td>
</tr>
<tr>
<td>-4</td>
<td>-0.0664</td>
<td>(0.0437)</td>
</tr>
<tr>
<td>-3</td>
<td>-0.0144</td>
<td>(0.0302)</td>
</tr>
<tr>
<td>-2</td>
<td>0.0223</td>
<td>(0.0442)</td>
</tr>
<tr>
<td>0</td>
<td>2.117***</td>
<td>(0.0622)</td>
</tr>
<tr>
<td>1</td>
<td>1.857***</td>
<td>(0.0720)</td>
</tr>
</tbody>
</table>
```
Example from Cheng and Hoekstra (2013):
Here’s an example using data from Cheng and Hoekstra (2013)

```r
# Castle Data
castle <- haven::read_dta("https://github.com/scunning1975/mixtape/raw/master/castle.dta")

did2s(
  data = castle,
  yname = "l_homicide",
  first_stage = ~ 0 | sid + year,
  second_stage = ~ i(post, ref=0),
  treatment = "post",
  cluster_var = "state", weights = "popwt"
)
```

```
#> Running Two-stage Difference-in-Differences
#> • first stage formula ~ 0 | sid + year
#> • second stage formula ~ i(post, ref = 0)
#> • The indicator variable that denotes when treatment is on is 'post'
#> • Standard errors will be clustered by 'state'
#> OLS estimation, Dep. Var.: l_homicide
#> Observations: 550
#> Standard-errors: Custom
```
event_study

Estimate event-study coefficients using TWFE and 5 proposed improvements.

Description

Uses the estimation procedures recommended from Borusyak, Jaravel, Spiess (2021); Callaway and Sant’Anna (2020); Gardner (2021); Roth and Sant’Anna (2021); Sun and Abraham (2020)

Usage

```
event_study(
  data,            # The dataframe containing all the variables
  yname,          # Variable name for outcome variable
  idname,         # Variable name for unique unit id
  gname,          # Variable name for unit-specific date of initial treatment (never-treated should be zero or NA)
  tname,          # Variable name for calendar period
  xformla = NULL, # A formula for the covariates to include in the model. It should be of the form ~ X1 + X2. Default is NULL.
  horizon = NULL, # Numeric. Vector of length 2. First element is min and second element is max of event_time to plot
  weights = NULL  # Variable name for estimation weights. This is used in estimating Y(0) and also augments treatment effect weights
)
```

```
plot_event_study(out, seperate = TRUE, horizon = NULL)
```

Arguments

- **data**: The dataframe containing all the variables
- **yname**: Variable name for outcome variable
- **idname**: Variable name for unique unit id
- **gname**: Variable name for unit-specific date of initial treatment (never-treated should be zero or NA)
- **tname**: Variable name for calendar period
- **xformla**: A formula for the covariates to include in the model. It should be of the form ~ X1 + X2. Default is NULL.
- **horizon**: Numeric. Vector of length 2. First element is min and second element is max of event_time to plot
- **weights**: Variable name for estimation weights. This is used in estimating Y(0) and also augments treatment effect weights
- **out**: Output from `event_study()`
- **seperate**: Logical. Should the estimators be on separate plots? Default is TRUE.

---

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|----------|
| post::1  | 0.075142   | 0.03538 | 2.1239   | 0.034127 * |

---

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

RMSE: 263.4  Adj. R2: 0.052465
Value

`event_study` returns a data.frame of point estimates for each estimator

`plot_event_study` returns a ggplot object that can be fully customized

Examples

```r
out = event_study(
  data = did2s::df_het, yname = "dep_var", idname = "unit",
  tname = "year", gname = "g"
)
plot_event_study(out)
```

---

## gen_data

**Generate TWFE data**

### Description

Generate TWFE data

### Usage

```r
gen_data(
  g1 = 2000, 
  g2 = 2010, 
  g3 = 0, 
  panel = c(1990, 2020),
  te1 = 2,
  te2 = 2,
  te3 = 2,
  te_m1 = 0,
  te_m2 = 0,
  te_m3 = 0
)
```

### Arguments

- **g1**: treatment date for group 1. For no treatment, set `g = 0`.
- **g2**: treatment date for group 2. For no treatment, set `g = 0`.
- **g3**: treatment date for group 3. For no treatment, set `g = 0`.
- **panel**: numeric vector of size 2, start and end years for panel
- **te1**: treatment effect for group 1. Will ignore for that group if `g = 0`.
- **te2**: treatment effect for group 1. Will ignore for that group if `g = 0`.
- **te3**: treatment effect for group 1. Will ignore for that group if `g = 0`. 
Dataframe of generated data

Examples

# Homogeneous treatment effect
df_hom <- gen_data(panel = c(1990, 2020),
g1 = 2000, g2 = 2010, g3 = 0,
te1 = 2, te2 = 2, te3 = 0,
te_m1 = 0, te_m2 = 0, te_m3 = 0)

# Heterogeneous treatment effect
df_het <- gen_data(panel = c(1990, 2020),
g1 = 2000, g2 = 2010, g3 = 0,
te1 = 2, te2 = 1, te3 = 0,
te_m1 = 0.05, te_m2 = 0.15, te_m3 = 0)
Index

* datasets
  castle, 2
  df_het, 2
  df_hom, 3

castle, 2
df_het, 2
df_hom, 3
did2s, 4

event_study, 8
event_study(), 8

fixest::coefplot, 5
fixest::esttable, 5
fixest::feols, 5
fixest::i, 5

gen_data, 9

plot_event_study(event_study), 8