

Package ‘interplot’

June 24, 2026

Title Plot the Effects of Variables in Interaction Terms

Version 1.2.0

Maintainer Yue Hu <yuehu@tsinghua.edu.cn>

Description Plots the conditional coefficients (‘‘marginal effects’’) of variables included in multiplicative interaction terms.

BugReports <https://github.com/sammo3182/interplot/issues>

Encoding UTF-8

Depends R (>= 4.1.0), ggplot2

Imports stats, abind, arm, dplyr, purrr, lme4, interactionTest

License MIT + file LICENSE

Suggests knitr, rmarkdown, mitools, gridExtra, merTools, brms, splines, testthat (>= 3.0.0)

VignetteBuilder knitr

RoxygenNote 8.0.0

Config/testthat/edition 3

NeedsCompilation no

Author Frederick Solt [aut],
Yue Hu [aut, cre],
Brenton Kenkel [ctb]

Repository CRAN

Date/Publication 2026-06-24 10:10:02 UTC

Contents

bin_layer	2
interplot	3
interplot.brmsfit	7
interplot.default	10
interplot.lmerMod	13
interplot.lmmi	16

interplot.mlmmi	19
interplot.plot	22
jn_interval	25
jn_layer	26

Index	27
--------------	-----------

bin_layer	<i>Add a Binning Diagnostic to an interplot</i>
-----------	---

Description

Overlays within-bin estimates of the conditional effect of var1 on an interplot output, following the binning estimator of Hainmueller, Mummolo, and Xu (2019). It provides a visual check of the linear-interaction- effect (LIE) assumption: if the moderating relationship is truly linear, the binned point estimates fall on the interplot line; systematic departures signal a nonlinear conditional effect.

Usage

```
bin_layer(  
  m,  
  var1,  
  var2,  
  ci = 0.95,  
  bins = 3,  
  point_color = "#BD472A",  
  point_shape = 18  
)
```

Arguments

m	A model object of class <code>lm</code> or <code>glm</code> including the interaction of <code>var1</code> and <code>var2</code> .
var1	The name (as a string) of the variable whose conditional effect is plotted.
var2	The name (as a string) of the moderating variable.
ci	A numeric value defining the confidence level. The default is 0.95.
bins	The number of moderator bins (quantile groups). The default is 3 (low / medium / high terciles).
point_color	Color of the binned points and whiskers. Default <code>"#BD472A"</code> .
point_shape	Plotting shape of the binned points. Default 18 (filled diamond).

Details

For each quantile bin of `var2`, the model is refitted on the observations in that bin with `var2` centered at the bin median. The coefficient on `var1` is then its marginal effect evaluated at the bin median, estimated from only that bin’s data; this is algebraically the Hainmueller-Mummolo-Xu L-estimator. Each estimate is drawn as a dot-and-whisker at the bin median.

Bins with singular or failed fits are dropped with a warning.

Value

A list of ggplot2 layers, to be added to an interplot plot with `+`.

Source

Hainmueller, Jens, Jonathan Mummolo, and Yiqing Xu. 2019. "How Much Should We Trust Estimates from Multiplicative Interaction Models? Simple Tools to Improve Empirical Practice." *Political Analysis* 27(2): 163–192.

Examples

```
m <- lm(mpg ~ wt * cyl, data = mtcars)
interplot(m, "cyl", "wt") + bin_layer(m, "cyl", "wt")
```

interplot

Plot Conditional Coefficients of a Variable in an Interaction Term

Description

`interplot` is a generic function to produce a plot of the coefficient estimates of one variable in a two-way interaction conditional on the values of the other variable in the interaction term. The function invokes particular methods which depend on the `class` of the first argument.

Usage

```
interplot(
  m,
  var1,
  var2,
  plot = TRUE,
  steps = NULL,
  ci = 0.95,
  adjCI = FALSE,
  hist = FALSE,
  var2_dt = NA,
  predPro = FALSE,
  var2_vals = NULL,
  point = FALSE,
  sims = 1000,
  xmin = NA,
  xmax = NA,
  ercolor = NA,
  esize = 0.5,
  ralpha = 0.5,
  rfill = "grey70",
  stats_cp = "none",
```

```

    txt_caption = NULL,
    ...
)

```

Arguments

<code>m</code>	A model object including an interaction term, or, alternately, a data frame generated by an earlier call to <code>interplot</code> using the argument <code>plot = FALSE</code> .
<code>var1</code>	The name (as a string) of the variable of interest in the interaction term; its conditional coefficient estimates will be plotted.
<code>var2</code>	The name (as a string) of the other variable in the interaction term
<code>plot</code>	A logical value indicating whether the output is a plot or a dataframe including the conditional coefficient estimates of <code>var1</code> , their upper and lower bounds, and the corresponding values of <code>var2</code> .
<code>steps</code>	Desired length of the sequence. A non-negative number, which for <code>seq</code> and <code>seq.int</code> will be rounded up if fractional. The default is 100 or the unique categories in the <code>var2</code> (when it is less than 100. Also see unique).
<code>ci</code>	A numeric value defining the confidence intervals. The default value is 95% (0.95).
<code>adjCI</code>	A logical value indication if applying the adjustment of confidence intervals to control the false discovery rate following the Esarey and Sumner (2017) procedure. (See also Benjamini and Hochberg 1995.) The default is <code>FALSE</code> ; the plot presents the confidence intervals suggested by Brambor, Clark, and Golder (2006). The functions dealing with multilevel model outputs in this package do not equip with this argument, because there is the controversy on the accurate degrees of freedom for multilevel models, esp. when random effect is engaged and the degrees of freedom is a necessary information to conduct the CI adjustment. See e.g., https://stat.ethz.ch/pipermail/r-help/2006-May/094765.html and https://stat.ethz.ch/pipermail/r-sig-mixed-models/2008q1/000517.html .
<code>hist</code>	A logical value indicating if there is a histogram of ‘ <code>var2</code> ’ added at the bottom of the conditional effect plot.
<code>var2_dt</code>	A numerical value indicating the frequency distribution of ‘ <code>var2</code> ’. It is only used when ‘ <code>hist == TRUE</code> ’. When the object is a model, the default is the distribution of ‘ <code>var2</code> ’ of the model.
<code>predPro</code>	A logical value with default of ‘ <code>FALSE</code> ’. When the ‘ <code>m</code> ’ is an object of class ‘ <code>glm</code> ’ or ‘ <code>glmerMod</code> ’ and the argument is set to ‘ <code>TRUE</code> ’, the function will plot predicted probabilities at the values given by ‘ <code>var2_vals</code> ’.
<code>var2_vals</code>	A numerical value indicating the values the predicted probabilities are estimated, when ‘ <code>predPro</code> ’ is ‘ <code>TRUE</code> ’.
<code>point</code>	A logical value determining the format of plot. By default, the function produces a line plot when <code>var2</code> takes on ten or more distinct values and a point (dot-and-whisker) plot otherwise; option <code>TRUE</code> forces a point plot.
<code>sims</code>	Number of independent simulation draws used to calculate upper and lower bounds of coefficient estimates: lower values run faster; higher values produce smoother curves.

xmin	A numerical value indicating the minimum value shown of x shown in the graph. Rarely used.
xmax	A numerical value indicating the maximum value shown of x shown in the graph. Rarely used.
ercolor	A character value indicating the outline color of the whisker or ribbon.
esize	A numerical value indicating the size of the whisker or ribbon.
ralpha	A numerical value indicating the transparency of the ribbon.
rfill	A character value indicating the filling color of the ribbon.
stats_cp	A character value indicating what statistics to present as the plot note. Three options are available: "none", "ci", and "ks". The default is "none". See the Details for more information.
txt_caption	A character string to add a note for the plot, a value will sending to <code>ggplot2::labs(caption = txt_caption)</code> .
...	Additional arguments passed to the specific <code>interplot</code> method. For <code>lm</code> and <code>glm</code> objects (see interplot.default) these include <code>var3</code> , <code>var3_vals</code> , and <code>facet</code> for three-way interactions. Other <code>ggplot</code> aesthetics arguments for points or lines may also be supplied.

Details

`interplot` visualizes the changes in the coefficient of one term in a two-way interaction conditioned by the other term. In the current version, the function works with interactions in the following classes of models:

- Ordinary linear models (object class: `lm`);
- Generalized linear models (object class: `glm`);
- Linear mixed-effects models (object class: `lmerMod`);
- Generalized linear mixed-effects models (object class: `glmerMod`);
- Ordinary linear models with imputed data (object class: `list`);
- Generalized linear models with imputed data (object class: `list`);
- Linear mixed-effects models with imputed data (object class: `list`);
- Generalized linear mixed-effects models with imputed data (object class: `list`).

For `lm` and `glm` objects, `interplot` additionally supports three-way interactions (supply a third variable via `var3`; see [interplot.default](#)) and nonlinear conditional effects, in which `var1` interacts with a polynomial (e.g. `I(var2^2)`) or spline (e.g. `splines::ns(var2)`) of the moderator. The companion function [bin_layer](#) adds the Hainmueller, Mummolo, and Xu (2019) binning diagnostic as a composable overlay.

The examples below illustrate how methods invoked by this generic deal with different type of objects.

Because the output function is based on [ggplot](#), any additional arguments and layers supported by `ggplot2` can be added with the `+`.

`interplot` visualizes the conditional effect based on simulated marginal effects. The simulation provides a probabilistic distribution of moderation effect of the conditioning variable (`var2`) at every

preset values (including the minimum and maximum values) of the conditioned variable (`var1`), denoted as E_{min} and E_{max} . This output allows the function to further examine the conditional effect statistically in two ways. One is to examine if the distribution of $E_{max} - E_{min}$ covers zero. The other is to directly compare E_{min} and E_{max} through statistical tools for distributional comparisons. Users can choose either method by setting the argument `stats_cp` to "ci" or "ks".

- "ci" provides the confidence interval of the difference of $E_{max} - E_{min}$. An interval including 0 suggests no statistical difference before and after the conditional effect is applied, and vice versa.
- "ks" presents the result of a two-sample Kolmogorov-Smirnov test of the simulated distributions of E_{min} and E_{max} . The output includes a D statistics and a p-value of the null hypothesis that the two distributions come from the same distribution at the 0.05 level.

See an illustration in the package vignette.

Value

The function returns a `ggplot` object.

Source

Aiken, Leona S., and Stephen G. West. 1991. "Multiple Regression: Testing and Interpreting Interactions". Newbury Park, CA: Sage.

Benjamini, Yoav, and Yosef Hochberg. 1995. "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing". *Journal of the Royal Statistical Society, Series B* 57(1): 289–300.

Brambor, Thomas, William Roberts Clark, and Matt Golder. "Understanding interaction models: Improving empirical analyses". *Political Analysis* 14.1 (2006): 63-82.

Esarey, Justin, and Jane Lawrence Sumner. 2015. "Marginal Effects in Interaction Models: Determining and Controlling the False Positive Rate". URL: <https://jee3.web.rice.edu/interaction-overconfidence.pdf>.

Examples

```
data(mtcars)
m_cyl <- lm(mpg ~ wt * cyl, data = mtcars)
library(interplot)

# Plot interactions with a continuous conditioning variable
interplot(m = m_cyl, var1 = 'cyl', var2 = 'wt') +
  xlab('Automobile Weight (thousands lbs)') +
  ylab('Estimated Coefficient for Number of Cylinders') +
  ggtitle('Estimated Coefficient of Engine Cylinders\non Mileage by Automobile Weight') +
  theme(plot.title = element_text(face='bold'))

# Plot interactions with a categorical conditioning variable
interplot(m = m_cyl, var1 = 'wt', var2 = 'cyl') +
  xlab('Number of Cylinders') +
  ylab('Estimated Coefficient for Automobile Weight (thousands lbs)') +
```

```

ggtitle('Estimated Coefficient of Automobile Weight \non Mileage by Engine Cylinders') +
theme(plot.title = element_text(face='bold'))

# Three-way interaction: effect of wt across hp, faceted by cyl
m_3 <- lm(mpg ~ wt * hp * cyl, data = mtcars)
interplot(m_3, var1 = 'wt', var2 = 'hp', var3 = 'cyl')

# Nonlinear conditional effect: wt interacted with a quadratic in cyl
m_q <- lm(mpg ~ wt * (cyl + I(cyl^2)), data = mtcars)
interplot(m_q, var1 = 'wt', var2 = 'cyl')

```

interplot.brmsfit	<i>Plot Conditional Coefficients in Bayesian Models with Interaction Terms</i>
-------------------	--

Description

interplot.brmsfit is a method to calculate conditional coefficient estimates from the posterior draws of a Bayesian regression model fitted with [brm](#) that includes a two-way interaction term.

Usage

```

## S3 method for class 'brmsfit'
interplot(
  m,
  var1,
  var2,
  plot = TRUE,
  steps = NULL,
  ci = 0.95,
  adjCI = FALSE,
  hist = FALSE,
  var2_dt = NA,
  predPro = FALSE,
  var2_vals = NULL,
  point = FALSE,
  sims = 1000,
  xmin = NA,
  xmax = NA,
  ercolor = NA,
  esize = 0.5,
  ralpha = 0.5,
  rfill = "grey70",
  stats_cp = "none",
  txt_caption = NULL,
  facet_labs = NULL,
  var3 = NULL,

```

```

    var3_vals = NULL,
    facet = TRUE,
    ...
)

```

Arguments

<code>m</code>	A model object of class <code>brmsfit</code> including an interaction term.
<code>var1</code>	The name (as a string) of the variable of interest in the interaction term; its conditional coefficient estimates will be plotted.
<code>var2</code>	The name (as a string) of the other variable in the interaction term.
<code>plot</code>	A logical value indicating whether the output is a plot or a dataframe including the conditional coefficient estimates of <code>var1</code> , their upper and lower bounds, and the corresponding values of <code>var2</code> .
<code>steps</code>	Desired length of the sequence. A non-negative number, which for <code>seq</code> and <code>seq.int</code> will be rounded up if fractional. The default is 100 or the unique categories in the <code>var2</code> (when it is less than 100. Also see unique).
<code>ci</code>	A numeric value defining the credible interval. The default value is 95% (0.95). For <code>brmsfit</code> objects the interval summarizes the posterior draws, so it is an (equal-tailed) Bayesian credible interval rather than a frequentist confidence interval.
<code>adjCI</code>	The false-discovery-rate adjustment of Esarey and Sumner (2017) is a frequentist correction and does not apply to Bayesian posteriors. The argument is ignored (with a warning) for <code>brmsfit</code> objects.
<code>hist</code>	A logical value indicating if there is a histogram of ‘ <code>var2</code> ’ added at the bottom of the conditional effect plot.
<code>var2_dt</code>	A numerical value indicating the frequency distribution of ‘ <code>var2</code> ’. It is only used when ‘ <code>hist == TRUE</code> ’. When the object is a model, the default is the distribution of ‘ <code>var2</code> ’ of the model.
<code>predPro</code>	A logical value with default of ‘ <code>FALSE</code> ’. When the ‘ <code>m</code> ’ is fitted with a Bernoulli or binomial family and the argument is set to ‘ <code>TRUE</code> ’, the function plots the posterior expected predicted probabilities at the values given by ‘ <code>var2_vals</code> ’, computed with posterior_epred .
<code>var2_vals</code>	A numerical value indicating the values the predicted probabilities are estimated, when ‘ <code>predPro</code> ’ is ‘ <code>TRUE</code> ’.
<code>point</code>	A logical value determining the format of plot. By default, the function produces a line plot when <code>var2</code> takes on ten or more distinct values and a point (dot-and-whisker) plot otherwise; option <code>TRUE</code> forces a point plot.
<code>sims</code>	Ignored for <code>brmsfit</code> objects: the posterior draws produced by the sampler are used directly instead of re-simulating the coefficients.
<code>xmin</code>	A numerical value indicating the minimum value shown of <code>x</code> shown in the graph. Rarely used.
<code>xmax</code>	A numerical value indicating the maximum value shown of <code>x</code> shown in the graph. Rarely used.

ercolor	A character value indicating the outline color of the whisker or ribbon.
esize	A numerical value indicating the size of the whisker or ribbon.
ralpha	A numerical value indicating the transparency of the ribbon.
rfill	A character value indicating the filling color of the ribbon.
stats_cp	A character value indicating what statistics to present as the plot note. Three options are available: "none", "ci", and "ks". The default is "none". See the Details for more information.
txt_caption	A character string to add a note for the plot, a value will sending to <code>ggplot2::labs(caption = txt_caption)</code> .
facet_labs	An optional character vector of facet labels to be used when plotting an interaction with a factor variable.
var3	An optional name (as a string) of a third variable for a three-way interaction $var1 * var2 * var3$. When supplied, the conditional effect of <code>var1</code> across <code>var2</code> is shown at several values/levels of <code>var3</code> . The default NULL gives the standard two-way behavior. Requires continuous <code>var1</code> and <code>var2</code> .
var3_vals	An optional numeric vector giving the values of a continuous <code>var3</code> to condition on. The default is the mean and the mean plus or minus one standard deviation. Ignored when <code>var3</code> is a factor.
facet	A logical value, used only with <code>var3</code> . TRUE (default) draws one panel per value/level of <code>var3</code> ; FALSE overlays the curves colored by <code>var3</code> .
...	Other ggplot aesthetics arguments for points in the dot-whisker plot or lines in the line-ribbon plots. Not currently used.

Details

`interplot.brmsfit` is an S3 method of `interplot` for models fitted with `brm`. Unlike the frequentist methods, it does not call `arm::sim`: the posterior draws of the population-level (fixed) effects are extracted directly with `as.matrix` and the conditional coefficient $b_{var1} + b_{var1:var2} \cdot var2$ is computed for every draw. Point estimates are posterior means and the bounds are posterior quantiles.

Because the output function is based on `ggplot`, any additional arguments and layers supported by `ggplot2` can be added with the `+`.

The `'brms'` package is only suggested by `'interplot'`; it must be installed for this method to run.

Value

The function returns a `ggplot` object, or a list with the data frame of conditional coefficients when `plot = FALSE`.

Examples

```
## Not run:
library(brms)
data(mtcars)

# A Bayesian linear model with a two-way interaction
```

```

m_brms <- brm(mpg ~ wt * cyl, data = mtcars, chains = 2, refresh = 0)

# Identical interface; the band is a 95% posterior credible interval
interplot(m_brms, var1 = "cyl", var2 = "wt")

# Posterior predicted probabilities for a Bernoulli model
m_brms_bin <- brm(am ~ wt * cyl, data = mtcars,
                  family = bernoulli(), chains = 2, refresh = 0)
interplot(m_brms_bin, var1 = "wt", var2 = "cyl",
          predPro = TRUE, var2_vals = c(4, 6, 8))

## End(Not run)

```

interplot.default	<i>Plot Conditional Coefficients in (Generalized) Linear Models with Interaction Terms</i>
-------------------	--

Description

interplot.default is a method to calculate conditional coefficient estimates from the results of (generalized) linear regression models with interaction terms.

Usage

```

## Default S3 method:
interplot(
  m,
  var1,
  var2,
  plot = TRUE,
  steps = NULL,
  ci = 0.95,
  adjCI = FALSE,
  hist = FALSE,
  var2_dt = NA,
  predPro = FALSE,
  var2_vals = NULL,
  point = FALSE,
  sims = 1000,
  xmin = NA,
  xmax = NA,
  ercolor = NA,
  esize = 0.5,
  ralpha = 0.5,
  rfill = "grey70",
  stats_cp = "none",
  txt_caption = NULL,

```

```

    facet_labs = NULL,
    var3 = NULL,
    var3_vals = NULL,
    facet = TRUE,
    ...
)

```

Arguments

<code>m</code>	A model object including an interaction term, or, alternately, a data frame recording conditional coefficients.
<code>var1</code>	The name (as a string) of the variable of interest in the interaction term; its conditional coefficient estimates will be plotted.
<code>var2</code>	The name (as a string) of the other variable in the interaction term.
<code>plot</code>	A logical value indicating whether the output is a plot or a dataframe including the conditional coefficient estimates of <code>var1</code> , their upper and lower bounds, and the corresponding values of <code>var2</code> .
<code>steps</code>	Desired length of the sequence. A non-negative number, which for <code>seq</code> and <code>seq.int</code> will be rounded up if fractional. The default is 100 or the unique categories in the <code>var2</code> (when it is less than 100. Also see unique).
<code>ci</code>	A numeric value defining the confidence intervals. The default value is 95% (0.95).
<code>adjCI</code>	A logical value indication if applying the adjustment of confidence intervals to control the false discovery rate following the Esarey and Sumner (2017) procedure. (See also Benjamini and Hochberg 1995.) The default is FALSE; the plot presents the confidence intervals suggested by Brambor, Clark, and Golder (2006).
<code>hist</code>	A logical value indicating if there is a histogram of ‘ <code>var2</code> ’ added at the bottom of the conditional effect plot.
<code>var2_dt</code>	A numerical value indicating the frequency distribution of ‘ <code>var2</code> ’. It is only used when ‘ <code>hist == TRUE</code> ’. When the object is a model, the default is the distribution of ‘ <code>var2</code> ’ of the model.
<code>predPro</code>	A logical value with default of ‘FALSE’. When the ‘ <code>m</code> ’ is an object of class ‘ <code>glm</code> ’ and the argument is set to ‘TRUE’, the function will plot predicted probabilities at the values given by ‘ <code>var2_vals</code> ’.
<code>var2_vals</code>	A numerical value indicating the values the predicted probabilities are estimated, when ‘ <code>predPro</code> ’ is ‘TRUE’.
<code>point</code>	A logical value determining the format of plot. By default, the function produces a line plot when <code>var2</code> takes on ten or more distinct values and a point (dot-and-whisker) plot otherwise; option TRUE forces a point plot.
<code>sims</code>	Number of independent simulation draws used to calculate upper and lower bounds of coefficient estimates: lower values run faster; higher values produce smoother curves.
<code>xmin</code>	A numerical value indicating the minimum value shown of <code>x</code> shown in the graph. Rarely used.

xmax	A numerical value indicating the maximum value shown of x shown in the graph. Rarely used.
ercolor	A character value indicating the outline color of the whisker or ribbon.
esize	A numerical value indicating the size of the whisker or ribbon.
ralpha	A numerical value indicating the transparency of the ribbon.
rfill	A character value indicating the filling color of the ribbon.
stats_cp	A character value indicating what statistics to present as the plot note. Three options are available: "none", "ci", and "ks". The default is "none". See the Details for more information.
txt_caption	A character string to add a note for the plot, a value will sending to <code>ggplot2::labs(caption = txt_caption)</code> .
facet_labs	An optional character vector of facet labels to be used when plotting an interaction with a factor variable.
var3	An optional name (as a string) of a third variable for a three-way interaction $\text{var1} * \text{var2} * \text{var3}$. When supplied, the conditional effect of <code>var1</code> across <code>var2</code> is shown at several values/levels of <code>var3</code> . The default NULL gives the standard two-way behavior. Requires continuous <code>var1</code> and <code>var2</code> .
var3_vals	An optional numeric vector giving the values of a continuous <code>var3</code> to condition on. The default is the mean and the mean plus or minus one standard deviation. Ignored when <code>var3</code> is a factor.
facet	A logical value, used only with <code>var3</code> . TRUE (default) draws one panel per value/level of <code>var3</code> ; FALSE overlays the curves colored by <code>var3</code> .
...	Other ggplot aesthetics arguments for points in the dot-whisker plot or lines in the line-ribbon plots. Not currently used.

Details

`interplot.default` is a S3 method from the `interplot`. It works on two classes of objects:

- Ordinary linear models (object class: `lm`);
- Generalized linear models (object class: `glm`).

Because the output function is based on `ggplot`, any additional arguments and layers supported by `ggplot2` can be added with the `+`.

`interplot` visualizes the conditional effect based on simulated marginal effects. The simulation provides a probabilistic distribution of moderation effect of the conditioning variable (`var2`) at every preset values (including the minimum and maximum values) of the conditioned variable (`var1`), denoted as E_{min} and E_{max} . This output allows the function to further examine the conditional effect statistically in two ways. One is to examine if the distribution of $E_{max} - E_{min}$ covers zero. The other is to directly compare E_{min} and E_{max} through statistical tools for distributional comparisons. Users can choose either method by setting the argument `stats_cp` to "ci" or "ks".

- "ci" provides the confidence interval of the difference of $E_{max} - E_{min}$. An interval including 0 suggests no statistical difference before and after the conditional effect is applied, and vice versa.

- "ks" presents the result of a two-sample Kolmogorov-Smirnov test of the simulated distributions of Emin and Emax. The output includes a D statistics and a p-value of the null hypothesis that the two distributions come from the same distribution at the 0.05 level.

See an illustration in the package vignette.

Value

The function returns a ggplot object.

Source

Benjamini, Yoav, and Yosef Hochberg. 1995. "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing". *Journal of the Royal Statistical Society, Series B* 57(1): 289–300.

Brambor, Thomas, William Roberts Clark, and Matt Golder. "Understanding interaction models: Improving empirical analyses". *Political Analysis* 14.1 (2006): 63-82.

Esarey, Justin, and Jane Lawrence Sumner. 2015. "Marginal Effects in Interaction Models: Determining and Controlling the False Positive Rate". URL: <https://jee3.web.rice.edu/interaction-overconfidence.pdf>.

interplot.lmerMod	<i>Plot Conditional Coefficients in Mixed-Effects Models with Interaction Terms</i>
-------------------	---

Description

interplot.mlm is a method to calculate conditional coefficient estimates from the results of multi-level (mixed-effects) regression models with interaction terms.

Usage

```
## S3 method for class 'lmerMod'
interplot(
  m,
  var1,
  var2,
  plot = TRUE,
  steps = NULL,
  ci = 0.95,
  adjCI = FALSE,
  hist = FALSE,
  var2_dt = NA,
  predPro = FALSE,
  var2_vals = NULL,
  point = FALSE,
  sims = 5000,
```

```

    xmin = NA,
    xmax = NA,
    ercolor = NA,
    esize = 0.5,
    ralpha = 0.5,
    rfill = "grey70",
    stats_cp = "none",
    txt_caption = NULL,
    facet_labs = NULL,
    var3 = NULL,
    var3_vals = NULL,
    facet = TRUE,
    ...
  )

```

Arguments

<code>m</code>	A model object including an interaction term, or, alternately, a data frame recording conditional coefficients.
<code>var1</code>	The name (as a string) of the variable of interest in the interaction term; its conditional coefficient estimates will be plotted.
<code>var2</code>	The name (as a string) of the other variable in the interaction term.
<code>plot</code>	A logical value indicating whether the output is a plot or a dataframe including the conditional coefficient estimates of <code>var1</code> , their upper and lower bounds, and the corresponding values of <code>var2</code> .
<code>steps</code>	Desired length of the sequence. A non-negative number, which for <code>seq</code> and <code>seq.int</code> will be rounded up if fractional. The default is 100 or the unique categories in the <code>var2</code> (when it is less than 100. Also see unique).
<code>ci</code>	A numeric value defining the confidence intervals. The default value is 95% (0.95).
<code>adjCI</code>	Not working for ‘lmer’ outputs yet.
<code>hist</code>	A logical value indicating if there is a histogram of ‘ <code>var2</code> ’ added at the bottom of the conditional effect plot.
<code>var2_dt</code>	A numerical value indicating the frequency distribution of ‘ <code>var2</code> ’. It is only used when ‘ <code>hist == TRUE</code> ’. When the object is a model, the default is the distribution of ‘ <code>var2</code> ’ of the model.
<code>predPro</code>	A logical value with default of ‘ <code>FALSE</code> ’. When the ‘ <code>m</code> ’ is an object of class ‘ <code>glmerMod</code> ’ and the argument is set to ‘ <code>TRUE</code> ’, the function will plot predicted probabilities at the values given by ‘ <code>var2_vals</code> ’.
<code>var2_vals</code>	A numerical value indicating the values the predicted probabilities are estimated, when ‘ <code>predPro</code> ’ is ‘ <code>TRUE</code> ’.
<code>point</code>	A logical value determining the format of plot. By default, the function produces a line plot when <code>var2</code> takes on ten or more distinct values and a point (dot-and-whisker) plot otherwise; option <code>TRUE</code> forces a point plot.

sims	Number of independent simulation draws used to calculate upper and lower bounds of coefficient estimates: lower values run faster; higher values produce smoother curves.
xmin	A numerical value indicating the minimum value shown of x shown in the graph. Rarely used.
xmax	A numerical value indicating the maximum value shown of x shown in the graph. Rarely used.
ercolor	A character value indicating the outline color of the whisker or ribbon.
esize	A numerical value indicating the size of the whisker or ribbon.
ralpha	A numerical value indicating the transparency of the ribbon.
rfill	A character value indicating the filling color of the ribbon.
stats_cp	A character value indicating what statistics to present as the plot note. Three options are available: "none", "ci", and "ks". The default is "none". See the Details for more information.
txt_caption	A character string to add a note for the plot, a value will sending to <code>ggplot2::labs(caption = txt_caption)</code> .
facet_labs	An optional character vector of facet labels to be used when plotting an interaction with a factor variable.
var3	An optional name (as a string) of a third variable for a three-way interaction <code>var1 * var2 * var3</code> . When supplied, the conditional effect of <code>var1</code> across <code>var2</code> is shown at several values/levels of <code>var3</code> . The default NULL gives the standard two-way behavior. Requires continuous <code>var1</code> and <code>var2</code> .
var3_vals	An optional numeric vector giving the values of a continuous <code>var3</code> to condition on. The default is the mean and the mean plus or minus one standard deviation. Ignored when <code>var3</code> is a factor.
facet	A logical value, used only with <code>var3</code> . TRUE (default) draws one panel per value/level of <code>var3</code> ; FALSE overlays the curves colored by <code>var3</code> .
...	Other ggplot aesthetics arguments for points in the dot-whisker plot or lines in the line-ribbon plots. Not currently used.

Details

`interplot.mlm` is a S3 method from the `interplot`. It works on mixed-effects objects with class `lmerMod` and `glmerMod`.

Because the output function is based on [ggplot](#), any additional arguments and layers supported by `ggplot2` can be added with the `+`.

`interplot` visualizes the conditional effect based on simulated marginal effects. The simulation provides a probabilistic distribution of moderation effect of the conditioning variable (`var2`) at every preset values (including the minimum and maximum values) of the conditioned variable (`var1`), denoted as E_{min} and E_{max} . This output allows the function to further examine the conditional effect statistically in two ways. One is to examine if the distribution of $E_{max} - E_{min}$ covers zero. The other is to directly compare E_{min} and E_{max} through statistical tools for distributional comparisons. Users can choose either method by setting the argument `stats_cp` to "ci" or "ks".

- "ci" provides the confidence interval of the difference of $E_{max} - E_{min}$. An interval including 0 suggests no statistical difference before and after the conditional effect is applied, and vice versa.
- "ks" presents the result of a two-sample Kolmogorov-Smirnov test of the simulated distributions of E_{min} and E_{max} . The output includes a D statistics and a p-value of the null hypothesis that the two distributions come from the same distribution at the 0.05 level.

See an illustration in the package vignette.

Value

The function returns a ggplot object.

interplot.lmmi	<i>Plot Conditional Coefficients in (Generalized) Linear Models with Imputed Data and Interaction Terms</i>
----------------	---

Description

interplot.mi is a method to calculate conditional coefficient estimates from the results of (generalized) linear regression models with interaction terms and multiply imputed data.

Usage

```
## S3 method for class 'lmmi'
interplot(
  m,
  var1,
  var2,
  plot = TRUE,
  steps = NULL,
  ci = 0.95,
  adjCI = FALSE,
  hist = FALSE,
  var2_dt = NA,
  predPro = FALSE,
  var2_vals = NULL,
  point = FALSE,
  sims = 5000,
  xmin = NA,
  xmax = NA,
  ercolor = NA,
  esize = 0.5,
  ralpha = 0.5,
  rfill = "grey70",
  stats_cp = "none",
  txt_caption = NULL,
```

```

    facet_labs = NULL,
    var3 = NULL,
    var3_vals = NULL,
    facet = TRUE,
    ...
)

```

Arguments

<code>m</code>	A model object including an interaction term, or, alternately, a data frame recording conditional coefficients.
<code>var1</code>	The name (as a string) of the variable of interest in the interaction term; its conditional coefficient estimates will be plotted.
<code>var2</code>	The name (as a string) of the other variable in the interaction term.
<code>plot</code>	A logical value indicating whether the output is a plot or a dataframe including the conditional coefficient estimates of <code>var1</code> , their upper and lower bounds, and the corresponding values of <code>var2</code> .
<code>steps</code>	Desired length of the sequence. A non-negative number, which for <code>seq</code> and <code>seq.int</code> will be rounded up if fractional. The default is 100 or the unique categories in the <code>var2</code> (when it is less than 100. Also see unique).
<code>ci</code>	A numeric value defining the confidence intervals. The default value is 95% (0.95).
<code>adjCI</code>	A logical value indication if applying the adjustment of confidence intervals to control the false discovery rate following the Esarey and Sumner (2017) procedure. (See also Benjamini and Hochberg 1995.) The default is FALSE; the plot presents the confidence intervals suggested by Brambor, Clark, and Golder (2006).
<code>hist</code>	A logical value indicating if there is a histogram of ‘ <code>var2</code> ’ added at the bottom of the conditional effect plot.
<code>var2_dt</code>	A numerical value indicating the frequency distribution of ‘ <code>var2</code> ’. It is only used when ‘ <code>hist == TRUE</code> ’. When the object is a model, the default is the distribution of ‘ <code>var2</code> ’ of the model.
<code>predPro</code>	A logical value with default of ‘FALSE’. When the ‘ <code>m</code> ’ is an object of class ‘ <code>glm</code> ’ and the argument is set to ‘TRUE’, the function will plot predicted probabilities at the values given by ‘ <code>var2_vals</code> ’.
<code>var2_vals</code>	A numerical value indicating the values the predicted probabilities are estimated, when ‘ <code>predPro</code> ’ is ‘TRUE’.
<code>point</code>	A logical value determining the format of plot. By default, the function produces a line plot when <code>var2</code> takes on ten or more distinct values and a point (dot-and-whisker) plot otherwise; option TRUE forces a point plot.
<code>sims</code>	Number of independent simulation draws used to calculate upper and lower bounds of coefficient estimates: lower values run faster; higher values produce smoother curves.
<code>xmin</code>	A numerical value indicating the minimum value shown of <code>x</code> shown in the graph. Rarely used.

<code>xmax</code>	A numerical value indicating the maximum value shown of <code>x</code> shown in the graph. Rarely used.
<code>ercolor</code>	A character value indicating the outline color of the whisker or ribbon.
<code>esize</code>	A numerical value indicating the size of the whisker or ribbon.
<code>ralpha</code>	A numerical value indicating the transparency of the ribbon.
<code>rfill</code>	A character value indicating the filling color of the ribbon.
<code>stats_cp</code>	A character value indicating what statistics to present as the plot note. Three options are available: "none", "ci", and "ks". The default is "none". See the Details for more information.
<code>txt_caption</code>	A character string to add a note for the plot, a value will sending to <code>ggplot2::labs(caption = txt_caption)</code> .
<code>facet_labs</code>	An optional character vector of facet labels to be used when plotting an interaction with a factor variable.
<code>var3</code>	An optional name (as a string) of a third variable for a three-way interaction <code>var1 * var2 * var3</code> , pooled across imputations. When supplied, the conditional effect of <code>var1</code> across <code>var2</code> is shown at several values/levels of <code>var3</code> . The default NULL gives the standard two-way behavior. Requires continuous <code>var1</code> and <code>var2</code> .
<code>var3_vals</code>	An optional numeric vector giving the values of a continuous <code>var3</code> to condition on. The default is the mean and the mean plus or minus one standard deviation. Ignored when <code>var3</code> is a factor.
<code>facet</code>	A logical value, used only with <code>var3</code> . TRUE (default) draws one panel per value/level of <code>var3</code> ; FALSE overlays the curves colored by <code>var3</code> .
<code>...</code>	Other ggplot aesthetics arguments for points in the dot-whisker plot or lines in the line-ribbon plots. Not currently used.

Details

`interplot.lmmi` and `interplot.glmmi` are S3 methods from the `interplot`. This function can work on interactions from results in the class of `list` generated by `mi` tools.

Because the output function is based on `ggplot`, any additional arguments and layers supported by `ggplot2` can be added with the `+`.

`interplot` visualizes the conditional effect based on simulated marginal effects. The simulation provides a probabilistic distribution of moderation effect of the conditioning variable (`var2`) at every preset values (including the minimum and maximum values) of the conditioned variable (`var1`), denoted as E_{min} and E_{max} . This output allows the function to further examine the conditional effect statistically in two ways. One is to examine if the distribution of $E_{max} - E_{min}$ covers zero. The other is to directly compare E_{min} and E_{max} through statistical tools for distributional comparisons. Users can choose either method by setting the argument `stats_cp` to "ci" or "ks".

- "ci" provides the confidence interval of the difference of $E_{max} - E_{min}$. An interval including 0 suggests no statistical difference before and after the conditional effect is applied, and vice versa.
- "ks" presents the result of a two-sample Kolmogorov-Smirnov test of the simulated distributions of E_{min} and E_{max} . The output includes a D statistics and a p-value of the null hypothesis that the two distributions come from the same distribution at the 0.05 level.

See an illustration in the package vignette.

Value

The function returns a ggplot object.

Source

Benjamini, Yoav, and Yosef Hochberg. 1995. "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing". *Journal of the Royal Statistical Society, Series B* 57(1): 289–300.

Brambor, Thomas, William Roberts Clark, and Matt Golder. "Understanding interaction models: Improving empirical analyses". *Political Analysis* 14.1 (2006): 63-82.

Esarey, Justin, and Jane Lawrence Sumner. 2015. "Marginal Effects in Interaction Models: Determining and Controlling the False Positive Rate". URL: <https://jee3.web.rice.edu/interaction-overconfidence.pdf>.

Examples

```
library(interplot)
library(mitools)

data(smi)
model1 <- with(smi, glm(drinkreg ~ wave * sex, family = binomial()))

interplot(model1, var1 = "sex", var2 = "wave")
```

interplot.mlmmi	<i>Plot Conditional Coefficients in Mixed-Effects Models with Imputed Data and Interaction Terms</i>
-----------------	--

Description

interplot.mlmmi is a method to calculate conditional coefficient estimates from the results of multilevel (mixed-effects) regression models with interaction terms and multiply imputed data.

Usage

```
## S3 method for class 'mlmmi'
interplot(
  m,
  var1,
  var2,
  plot = TRUE,
  steps = NULL,
  ci = 0.95,
  adjCI = FALSE,
  hist = FALSE,
  var2_dt = NA,
```

```

    predPro = FALSE,
    var2_vals = NULL,
    point = FALSE,
    sims = 5000,
    xmin = NA,
    xmax = NA,
    ercolor = NA,
    esize = 0.5,
    ralpha = 0.5,
    rfill = "grey70",
    stats_cp = "none",
    txt_caption = NULL,
    facet_labs = NULL,
    var3 = NULL,
    var3_vals = NULL,
    facet = TRUE,
    ...
)

```

Arguments

<code>m</code>	A model object including an interaction term, or, alternately, a data frame recording conditional coefficients.
<code>var1</code>	The name (as a string) of the variable of interest in the interaction term; its conditional coefficient estimates will be plotted.
<code>var2</code>	The name (as a string) of the other variable in the interaction term.
<code>plot</code>	A logical value indicating whether the output is a plot or a dataframe including the conditional coefficient estimates of <code>var1</code> , their upper and lower bounds, and the corresponding values of <code>var2</code> .
<code>steps</code>	Desired length of the sequence. A non-negative number, which for <code>seq</code> and <code>seq.int</code> will be rounded up if fractional. The default is 100 or the unique categories in the <code>var2</code> (when it is less than 100. Also see unique).
<code>ci</code>	A numeric value defining the confidence intervals. The default value is 95% (0.95).
<code>adjCI</code>	Not working for 'lmer' outputs yet.
<code>hist</code>	A logical value indicating if there is a histogram of ' <code>var2</code> ' added at the bottom of the conditional effect plot.
<code>var2_dt</code>	A numerical value indicating the frequency distribution of ' <code>var2</code> '. It is only used when ' <code>hist == TRUE</code> '. When the object is a model, the default is the distribution of ' <code>var2</code> ' of the model.
<code>predPro</code>	A logical value with default of 'FALSE'. When the ' <code>m</code> ' is an object of class 'glmerMod' and the argument is set to 'TRUE', the function will plot predicted probabilities at the values given by ' <code>var2_vals</code> '.
<code>var2_vals</code>	A numerical value indicating the values the predicted probabilities are estimated, when ' <code>predPro</code> ' is 'TRUE'.

point	A logical value determining the format of plot. By default, the function produces a line plot when var2 takes on ten or more distinct values and a point (dot-and-whisker) plot otherwise; option TRUE forces a point plot.
sims	Number of independent simulation draws used to calculate upper and lower bounds of coefficient estimates: lower values run faster; higher values produce smoother curves.
xmin	A numerical value indicating the minimum value shown of x shown in the graph. Rarely used.
xmax	A numerical value indicating the maximum value shown of x shown in the graph. Rarely used.
ercolor	A character value indicating the outline color of the whisker or ribbon.
esize	A numerical value indicating the size of the whisker or ribbon.
ralpha	A numerical value indicating the transparency of the ribbon.
rfill	A character value indicating the filling color of the ribbon.
stats_cp	A character value indicating what statistics to present as the plot note. Three options are available: "none", "ci", and "ks". The default is "none". See the Details for more information.
txt_caption	A character string to add a note for the plot, a value will sending to <code>ggplot2::labs(caption = txt_caption)</code> .
facet_labs	An optional character vector of facet labels to be used when plotting an interaction with a factor variable.
var3	An optional name (as a string) of a third variable for a three-way interaction $\text{var1} * \text{var2} * \text{var3}$, pooled across imputations. When supplied, the conditional effect of var1 across var2 is shown at several values/levels of var3. The default NULL gives the standard two-way behavior. Requires continuous var1 and var2.
var3_vals	An optional numeric vector giving the values of a continuous var3 to condition on. The default is the mean and the mean plus or minus one standard deviation. Ignored when var3 is a factor.
facet	A logical value, used only with var3. TRUE (default) draws one panel per value/level of var3; FALSE overlays the curves colored by var3.
...	Other ggplot aesthetics arguments for points in the dot-whisker plot or lines in the line-ribbon plots. Not currently used.

Details

`interplot.mlmmi` and `interplot.gmlmmi` are S3 methods from the `interplot`. It works on lists of mixed-effects objects with class `lmerMod` and `glmerMod` generated by `mitools` and `lme4`.

Because the output function is based on [ggplot](#), any additional arguments and layers supported by `ggplot2` can be added with the `+`.

`interplot` visualizes the conditional effect based on simulated marginal effects. The simulation provides a probabilistic distribution of moderation effect of the conditioning variable (`var2`) at every preset values (including the minimum and maximum values) of the conditioned variable (`var1`), denoted as E_{min} and E_{max} . This output allows the function to further examine the conditional effect statistically in two ways. One is to examine if the distribution of $E_{max} - E_{min}$ covers

zero. The other is to directly compare E_{min} and E_{max} through statistical tools for distributional comparisons. Users can choose either method by setting the argument `stats_cp` to "ci" or "ks".

- "ci" provides the confidence interval of the difference of $E_{max} - E_{min}$. An interval including 0 suggests no statistical difference before and after the conditional effect is applied, and vice versa.
- "ks" presents the result of a two-sample Kolmogorov-Smirnov test of the simulated distributions of E_{min} and E_{max} . The output includes a D statistics and a p-value of the null hypothesis that the two distributions come from the same distribution at the 0.05 level.

See an illustration in the package vignette.

Value

The function returns a ggplot object.

interplot.plot

Plot Conditional Coefficients in Models with Interaction Terms

Description

Graph based on the data frame of statistics about the conditional effect of an interaction.

Usage

```
## S3 method for class 'plot'
interplot(
  m,
  var1 = NULL,
  var2 = NULL,
  plot = TRUE,
  steps = NULL,
  ci = 0.95,
  adjCI = FALSE,
  hist = FALSE,
  var2_dt = NULL,
  predPro = FALSE,
  var2_vals = NULL,
  point = FALSE,
  sims = 5000,
  xmin = NA,
  xmax = NA,
  ercolor = NA,
  esize = 0.5,
  ralpha = 0.5,
  rfill = "grey70",
  stats_cp = "none",
  txt_caption = NULL,
```

```

    ci_diff = NULL,
    ks_diff = NULL,
    overlay = FALSE,
    ...
)

```

Arguments

<code>m</code>	A model object including an interaction term, or, alternately, a data frame recording conditional coefficients. This data frame should includes four columns: • <code>fake</code>: The sequence of <code>var1</code> (the item whose effect will be conditioned on in the interaction); • <code>coef1</code>: The point estimates of the coefficient of <code>var1</code> at each break point. • <code>ub</code>: The upper bound of the simulated 95% CI. • <code>lb</code>: The lower bound of the simulated 95% CI.
<code>var1</code>	The name (as a string) of the variable of interest in the interaction term; its conditional coefficient estimates will be plotted.
<code>var2</code>	The name (as a string) of the other variable in the interaction term.
<code>plot</code>	A logical value indicating whether the output is a plot or a dataframe including the conditional coefficient estimates of <code>var1</code> , their upper and lower bounds, and the corresponding values of <code>var2</code> .
<code>steps</code>	Desired length of the sequence. A non-negative number, which for <code>seq</code> and <code>seq.int</code> will be rounded up if fractional. The default is 100 or the unique categories in the <code>var2</code> (when it is less than 100. Also see unique).
<code>ci</code>	is a numeric value inherited from the data wrangling functions in this package. Adding it here is just for the method consistency.
<code>adjCI</code>	Succeeded from the data management functions in ‘interplot’ package.
<code>hist</code>	A logical value indicating if there is a histogram of ‘ <code>var2</code> ’ added at the bottom of the conditional effect plot.
<code>var2_dt</code>	A numerical value indicating the frequency distribution of ‘ <code>var2</code> ’. It is only used when ‘ <code>hist == TRUE</code> ’. When the object is a model, the default is the distribution of ‘ <code>var2</code> ’ of the model.
<code>predPro</code>	A logical value with default of ‘ <code>FALSE</code> ’. When the ‘ <code>m</code> ’ is an output of a general linear model (class ‘ <code>glm</code> ’ or ‘ <code>glmerMod</code> ’) and the argument is set to ‘ <code>TRUE</code> ’, the function will plot predicted probabilities at the values given by ‘ <code>var2_vals</code> ’.
<code>var2_vals</code>	A numerical value indicating the values the predicted probabilities are estimated, when ‘ <code>predPro</code> ’ is ‘ <code>TRUE</code> ’.
<code>point</code>	A logical value determining the format of plot. By default, the function produces a line plot when <code>var2</code> takes on ten or more distinct values and a point (dot-and-whisker) plot otherwise; option <code>TRUE</code> forces a point plot.
<code>sims</code>	Number of independent simulation draws used to calculate upper and lower bounds of coefficient estimates: lower values run faster; higher values produce smoother curves.
<code>xmin</code>	A numerical value indicating the minimum value shown of <code>x</code> shown in the graph. Rarely used.

<code>xmax</code>	A numerical value indicating the maximum value shown of <code>x</code> shown in the graph. Rarely used.
<code>ercolor</code>	A character value indicating the outline color of the whisker or ribbon.
<code>esize</code>	A numerical value indicating the size of the whisker or ribbon.
<code>ralpha</code>	A numerical value indicating the transparency of the ribbon.
<code>rfill</code>	A character value indicating the filling color of the ribbon.
<code>stats_cp</code>	A character value indicating what statistics to present as the plot note. Three options are available: "none", "ci", and "ks". The default is "none". See the Details for more information.
<code>txt_caption</code>	A character string to add a note for the plot, a value will sending to <code>ggplot2::labs(caption = txt_caption)</code> .
<code>ci_diff</code>	A numerical vector with a pair of values indicating the confidence intervals of the difference between <code>var1</code> and <code>var2</code> .
<code>ks_diff</code>	A <code>ks.test</code> object of the effect of <code>var1</code> conditioned on <code>var2</code> .
<code>overlay</code>	A logical value. When TRUE, the conditional-effect curves are colored by the grouping column value in a single panel (used by three-way interaction overlays). The default is FALSE.
<code>...</code>	Other ggplot aesthetics arguments for points in the dot-whisker plot or lines in the line-ribbon plots. Not currently used.

Details

`interplot.plot` is a S3 method from the `interplot`. It generates plots of conditional coefficients. Because the output function is based on `ggplot`, any additional arguments and layers supported by `ggplot2` can be added with the `+`.

`interplot` visualizes the conditional effect based on simulated marginal effects. The simulation provides a probabilistic distribution of moderation effect of the conditioning variable (`var2`) at every preset values (including the minimum and maximum values) of the conditioned variable (`var1`), denoted as E_{min} and E_{max} . This output allows the function to further examine the conditional effect statistically in two ways. One is to examine if the distribution of $E_{max} - E_{min}$ covers zero. The other is to directly compare E_{min} and E_{max} through statistical tools for distributional comparisons. Users can choose either method by setting the argument `stats_cp` to "ci" or "ks".

- "ci" provides the confidence interval of the difference of $E_{max} - E_{min}$. An interval including 0 suggests no statistical difference before and after the conditional effect is applied, and vice versa.
- "ks" presents the result of a two-sample Kolmogorov-Smirnov test of the simulated distributions of E_{min} and E_{max} . The output includes a D statistics and a p-value of the null hypothesis that the two distributions come from the same distribution at the 0.05 level.

See an illustration in the package vignette.

Value

The function returns a `ggplot` object.

jn_interval

*Compute Johnson-Neyman Interval for Interaction Effects***Description**

Identifies the values of the moderating variable at which the conditional effect of the other variable transitions between statistical significance and non-significance.

Usage

```
jn_interval(m, var1, var2, ci = 0.95)
```

Arguments

<code>m</code>	A model object (lm, glm, lmerMod, or glmerMod).
<code>var1</code>	The name (as a string) of the variable whose conditional effect is of interest.
<code>var2</code>	The name (as a string) of the moderating variable.
<code>ci</code>	A numeric value defining the confidence level. The default is 0.95.

Details

The Johnson-Neyman (JN) technique finds the values of the moderating variable (`var2`) at which the conditional effect of `var1` is exactly at the boundary of statistical significance. This is computed analytically from the regression coefficients and their variance-covariance matrix.

For linear mixed-effects models (lmerMod, glmerMod), a normal approximation (z-distribution) is used instead of the t-distribution due to the controversy over appropriate degrees of freedom.

The function does not support factor variables or quadratic terms (`var1 == var2`).

Value

An object of class `jn_interval` containing:

bounds All Johnson-Neyman bounds (may be 0, 1, or 2 values).

bounds_in_range Bounds that fall within the observed data range of `var2`.

var2_range The range of the moderating variable in the data.

sig_pattern One of "always", "never", "between", "outside", "below", or "above", indicating where the effect is statistically significant.

ns_regions List of (xmin, xmax) pairs marking non-significant regions.

Examples

```
m <- lm(mpg ~ wt * cyl, data = mtcars)
jn <- jn_interval(m, "cyl", "wt")
print(jn)

# Add JN bounds to an interplot
interplot(m, "cyl", "wt") + jn_layer(jn)
```

jn_layer

*Add Johnson-Neyman Bounds to an interplot***Description**

Returns a list of ggplot2 layers that overlay Johnson-Neyman significance boundaries on an interplot output. Add to any interplot with +.

Usage

```
jn_layer(
  jn,
  line_color = "red",
  linetype = "dashed",
  shade_color = "grey80",
  shade_alpha = 0.15,
  label = TRUE
)
```

Arguments

jn	A jn_interval object produced by jn_interval .
line_color	Color of the boundary lines. Default "red".
linetype	Line type for boundaries. Default "dashed".
shade_color	Fill color for non-significant regions. Default "grey80".
shade_alpha	Transparency of shading. Default 0.15.
label	Logical; if TRUE (default), annotate the JN bound values.

Value

A list of ggplot2 layers.

Examples

```
m <- lm(mpg ~ wt * cyl, data = mtcars)
jn <- jn_interval(m, "cyl", "wt")
interplot(m, "cyl", "wt") + jn_layer(jn)
```

Index

`bin_layer`, [2](#), [5](#)

`brm`, [7](#), [9](#)

`class`, [3](#)

`ggplot`, [5](#), [9](#), [12](#), [15](#), [18](#), [21](#), [24](#)

`interplot`, [3](#)

`interplot.brmsfit`, [7](#)

`interplot.default`, [5](#), [10](#)

`interplot.lmerMod`, [13](#)

`interplot.lmmi`, [16](#)

`interplot.mlmmi`, [19](#)

`interplot.plot`, [22](#)

`jn_interval`, [25](#), [26](#)

`jn_layer`, [26](#)

`posterior_epred`, [8](#)

`unique`, [4](#), [8](#), [11](#), [14](#), [17](#), [20](#), [23](#)