Package 'RobustMediate'

November 2, 2026

Title Causal Mediation Analysis with Diagnostics and Sensitivity Analysis

Version 0.3.1

Description Provides tools for causal mediation analysis with continuous treatments. Estimates natural direct and indirect effects by g-computation over a user-defined treatment grid, with bootstrap intervals that can resample whole clusters, and supports nonlinear dose-response through spline terms in the model formulas. Includes balance diagnostics based on standardized mean differences and stabilized inverse probability weights. Provides sensitivity analysis for confounding of the mediator-outcome relationship using the error correlation parameter of Imai, Keele, and Yamamoto (2010) <doi:10.1214/10-STS321>, and path-wise and indirect-effect robustness values in the partial R-squared metric of Cinelli and Hazlett (2020) <doi:10.1111/rssb.12348>, for both joint-significance and Sobel tests of the indirect effect. Additional utilities include visualization of dose-response mediation functions and formatted outputs for applied research.

License MIT + file LICENSE

URL https://github.com/subirhait/RobustMediate

BugReports https://github.com/subirhait/RobustMediate/issues

Encoding UTF-8

LazyData true

Depends R (>= 4.1.0)

Imports cli, ggplot2 (>= 3.4.0), rlang, scales, stats, broom

Suggests covr, dplyr, knitr, lme4, pkgdown, rmarkdown, splines, testthat (>= 3.0.0), tidyr

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

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

Maintainer Subir Hait <haitsubi@msu.edu>

Config/roxygen2/version 8.1.1

Repository CRAN

Date/Publication 2026-11-02 10:10:02 UTC

Contents

RobustMediate-package . . . 2
adjust_paths . . . 4
as.data.frame.robmedfit . . . 4
augment.robmedfit . . . 5
compare_fits . . . 6
diagnose . . . 7
effect_curvature . . . 8
glance.robmedfit . . . 9
plot.robmedfit . . . 10
plot_balance . . . 10
plot_curvature . . . 11
plot_mediation . . . 12
plot_paths . . . 13
plot_sensitivity . . . 14
print.robmedfit . . . 14
print.robmed_paths . . . 15
print.robmed_sens . . . 15
robustmediate . . . 16
sensitivity_paths . . . 18
sensitivity_rho . . . 20
sim_mediation . . . 22
summary.robmedfit . . . 23
tidy.robmedfit . . . 24

Index **25**

`RobustMediate-package` *RobustMediate: Causal Mediation Analysis with Diagnostics and Sensitivity Analysis*

Description

RobustMediate provides a workflow for causal mediation analysis with continuous treatments: g-computation estimates of natural direct and indirect effects, balance diagnostics, and sensitivity analysis.

Main functions include:

- **‘robustmediate()‘** - Fits treatment, mediator, and outcome models and stores precomputed results for downstream plotting and reporting.

- **‘plot_balance()‘** - Displays covariate balance before and after weighting for both the treatment and mediator pathways using standardized mean differences.

- **‘plot_mediation()‘** - Plots estimated natural direct effects (NDE) and natural indirect effects (NIE) over the treatment range, with pointwise uncertainty bands.

- **‘sensitivity_rho()‘** and **‘plot_sensitivity()‘** - Sensitivity of the indirect effect to confounding of the mediator-outcome relationship, using the correlation parameter ‘rho‘ of Imai, Keele, and Yamamoto (2010).

- **‘diagnose()‘** - Produces a formatted diagnostic summary of balance, mediation effects, and sensitivity results.

Getting started

“‘r library(RobustMediate)

data(sim_mediation)

fit <- robustmediate(X ~ Z1 + Z2 + Z3, M ~ X + Z1 + Z2 + Z3, Y ~ X + M + Z1 + Z2 + Z3, data = sim_mediation, R = 500)

plot(fit) plot(fit, type = "balance") plot(fit, type = "sensitivity") diagnose(fit) “‘

Sensitivity interpretation

‘sensitivity_rho()‘ reports how the indirect effect changes as the correlation ‘rho‘ between the mediator-model and outcome-model errors moves away from zero, and the value of ‘rho‘ at which it reaches zero. It applies to linear models only and assumes the treatment is ignorable given the covariates.

Author(s)

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

Authors:

- Subir Hait <haitsubi@msu.edu> (ORCID)

References

Imai, K., Keele, L., & Yamamoto, T. (2010). Identification, inference and sensitivity analysis for causal mediation effects. *Statistical Science*, 25(1), 51–71.

See Also

Useful links:

- https://github.com/subirhait/RobustMediate
- Report bugs at https://github.com/subirhait/RobustMediate/issues

adjust_paths *Adjust Both Paths for a Confounder of Given Strength*

Description

Returns the adjusted path t-statistics and the adjusted Sobel statistic for a hypothetical confounder with the given partial R^2 values. Use it when the three values should differ, for example to explore a confounder that is strongly related to the mediator but only weakly to the treatment. See [sensitivity_paths()] for the definitions.

Usage

```
adjust_paths(x, r2_xu, r2_mu, r2_yu)
```

Arguments

x	A 'robmed_paths' object from [sensitivity_paths()].
r2_xu, r2_mu, r2_yu	Partial R^2 of the confounder with the treatment, the mediator, and the outcome. Vectors are recycled.

Value

A data frame with the inputs and columns 't_a', 't_b', 'z_sobel', 'sig_joint', and 'sig_sobel'.

Examples

```
data(sim_mediation)
m_fit <- lm(M ~ X + Z1 + Z2 + Z3, data = sim_mediation)
y_fit <- lm(Y ~ X + M + Z1 + Z2 + Z3, data = sim_mediation)
sp <- sensitivity_paths(mediator_model = m_fit, outcome_model = y_fit,
                        treatment = "X", mediator = "M")
adjust_paths(sp, r2_xu = 0.02, r2_mu = 0.10, r2_yu = 0.05)
```

---

as.data.frame.robmedfit

Coerce effects curve to a data frame

Description

Convenience accessor returning the NDE/NIE/TE curve data frame.

Usage

```
## S3 method for class 'robmedfit'
as.data.frame(x, ...)
```

Arguments

x	A ‘robmedfit‘ object.
...	Ignored.

Value

A data frame with columns ‘dose‘, ‘estimand‘, ‘estimate‘, ‘lower‘, ‘upper‘.

---

augment.robmedfit *Augment data with fitted values from a robmedfit object (broom-compatible)*

Description

Returns the original data augmented with the balance weights, fitted mediator values, and fitted outcome values from the pathway models.

Usage

```
## S3 method for class 'robmedfit'
augment(x, data = NULL, ...)
```

Arguments

x	A ‘robmedfit‘ object.
data	Optional data frame to augment. Defaults to the data stored in the outcome model’s ‘$data‘ slot.
...	Ignored.

Value

The original data frame with additional columns:

‘.ipw_weight‘ Stabilised inverse probability weights from the treatment model (the weights used in the balance diagnostics).

‘.fitted_mediator‘ Predicted mediator values.

‘.fitted_outcome‘ Predicted outcome values.

‘.resid_mediator‘ Residuals from the mediator model.

‘.resid_outcome‘ Residuals from the outcome model.

Examples

```
fit <- robustmediate(
  treatment_formula = X ~ Z1 + Z2 + Z3,
  mediator_formula  = M ~ X + Z1 + Z2 + Z3,
  outcome_formula   = Y ~ X + M + Z1 + Z2 + Z3,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
aug <- augment(fit)
hist(aug$.ipw_weight)
```

---

compare_fits *Compare Two robmedfit Objects Side by Side*

Description

Overlays the NDE/NIE/TE curves from two ‘robmedfit‘ objects on the same panel. Useful for comparing model specifications (e.g. different covariate sets).

Usage

```
compare_fits(
  fit1,
  fit2,
  label1 = "Model 1",
  label2 = "Model 2",
  estimands = c("NDE", "NIE")
)
```

Arguments

fit1	First ‘robmedfit‘ object.
fit2	Second ‘robmedfit‘ object.
label1	Label for ‘fit1‘. Default ‘"Model 1"‘.
label2	Label for ‘fit2‘. Default ‘"Model 2"‘.
estimands	Estimands to display. Default ‘c("NDE","NIE")‘.

Value

A ‘ggplot2‘ object.

Examples

```
fit_a <- robustmediate(
  treatment_formula = X ~ Z1 + Z2 + Z3,
  mediator_formula  = M ~ X + Z1 + Z2 + Z3,
  outcome_formula   = Y ~ X + M + Z1 + Z2 + Z3,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
fit_b <- robustmediate(
  treatment_formula = X ~ Z1 + Z2,
  mediator_formula  = M ~ X + Z1 + Z2,
  outcome_formula   = Y ~ X + M + Z1 + Z2,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
compare_fits(fit_a, fit_b, label1 = "With Z3", label2 = "Without Z3")
```

---

diagnose *Diagnose a robmedfit Object*

Description

Prints a formatted diagnostics report covering balance, mediation effects, and sensitivity robustness. The output is structured so that it can be used directly (or with minimal editing) in the Results section of an applied paper. Returns the underlying results invisibly.

Usage

```
diagnose(x, ...)
```

Arguments

x	A ‘robmedfit‘ object.
...	Ignored.

Value

Invisibly returns a list with elements ‘balance‘, ‘effects‘, and ‘sensitivity‘.

`effect_curvature` *Curvature of an Effect Curve, with Bootstrap Intervals*

Description

Describes the shape of an estimated effect curve (NIE, NDE, or TE) over the treatment grid: the second derivative of the curve at each interior dose, with percentile bootstrap intervals taken from the same bootstrap replicates as the effect estimates.

This is a description of the dose-response shape. It is not a sensitivity analysis for unmeasured confounding; for that see [sensitivity_rho()] and [sensitivity_paths()].

'sensitivity_curvature()' is deprecated; use 'effect_curvature()'.

Usage

```
effect_curvature(x, estimand = c("NIE", "NDE", "TE"), alpha = NULL)

sensitivity_curvature(x, estimand = c("NIE", "NDE", "TE"))
```

Arguments

`x`	A 'robmedfit' object.
`estimand`	'"NIE"' (default), '"NDE"', or '"TE"'.
`alpha`	Level for the curvature intervals. Defaults to the level used in the fit.

Details

When the mediator and outcome models are linear (identity links) and the treatment and mediator enter only as main effects, every effect curve is a straight line by construction. The function then returns curvature exactly zero, with no interval, and says so in a message, instead of differencing rounding error. To let the curve bend, put a spline of the treatment in the model formulas, for example 'M ~ splines::ns(X, df = 3) + Z1'.

Otherwise the second derivative is computed by second differences on the (possibly unequally spaced) dose grid, for the point estimate and for each bootstrap replicate. Values at the two end doses are 'NA'.

The curvature is that of the fitted models, so it depends on the spline chosen. Natural splines are linear beyond their boundary knots, so with few degrees of freedom the curvature is pulled toward zero near the ends of the treatment range. The intervals reflect sampling variability only, not this modelling choice; compare a few spline settings before interpreting shape. Fits created before version 0.3.0 do not store replicate curves, so intervals are 'NA'.

Value

A data frame of class '"robmed_curvature"' with columns 'dose', 'estimate', 'lower', 'upper' (the effect curve and its interval), 'curvature', 'curv_lower', 'curv_upper', and 'ci_includes_zero' (whether the effect interval includes zero). Attributes 'estimand' and 'linear_by_construction' record the settings.

See Also

[plot_curvature()]

Examples

```
data(sim_mediation)
fit <- robustmediate(
  X ~ Z1 + Z2, M ~ splines::ns(X, df = 3) + Z1 + Z2, Y ~ X + M + Z1 + Z2,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
curv <- effect_curvature(fit, estimand = "NIE")
plot_curvature(curv, ref_dose = fit$meta$ref_dose)
```

---

glance.robmedfit *Glance at a robmedfit object (broom-compatible)*

Description

Returns a one-row summary of the fit: sample size, bootstrap reps, reference dose, percentage mediated, the value of ‘rho‘ at which the indirect effect is zero (‘NA‘ when the sensitivity analysis does not apply), and maximum post-weighting SMDs.

Usage

```
## S3 method for class 'robmedfit'
glance(x, ...)
```

Arguments

x	A ‘robmedfit‘ object.
...	Ignored.

Value

A one-row data frame.

Examples

```
fit <- robustmediate(
  treatment_formula = X ~ Z1 + Z2 + Z3,
  mediator_formula  = M ~ X + Z1 + Z2 + Z3,
  outcome_formula   = Y ~ X + M + Z1 + Z2 + Z3,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
```

```
glance(fit)
```

`plot.robmedfit` *Plot a robmedfit object*

Description

Dispatches to the appropriate plot function based on ‘type‘.

Usage

```
## S3 method for class 'robmedfit'
plot(x, type = c("mediation", "balance", "sensitivity", "curvature"), ...)
```

Arguments

`x`	A ‘robmedfit‘ object.
`type`	Which plot to produce. One of: **‘"mediation"‘** Dose-response curve of NDE/NIE/TE (default). **‘"balance"‘** Dual love plot of covariate balance. **‘"sensitivity"‘** Indirect effect as a function of ‘rho‘ (see [sensitivity_rho()]). **‘"curvature"‘** Effect curve and its curvature (see [effect_curvature()]).
`...`	Passed to the underlying plot function.

Value

A ‘ggplot2‘ object.

`plot_balance` *Love Plot: Balance Diagnostics for Both Pathways*

Description

Produces a love plot of standardised mean differences (SMDs) before and after weighting by the stabilised treatment-model weights. For the treatment pathway, covariates are compared between units above and below the median treatment. For the mediator pathway, they are compared between units above and below their predicted mediator value; this is a rough screening check, not a formal balance criterion for the mediator model.

Usage

```
plot_balance(x, threshold = 0.1, pathways = c("treatment", "mediator"), ...)
```

Arguments

x	A ‘robmedfit‘ object.
threshold	Absolute SMD threshold displayed as dashed reference lines. A common rule of thumb is |SMD| < 0.10. Default ‘0.1‘.
pathways	Character vector indicating which pathways to show. Options: ‘"treatment"‘, ‘"mediator"‘, or both (default).
...	Ignored (for S3 consistency).

Value

A ‘ggplot2‘ object. Add layers or themes as usual.

Examples

```
fit <- robustmediate(
  treatment_formula = X ~ Z1 + Z2 + Z3,
  mediator_formula  = M ~ X + Z1 + Z2 + Z3,
  outcome_formula   = Y ~ X + M + Z1 + Z2 + Z3,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
plot_balance(fit)
plot_balance(fit, threshold = 0.05, pathways = "treatment")
```

---

plot_curvature *Plot an Effect Curve and Its Curvature*

Description

Two panels: the effect curve with its bootstrap interval, and its second derivative with a bootstrap interval. Doses where the effect interval includes zero are shaded.

Usage

```
plot_curvature(x, estimand = NULL, ref_dose = NULL, ...)
```

Arguments

x	Output of [effect_curvature()].
estimand	Label for the estimand. Defaults to the one stored in ‘x‘.
ref_dose	Optional reference dose, drawn as a vertical line.
...	Ignored.

Value

A ‘ggplot2‘ object.

Examples

```
data(sim_mediation)
fit <- robustmediate(
  X ~ Z1 + Z2, M ~ splines::ns(X, df = 3) + Z1 + Z2, Y ~ X + M + Z1 + Z2,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
plot_curvature(effect_curvature(fit), ref_dose = fit$meta$ref_dose)
```

---

plot_mediation *Dose-Response Curve: Natural Direct and Indirect Effects*

Description

Plots NDE, NIE, and (optionally) total effect as smooth spline curves over the full range of treatment values, with pointwise bootstrap confidence bands. This is the signature visualisation of **Robust-Mediate** and is publication-ready out of the box.

Usage

```
plot_mediation(
  x,
  estimands = c("NDE", "NIE"),
  show_total = FALSE,
  facet = FALSE,
  ...
)
```

Arguments

x	A ‘robmedfit‘ object.
estimands	Character vector of estimands to display. Any subset of ‘c("NDE", "NIE", "TE")‘. Default ‘c("NDE","NIE")‘.
show_total	Shorthand for adding ‘"TE"‘ to ‘estimands‘. Default ‘FALSE‘.
facet	Logical. Split estimands into separate facets? Default ‘FALSE‘.
...	Ignored.

Value

A ‘ggplot2‘ object.

Examples

```
fit <- robustmediate(
  treatment_formula = X ~ Z1 + Z2 + Z3,
  mediator_formula  = M ~ X + Z1 + Z2 + Z3,
  outcome_formula   = Y ~ X + M + Z1 + Z2 + Z3,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
plot_mediation(fit)
plot_mediation(fit, estimands = c("NDE", "NIE", "TE"), facet = TRUE)
```

`plot_paths` *Plot Adjusted Path and Sobel Statistics Against Confounder Strength*

Description

Plots the adjusted t-statistics of the two paths and the adjusted Sobel statistic as the confounder strength ρ grows (all three partial R^2 values equal to ρ). Dashed vertical lines mark the robustness values for the joint test and the Sobel test.

Usage

```
plot_paths(x, ...)
```

Arguments

`x`	A ‘robmed_paths‘ object from [sensitivity_paths()].
`...`	Ignored.

Value

A ‘ggplot2‘ object.

Examples

```
data(sim_mediation)
m_fit <- lm(M ~ X + Z1 + Z2 + Z3, data = sim_mediation)
y_fit <- lm(Y ~ X + M + Z1 + Z2 + Z3, data = sim_mediation)
plot_paths(sensitivity_paths(mediator_model = m_fit, outcome_model = y_fit,
                             treatment = "X", mediator = "M"))
```

plot_sensitivity *Plot the Indirect Effect as a Function of rho*

Description

Plots the indirect effect from [sensitivity_rho()] against the sensitivity parameter ‘rho‘, with the point where the indirect effect is zero marked.

Usage

```
plot_sensitivity(x, ...)
```

Arguments

x	A ‘robmedfit‘ object or a ‘robmed_sens‘ object.
...	Passed to [sensitivity_rho()] when ‘x‘ is a ‘robmedfit‘ object.

Value

A ‘ggplot2‘ object.

Examples

```
data(sim_mediation)
fit <- robustmediate(
  X ~ Z1 + Z2, M ~ X + Z1 + Z2, Y ~ X + M + Z1 + Z2,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
plot_sensitivity(fit)
```

print.robmedfit *Print a robmedfit object*

Description

Print a robmedfit object

Usage

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

Arguments

x	A ‘robmedfit‘ object returned by [robustmediate()].
...	Ignored.

Value

The input object, invisibly.

---

print.robmed_paths *Print a robmed_paths object*

Description

Print a robmed_paths object

Usage

```
## S3 method for class 'robmed_paths'
print(x, digits = 3, ...)
```

Arguments

x	A ‘robmed_paths‘ object from [sensitivity_paths()].
digits	Number of digits to print. Default ‘3‘.
...	Ignored.

Value

The input object, invisibly.

---

print.robmed_sens *Print a robmed_sens object*

Description

Print a robmed_sens object

Usage

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

Arguments

x	A ‘robmed_sens‘ object from [sensitivity_rho()].
...	Ignored.

Value

The input object, invisibly.

`robustmediate` *Robust Causal Mediation Analysis*

Description

Fits treatment, mediator, and outcome models for causal mediation analysis with continuous treatments and returns a precomputed ‘robmedfit‘ object for plotting and diagnostics.

Natural direct and indirect effects are estimated by g-computation from the fitted mediator and outcome models, averaging over the observed covariates. When the outcome model is linear in the mediator (identity link, mediator entering as a linear term), the predicted mediator mean is plugged in, which is exact. Otherwise the outcome prediction is averaged over the mediator's distribution: exactly for a binary mediator, and by a 15-point quantile rule for a Gaussian mediator. Stabilised inverse probability weights from the treatment model are used for the balance diagnostics only.

Usage

```
robustmediate(
  treatment_formula,
  mediator_formula,
  outcome_formula,
  data,
  ref_dose = NULL,
  dose_grid = NULL,
  R = 500,
  alpha = 0.05,
  covariates = NULL,
  cluster_var = NULL,
  family_treatment = stats::gaussian(),
  family_mediator = stats::gaussian(),
  family_outcome = stats::gaussian(),
  spline_df = NULL,
  rho_seq = seq(-0.9, 0.9, by = 0.01),
  verbose = TRUE
)
```

Arguments

`treatment_formula`
: Formula for the treatment model (for example, ‘X ~ Z1 + Z2‘).

`mediator_formula`
: Formula for the mediator model (for example, ‘M ~ X + Z1 + Z2‘).

`outcome_formula`
: Formula for the outcome model (for example, ‘Y ~ X + M + Z1 + Z2‘).

`data`	A data frame containing all analysis variables.
`ref_dose`	Reference dose value. Defaults to the sample mean of the treatment variable.
`dose_grid`	Numeric vector of dose values over which NDE, NIE, and TE are evaluated. Defaults to 100 evenly spaced points across the observed treatment range.
`R`	Number of bootstrap replicates. Default is ‘500‘.
`alpha`	Significance level. Default is ‘0.05‘.
`covariates`	Covariate names for balance diagnostics. If ‘NULL‘, covariates are inferred from the treatment formula.
`cluster_var`	Optional clustering variable name. When supplied, the bootstrap resamples whole clusters, so intervals reflect within-cluster dependence. ‘NULL‘ assumes independent observations.
`family_treatment`	GLM family for the treatment model. Default is ‘stats::gaussian()‘.
`family_mediator`	GLM family for the mediator model. Default is ‘stats::gaussian()‘.
`family_outcome`	GLM family for the outcome model. Default is ‘stats::gaussian()‘.
`spline_df`	Deprecated and ignored. To allow a nonlinear dose-response, put a spline of the treatment in the formulas, for example ‘M ~ splines::ns(X, df = 3) + Z1‘.
`rho_seq`	Values of the sensitivity parameter ‘rho‘ passed to [sensitivity_rho()]. Default is ‘seq(-0.9, 0.9, by = 0.01)‘.
`verbose`	Logical; if ‘TRUE‘, display progress messages.

Value

An object of class ‘"robmedfit"‘ containing:

‘models‘ Fitted treatment, mediator, and outcome models.

‘balance‘ Balance statistics before and after weighting.

‘effects‘ Dose-response summaries for NDE, NIE, and TE, including bootstrap intervals.

‘sensitivity‘ A ‘robmed_sens‘ object from [sensitivity_rho()], or ‘NULL‘ when the models are not linear (the closed-form result does not apply).

‘cluster‘ Cluster variable, number of clusters, and cluster sizes, or ‘NULL‘ if clustering was not used.

‘meta‘ Call, variable names, dose settings, bootstrap settings, and sample information.

Examples

```
n <- 400
Z1 <- rnorm(n)
Z2 <- rbinom(n, 1, 0.5)
X  <- 0.5 * Z1 + 0.3 * Z2 + rnorm(n)
M  <- 0.4 * X + 0.2 * Z1 + rnorm(n)
Y  <- 0.3 * X + 0.5 * M + 0.1 * Z1 + rnorm(n)
dat <- data.frame(Y, X, M, Z1, Z2)
```

```
fit <- robustmediate(
  treatment_formula = X ~ Z1 + Z2,
  mediator_formula = M ~ X + Z1 + Z2,
  outcome_formula = Y ~ X + M + Z1 + Z2,
  data = dat,
  R = 10, verbose = FALSE
)

print(fit)
```

`sensitivity_paths` *Path-Wise and Indirect-Effect Robustness Values*

Description

Sensitivity analysis for the two paths of a linear mediation model and for the indirect effect, in the partial R^2 metric of Cinelli and Hazlett (2020).

One unmeasured confounder 'U' is described by three partial R^2 values, all conditional on the observed covariates 'C':

- 'r2_xu': $R^2_{X\sim U|C}$, with the treatment;
- 'r2_mu': $R^2_{M\sim U|X,C}$, with the mediator;
- 'r2_yu': $R^2_{Y\sim U|M,X,C}$, with the outcome.

The a-path (treatment to mediator) depends on 'r2_xu' and 'r2_mu'. The b-path (mediator to outcome) depends on 'r2_mu' and 'r2_yu'. Because 'r2_mu' enters both paths, one confounder related to the mediator weakens both paths at once.

A robustness value (RV) is the smallest ρ such that a confounder with every partial R^2 equal to ρ changes the conclusion. The function reports:

Path RVs 'rv_q1' reduces the path estimate to zero; 'rv_alpha' makes the path test non-significant (Cinelli and Hazlett, 2020).

Indirect, joint test The indirect effect is declared significant when both paths are. Its RV equals the smaller of the two path RVs.

Indirect, Monte Carlo CI The indirect effect is declared significant when the Monte Carlo confidence interval for ab (MacKinnon, Lockwood, and Williams, 2004), built from the adjusted path estimates and standard errors, excludes zero. The draws are fixed by 'mc_seed', so the value is reproducible.

Indirect, Sobel test The indirect effect is tested by $z = t_a t_b / \sqrt{t_a^2 + t_b^2}$. Because this z is below both path t-statistics, its RV is below the smaller path RV. Most of that gap reflects the known conservativeness of the Sobel test rather than confounding.

In simulations with normal, constant-variance errors the Monte Carlo and percentile bootstrap RVs were close to the joint-test RV (median shortfall at most about 8 percent, largest when both paths were weak and similar in strength), while the Sobel RV could be much smaller. When the errors are heteroskedastic, the OLS standard errors understate sampling variability and every OLS-based RV (joint, Monte Carlo, Sobel) overstates robustness; the bootstrap RV ('boot > 0') does not rely on the OLS standard errors in the same way and is then the better summary.

Usage

```
sensitivity_paths(
  x = NULL,
  alpha = 0.05,
  mediator_model = NULL,
  outcome_model = NULL,
  treatment = NULL,
  mediator = NULL,
  rho_grid = seq(0, 0.6, by = 0.005),
  mc_draws = 20000L,
  mc_seed = 1L,
  boot = 0L,
  boot_seed = 1L
)
```

Arguments

x	A ‘robmedfit‘ object. Alternatively leave ‘x‘ as ‘NULL‘ and supply ‘mediator_model‘, ‘outcome_model‘, ‘treatment‘, and ‘mediator‘.
alpha	Significance level. Default ‘0.05‘.
mediator_model, outcome_model	Fitted ‘lm‘ or Gaussian ‘glm‘ models for the mediator and the outcome. Used when ‘x‘ is ‘NULL‘.
treatment, mediator	Names of the treatment and mediator variables. Used when ‘x‘ is ‘NULL‘.
rho_grid	Values of ρ for the robustness curve stored in the result and used by [plot_paths()].
mc_draws	Number of Monte Carlo draws for the Monte Carlo CI. Default ‘20000‘.
mc_seed	Seed for the Monte Carlo draws. The global random-number state is left unchanged.
boot	Number of bootstrap resamples for a percentile-bootstrap robustness value. Default ‘0‘ (not computed). In each resample both paths are refitted and the bias adjustment is applied with that resample's standard errors, as in Zhang and Ding. Use at least 1,000 for reporting.
boot_seed	Seed for the bootstrap. The global random-number state is left unchanged.

Details

The adjusted t-statistic for a coefficient with t-statistic t and residual degrees of freedom df, given treatment-side and outcome-side partial R^2 values R^2_D and R^2_Y, is

$$t_{adj} = \frac{|t| - \sqrt{df}\sqrt{R^2_Y R^2_D/(1-R^2_D)}}{\sqrt{(1-R^2_Y)/(1-R^2_D)}\sqrt{df/(df-1)}}$$

assuming the confounder biases the estimate toward zero. Path tests use the t critical value with $df - 1$ degrees of freedom; the Sobel test uses the normal critical value.

The result applies to linear (Gaussian, identity-link) mediator and outcome models with the treatment and mediator entering as main effects.

Value

An object of class ‘"robmed_paths"‘: a list with

‘table‘ Data frame with one row per path and per indirect-effect test: ‘estimate‘, ‘se‘, ‘stat‘, ‘df‘, ‘rv_q1‘, ‘rv_alpha‘.

‘overstatement‘ RV(joint) / RV(Sobel).

‘ratio_mc‘ RV(joint) / RV(Monte Carlo).

‘ratio_boot‘, ‘se_ratio‘ When ‘boot > 0‘: RV(joint) / RV(bootstrap), and the ratio of the bootstrap standard deviation of each path estimate to its OLS standard error. Ratios well above 1 (for example, from heteroskedastic errors) mean the OLS-based robustness values, including the joint, Monte Carlo, and Sobel values, overstate robustness.

‘curve‘ Data frame of adjusted ‘t_a‘, ‘t_b‘, and Sobel ‘z‘ over ‘rho_grid‘.

‘inputs‘ t-statistics, degrees of freedom, and critical values.

References

Cinelli, C., & Hazlett, C. (2020). Making sense of sensitivity: Extending omitted variable bias. *Journal of the Royal Statistical Society: Series B*, 82(1), 39–67. doi:10.1111/rssb.12348

MacKinnon, D. P., Lockwood, C. M., & Williams, J. (2004). Confidence limits for the indirect effect: Distribution of the product and resampling methods. *Multivariate Behavioral Research*, 39(1), 99–128.

Sobel, M. E. (1982). Asymptotic confidence intervals for indirect effects in structural equation models. *Sociological Methodology*, 13, 290–312.

See Also

[adjust_paths()], [plot_paths()], [sensitivity_rho()]

Examples

```
data(sim_mediation)
m_fit <- lm(M ~ X + Z1 + Z2 + Z3, data = sim_mediation)
y_fit <- lm(Y ~ X + M + Z1 + Z2 + Z3, data = sim_mediation)
sp <- sensitivity_paths(mediator_model = m_fit, outcome_model = y_fit,
                        treatment = "X", mediator = "M")
sp
plot_paths(sp)
```

`sensitivity_rho` *Sensitivity of the Indirect Effect to Mediator-Outcome Confounding*

Description

Implements the sensitivity analysis of Imai, Keele, and Yamamoto (2010) for the linear structural equation model. The sensitivity parameter ‘rho‘ is the correlation between the error terms of the mediator model and the outcome model. Sequential ignorability implies ‘rho = 0‘. A nonzero ‘rho‘ arises when an unmeasured pre-treatment variable affects both the mediator and the outcome.

For each value of ‘rho‘, the indirect effect per unit change in treatment is

$$\delta(\rho) = \frac{\beta_2 \sigma_1}{\sigma_2} \left\{ \tilde{\rho} - \rho \sqrt{\frac{1 - \tilde{\rho}^2}{1 - \rho^2}} \right\}$$

where β_2 is the treatment coefficient in the mediator model, σ_2 is the residual standard deviation of the mediator model, σ_1 is the residual standard deviation of the outcome model with the mediator removed, and $\tilde{\rho}$ is the correlation between those two sets of residuals. At ‘rho = 0‘ this equals the product of the treatment coefficient in the mediator model and the mediator coefficient in the outcome model. The indirect effect is zero exactly when ‘rho‘ equals $\tilde{\rho}$.

The same threshold can be stated in variance-explained terms. If an unmeasured confounder explains a proportion R^{2*}_M of the residual variance of the mediator and R^{2*}_Y of the residual variance of the outcome, then $\rho^2 = R^{2*}_M R^{2*}_Y$.

Usage

```
sensitivity_rho(x, rho_seq = seq(-0.9, 0.9, by = 0.01), contrast = NULL)
```

Arguments

x	A ‘robmedfit‘ object returned by [robustmediate()].
rho_seq	Values of ‘rho‘ at which to evaluate the indirect effect. Default ‘seq(-0.9, 0.9, by = 0.01)‘.
contrast	Size of the treatment contrast. Default is the focal contrast reported by [robustmediate()] (focal dose minus reference dose).

Details

The closed form holds only for the linear model. The function therefore requires a Gaussian identity-link mediator and outcome model, a treatment that enters both models as a single linear term, and no treatment-by-mediator interaction. It treats the treatment assignment as ignorable given the covariates; it addresses only confounding of the mediator-outcome relationship. Results are point estimates; they do not include sampling uncertainty.

Value

An object of class ‘"robmed_sens"‘: a list with

‘curve‘ Data frame with columns ‘rho‘, ‘nie_unit‘ (indirect effect per unit of treatment), and ‘nie‘ (indirect effect for ‘contrast‘).

‘rho_zero‘ Value of ‘rho‘ at which the indirect effect is zero.

‘r2_product_zero‘ $R^{2*}_M R^{2*}_Y$ at which the indirect effect is zero (equal to ‘rho_zero^2‘).

‘nie_unit_0‘, ‘contrast‘ Indirect effect per unit at ‘rho = 0‘, and the contrast used.

References

Imai, K., Keele, L., & Yamamoto, T. (2010). Identification, inference and sensitivity analysis for causal mediation effects. *Statistical Science*, 25(1), 51–71.

Imai, K., Keele, L., & Tingley, D. (2010). A general approach to causal mediation analysis. *Psychological Methods*, 15(4), 309–334.

Examples

```
data(sim_mediation)
fit <- robustmediate(
  X ~ Z1 + Z2, M ~ X + Z1 + Z2, Y ~ X + M + Z1 + Z2,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
sens <- sensitivity_rho(fit)
sens
plot_sensitivity(fit)
```

`sim_mediation` *Simulated Mediation Study Data*

Description

A synthetic dataset mimicking a clustered education study with a continuous treatment (tutoring hours), a continuous mediator (mid-year test score), and a continuous outcome (end-of-year test score). Designed to illustrate **RobustMediate** with realistic effect sizes and non-trivial confounding.

Usage

```
sim_mediation
```

Format

A data frame with 600 rows (30 schools x 20 students) and 7 columns:

school Factor. School identifier (30 levels). Use as ‘cluster_var‘.

Y Numeric. End-of-year test score (outcome).

X Numeric. Tutoring hours received (continuous treatment, >= 0).

M Numeric. Mid-year test score (mediator).

Z1 Numeric. Prior achievement (continuous covariate).

Z2 Integer (0/1). Free-lunch status (binary covariate).

Z3 Numeric. Parental education index (continuous covariate).

True parameter targets

The data-generating process sets:

- **NDE** (X → Y direct path) ~= **0.25**
- **NIE** (X → M → Y path) ~= **0.35**
- **TE** ~= **0.60**
- **

Use these as a ground truth to assess estimation accuracy.

Source

Generated via 'data-raw/generate_sim_data.R'. See that script for the full data-generating process.

Examples

```
data(sim_mediation)
str(sim_mediation)
summary(sim_mediation[, c("Y","X","M")])


fit <- robustmediate(
  treatment_formula = X ~ Z1 + Z2 + Z3,
  mediator_formula  = M ~ X + Z1 + Z2 + Z3,
  outcome_formula   = Y ~ X + M + Z1 + Z2 + Z3,
  data        = sim_mediation,
  cluster_var = "school",
  R           = 10,
  verbose     = FALSE
)
diagnose(fit)
```

---

summary.robmedfit *Summary method for robmedfit objects*

Description

Summary method for robmedfit objects

Usage

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

Arguments

object	A 'robmedfit' object.
...	Ignored.

Value

A list with effect, balance, and sensitivity summaries, invisibly.

`tidy.robmedfit` *Tidy a robmedfit object (broom-compatible)*

Description

Returns a tidy data frame of the mediation effect estimates (NDE, NIE, TE) at the reference dose, with confidence intervals. Compatible with ‘broom::tidy()‘ and the broader ‘tidymodels‘ ecosystem.

Usage

```
## S3 method for class 'robmedfit'
tidy(x, conf.int = TRUE, ...)
```

Arguments

`x`	A ‘robmedfit‘ object.
`conf.int`	Logical. Include confidence interval columns? Default ‘TRUE‘.
`...`	Ignored.

Value

A data frame with columns ‘term‘, ‘estimate‘, ‘conf.low‘, ‘conf.high‘, and ‘ref_dose‘.

Examples

```
fit <- robustmediate(
  treatment_formula = X ~ Z1 + Z2 + Z3,
  mediator_formula  = M ~ X + Z1 + Z2 + Z3,
  outcome_formula   = Y ~ X + M + Z1 + Z2 + Z3,
  data = sim_mediation, R = 10, verbose = FALSE,
  dose_grid = seq(-2, 2, length.out = 25)
)
tidy(fit)
```

Index

∗ **datasets**
sim_mediation, 22

adjust_paths, 4
as.data.frame.robmedfit, 4
augment.robmedfit, 5

compare_fits, 6

diagnose, 7

effect_curvature, 8

glance.robmedfit, 9

plot.robmedfit, 10
plot_balance, 10
plot_curvature, 11
plot_mediation, 12
plot_paths, 13
plot_sensitivity, 14
print.robmed_paths, 15
print.robmed_sens, 15
print.robmedfit, 14

RobustMediate *(*RobustMediate-package*)*, 2
robustmediate, 16
RobustMediate-package, 2

sensitivity_curvature
*(*effect_curvature*)*, 8
sensitivity_paths, 18
sensitivity_rho, 20
sim_mediation, 22
summary.robmedfit, 23

tidy.robmedfit, 24