Package ‘lavinteract’

November 4, 2025

Type Package

Title Post-Estimation Utilities for 'lavaan' Fitted Models

Version 0.2.2

Description Companion toolbox for structural equation models fitted with 'lavaan'. Provides post-estimation diagnostics and graphics that operate directly on a fitted object using its estimates and covariance, and refits auxiliary models when needed. The package relies on 'lavaan' (Rosseel, 2012) <doi:10.18637/jss.v048.i02>.

URL `https://github.com/g-corbelli/lavinteract`

BugReports `https://github.com/g-corbelli/lavinteract/issues`

License GPL-3

Encoding UTF-8

Imports lavaan, rlang, ggplot2, stats

Suggests testthat (>= 3.0.0), knitr, rmarkdown

Language en-US

NeedsCompilation no

Maintainer Giuseppe Corbelli <`giuseppe.corbelli@uniroma1.it`>

Config/testthat/edition 3

RoxygenNote 7.3.2

Author Giuseppe Corbelli [aut, cre] (ORCID: <`https://orcid.org/0000-0002-2864-3548`>)

Repository CRAN

Date/Publication 2025-11-04 11:10:02 UTC

Contents

`lavinteract` *Post-Estimation Utilities for 'lavaan' Fitted Models*

Description

Companion toolbox for structural equation models fitted with 'lavaan'. Operates directly on a fitted object using its estimates and covariance. Refits auxiliary models when needed to compute estimates, diagnostics, and plots.

Details

The functions are:

- `lav_slopes`: simple slopes and interaction plots from a fitted 'lavaan' model.
- `lav_vif`: variance inflation factors for structural predictors with measurement preserved.

Note

The development of this package grew from ongoing discussions and interactions (sic) with colleagues, in particular Dr. Cataldo Giuliano Gemmano, whose steady feedback and support helped shape it.

Author(s)

Giuseppe Corbelli (<giuseppe.corbelli@uniroma1.it>, or <giuseppe.corbelli@uninettunouniversity.net>)

See Also

Useful links:

- `https://github.com/g-corbelli/lavinteract`
- Report bugs at `https://github.com/g-corbelli/lavinteract/issues`

`lav_slopes` *Simple slopes and interaction plots for fitted 'lavaan' models*

Description

Computes conditional (simple) slopes of a focal predictor across values of a moderator from a fitted 'lavaan' model that includes their explicit product term. Plots predicted lines with Wald confidence ribbons, and print an APA-style test of the interaction for easy reporting and interpretation, plus a simple-slopes table.

Usage

```
lav_slopes(
  fit,
  outcome,
  pred,
  modx,
  interaction,
  data = NULL,
  modx.values = NULL,
  modx.labels = NULL,
  pred.range = NULL,
  conf.level = 0.95,
  x.label = NULL,
  y.label = NULL,
  legend.title = NULL,
  colors = NULL,
  line.size = 0.80,
  alpha = 0.20,
  table = TRUE,
  digits = 2,
  modx_n_unique_cutoff = 4L,
  return_data = FALSE
)

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

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

Arguments

`fit` A fitted 'lavaan' object that includes the product term (required).

`outcome` Character. Name of the dependent variable in `fit` (required).

`pred` Character. Name of the focal predictor whose simple slopes are probed (required).

`modx` Character. Name of the moderator (required).

`interaction` Character. Name of the product term in `fit` (e.g., `"X_Z"`) (required).

`data` `data.frame`. Raw data. If `NULL`, the function tries to pull data from `fit` via `lavInspect`.

`modx.values` Numeric or character vector. Values or levels of the moderator at which to compute slopes; derived automatically when `NULL`.

`modx.labels` Character vector. Legend/table labels for `modx.values` (default: the character form of `modx.values`).

`pred.range` Numeric length-2. Range `c(min, max)` for the x-axis; uses observed range in `data` when available, else `c(-2, 2)`.

`conf.level` Numeric in (0,1). Confidence level for CIs and ribbons (default: 0.95).

`x.label` Character. X-axis label (default: `pred`).

`y.label` Character. Y-axis label (default: `outcome`).

`legend.title` Character. Legend title; if `NULL`, the legend shows only levels (default: NULL).

`colors` Character vector. Colors for lines and ribbons; named vector recommended with names matching `modx.labels` (default: Okabe-Ito palette).

`line.size` Numeric > 0. Line width (default: 0.80).

`alpha` Numeric in `(0,1)`. Ribbon opacity (default 0.20).

`table` Logical. Print APA-style interaction test and simple-slopes table (default: `TRUE`).

`digits` Integer >= `0`. Decimal digits in printed output (default: 2).

`modx_n_unique_cutoff`
Integer >= 1. Threshold for treating a numeric moderator as continuous and using mean ± SD (default: 4).

`return_data` Logical. If `TRUE`, include the plotting data.frame in the returned list (default: FALSE).

`x` A 'lav_slopes' object.

`...` Additional arguments; unused.

`object` A 'lav_slopes' object.

Details

The model should include a main effect for the predictor, a main effect for the moderator, and their product term. The simple slope of the predictor at a given moderator value combines the predictor main effect with the interaction term. The moderator can be continuous or categorical. Standard errors use the delta method with the model covariance matrix of the estimates.

Value

A list with elements:

`plot` `ggplot` object with lines and confidence ribbons.

`slope_table` Data frame with moderator levels, simple slopes, SE, z, and CI.

`plot_data` Only when `return_data = TRUE`: data used to build the plot.

Notes

Estimates are unstandardized; a standardized beta for the interaction is also reported for reference. Wald tests assume large-sample normality of estimates.

Examples

```
set.seed(42)
X <- rnorm(100); Z <- rnorm(100); X_Z <- X*Z
Y <- 0.6*X + 0.6*Z + 0.3*X_Z + rnorm(100, sd = 0.7)
dataset <- data.frame(Y, X, Z, X_Z)
fit <- lavaan::sem("Y ~ X + Z + X_Z", data = dataset)
```

```
lav_slopes(
fit = fit,
data = dataset,
outcome = "Y",
pred = "X",
modx = "Z",
interaction = "X_Z")
```

lav_vif *Variance Inflation Factors for 'lavaan' Structural Predictors*

Description

Compute VIF for each predictor that appears in structural regressions with two or more predictors, refitting the necessary sub-models so that latent predictors are handled at the latent level (i.e., with their original measurement models). It returns also the R^2 of each eligible endogenous variable from the original fit for context.

Usage

```
lav_vif(
  fit,
  data = NULL,
  quiet = TRUE
)

## S3 method for class 'lav_vif'
print(x, digits = 3, cutoff = c(5, 10), ...)

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

Arguments

`fit`	A fitted `lavaan` object.
`data`	Optional. The data frame used to fit `fit`. If `NULL`, the function attempts to extract the data from `fit` via `lavInspect(fit, "data")` then `"data.original"`.
`quiet`	Logical. If `TRUE` suppresses lavaan refit messages.
`x`	A 'lav_vif' object.
`digits`	Integer number of digits to print.
`cutoff`	Numeric length-2 thresholds used to flag VIF values.
`...`	Passed to ‘print.lav_vif()‘ (e.g., ‘digits‘, ‘cutoff‘).
`object`	A 'lav_vif' object.

Details

Each auxiliary refitted model:

- includes the original measurement model for any latent predictors;
- includes any residual covariances among those indicators that were specified in the original model;
- regresses the focal predictor on the remaining predictors at the latent level when applicable.

VIF_i = 1 / (1 - R^2_i) generalizes VIF to SEM while respecting measurement models.

The function reuses the estimator, missing-data handling, and several options from `fit`.

Value

A list with:

- `vif_table`: data.frame with columns `outcome`, `predictor`, `group`, `r2_predictor`, `vif`, `k_predictors`.
- `outcome_r2`: data.frame with R^2 per eligible endogenous `outcome` and `group` from the original `fit`.

Examples

```
set.seed(42)
x1 <- rnorm(100); x2 <- 0.85*x1 + rnorm(100, sd = sqrt(1 - 0.85^2)); x3 <- rnorm(100)
y  <- 0.5*x1 + 0.3*x2 + 0.1*x3 + rnorm(100, sd = 0.7)
dataset <- data.frame(y, x1, x2, x3)
fit <- lavaan::sem(”y ~ x1 + x2 + x3”, data = dataset)
lav_vif(
fit = fit,
data = dataset)
```

Index