Package: pstratreg (via r-universe)

November 7, 2026

Type Package

Title Estimate Bounds via Principal Stratification

Version 1.0.2

Description When studying the causal effect of a treatment on an outcome, post-treatment events may render the outcome undefined. For example, the effect of job training on hourly wage is ill-defined for those who would not be employed without job training. This package estimates average causal effects among those whose outcome would be defined regardless of the intervention, producing a set-identified result. For more details, see Lundberg and Cho (2025) <doi:10.48550/arXiv.2508.14770>.

License MIT + file LICENSE

Encoding UTF-8

Imports dplyr, foreach, methods, purrr, stats, tidyr, tidyselect

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Ian Lundberg [aut, cre], Soonhong Cho [aut]

Maintainer Ian Lundberg <ianlundberg@ucla.edu>

Depends R (>= 4.1.0)

Repository https://cran.r-universe.dev

Date/Publication 2026-11-07 21:01:15 UTC

RemoteUrl https://github.com/cran/pstratreg

RemoteRef HEAD

RemoteSha 1894ff422541611e8cf74ff61a388d809d09d5a2

Contents

print.pstratreg 2
pstratreg 2

pstratreg_se 4
pstratreg_sim 5
summary.pstratreg 6

Index **7**

`print.pstratreg` *Print*

Description

Print results from principal stratification regression.

Usage

```
## S3 method for class 'pstratreg'
print(x, ...)
```

Arguments

`x`	An object of class `pstratreg`, such as the output of a call to `pstratreg()`
`...`	Other arguments to `print` commands

Value

No return value, called for side effects.

`pstratreg` *Estimate Principal Stratification Regression Bounds*

Description

Uses principal stratification and parametric models to bound the average causal effect among those who would have a valid outcome under either treatment condition

Usage

```
pstratreg(
  formula_y,
  formula_s,
  formula_sq_resid = NULL,
  family_y = "gaussian",
  homoskedastic = TRUE,
  data,
  weights = NULL,
  treatment_name,
```

```
  monotonicity_positive = FALSE,
  monotonicity_negative = FALSE,
  mean_dominance_y1_positive = FALSE,
  mean_dominance_y1_negative = FALSE,
  mean_dominance_y0_positive = FALSE,
  mean_dominance_y0_negative = FALSE,
  aggregate = TRUE,
  group_vars = NULL
)
```

Arguments

formula_y	A model formula for the outcome
formula_s	A model formula for the binary indicator for whether the outcome exists
formula_sq_resid	A model formula for the log mean of squared residuals for a variance function regression. Should only use variables that are used in formula_y. Because the Gamma regression for squared residuals is more computationally demanding, one might want a simpler formula_sq_resid than formula_y, perhaps to allow only highly structured heteroskedasticity. Defaults to match formula_y. No outcome need be specified in this formula. Can be NULL if 'homoskedastic = TRUE' because then there is no model for the conditional variance, which is assumed to be constant.
family_y	Character family of the outcome, either "gaussian" or "binomial"
homoskedastic	A logical for whether homoskedasticity of the outcome should be assumed. If FALSE, estimates by variance function regression for log squared residuals with the same formula as in formula_y.
data	A data frame
weights	A numeric vector of weights, of length nrow(data)
treatment_name	A character string for the name of the treatment variable in data,
monotonicity_positive	A logical. Whether to assume M1 >= M0, so that treatment never causes the outcome to be undefined
monotonicity_negative	A logical. Whether to assume M1 <= M0, so that lack of treatment never causes the outcome to be undefined
mean_dominance_y1_positive	A logical. Whether to assume E(Y1 | S1 = 1, S0 = 1) >= E(Y1 | S1 = 1, S0 = 0), so that the always-survivor treated units have mean outcomes at least as great as the observed-survivor treated units
mean_dominance_y1_negative	A logical. Whether to assume E(Y1 | S1 = 1, S0 = 1) <= E(Y1 | S1 = 1, S0 = 0), so that the always-survivor treated units have mean outcomes no greater the observed-survivor treated units
mean_dominance_y0_positive	A logical. Whether to assume E(Y0 | S1 = 1, S0 = 1) >= E(Y0 | S1 = 0, S0 = 1), so that the always-survivor untreated units have mean outcomes at least as great as the observed-survivor untreated units

mean_dominance_y0_negative
: A logical. Whether to assume E(Y0 | S1 = 1, S0 = 1) <= E(Y0 | S1 = 0, S0 = 1), so that the always-survivor untreated units have mean outcomes no greater the observed-survivor untreated units

aggregate
: A logical. Whether to aggregate results or return one estimate per row of data

group_vars
: Character vector of names of variables in data by which to group when aggregating results. Only relevant when aggregate = TRUE

Value

An object of class pstratreg, which is a list containing:

estimates_y
: A data frame or tibble containing point bounds for the outcome values among those whose outcome exists regardless.

estimates_s
: A data frame or tibble containing point estimates for outcome existence.

fit_s
: The glm object for the model of outcome existence.

fit_y
: The glm or lm object for the model of outcome values conditional on existence.

fit_sq_resid
: The glm object for the model of squared outcome residuals under heteroskedasticity (if homoskedastic = FALSE).

call
: A list of arguments passed to the function.

Examples

```
sim <- pstratreg_sim(n = 100)
result <- pstratreg(
     formula_y = formula(y ~ x + a),
     formula_s = formula(s ~ x + a),
     data = sim,
     treatment_name = "a"
)
result <- pstratreg(
     formula_y = formula(y ~ x + a),
     formula_s = formula(s ~ x + a),
     data = sim,
     treatment_name = "a",
     aggregate = TRUE,
     homoskedastic = FALSE,
     monotonicity_negative = TRUE
)
```

---

pstratreg_se *Standard errors for pstratreg*

Description

Calculate standard errors by a nonparametric bootstrap, and 95

Usage

```
pstratreg_se(x, r = 100)
```

Arguments

x	An object of class pstratreg, such as the output of a call to pstratreg()
r	Number of bootstrap replicates

Value

Data frame with estimates, standard errors, and confidence intervals for many quantities of interest.

Examples

```
sim <- pstratreg_sim(n = 100)
result <- pstratreg(
     formula_y = formula(y ~ x + a),
     formula_s = formula(s ~ x + a),
     data = sim,
     treatment_name = "a"
)
se <- pstratreg_se(result, r = 10)
```

---

pstratreg_sim *Simulate Data*

Description

Simulate data to illustrate principal stratification regression bounding.

Usage

```
pstratreg_sim(n = 100)
```

Arguments

n	Integer number of cases

Value

Simulated data frame

Examples

```
pstratreg_sim(n = 100)
```

summary.pstratreg *Summary*

Description

Summarize results from principal stratification regression.

Usage

```
## S3 method for class 'pstratreg'
summary(object, ...)
```

Arguments

object	An object of class pstratreg, such as the output of a call to pstratreg()
...	Other arguments to summary commands

Value

No return value, called for side effects.

Index

print.pstratreg, 2
pstratreg, 2
pstratreg_se, 4
pstratreg_sim, 5

summary.pstratreg, 6