Package ‘pequod’

October 14, 2022

Type Package
Title Moderated Regression Package
Version 0.0-5
Date 2016-2-28
Author Alberto Mirisola & Luciano Seta
Depends ggplot2, car
Imports methods
Maintainer Alberto Mirisola <alberto.mirisola@gmail.com>
Description Moderated regression with mean and residual centering and simple slopes analysis.
License GPL-2
LazyLoad yes
NeedsCompilation no
Repository CRAN
Date/Publication 2016-02-29 01:28:20

R topics documented:

lmres ......................................................... 2
PlotSlope ..................................................... 4
simpleSlope ................................................... 7
summary.lmres .............................................. 9
summary.simpleSlope ..................................... 10

Index 12
Description

Fit moderated linear regression with both residual centering and mean centering methods.

Usage

lmres(formula, data, residual_centering, centered, ...)
# Default S3 method:
lmres(formula, data, residual_centering=FALSE, centered = "none", ...)

Arguments

formula an object of class "formula": a symbolic description of the model to be fitted.
data a data frame
centered variables wich must be centered
residual_centering "FALSE" generate a moderated using standard lm regression, "TRUE" generate a moderated regression with residuals centering

Details

Moderated regression without residual centering : For any interaction term, the product is computed and entered in the final model. In order to perform a mean centered moderated regression, predictors must be centered Moderated regression with residual centering: For any interaction term with order n, a regression with low order terms (n-1) is computed, and Interaction residuals are entered in the final model.

Value

lmres returns an object of class "lmres".
An object of class "lmres" is a list containing at least the following components:

regr.order the numeric order of the fitted linear model
formula.StepI the formula of the first order regression
formula.StepII (only where relevant) the formula of the second order regression
formula.Stepfin the formula of the x (max(x)=3) order regression
beta.StepI a named vector of standardized coefficients for the first order regression
beta.StepII (only where relevant) a named vector of standardized coefficients for the second order regression
beta.Stepfin a named vector of standardized coefficients for the x (max(x)=3) order regression
StepI a lm object for the first order regression
StepII (only where relevant) a lm object for the second order regression
Stepfin a lm object for the x (max(x)=3) order regression
F_change is a list containing F change statistics

Author(s)
Alberto Mirisola and Luciano Seta

References

See Also
“summary.lmres”

Examples

```r
## moderated regression with mean centering
library(car)
data(Ginzberg)
model1<-lmres(adjdep~adjsimp*adjfatal, centered=c("adjsimp", "adjfatal"),
data=Ginzberg)

## moderated regression with mean centering
library(car)
data(Ginzberg)
model2<-lmres(adjdep~adjsimp*adjfatal, centered=c("adjsimp", "adjfatal"),
data=Ginzberg)

## three way interaction with mean centering
library(car)
data(Highway1)
model3<-lmres(rate~len*trks*sigs1, centered=c("len","trks","sigs1"),data=Highway1)
```

## The function is currently defined as
function (formula, data, centered, ...)
UseMethod("lmres")
PlotSlope

Simple slopes plot

Description

Simple slope plot for two and three way interactions.

Usage

PlotSlope(object, namemod = "default", namex = "default", namey = "default", limitx = "default", limity = "default")

Arguments

object
an object of class "simpleSlope".
namemod
a character vector of the moderator points. If "default" is used, default setting is printed.
namex
name of the predictor. If "default" is used, the predictor name in the dataframe is printed.
namey
name of the dependent variable. If "default" is used, the dataframe name is printed.
limitx
a numeric vector for setting limits of x axis.
limity
a numeric vector for setting limits of y axis.

Details

Plot for Simple slope analysis.

Value

PlotSlope returns an object of class "ggplot".

Author(s)

Alberto Mirisola and Luciano Seta

Examples

```r
## Default plot for three way interaction
library(car)
data(Highway1)
model3<-lmres(rate~len*trks*sigs1, centered=c("len","trks","sigs1"),data=Highway1)
S_slopes<-simpleSlope(model3,pred="len",mod1="trks", mod2="sigs1")
Plot<-PlotSlope(S_slopes)
```
## Personalized plot for three way interaction

```r
library(car)
data(Highway1)
model3 <- lm(rate ~ len * trks * sigs1, centered = c("len", "trks", "sigs1"), data = Highway1)
S_slopes <- simpleSlope(model3, pred = "len", mod1 = "trks", mod2 = "sigs1")
Plot <- PlotSlope(S_slopes, namemod = c("Low truck volume (-1SD), Low number of signals per mile (-1 SD)", "Low truck volume (-1SD), High number of signals per mile (+1 SD)", "High truck volume (+1SD), Low number of signals per mile (-1 SD)", "High truck volume (+1SD), High number of signals per mile (+1 SD)"),
name = "length of the Highway1\nsegment in miles",
namey = "1973 accident rate \nper million vehicle miles",
limitx = c(-9, 9), limity = c(-2, 9))
```

## The function is currently defined as

```r
function(object, namemod = "default",
namex = "default", namey = "default", limitx = "default",
limity = "default") {

pmatr <- object$Points
nomY <- object$xnomY
nomX <- object$xnomX
X_1L <- object$x_1L
X_1H <- object$x_1H

if (object$orde == 2) {
  nam <- dimnames(object$simple_slope)[1]
  nam <- nam[[1]]
  r1 <- nam[1]
  r2 <- nam[2]

  xini <- rep(X_1L, 4)
  xend <- rep(X_1H, 4)
  fact <- c(5, 6)
  mat <- cbind(fact, xini, pmatr[, 1], xend, pmatr[, 2])
  mat <- as.data.frame(mat)
  names(mat) <- c("fact", "xini", "yini", "xend", "yend")
  p <- ggplot(mat, aes(x = xini, y = yini))
  p1 <- p + geom_segment(aes(xend = xend, yend = yend))
  p1 <- p1 + scale_x_continuous(nomX) + scale_y_continuous(nomY)
  p1 <- p1 + geom_point(size = 3, aes(shape = factor(fact))) +
  geom_point(aes(x = xend, y = yend, shape = factor(fact)),
  size = 3)

  if (length(namemod) == 1) {
    p1 <- p1 + scale_shape(name = "Moderator", breaks = c(5, 6),
    labels = c(r1, r2))
  }
  if (length(namemod) > 1) {
    if (length(namemod) != 2) {
      stop("length of namemod vector must be = 2")
    }
  }
}
```
p1 <- p1 + scale_shape(name = "Moderator", breaks = c(5, 6), labels = namemod)

if (namex != "default") {
  if (length(limitx) == 2) {
    p1 <- p1 + scale_x_continuous(namex, limits = limitx)
  } else {
    p1 <- p1 + scale_x_continuous(namex)
  }
}

if (namey != "default") {
  if (length(limity) == 2) {
    p1 <- p1 + scale_y_continuous(namey, limits = limity)
  } else {
    p1 <- p1 + scale_y_continuous(namey)
  }
}

return(p1)

if (object$orde == 3) {
  nam <- dimnames(object$simple_slope)[1]
  nam <- nam[[1]]
  r1 <- nam[1]
  r2 <- nam[2]
  r3 <- nam[3]
  r4 <- nam[4]

  xini <- rep(X_1L, 4)
  xend <- rep(X_1H, 4)
  fact <- c(5, 6, 7, 8)
  mat <- cbind(fact, xini, pmatr[, 1], xend, pmatr[, 2])
  mat <- as.data.frame(mat)
  names(mat) <- c("fact", "xini", "yini", "xend", "yend")
  p <- ggplot(mat, aes(x = xini, y = yini))
  p1 <- p + geom_segment(aes(xend = xend, yend = yend))
  p1 <- p1 + geom_point(size = 3, aes(shape = factor(fact))) +
    geom_point(aes(x = xend, y = yend, shape = factor(fact)), size = 3)
  if (length(namemod) == 1) {
    p1 <- p1 + scale_shape(name = "Moderators Combination",}
simpleSlope

    Simple slopes analysis for Moderated regression

Description


Usage

    simpleSlope(object, pred, mod1, mod2, coded, ...)
    ## Default S3 method:
    simpleSlope(object, pred, mod1, mod2="none", coded = "none", ...)
**Arguments**

- **object**: an object of class "lmres": a moderated regression function.
- **pred**: name of the predictor variable
- **mod1**: name of the first moderator variable
- **mod2**: name of the second moderator variable. Default "none" is used in order to analyzing two way interaction.
- **coded**: a character vector of coded variables

**Details**

Simple slope analysis for moderated regression. If two way interaction is analyzed, the function computes simple slope analysis and region of significance (Bauer & Curran, 2005). If three way interaction is analyzed, the function compute simple slope analysis and difference slope test (Dawson and Richter, 2006).

**Value**

simpleSlope returns an object of class "simpleSlope".

An object of class "simpleSlope" is a list containing at least the following components:

- **nomY**: the name of dependent variable
- **orde**: it’s 2 for two way interaction, it’s 3 for three way interaction
- **points**: Estimated points of dependent variable as a function of levels of moderators and predictor
- **simple_slope**: a matrix summarizing simple slopes results
- **delta_slope**: (only for three way interaction) a matrix summarizing difference slope tests
- **Df**: degree of freedom
- **conf95**: (only for two way interaction) confidence interval of moderator region of significance

**Author(s)**

Alberto Mirisola and Luciano Seta

**References**

See Also

“summary.simpleSlope”

Examples

```r
## simple slope for three way interaction
library(car)
data(Highway1)
model3<-lmres(rate~len*trks*sigs1, centered=c("len","trks","sigs1"),data=Highway1)
S_slopes<-simpleSlope(model3,pred="len",mod1="trks",mod2="sigs1")

## The function is currently defined as
function (object, pred, mod1, mod2, coded, ...)
UseMethod("simpleSlope")
```

summary.lmres

**summary for lmres object**

Description

return simple and nested summaries

Usage

```r
## S3 method for class 'lmres'
summary(object, type = "default", ...)
```

Arguments

- **object**: a lmres object
- **type**: "default" generate a lm output, "nested" generate a hierarchical regression output
- ...

Value

The function summary is used to obtain a simple of nested summary of the results.

Author(s)

Alberto Mirisola and Luciano Seta
### Examples

```r
library(car)
data(Ginzberg)
model1 <- lmres(adjdep ~ adjsimp*adjfatal, centered=c("adjsimp", "adjfatal"),
data=Ginzberg)

summary(model1)
summary(model1, type="nested")

## The function is currently defined as
function (object, type = "default", ...)

summary.simpleSlope  summary for simpleSlope object

### Description

return summaries for simpleSlope object

### Usage

```r
## S3 method for class 'simpleSlope'
summry(object,...)
```

### Arguments

object,...  a simpleSlope object

### Value

The function summary is used to obtain the summary of the simpleSlope results.

### Author(s)

Alberto Mirisola and Luciano Setà

### Examples

```r
library(car)
data(Highway1)
model3 <- lmres(rate ~ len*trks*sigs1, centered=c("len","trks","sigs1"),data=Highway1)

S_slopes <- simpleSlope(model3, pred="len", mod1="trks", mod2="sigs1")

summary(S_slopes)

## The function is currently defined as
```
summary.simpleSlope

function (object,...)
Index

lmres, 2
PlotSlope, 4
print.lmres(lmres), 2
print.simpleSlope(simpleSlope), 7
print.summary.lmres(summary.lmres), 9
print.summary.simpleSlope
  (summary.simpleSlope), 10

simpleSlope, 7
summary.lmres, 9
summary.simpleSlope, 10