

Package ‘probemod’

April 22, 2015

Title Statistical Tools for Probing Moderation Effects

Version 0.2.1

Description Contains functions that are useful for probing moderation effects (or interactions) including techniques such as pick-a-point (also known as spotlight analysis) and Johnson-Neyman (also known as floodlight analysis). Plot function is also provided to facilitate visualization of results from each of these techniques.

Depends R (>= 3.1.2)

License GPL-3

LazyData no

Author Jiat Chow Tan [aut, cre]

Maintainer Jiat Chow Tan <w110013@ntu.edu.sg>

NeedsCompilation no

Repository CRAN

Date/Publication 2015-04-22 22:01:37

R topics documented:

jn	2
pickapoint	3
plot.jn	4
plot.pickapoint	5
print.jn	6
print.pickapoint	6
Index	8

 jn *Johnson-Neyman Technique*

Description

Probe moderation effect using the Johnson-Neyman technique

Usage

```
jn(model, dv, iv, mod, mrange, alpha = 0.05, yas = "none")
```

Arguments

model	Regression model (lm, glm, list).
dv	Dependent variable (character).
iv	Independent variable (character).
mod	Moderator variable(s) (character or character vector).
mrange	Range of values that jn should examine for moderator variable. Uses the current range of moderator values by default (numeric vector).
alpha	Alpha level to use (numeric).
yas	Show y (or conditional effect) as: "none", "ratio", "probability", "percentage", yas="none" by default.

Value

A list with the elements

References

Spiller, S. A., Fitzsimons, G. J., Lynch, J. G., Jr., & McClelland, G. H. (2013). Spotlights, floodlights, and the magic number zero: Simple effects tests in moderated regression. *Journal of Marketing Research*, 50(2), 277-288.

Bauer, D. J., & Curran, P. J. (2005). Probing interactions in fixed and multilevel regression: Inferential and graphical techniques. *Multivariate Behavioral Research*, 40(3), 373-400.

Examples

```
## Not run:
myModel <- lm('DV ~ IV + MOD', data=someData)
jnresults <- jn(myModel, dv='DV', iv='IV', mod='MOD')
jnresults <- jn(myModel, dv='DV', iv='IV', mod='MOD', alpha=.01)
plot(jnresults)

## End(Not run)
```

pickapoint *Pick-A-Point Technique*

Description

Probe moderation effect using the Pick-A-Point technique

Usage

```
pickapoint(model, dv, iv, mod, points, method = "meansd", alpha = 0.05,
  yas = "none")
```

Arguments

model	Regression model (lm, glm, list).
dv	Dependent variable (character).
iv	Independent variable (character).
mod	Moderator variable(s) (character or character vector).
points	List of points to test for each moderator variable (list).
method	Method to use. Possible values are: "meansd", "percentiles", method="meansd" by default.
alpha	Alpha level to use (numeric).
yas	Show y (or conditional effect) as: "none", "ratio", "probability", "percentage", yas="none" by default.

Value

A list with the elements

References

Spiller, S. A., Fitzsimons, G. J., Lynch, J. G., Jr, & McClelland, G. H. (2013). Spotlights, floodlights, and the magic number zero: Simple effects tests in moderated regression. *Journal of Marketing Research*, 50(2), 277-288.

Aiken, L. S., & West, S. G. (1991). *Multiple regression: Testing and interpreting interactions*. Thousand Oaks, CA: Sage Publications.

Examples

```
## Not run:
myModel <- lm('dv ~ iv + mod', data=someData)
pickapoint(myModel, dv='DV', iv='IV', mod='MOD')
pickapoint(myModel, dv='DV', iv='IV', mod='MOD', alpha=.01)
pickapoint(myModel, dv='DV', iv='IV', mod='MOD', method='percentiles')
pickapoint(myModel, dv='DV', iv='IV', mod='MOD', points=c(1,2,3))

## End(Not run)
```

`plot.jn`*Plot Function For Johnson-Neyman*

Description

Plot function for objects of class "jn".

Usage

```
## S3 method for class 'jn'  
plot(x, xlab = "", ylab = "", xlim = 0, ylim = 0,  
      axlwd = 10, celwd = 10, cblwd = 8, ...)
```

Arguments

<code>x</code>	An object of class "jn".
<code>xlab</code>	A title for the x axis (character).
<code>ylab</code>	A title for the y axis (character).
<code>xlim</code>	Coordinate range for x axis (numeric vector). Determined by the range of the given data by default.
<code>ylim</code>	Coordinate range for y axis (numeric vector). Determined by the range of the given data by default.
<code>axlwd</code>	Axis line width (numeric vector). <code>axlwd=10</code> by default.
<code>celwd</code>	Conditional effect line width (numeric vector). <code>celwd=10</code> by default.
<code>cblwd</code>	Conditional band line width (numeric vector). <code>cblwd=8</code> by default.
<code>...</code>	Additional arguments (not supported yet).

Value

none

Examples

```
## Not run:  
myModel <- lm('DV ~ IV + MOD', data=someData)  
jnresults <- jn(myModel, dv='DV', iv='IV', mod='MOD')  
plot(jnresults)  
  
## End(Not run)
```

plot.pickapoint *Plot Function For Pick-a-Point*

Description

Plot function for objects of class "pickapoint".

Usage

```
## S3 method for class 'pickapoint'  
plot(x, xlab = "", ylab = "", xlim = 0, ylim = 0,  
     axlwd = 10, cesize = 1.2, cilwd = 5, ...)
```

Arguments

x	An object of class "pickapoint".
xlab	A title for the x axis (character).
ylab	A title for the y axis (character).
xlim	Coordinates range for x axis (numeric vector). Determined by the range of the given data by default.
ylim	Coordinates range for y axis (numeric vector). Determined by the range of the given data by default.
axlwd	Axis line width (numeric vector). axlwd=10 by default.
cesize	Size of the conditional effect marker (numeric vector). cesize=1.2 by default.
cilwd	Conditional interval line width (numeric vector). cilwd=5 by default.
...	Additional arguments (not supported yet).

Value

none

Examples

```
## Not run:  
myModel <- lm('DV ~ IV + MOD', data=someData)  
papresults <- pickapoint(myModel, dv='DV', iv='IV', mod='MOD')  
plot(papresults)  
  
## End(Not run)
```

print.jn

Print Function For Johnson-Neyman

Description

Prints function for objects of class "jn"

Usage

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

Arguments

x An object of class "jn".
... Additional arguments (not supported yet).

Value

none

Examples

```
## Not run:  
myModel <- lm('DV ~ IV + MOD', data=someData)  
jnresults <- jn(myModel, dv='DV', iv='IV', mod='MOD')  
jnresults  
  
## End(Not run)
```

print.pickapoint*Print Function For Pick-A-Point*

Description

Print function for objects of class "pickapoint"

Usage

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

Arguments

x An object of class "pickapoint".
... Additional arguments (not supported yet).

print.pickapoint

7

Value

none

Examples

```
## Not run:  
myModel <- lm('dv ~ iv + mod', data=someData)  
papresults <- pickapoint(myModel, dv='DV', iv='IV', mod='MOD')  
papresults
```

```
## End(Not run)
```

Index

floodlight (jn), 2

jn, 2

pickapoint, 3

plot.jn, 4

plot.pickapoint, 5

print.jn, 6

print.pickapoint, 6

spotlight (pickapoint), 3