

Package ‘NonlinearDiD’

May 5, 2026

Type Package

Title Staggered Difference-in-Differences with Nonlinear Outcomes

Version 0.1.0

Description Implements difference-in-differences estimators for staggered treatment adoption with binary, count, and other nonlinear outcomes. Extends Callaway and Sant'Anna (2021) <[doi:10.1016/j.jeconom.2020.12.001](https://doi.org/10.1016/j.jeconom.2020.12.001)> to handle the fundamental identification challenges that arise with nonlinear outcome models (logit, probit, Poisson) in heterogeneous treatment timing designs. Provides group-time average treatment effects on the treated (ATT), aggregation schemes, and pre-treatment parallel trends tests appropriate for nonlinear settings. Methods include doubly-robust semiparametric estimators, nonparametric bounds, and an odds-ratio DiD approach for binary outcomes. Methods extend Callaway and Sant'Anna (2021) <[doi:10.1016/j.jeconom.2020.12.001](https://doi.org/10.1016/j.jeconom.2020.12.001)>, Roth and Sant'Anna (2023) <[doi:10.3982/ECTA19255](https://doi.org/10.3982/ECTA19255)>

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.0.0)

Imports stats, utils, MASS, sandwich, lmtest, ggplot2, Rcpp (>= 1.0.0)

LinkingTo Rcpp

Suggests did, dplyr, knitr, rmarkdown, testthat (>= 3.0.0), covr

VignetteBuilder knitr

Config/testthat/edition 3

URL <https://github.com/causalfragility-lab/NonlinearDiD>

BugReports <https://github.com/causalfragility-lab/NonlinearDiD/issues>

NeedsCompilation yes

Author Subir Hait [aut, cre] (ORCID: <<https://orcid.org/0009-0004-9871-9677>>)

Maintainer Subir Hait <haitsubi@msu.edu>

Repository CRAN

Date/Publication 2026-05-05 19:30:02 UTC

Contents

NonlinearDiD-package	2
binary_did_dr	3
binary_did_logit	5
binary_did_probit	6
count_did_poisson	7
nonlineardid_methods	8
nonlinear_aggte	8
nonlinear_attgt	10
nonlinear_bounds	12
nonlinear_pretest	14
odds_ratio_did	15
plot.nonlinear_aggte	16
plot.nonlinear_attgt	17
sim_binary_panel	17
sim_count_panel	19
Index	20

NonlinearDiD-package *NonlinearDiD: Staggered DiD with Nonlinear Outcomes*

Description

The **NonlinearDiD** package extends the Callaway and Sant’Anna (2021) staggered difference-in-differences framework to handle nonlinear outcome models, including binary (logit/probit), count (Poisson/NegBin), and odds-ratio estimands.

The Core Problem

The canonical CS2021 framework assumes parallel trends on the mean scale of a continuous outcome. For binary and count outcomes, this assumption is not scale-invariant: parallel trends in $P(Y=1)$ does NOT imply parallel trends in log-odds, pre-trend tests depend on which scale is used, and treatment effect estimates conflate true effects with Jensen’s inequality.

Main Functions

- `nonlinear_attgt()` – Estimate ATT(g,t) under nonlinear outcome models
- `nonlinear_aggte()` – Aggregate: event-study, group, calendar, overall
- `nonlinear_pretest()` – Pre-treatment parallel trends test
- `binary_did_logit()` – 2x2 DiD with logit outcome
- `binary_did_probit()` – 2x2 DiD with probit outcome
- `binary_did_dr()` – Doubly-robust binary DiD
- `count_did_poisson()` – Poisson QMLE DiD for count outcomes
- `odds_ratio_did()` – Odds-ratio DiD (scale-free)
- `nonlinear_bounds()` – Nonparametric Manski / PT bounds

- `sim_binary_panel()` – Simulate binary staggered panel data
- `sim_count_panel()` – Simulate count staggered panel data

Quick Start

```
library(NonlinearDiD)
dat <- sim_binary_panel(n = 500, nperiods = 8, seed = 42)
res <- nonlinear_attgt(dat, yname = "y", tname = "period",
                      idname = "id", gname = "g",
                      outcome_model = "logit")
agg <- nonlinear_aggte(res, type = "dynamic")
plot(agg)
nonlinear_pretest(res)
```

Author(s)

Maintainer: Subir Hait <haitsubi@msu.edu> ([ORCID](#))

References

- Callaway, B., & Sant'Anna, P. H. C. (2021). Difference-in-differences with multiple time periods. *Journal of Econometrics*, 225(2), 200-230.
- Roth, J., & Sant'Anna, P. H. C. (2023). When is parallel trends sensitive to functional form? *Econometrica*, 91(2), 737-747.
- Wooldridge, J. M. (2023). Simple approaches to nonlinear difference-in-differences with panel data. *The Econometrics Journal*, 26(3).

See Also

Useful links:

- <https://github.com/causalfragility-lab/NonlinearDiD>
- Report bugs at <https://github.com/causalfragility-lab/NonlinearDiD/issues>

binary_did_dr

Doubly-Robust Binary DiD

Description

Doubly-robust estimator for binary outcomes combining a nonlinear outcome regression model with inverse probability weighting via propensity score. Consistent if EITHER the outcome model OR the propensity score is correctly specified.

Usage

```
binary_did_dr(
  data,
  yname,
  tname,
  idname,
  treat_period,
  control_period,
  dname = NULL,
  gname = NULL,
  xformula = ~1,
  outcome_model = c("logit", "probit"),
  se_type = c("robust", "cluster", "analytical"),
  cluster_var = NULL
)
```

Arguments

<code>data</code>	A data frame (long format).
<code>yname</code>	Character. Binary outcome variable name.
<code>tname</code>	Character. Time period variable name.
<code>idname</code>	Character. Unit ID variable name.
<code>treat_period</code>	Numeric. The treatment (post) period.
<code>control_period</code>	Numeric. The pre-treatment baseline period.
<code>dname</code>	Character. Treatment indicator variable name (optional).
<code>gname</code>	Character. Cohort variable name (optional).
<code>xformula</code>	One-sided formula for covariates. Default <code>~1</code> .
<code>outcome_model</code>	Character. "logit" (default) or "probit".
<code>se_type</code>	Character. SE type: "robust" (default), "cluster", or "analytical".
<code>cluster_var</code>	Character. Clustering variable (if <code>se_type = "cluster"</code>).

Value

A list of class `binary_did_dr`.

Examples

```
dat <- sim_binary_panel(n = 500, nperiods = 4, prop_treated = 0.5)
dat2 <- dat[dat$period %in% c(2, 3), ]
res <- binary_did_dr(dat2, "y", "period", "id", 3, 2, gname = "g",
  outcome_model = "logit")
print(res)
```

binary_did_logit	<i>Binary Outcome DiD: Logit Estimator</i>
------------------	--

Description

Estimates a 2x2 difference-in-differences model with a binary outcome using logistic regression on the log-odds scale, reporting both the log-odds DiD coefficient and the average partial effect (APE) on the probability scale.

Usage

```
binary_did_logit(
  data,
  yname,
  tname,
  idname,
  treat_period,
  control_period,
  dname = NULL,
  gname = NULL,
  xformula = ~1,
  se_type = c("robust", "cluster", "analytical"),
  cluster_var = NULL
)
```

Arguments

data	A data frame (long format).
yname	Character. Binary outcome variable name.
tname	Character. Time period variable name.
idname	Character. Unit ID variable name.
treat_period	Numeric. The treatment (post) period.
control_period	Numeric. The pre-treatment baseline period.
dname	Character. Treatment indicator variable name (optional).
gname	Character. Cohort variable name (optional).
xformula	One-sided formula for covariates. Default ~1.
se_type	Character. SE type: "robust" (default), "cluster", or "analytical".
cluster_var	Character. Clustering variable (if se_type = "cluster").

Value

A list of class `binary_did_logit`.

Examples

```

dat <- sim_binary_panel(n = 500, nperiods = 4, prop_treated = 0.5)
dat2 <- dat[dat$period %in% c(2, 3), ]
res <- binary_did_logit(dat2, yname = "y", tname = "period",
                        idname = "id", treat_period = 3,
                        control_period = 2, gname = "g")

print(res)

```

binary_did_probit	<i>Binary Outcome DiD: Probit Estimator</i>
-------------------	---

Description

Estimates 2x2 DiD with binary outcome using probit regression. Parallel trends assumed on the probit (inverse-normal) scale.

Usage

```

binary_did_probit(
  data,
  yname,
  tname,
  idname,
  treat_period,
  control_period,
  dname = NULL,
  gname = NULL,
  xformula = ~1,
  se_type = c("robust", "cluster", "analytical"),
  cluster_var = NULL
)

```

Arguments

<code>data</code>	A data frame (long format).
<code>yname</code>	Character. Binary outcome variable name.
<code>tname</code>	Character. Time period variable name.
<code>idname</code>	Character. Unit ID variable name.
<code>treat_period</code>	Numeric. The treatment (post) period.
<code>control_period</code>	Numeric. The pre-treatment baseline period.
<code>dname</code>	Character. Treatment indicator variable name (optional).
<code>gname</code>	Character. Cohort variable name (optional).
<code>xformula</code>	One-sided formula for covariates. Default ~1.
<code>se_type</code>	Character. SE type: "robust" (default), "cluster", or "analytical".
<code>cluster_var</code>	Character. Clustering variable (if <code>se_type = "cluster"</code>).

Value

A list of class `binary_did_probit`.

Examples

```
dat <- sim_binary_panel(n = 500, nperiods = 4, prop_treated = 0.5)
dat2 <- dat[dat$period %in% c(2, 3), ]
res <- binary_did_probit(dat2, "y", "period", "id", 3, 2, gname = "g")
print(res)
```

count_did_poisson	<i>Count Outcome DiD: Poisson Estimator</i>
-------------------	---

Description

Estimates DiD for count outcomes using a Poisson quasi-maximum likelihood (QMLE) estimator with a log-linear parallel trends assumption. The treatment effect is a multiplicative rate ratio.

Usage

```
count_did_poisson(
  data,
  yname,
  tname,
  idname,
  treat_period,
  control_period,
  dname = NULL,
  gname = NULL,
  xformula = ~1,
  offset = NULL,
  se_type = c("robust", "cluster", "analytical"),
  cluster_var = NULL
)
```

Arguments

<code>data</code>	A data frame (long format).
<code>yname</code>	Character. Binary outcome variable name.
<code>tname</code>	Character. Time period variable name.
<code>idname</code>	Character. Unit ID variable name.
<code>treat_period</code>	Numeric. The treatment (post) period.
<code>control_period</code>	Numeric. The pre-treatment baseline period.
<code>dname</code>	Character. Treatment indicator variable name (optional).
<code>gname</code>	Character. Cohort variable name (optional).

xformula	One-sided formula for covariates. Default ~1.
offset	Character. Name of offset variable. Default NULL.
se_type	Character. SE type: "robust" (default), "cluster", or "analytical".
cluster_var	Character. Clustering variable (if se_type = "cluster").

Value

A list of class count_did_poisson.

Examples

```
dat <- sim_count_panel(n = 400, nperiods = 6, prop_treated = 0.4)
dat2 <- dat[dat$period %in% c(2, 4), ]
res <- count_did_poisson(dat2, "y", "period", "id", 4, 2, gname = "g")
print(res)
```

nonlineardid_methods *S3 Methods for NonlinearDiD Objects*

Description

Print, summary, and plot methods for nonlinear_attgt and nonlinear_aggte objects.

nonlinear_aggte *Aggregate ATT(g,t) Estimates for Nonlinear DiD*

Description

Aggregates the group-time average treatment effects from [nonlinear_attgt](#) into interpretable summary parameters. Provides event-study (dynamic), group-level, calendar-time, and overall ATT aggregations - each appropriate for nonlinear settings.

Usage

```
nonlinear_aggte(
  obj,
  type = c("dynamic", "group", "calendar", "simple"),
  na.rm = TRUE,
  min_periods = 1L,
  weights = c("equal", "sample")
)
```

Arguments

<code>obj</code>	An object of class <code>nonlinear_attgt</code> from <code>nonlinear_attgt</code> .
<code>type</code>	Character. The aggregation type: <ul style="list-style-type: none"> • "dynamic": Event-study / dynamic treatment effects. Averages ATT(g,t) across groups g for each relative time $e = t - g$. • "group": Group-specific ATT. Averages over post-treatment periods within each treated cohort g. • "calendar": Calendar-time ATT. Averages over groups for each calendar time t. • "simple": Overall average ATT, weighted by cohort size.
<code>na.rm</code>	Logical. Remove NA ATT(g,t) estimates. Default TRUE.
<code>min_periods</code>	Integer. Minimum number of ATT(g,t) observations required for an aggregated estimate to be reported. Default 1.
<code>weights</code>	Character. Weighting scheme for aggregation: <ul style="list-style-type: none"> • "equal": Equal-weight across (g,t) cells (default). • "sample": Weight by treated sample size in each (g,t).

Value

An object of class `nonlinear_aggte` with slots:

agg Data frame with aggregated ATT, SE, and CI.

type The aggregation type used.

overall_att Scalar overall ATT estimate.

overall_se SE for overall ATT.

Examples

```
set.seed(1)
dat <- sim_binary_panel(n = 400, nperiods = 8, prop_treated = 0.5)
res <- nonlinear_attgt(dat, yname = "y", tname = "period",
                      idname = "id", gname = "g",
                      outcome_model = "logit")
agg <- nonlinear_aggte(res, type = "dynamic")
plot(agg)
```

Description

Computes group-time average treatment effects on the treated (ATT(g,t)) for staggered difference-in-differences designs with nonlinear outcomes.

This function extends Callaway & Sant’Anna (2021) to handle binary, count, and other nonlinear outcomes where the standard linear parallel trends assumption is misspecified. The key methodological contributions are:

1. **Parallel trends on the latent index** (for logit/probit): Instead of assuming parallel trends in $E[Y]$, we assume parallel trends in the latent utility $F^{-1}(E[Y])$.
2. **Doubly-robust nonlinear estimator**: Combines outcome regression (nonlinear model) with propensity score weighting, inheriting DR properties in the nonlinear setting.
3. **Odds-ratio DiD**: A scale-free estimand appropriate for binary outcomes that does not require parallel trends in probabilities.
4. **Nonparametric bounds**: When no functional form is assumed, provides sharp bounds on ATT(g,t).

Usage

```
nonlinear_attgt(
  data,
  yname,
  tname,
  idname,
  gname,
  xformula = ~1,
  outcome_model = c("logit", "probit", "poisson", "negbin", "linear"),
  estimand = c("att", "odds_ratio", "ape"),
  control_group = c("nevertreated", "notyetreated"),
  doubly_robust = TRUE,
  boot = FALSE,
  nboot = 999,
  boot_type = c("multiplier", "empirical"),
  alpha = 0.05,
  parallel = FALSE,
  pl_cores = 2L,
  anticipation = 0L
)
```

Arguments

data	A data frame in long format (one row per unit-period).
yname	Character. Name of the outcome variable column.

tname	Character. Name of the time period column.
idname	Character. Name of the unit identifier column.
gname	Character. Name of the treatment cohort column (the period when a unit first receives treatment; 0 or Inf for never-treated units).
xformula	A one-sided formula for covariates (e.g., $\sim x_1 + x_2$). Default is ~ 1 (intercept only).
outcome_model	Character. The outcome model to use. One of: <ul style="list-style-type: none"> • "logit": Logistic regression (for binary Y) • "probit": Probit regression (for binary Y) • "poisson": Poisson regression (for count Y) • "negbin": Negative binomial (for overdispersed count Y) • "linear": Linear model (reproduces CS2021 when combined with doubly_robust = TRUE)
estimand	Character. The treatment effect estimand: <ul style="list-style-type: none"> • "att": Average treatment effect on the treated (default) • "odds_ratio": Odds ratio DiD (binary outcomes only) • "ape": Average partial effect on the probability scale
control_group	Character. Which units serve as the control group: <ul style="list-style-type: none"> • "nevertreated": Use never-treated units only (default) • "notyetreated": Use not-yet-treated units
doubly_robust	Logical. If TRUE (default), uses the doubly-robust estimator that combines propensity score weighting with outcome regression. More robust to model misspecification.
boot	Logical. If TRUE, uses bootstrap for inference. Default FALSE.
nboot	Integer. Number of bootstrap iterations. Default 999.
boot_type	Character. Type of bootstrap: "multiplier" (default, fast) or "empirical".
alpha	Numeric. Significance level for confidence intervals. Default 0.05.
parallel	Logical. Use parallel processing for bootstrap. Default FALSE.
pl_cores	Integer. Number of cores for parallel processing.
anticipation	Integer. Number of periods of anticipation allowed. Default 0.

Value

An object of class `nonlinear_attgt` containing:

attgt Data frame of ATT(g,t) estimates, standard errors, and confidence intervals for each (group, time) pair.

call The matched call.

args List of arguments used.

boot_draws Matrix of bootstrap draws (if boot = TRUE).

References

- Callaway, B., & Sant'Anna, P. H. C. (2021). Difference-in-differences with multiple time periods. *Journal of Econometrics*, 225(2), 200-230.
- Wooldridge, J. M. (2023). Simple approaches to nonlinear difference-in-differences with panel data. *The Econometrics Journal*, 26(3).
- Roth, J., & Sant'Anna, P. H. C. (2023). When is parallel trends sensitive to functional form? *Econometrica*, 91(2), 737-747.

Examples

```
# Simulate binary panel data
set.seed(42)
dat <- sim_binary_panel(n = 500, nperiods = 6, prop_treated = 0.4)

# Estimate ATT(g,t) with logistic outcome model
result <- nonlinear_attgt(
  data = dat,
  yname = "y",
  tname = "period",
  idname = "id",
  gname = "g",
  outcome_model = "logit",
  control_group = "nevertreated"
)

summary(result)
plot(result)
```

 nonlinear_bounds

Nonparametric Bounds for Binary Outcomes in Staggered DiD

Description

Computes sharp nonparametric bounds on the ATT for binary outcomes in staggered difference-in-differences designs, following the partial identification approach. These bounds require NO functional form assumptions on the outcome model - only an assumption about the direction or magnitude of selection.

The key insight for binary outcomes: Since Y is binary (0 or 1), the ATT is bounded by:

- Lower bound: counterfactual never exceeds observed (pessimistic)
- Upper bound: counterfactual never falls below observed (optimistic)

Under a Manski-style no-assumptions bound, plus refinements using the parallel trends assumption as a restriction.

Usage

```

nonlinear_bounds(
  data,
  yname,
  tname,
  idname,
  gname,
  xformula = ~1,
  control_group = c("nevertreated", "notyetreated"),
  bound_type = c("pt_only", "manski", "pt_monotone"),
  alpha = 0.05
)

```

Arguments

data	A long-format panel data frame.
yname	Character. Name of binary outcome variable (0/1).
tname	Character. Name of time period column.
idname	Character. Name of unit identifier.
gname	Character. Name of treatment cohort column.
xformula	One-sided formula for covariates. Default ~ 1.
control_group	Character. "nevertreated" (default) or "notyetreated".
bound_type	Character. Type of bound: <ul style="list-style-type: none"> • "manski": No-assumptions Manski bounds (widest) • "pt_monotone": Tighten using parallel trends + monotone treatment response • "pt_only": Use only parallel trends restriction
alpha	Numeric. Significance level for confidence intervals on bounds.

Value

A data frame of sharp bounds (lb, ub) for ATT(g,t), with bootstrap confidence intervals.

References

Manski, C. F. (1990). Nonparametric bounds on treatment effects. *American Economic Review*, 80(2), 319-323.

Callaway, B. (2021). Bounds on distributional treatment effect parameters. *Journal of Econometrics*, 222(2), 1084-1111.

Examples

```

set.seed(5)
dat <- sim_binary_panel(n = 300, nperiods = 6)
bounds <- nonlinear_bounds(dat, "y", "period", "id", "g")
print(bounds)

```

nonlinear_pretest *Pre-Treatment Parallel Trends Test for Nonlinear DiD*

Description

Tests for pre-treatment violations of the parallel trends assumption in nonlinear staggered DiD settings. This is fundamentally different from the linear case because:

1. **Scale dependence:** Parallel trends on the probability scale does NOT imply parallel trends on the latent index scale (and vice versa). Tests are performed on the scale specified in `outcome_model`.
2. **Roth-Sant'Anna sensitivity:** Computes sensitivity of post-treatment estimates to violations of magnitude delta in pre-period, following Roth & Sant'Anna (2023).
3. **Joint test:** Provides a joint chi-squared test of all pre-period $ATT(g,t) = 0$, accounting for correlation across (g,t) cells.

Usage

```
nonlinear_pretest(
  obj,
  plot = TRUE,
  alpha = 0.05,
  type = c("joint", "individual", "honestdid")
)
```

Arguments

<code>obj</code>	An object of class <code>nonlinear_attgt</code> .
<code>plot</code>	Logical. If TRUE (default), produces a pre-trends plot.
<code>alpha</code>	Numeric. Significance level. Default 0.05.
<code>type</code>	Character. Type of pre-trends test: <ul style="list-style-type: none"> • "joint": Joint chi-squared test (default) • "individual": Individual t-tests per pre-period cell • "honestdid": Sensitivity analysis a la Roth-Sant'Anna

Value

A list with:

pretest_results Data frame of pre-period $ATT(g,t)$ with p-values.
joint_stat Joint test statistic.
joint_pval P-value for joint test.
conclusion Interpretive conclusion string.

References

Roth, J. (2022). Pretest with caution: Event-study estimates after testing for parallel trends. *American Economic Review: Insights*, 4(3), 305-322.

Roth, J., & Sant'Anna, P. H. C. (2023). When is parallel trends sensitive to functional form? *Econometrica*, 91(2), 737-747.

Examples

```
set.seed(99)
dat <- sim_binary_panel(n = 600, nperiods = 8, prop_treated = 0.5)
res <- nonlinear_attgt(dat, "y", "period", "id", "g",
                      outcome_model = "logit")
pt <- nonlinear_pretest(res)
print(pt)
```

odds_ratio_did

Odds-Ratio DiD for Binary Outcomes

Description

Estimates the odds-ratio difference-in-differences (OR-DiD) for binary outcomes. OR-DiD equals 1 under no treatment effect and is invariant to which group is labelled treatment.

Usage

```
odds_ratio_did(
  data,
  yname,
  tname,
  idname,
  treat_period,
  control_period,
  dname = NULL,
  gname = NULL,
  xformula = ~1
)
```

Arguments

<code>data</code>	A data frame (long format).
<code>yname</code>	Character. Binary outcome variable name.
<code>tname</code>	Character. Time period variable name.
<code>idname</code>	Character. Unit ID variable name.
<code>treat_period</code>	Numeric. The treatment (post) period.

control_period Numeric. The pre-treatment baseline period.
 dname Character. Treatment indicator variable name (optional).
 gname Character. Cohort variable name (optional).
 xformula One-sided formula for covariates. Default ~1.

Value

A list of class odds_ratio_did.

Examples

```
dat <- sim_binary_panel(n = 500, nperiods = 4, prop_treated = 0.5)
dat2 <- dat[dat$period %in% c(2, 3), ]
res <- odds_ratio_did(dat2, "y", "period", "id", 3, 2, gname = "g")
print(res)
```

plot.nonlinear_aggte *Plot Aggregated DiD Estimates*

Description

Plots event-study, group-level, calendar, or overall aggregated ATT estimates from [nonlinear_aggte](#).

Usage

```
## S3 method for class 'nonlinear_aggte'
plot(x, ...)
```

Arguments

x An object of class nonlinear_aggte.
 ... Additional arguments (unused).

Value

A ggplot2 object.

plot.nonlinear_attgt *Plot ATT(g,t) Estimates*

Description

Produces a faceted scatter plot of ATT(g,t) estimates with confidence intervals, one panel per treatment cohort.

Usage

```
## S3 method for class 'nonlinear_attgt'
plot(x, ..., alpha = 0.05, point_size = 2)
```

Arguments

x	An object of class nonlinear_attgt.
...	Additional arguments (unused).
alpha	Numeric. Significance level for CI. Default 0.05.
point_size	Numeric. Size of estimate points. Default 2.

Value

A ggplot2 object.

sim_binary_panel *Simulate Binary Panel Data with Staggered Treatment*

Description

Generates a simulated panel dataset with staggered treatment adoption and a binary outcome. Useful for testing and illustrating nonlinear DiD methods.

The data-generating process is:

$$Y_{it} = \mathbf{1}\{\alpha_i + \lambda_t + \delta_{it} \cdot D_{it} + \epsilon_{it} > 0\}$$

where α_i is a unit fixed effect, λ_t is a time fixed effect, δ_{it} is the treatment effect (heterogeneous across cohorts), and ϵ_{it} is logistic noise.

Usage

```
sim_binary_panel(
  n = 500L,
  nperiods = 6L,
  prop_treated = 0.5,
  n_cohorts = 3L,
  true_att = 0.3,
  base_prob = 0.3,
  unit_fe_sd = 0.5,
  add_covariates = TRUE,
  seed = NULL
)
```

Arguments

n	Integer. Number of units. Default 500.
nperiods	Integer. Number of time periods. Default 6.
prop_treated	Numeric. Proportion of units ever treated. Default 0.5.
n_cohorts	Integer. Number of treatment cohorts (groups). Default 3.
true_att	Numeric or vector. True ATT for each cohort. Default 0.3.
base_prob	Numeric. Baseline probability $P(Y=1)$ for untreated. Default 0.3.
unit_fe_sd	Numeric. Std. dev. of unit fixed effects. Default 0.5.
add_covariates	Logical. Add pre-treatment covariates. Default TRUE.
seed	Integer. Random seed. Default NULL.

Value

A data frame in long format. Columns: id (unit identifier), period (time period 1 to nperiods), y (binary outcome 0/1), g (treatment cohort; 0 = never treated), D (treatment indicator), x1 and x2 (covariates, if add_covariates = TRUE), and alpha_i (true unit fixed effect, for validation).

Examples

```
dat <- sim_binary_panel(n = 1000, nperiods = 8, prop_treated = 0.6,
                       n_cohorts = 4, true_att = c(0.2, 0.4, 0.3, 0.5))
head(dat)
table(dat$g)
```

sim_count_panel *Simulate Count Panel Data with Staggered Treatment*

Description

Generates simulated panel data with a count outcome (Poisson-distributed) and staggered treatment adoption. Treatment effect is multiplicative (rate ratio) on the count scale.

Usage

```
sim_count_panel(  
  n = 500L,  
  nperiods = 6L,  
  prop_treated = 0.5,  
  n_cohorts = 3L,  
  true_rr = 1.5,  
  base_rate = 5,  
  overdispersion = FALSE,  
  seed = NULL  
)
```

Arguments

n	Integer. Number of units. Default 500.
nperiods	Integer. Number of time periods. Default 6.
prop_treated	Numeric. Proportion of units ever treated. Default 0.5.
n_cohorts	Integer. Number of treatment cohorts. Default 3.
true_rr	Numeric or vector. True rate ratio for each cohort. Default 1.5 (50 percent increase in count).
base_rate	Numeric. Baseline Poisson rate. Default 5.
overdispersion	Logical. Add overdispersion (negative binomial). Default FALSE.
seed	Integer. Random seed.

Value

Long-format data frame with columns: id, period, y, g, D, x1.

Examples

```
dat <- sim_count_panel(n = 400, nperiods = 6, true_rr = 1.8)  
summary(dat$y)
```

Index

binary_did_dr, [3](#)
binary_did_logit, [5](#)
binary_did_probit, [6](#)

count_did_poisson, [7](#)

nonlinear_aggte, [8](#), [16](#)
nonlinear_attgt, [8](#), [9](#), [10](#)
nonlinear_bounds, [12](#)
nonlinear_pretest, [14](#)
NonlinearDiD (NonlinearDiD-package), [2](#)
NonlinearDiD-package, [2](#)
nonlineardid_methods, [8](#)

odds_ratio_did, [15](#)

plot.nonlinear_aggte, [16](#)
plot.nonlinear_attgt, [17](#)

sim_binary_panel, [17](#)
sim_count_panel, [19](#)