

Package ‘fixes’

May 10, 2026

Type Package

Title Tools for Creating and Visualizing Fixed-Effects Event Study Models

Version 0.8.1

Date 2026-05-10

Description Provides functions for creating, analyzing, and visualizing event study models using fixed-effects regression. Supports staggered adoption, multiple confidence intervals, flexible clustering, and panel/time transformations in a simple workflow.

Depends R (>= 4.1.0)

Imports dplyr, ggplot2, fixest, broom, tibble, rlang, Rcpp

License MIT + file LICENSE

Encoding UTF-8

Suggests knitr, rmarkdown, haven, testthat, plotly, tidyr, did, didimputation

VignetteBuilder knitr

URL <https://github.com/yo5uke/fixes>

BugReports <https://github.com/yo5uke/fixes/issues>

Config/roxygen2/version 8.0.0

RoxygenNote 7.3.3

LinkingTo Rcpp

NeedsCompilation yes

Author Yosuke Abe [aut, cre]

Maintainer Yosuke Abe <yosuke.abe0507@gmail.com>

Repository CRAN

Date/Publication 2026-05-10 05:30:02 UTC

Contents

autoplot.es_result	2
plot_att_gt	3
plot_es	4
plot_es_interactive	5
run_es	7

Index	10
--------------	-----------

autoplot.es_result	<i>Autoplot for event-study results</i>
--------------------	---

Description

S3 method that plots an `es_result` (from `run_es()`). It forwards arguments to `plot_es`.

Usage

```
## S3 method for class 'es_result'
autoplot(object, ci_level = 0.95, type = "ribbon", ...)
```

Arguments

<code>object</code>	An <code>es_result</code> returned by <code>run_es()</code> .
<code>ci_level</code>	Confidence level (numeric, e.g., 0.95). Passed to <code>plot_es()</code> .
<code>type</code>	Plot type: "ribbon" (default) or "errorbar". Passed to <code>plot_es()</code> .
<code>...</code>	Additional arguments forwarded to <code>plot_es</code> .

Value

A `ggplot2` `ggplot` object.

Examples

```
# res <- run_es(...)
# ggplot2::autoplot(res, ci_level = 0.95, type = "ribbon")
```

plot_att_gt

*Plot the ATT(g,t) matrix from a Callaway-Sant'Anna event study***Description**

Visualises the full cohort-by-period ATT(g,t) matrix stored in the `att_gt` attribute of an `es_result` object produced by `run_es(estimator = "cs")`. Two display styles are available:

- "heatmap": a tile plot with calendar time t on the x-axis and cohort g on the y-axis, colour-filled by the point estimate. Cells whose pointwise confidence interval excludes zero are marked with a filled dot; cells that are simultaneously significant (when bootstrap data are available) are additionally marked with an open diamond.
- "facet": one panel per cohort showing ATT(g,t) over calendar time t with a pointwise confidence ribbon, mirroring the style of `plot_es`. A lighter simultaneous CI ribbon is overlaid when bootstrap data are available.

Both types draw a vertical dashed line at $t = g$ (treatment onset) for each cohort.

Usage

```
plot_att_gt(
  x,
  type = c("heatmap", "facet"),
  ci_level = 0.95,
  zero_line = TRUE,
  theme = c("bw", "minimal", "classic"),
  color = "#B25D91FF",
  fill = "#B25D91FF",
  alpha = 0.2
)

## S3 method for class 'att_gt_result'
autoplot(object, ...)
```

Arguments

<code>x</code>	An <code>es_result</code> object returned by <code>run_es(estimator = "cs")</code> , or an <code>att_gt_result</code> data frame produced by extracting <code>attr(x, "att_gt")</code> and giving it class "att_gt_result".
<code>type</code>	"heatmap" (default) or "facet".
<code>ci_level</code>	Confidence level for pointwise intervals (default 0.95).
<code>zero_line</code>	Logical; draw a horizontal reference line at zero in the "facet" display (default TRUE).
<code>theme</code>	One of "bw" (default), "minimal", or "classic".
<code>color</code>	Line and point colour used in the "facet" display (default "#B25D91FF", matching <code>plot_es</code>).

fill	Ribbon fill colour in the "facet" display (default "#B25D91FF").
alpha	Ribbon transparency in the "facet" display (default 0.2).
object	An att_gt_result object (extracted from an es_result via attr(result, "att_gt")).
...	Passed to plot_att_gt .

Value

A [ggplot](#) object.

Bootstrap annotation

When `attr(x, "bootstrap")` is present (i.e., `run_es()` was called with `bootstrap = TRUE`), both plot types add simultaneous inference overlays sourced from the (g,t)-level bootstrap object.

See Also

[plot_es](#), [run_es](#)

Examples

```
## Not run:
cs_result <- run_es(data = mydata, outcome = y, time = year,
                   timing = g, unit = id, fe = ~id + year,
                   staggered = TRUE, estimator = "cs")
plot_att_gt(cs_result)
plot_att_gt(cs_result, type = "facet")

## End(Not run)
```

plot_es

Plot event-study results with ribbons or error bars

Description

Plot event-study results with ribbons or error bars

Usage

```
plot_es(
  data,
  ci_level = 0.95,
  type = "ribbon",
  vline_val = 0,
  vline_color = "#000",
  hline_val = 0,
  hline_color = "#000",
```

```

  linewidth = 1,
  pointsize = 2,
  alpha = 0.2,
  barwidth = 0.2,
  color = "#B25D91FF",
  fill = "#B25D91FF",
  theme_style = "bw",
  show_simultaneous = FALSE
)

```

Arguments

<code>data</code>	An object of class <code>es_result</code> returned by <code>run_es()</code> .
<code>ci_level</code>	Confidence level to display (e.g., 0.95).
<code>type</code>	One of "ribbon" (default) or "errorbar".
<code>vline_val, hline_val</code>	Numeric locations for vertical/horizontal reference lines (default 0).
<code>vline_color, hline_color</code>	Colors for reference lines.
<code>linewidth, pointsize, alpha, barwidth</code>	Styling parameters for lines/points/bands/bars.
<code>color, fill</code>	Optional, override line/point color and ribbon fill.
<code>theme_style</code>	One of "bw", "minimal", or "classic" for ggplot theme.
<code>show_simultaneous</code>	Logical; if TRUE, overlays the simultaneous bootstrap CI (lighter band, alpha 0.15) alongside the standard pointwise CI (alpha 0.3), with a legend distinguishing the two bands. Requires <code>bootstrap = TRUE</code> in the originating <code>run_es()</code> call. Default FALSE.

Value

A ggplot object.

Examples

```

# Assuming `res <- run_es(...)`
# p <- plot_es(res, ci_level = 0.95, type = "ribbon")
# print(p)

```

`plot_es_interactive` *Interactive event-study plot with hover details*

Description

Creates an interactive plotly visualization of event study results with hover-over displays showing coefficients, confidence intervals, and other details.

Usage

```
plot_es_interactive(
  data,
  ci_level = 0.95,
  vline_val = 0,
  hline_val = 0,
  vline_color = "#000",
  hline_color = "#000",
  color = "#B25D91FF",
  fill = "#B25D91FF",
  alpha = 0.2,
  linewidth = 2,
  markersize = 8,
  show_ribbon = TRUE,
  show_simultaneous = FALSE,
  height = NULL,
  width = NULL
)
```

Arguments

<code>data</code>	An object of class <code>es_result</code> returned by <code>run_es()</code> .
<code>ci_level</code>	Confidence level to display (e.g., 0.95). Default is 0.95.
<code>vline_val</code>	Numeric location for vertical reference line (default 0).
<code>hline_val</code>	Numeric location for horizontal reference line (default 0).
<code>vline_color</code>	Color for vertical reference line (default "#000").
<code>hline_color</code>	Color for horizontal reference line (default "#000").
<code>color</code>	Point and line color (default "#B25D91FF").
<code>fill</code>	Ribbon/band fill color (default "#B25D91FF").
<code>alpha</code>	Ribbon transparency (default 0.2).
<code>linewidth</code>	Line width (default 2).
<code>markersize</code>	Marker size (default 8).
<code>show_ribbon</code>	Logical; if TRUE, shows confidence interval as a ribbon band (default TRUE).
<code>show_simultaneous</code>	Logical; if TRUE, overlays a second (lighter) ribbon for the simultaneous bootstrap CI and extends the hover tooltip with simultaneous CI bounds. Requires <code>bootstrap = TRUE</code> in the originating <code>run_es()</code> call. Default FALSE.
<code>height</code>	Plot height in pixels (default NULL for auto).
<code>width</code>	Plot width in pixels (default NULL for auto).

Details

The hover tooltip displays:

- Relative time to treatment

- Point estimate (coefficient)
- Confidence interval bounds
- Standard error
- P-value
- Simultaneous CI bounds (when show_simultaneous = TRUE)

Value

A plotly object that can be displayed interactively.

Examples

```
## Not run:  
# Assuming res <- run_es(...)  
plot_es_interactive(res)  
plot_es_interactive(res, ci_level = 0.99, show_ribbon = FALSE)  
plot_es_interactive(res, show_simultaneous = TRUE)  
  
## End(Not run)
```

run_es

Event Study Estimation for Panel Data

Description

Runs an event study regression on panel data, supporting both classic (universal timing) and staggered (unit-varying timing via sunab). The function builds the design (lead/lag factor or sunab), estimates with **fixest**, and returns a tidy table with metadata.

Usage

```
run_es(  
  data,  
  outcome,  
  treatment = NULL,  
  time,  
  timing,  
  fe = NULL,  
  lead_range = NULL,  
  lag_range = NULL,  
  covariates = NULL,  
  cluster = NULL,  
  weights = NULL,  
  baseline = -1L,  
  interval = 1,  
  time_transform = FALSE,  
  unit = NULL,
```

```

staggered = FALSE,
method = c("classic", "sunab"),
estimator = c("twfe", "cs", "sa", "bjs"),
control_group = c("nevertreated", "notyettreated"),
anticipation = 0L,
conf.level = 0.95,
vcov = "HC1",
vcov_args = list(),
bootstrap = FALSE,
B = 999L,
alpha = 0.05,
boot_seed = NULL
)

```

Arguments

<code>data</code>	A <code>data.frame</code> containing panel data.
<code>outcome</code>	Unquoted outcome (name or expression, e.g., $\log(y)$).
<code>treatment</code>	Unquoted treatment indicator (0/1 or logical). Used only when <code>method = "classic"</code> .
<code>time</code>	Unquoted time variable (numeric or Date).
<code>timing</code>	For <code>classic</code> : a numeric/Date (universal) or a variable (unquoted) if <code>staggered = TRUE</code> . For <code>sunab</code> : an unquoted variable with adoption time. For estimator = <code>"cs"</code> : unquoted column giving each unit's first treatment period (NA = never treated).
<code>fe</code>	One-sided fixed-effects formula, e.g., $\sim id + year$. Can be NULL for no fixed effects. Ignored when estimator = <code>"cs"</code> .
<code>lead_range, lag_range</code>	Integers for pre/post windows. If NULL, determined automatically.
<code>covariates</code>	One-sided formula of additional controls, e.g., $\sim x1 + \log(x2)$.
<code>cluster</code>	Cluster specification (one-sided formula like $\sim id + year$, a single character column name, or a vector of length <code>nrow(data)</code>).
<code>weights</code>	Observation weights (a name/one-sided formula or a numeric vector of length <code>nrow(data)</code>).
<code>baseline</code>	Integer baseline period (default -1); reference period excluded from results for both <code>"classic"</code> and <code>"sunab"</code> methods.
<code>interval</code>	Numeric spacing of the time variable (default 1; ignored internally for Dates).
<code>time_transform</code>	Logical; if TRUE, creates consecutive integer time within unit.
<code>unit</code>	Unit identifier variable (required when estimator = <code>"cs"</code> or <code>time_transform = TRUE</code>); also used for metadata when supplied.
<code>staggered</code>	Logical; if TRUE, timing is a variable (classic) or is used by sunab.
<code>method</code>	Either <code>"classic"</code> or <code>"sunab"</code> (default: <code>"classic"</code>).
<code>estimator</code>	Estimation strategy: <code>"twfe"</code> (default, existing fixest-based paths), <code>"cs"</code> for the Callaway-Sant'Anna (2021) group-time ATT estimator, or <code>"sa"</code> for the Sun-Abraham (2021) interaction-weighted estimator.

control_group	For estimator = "cs": comparison group, either "nevertreated" (default) or "notyettreated".
anticipation	For estimator = "cs": number of anticipation periods before treatment (non-negative integer, default 0L).
conf.level	Numeric vector of confidence levels (default 0.95).
vcov	VCOV type passed to <code>fixest::vcov()</code> or used via <code>broom::tidy(vcov = ...)</code> . Default "HC1".
vcov_args	List of additional arguments forwarded to <code>fixest::vcov()</code> .
bootstrap	Logical; if TRUE and estimator = "cs", compute simultaneous confidence bands via the multiplier bootstrap (Algorithm 1, Callaway and Sant'Anna 2021). Adds <code>conf_low_sim</code> and <code>conf_high_sim</code> columns to the result and stores the full (g,t)-level bootstrap object as <code>attr(result, "bootstrap")</code> . Default FALSE.
B	Integer number of bootstrap draws (default 999). Used only when bootstrap = TRUE and estimator = "cs".
alpha	Significance level for the simultaneous band (default 0.05). Note: this is independent of <code>conf.level</code> , which governs the pointwise delta-method CIs.
boot_seed	Integer seed for the bootstrap RNG; NULL (default) does not set a seed. Pass an integer for reproducible results.

Value

A data frame of class "es_result" with columns:

- term, estimate, std.error, statistic, p.value
- conf_low_XX, conf_high_XX (for each requested conf.level)
- relative_time (integer; 0 = event), is_baseline (logical; marks the reference period)

Attributes include: lead_range, lag_range, baseline, interval, call, model_formula, conf.level, N, N_units, N_treated, N_nevertreated, fe, vcov_type, cluster_vars, staggered, sunab_used.

Key Features

- One-step event study: specify outcome, treatment, time, timing, fixed effects, and (optionally) covariates.
- Switch between Classic (factor expansion) and Staggered-SAFE (method = "sunab").
- Flexible clustering, weights, and VCOV choices (e.g., `vcov = "HC1" | "HC3" | "CR2" | "iid" ...`).
- Automatic lead/lag window detection and customizable baseline period.
- Returns an "es_result" object compatible with `print()` and `autoplot()`.

Index

`autoplot.att_gt_result (plot_att_gt)`, [3](#)
`autoplot.es_result`, [2](#)

`ggplot`, [4](#)

`plot_att_gt`, [3](#), [4](#)

`plot_es`, [2–4](#), [4](#)

`plot_es_interactive`, [5](#)

`run_es`, [4](#), [7](#)