Package ‘catviz’

May 8, 2026

Title Visualizing Causal Assignment Trees for CSDiD and DR-DDD Designs

Version 0.1.1

Description Tools for constructing, labeling, and visualizing Causal Assignment Trees (CATs) in settings with staggered adoption. Supports Callaway and Sant'Anna difference-in-differences (CSDiD) and doubly robust difference-in-difference-differences (DR-DDD) designs. The package helps clarify treatment timing, never-treated vs. not-yet-treated composition, and subgroup structure, and produces publication-quality diagrams and summary tables. Current functionality focuses on data-to-node mapping, node counts, cohort-year summaries, and high-quality tree plots suitable for empirical applications prior to estimation.
Methods are based on Callaway and Sant'Anna (2021) <doi:10.1016/j.jeconom.2020.12.001>, Sant'Anna and Zhao (2020) <doi:10.1016/j.jeconom.2020.06.003>, and Kilanko (2026) <`https://github.com/VictorKilanko/catviz`>.

URL `https://github.com/VictorKilanko/catviz`

BugReports `https://github.com/VictorKilanko/catviz/issues`

Encoding UTF-8

Depends R (>= 4.1.0)

Imports dplyr, ggplot2, glue, tidyr, purrr, tibble, grid, rlang

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.3

NeedsCompilation no

Author Victor Kilanko [aut, cre]

Maintainer Victor Kilanko <`victorkilanko@gmail.com`>

Repository CRAN

Date/Publication 2026-04-01 08:00:20 UTC

Contents

`cat_att_equation` *Show ATT contrast implied by CAT nodes and design*

Description

Returns LaTeX and plain-text versions of the ATT equation that match the Causal Assignment Tree diagram.

Usage

```
cat_att_equation(
  design = c("drddd", "csdid"),
  subgroup_value = 1,
  include_never_treated = TRUE
)
```

Arguments

`design`	Character; either `"drddd"` or `"csdid"`.
`subgroup_value`	Integer; 0 or 1 selecting the subgroup for CSDiD contrast.
`include_never_treated`	Logical; if `TRUE` (default), a note about never-treated controls is included in the output.

Value

A named list with four elements:

- `text` - plain-text representation of the ATT contrast equation.
- `tex` - LaTeX math string of the ATT contrast equation.
- `nodes` - character vector naming the CAT nodes involved in the contrast.
- `note` - plain-text note about the control group composition.

Examples

```
eq <- cat_att_equation(design = "csdid")
cat(eq$text)

eq2 <- cat_att_equation(design = "drddd")
cat(eq2$text)
```

`cat_balance_plot` *Love plot for balance*

Description

Love plot for balance

Usage

```
cat_balance_plot(balance_tbl)
```

Arguments

`balance_tbl`	A balance table as returned by `cat_balance_table()`, containing columns `covariate`, `smd`, and `group`.

Value

A ggplot object showing standardized mean differences by covariate.

Examples

```
df <- data.frame(
  id   = rep(1:4, each = 3),
  year = rep(2018:2020, 4),
  g    = c(rep(2019, 6), rep(Inf, 6)),
  age  = c(25, 25, 25, 30, 30, 30, 40, 40, 40, 35, 35, 35)
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
btbl <- cat_balance_table(spec, covariates = "age")
cat_balance_plot(btbl)
```

`cat_balance_table` *Standardized mean differences across CAT nodes or design groups*

Description

Standardized mean differences across CAT nodes or design groups

Usage

```
cat_balance_table(spec, covariates, by = c("node", "design"), weight = NULL)
```

Arguments

`spec`	A `cat_spec` object. Must be labeled (via `cat_label()`) when `by = "design"`.
`covariates`	Character vector of covariate names to assess.
`by`	Character; `"node"` (default) computes SMDs across CAT nodes; `"design"` compares Treated vs Never-Treated units.
`weight`	Optional name of a weight column in `spec$data`.

Value

A tibble with one row per covariate-group combination and four columns:

- `covariate` - name of the covariate.
- `group` - CAT node label or design group ("Treated" or "Never-Treated").
- `mean` - weighted mean of the covariate within the group.
- `smd` - standardized mean difference relative to the first (reference) group.

Examples

```
df <- data.frame(
  id   = rep(1:4, each = 3),
  year = rep(2018:2020, 4),
  g    = c(rep(2019, 6), rep(Inf, 6)),
  age  = c(25, 25, 25, 30, 30, 30, 40, 40, 40, 35, 35, 35)
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
cat_balance_table(spec, covariates = "age")
```

`cat_counts` *Count observations or units per node*

Description

Count observations or units per node

Usage

```
cat_counts(spec)
```

Arguments

`spec`	A cat_spec or labeled cat_spec object

Value

A tibble with counts per node

Examples

```
df <- data.frame(
  id   = rep(1:4, each = 3),
  year = rep(2018:2020, 4),
  g    = c(rep(2019, 6), rep(Inf, 6))
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
cat_counts(spec)
```

`cat_design_csdid` *Blueprint for Callaway-Sant'Anna DiD Tree (Staggered Adoption)*

Description

Creates the tree structure for CSDiD with multiple treatment cohorts and a never-treated comparison group.

Usage

```
cat_design_csdid(cohort_labels)
```

Arguments

`cohort_labels`	Character vector of cohort labels (e.g., "g = 2015 (A)")

Details

Tree structure: All Units |– Never-Treated (g = Inf) (last letter) +– Treated Cohorts |– g = g1 (A) |– g = g2 (B) |– g = g3 (C) +– ...

Value

A nested list representing the tree structure

Examples

```
tree <- cat_design_csdid(c("g = 2018 (A)", "g = 2019 (B)"))
tree$root
```

`cat_design_ddd` *Blueprint for Difference-in-Difference-in-Differences Tree*

Description

Creates the tree structure for DDD with multiple treatment cohorts, each split into treated (Q=1) and untreated (Q=0) subgroups, plus never-treated subgroups.

Usage

```
cat_design_ddd(cohort_labels)
```

Arguments

`cohort_labels`	Character vector of cohort labels (e.g., "g = 2015")

Details

Tree structure: All Units |– Treated Cohorts | |– g = g1 | | |– Q = 1 (A) | | +– Q = 0 (B) | |– g = g2 | | |– Q = 1 (C) | | +– Q = 0 (D) | +– ... +– Never-Treated (g = Inf) |– Q = 1 (penultimate letter) +– Q = 0 (last letter)

Value

A nested list representing the tree structure

Examples

```
tree <- cat_design_ddd(c("g = 2018", "g = 2019"))
tree$root
```

`cat_design_did` *Blueprint for 2x2 Difference-in-Differences Tree*

Description

Creates the tree structure for standard 2x2 DiD with one treated cohort and one control (never-treated) group.

Usage

```
cat_design_did()
```

Details

Tree structure: All Units |– Treated (g = g*) | |– Pre (C) | +– Post (D) +– Control (g = Inf) |– Pre (E) +– Post (F)

Value

A nested list representing the tree structure

Examples

```
tree <- cat_design_did()
tree$root
```

`cat_diag` *General diagnostic helper for CAT specifications*

Description

Automatically selects an appropriate diagnostic based on the method argument. Dispatches to:

- "event" -> `cat_event_table()` (event-time counts; no outcome needed)
- "drddd" -> `cat_pt_drddd()` (subgroup pretrend plot)
- "csdid" -> `cat_pt_csdid()` (parallel-gaps plot)

Usage

```
cat_diag(spec, outcome = NULL, method = c("event", "drddd", "csdid"), ...)
```

Arguments

`spec`	A `cat_spec` object
`outcome`	Outcome variable name (required for `"drddd"` and `"csdid"`)
`method`	Diagnostic type: `"event"`, `"drddd"`, or `"csdid"`
`...`	Additional arguments passed to the specific diagnostic function (e.g., `pre_window`)

Value

A list with diagnostic results (always includes `data`; includes `plot` for `"drddd"` and `"csdid"`)

Examples

```
df <- data.frame(id=rep(1:10,each=6), year=rep(2015:2020,10),
                 g=c(rep(2018,30),rep(Inf,30)), outcome=rnorm(60))
spec <- cat_spec(df, id="id", time="year", g="g")
result <- cat_diag(spec, method = "event")
```

`cat_event_table` *Count treated and control observations by event time*

Description

Summarizes the number of treated and control observations by event time `e = t - g`. Works directly with any `cat_spec` object; does not require any additional labeling.

Usage

```
cat_event_table(spec, event_window = -10:10)
```

Arguments

`spec`	A `cat_spec` object (from `cat_spec()`)
`event_window`	Integer vector of event times to include (default `-10:10`)

Value

A tibble with columns `e`, `n_treated`, `n_control`

Examples

```
df <- data.frame(
  id   = rep(1:10, each = 6),
  year = rep(2015:2020, 10),
  g    = c(rep(2018, 30), rep(Inf, 30))
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
cat_event_table(spec, event_window = -3:2)
```

`cat_label` *Label CAT nodes with cohort letters and canonical g labels*

Description

Adds three columns to `spec$data`:

- `.cohort_letter`: uppercase letter assigned chronologically to each finite g cohort
- `.g_label`: canonical label ("g = `2015`" or "g = `Inf`")
- `.g_pretty`: combined label ("g = `2015 (A)`")

Usage

```
cat_label(spec)
```

Arguments

`spec`	A `cat_spec` object (from `cat_spec()`)

Details

Never-treated units (g = `Inf` or g `%in%` `never_treated_values`) are labeled with the last letter in the sequence.

Value

The same `cat_spec` object with three new columns added to `spec$data`

Examples

```
df <- data.frame(
  id   = rep(1:4, each = 3),
  year = rep(2018:2020, 4),
  g    = c(rep(2019, 6), rep(Inf, 6))
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
spec <- cat_label(spec)
```

---

cat_plot_csdid *Plot CSDiD Causal Assignment Tree (IMPROVED - No Cut-offs)*

Description

Tree structure: All Units |– Treated Cohorts | |– g = g1 (A) | |– g = g2 (B) | |– g = g3 (C) | +– ... +– Never-Treated (g = Inf) (last letter)

Usage

```
cat_plot_csdid(spec, counts = TRUE, save_plot = NULL)
```

Arguments

`spec`	A cat_spec object (CSDID setup: no subgroup).
`counts`	Logical; include counts in node labels (default TRUE)
`save_plot`	Optional file path for plot (PNG, PDF, etc.)

Details

Letters A,B,C,... assigned in chronological order of g.

Value

A ggplot object representing the CSDID Causal Assignment Tree.

Examples

```
df <- data.frame(
  id   = rep(1:6, each = 4),
  year = rep(2017:2020, 6),
  g    = c(rep(2018, 8), rep(2019, 8), rep(Inf, 8))
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
cat_plot_csdid(spec)
```

---

cat_plot_ddd *Plot DDD-style Causal Assignment Tree (cohorts x subgroup) - IMPROVED VERSION*

Description

This version uses optimized spacing to prevent node overlap and create a publication-quality visualization.

Usage

```
cat_plot_ddd(spec, counts = TRUE, save_plot = NULL)
```

Arguments

`spec`	A `cat_spec` object with a subgroup variable (DDD setup).
`counts`	Logical; include sample size counts in node labels (default TRUE).
`save_plot`	Optional file path for saving the plot (PNG, PDF, etc.).

Value

A ggplot object representing the DDD Causal Assignment Tree.

Examples

```
df <- data.frame(
  id   = rep(1:6, each = 4),
  year = rep(2017:2020, 6),
  g    = c(rep(2018, 8), rep(2019, 8), rep(Inf, 8)),
  p    = rep(c(0L, 1L), 12)
)
spec <- cat_spec(df, id = "id", time = "year", g = "g", subgroup = "p")
cat_plot_ddd(spec)
```

`cat_plot_did` *Plot 2x2 DID Causal Assignment Tree (IMPROVED - No Cut-offs)*

Description

Structure: All Units |– Treated (A) | |– Pre (C) | +– Post (D) +– Control (B) |– Pre (E) +– Post (F)

Usage

```
cat_plot_did(spec, counts = TRUE, save_plot = NULL)
```

Arguments

`spec`	A cat_spec object (standard DID with no staggered timing).
`counts`	Logical; include sample size counts in node labels.
`save_plot`	Optional path to save.

Value

A ggplot object.

Examples

```
df <- data.frame(
  id   = rep(1:4, each = 2),
  year = rep(2019:2020, 4),
  g    = c(rep(2020, 4), rep(Inf, 4))
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
cat_plot_did(spec)
```

`cat_plot_tree` *Plot a Causal Assignment Tree (unified interface)*

Description

`cat_plot_tree()` is the recommended high-level function for visualizing any CAT design. It inspects the `cat_spec` object and automatically dispatches to the correct underlying plot function:

Usage

```
cat_plot_tree(spec, counts = TRUE, grayscale = FALSE, save_plot = NULL, ...)
```

Arguments

`spec`	A `cat_spec` object (from `cat_spec()`).
`counts`	Logical; include sample-size counts in node labels (default `TRUE`).
`grayscale`	Logical; use a grayscale color palette suitable for black-and-white publications (default `FALSE`). When `TRUE`, treated nodes are shown in dark gray and control nodes in light gray.
`save_plot`	Optional file path to save the plot (e.g. `"tree.png"`). Passed to `ggplot2::ggsave()`.
`...`	Additional arguments passed to the underlying plot function.

Details

Design	Condition	Underlying function
DDD / DR-DDD	subgroup column provided, multiple g values	`cat_plot_ddd()`
CSDiD	No subgroup, multiple g values	`cat_plot_csdid()`
2x2 DiD	No subgroup, exactly one finite g value	`cat_plot_did()`

Value

A `ggplot` object.

Examples

```
df <- data.frame(
  id   = rep(1:6, each = 4),
  year = rep(2017:2020, 6),
  g    = c(rep(2018, 8), rep(2019, 8), rep(Inf, 8))
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
cat_plot_tree(spec)

df$p <- rep(c(0L, 1L), 12)
spec_ddd <- cat_spec(df, id = "id", time = "year", g = "g", subgroup = "p")
cat_plot_tree(spec_ddd)

cat_plot_tree(spec, grayscale = TRUE)
```

`cat_pt_csdid` *CSDiD parallel-gaps diagnostic*

Description

Checks whether treated cohorts and the never-treated group have parallel pre-treatment trends. Plots the gap `mean(treated) - mean(never-treated)` across pre-treatment event-time periods.

Usage

```
cat_pt_csdid(spec, y, pre_window = -8:-1)
```

Arguments

`spec`	A `cat_spec` object
`y`	Outcome variable name
`pre_window`	Integer vector of pre-periods (default `-8:-1`)

Value

A list with `data` (tibble) and `plot` (ggplot)

Examples

```
set.seed(42)
df <- data.frame(
  id      = rep(1:10, each = 6),
  year    = rep(2015:2020, 10),
  g       = c(rep(2018, 30), rep(Inf, 30)),
  outcome = rnorm(60)
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")
result <- cat_pt_csdid(spec, y = "outcome", pre_window = -3:-1)
```

`cat_pt_drddd` *DR-DDD pretrend diagnostic: subgroup means in pre-periods*

Description

Plots mean outcomes for the treated subgroup (subgroup = 1) vs. the control subgroup (subgroup = 0) across pre-treatment event-time periods. Requires a `subgroup` variable in the `cat_spec`.

Usage

```
cat_pt_drddd(spec, y, pre_window = -8:-1)
```

Arguments

`spec`	A `cat_spec` object with a subgroup variable
`y`	Name of the outcome variable
`pre_window`	Integer vector of pre-periods (default `-8:-1`)

Value

A list with elements `data` (tibble) and `plot` (ggplot)

Examples

```
set.seed(42)
df <- data.frame(
  id      = rep(1:10, each = 6),
  year    = rep(2015:2020, 10),
  g       = c(rep(2018, 30), rep(Inf, 30)),
  p       = rep(c(0L, 1L), 30),
  outcome = rnorm(60)
)
spec <- cat_spec(df, id = "id", time = "year", g = "g", subgroup = "p")
result <- cat_pt_drddd(spec, y = "outcome", pre_window = -3:-1)
```

`cat_save_png` *Save a CAT ggplot as a high-quality PNG*

Description

Save a CAT ggplot as a high-quality PNG

Usage

```
cat_save_png(
  plot,
  filename = "CAT_plot.png",
  width = 10,
  height = 6,
  dpi = 400
)
```

Arguments

`plot`	A ggplot object (e.g., from cat_plot_tree())
`filename`	Path to save the PNG (e.g., "CAT_plot.png")
`width`	Width in inches (default = 10)
`height`	Height in inches (default = 6)
`dpi`	Resolution in dots per inch (default = 400)

Value

Invisibly returns the file path

Examples

```
## Not run:
  df <- data.frame(
    id   = rep(1:4, each = 3),
    year = rep(2018:2020, 4),
    g    = c(rep(2019, 6), rep(Inf, 6))
  )
  spec <- cat_spec(df, id = "id", time = "year", g = "g")
  p <- cat_plot_tree(spec)
  cat_save_png(p, filename = tempfile(fileext = ".png"))

## End(Not run)
```

`cat_spec` *Build a Causal Assignment Tree specification*

Description

`cat_spec()` is the entry point for `catviz`. It attaches internal standardised columns (`.id`, `.time`, `.g`, `.subgroup`, `.NT`, `.NYT`, `.node`) to your panel data and records the variable mapping so that all downstream functions know where to look.

Usage

```
cat_spec(
  data,
  id,
  time,
  g,
  subgroup = NULL,
  group_id = NULL,
  never_treated_values = c(0, Inf)
)
```

Arguments

`data`	A data frame (panel structure: one row per unit x time period).
`id`	Name of the **unit** identifier column (e.g. `"hospital_id"`).
`time`	Name of the **time** column (e.g. `"year"` or `"date"`).
`g`	Name of the **first treatment period** column. Units that never receive treatment should have `g = Inf` (or another value listed in `never_treated_values`).
`subgroup`	*(optional)* Name of a binary subgroup column (0/1), used for **DR-DDD** designs. Omit or set `NULL` for pure **CSDID**.
`group_id`	*(optional)* Name of a higher-level grouping column (e.g. `"state"`) when treatment is assigned at a level above the unit.
`never_treated_values`	Numeric vector of `g` values that indicate never-treated status. Default: `c(0, Inf)`.

Value

A `cat_spec` object: a list with elements

- `$data` - the augmented data frame
- `$meta` - a list of variable names and settings

Examples

```
df <- data.frame(
  id   = rep(1:4, each = 3),
  year = rep(2018:2020, 4),
  g    = c(rep(2019, 6), rep(Inf, 6))
)
spec <- cat_spec(df, id = "id", time = "year", g = "g")

df$p <- rep(c(0, 1), 6)
spec_ddd <- cat_spec(df, id = "id", time = "year", g = "g", subgroup = "p")
```

`generate_cohort_labels`

Generate cohort labels with letters

Description

Creates labeled cohort identifiers in the format "g = YYYY (A)" where letters are assigned chronologically.

Usage

```
generate_cohort_labels(g_values, start_letter = "A")
```

Arguments

`g_values`	Numeric vector of treatment years/periods
`start_letter`	Starting letter (default "A")

Value

Character vector of labeled cohorts

Examples

```
generate_cohort_labels(c(2015, 2016, 2019))
```

Index