

Package ‘JNplots’

July 21, 2025

Type Package

Title Visualize Outputs from the 'Johnson-Neyman' Technique

Version 0.1.1

Description Aids in the calculation and visualization of regions of non-significance using the 'Johnson-Neyman' technique and its extensions as described by Bauer and Curran (2005) <[doi:10.1207/s15327906mbr4003_5](https://doi.org/10.1207/s15327906mbr4003_5)> to assess the influence of categorical and continuous moderators. Allows correcting for phylogenetic relatedness.

License MIT + file LICENSE

URL <https://github.com/kenstoyama/JNplots>

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3.9000

Depends R (>= 3.5.0), ape, nlme, scales

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Ken Toyama [cre, aut] (ORCID: <<https://orcid.org/0000-0002-8331-4894>>)

Maintainer Ken Toyama <ken.toyama@mail.utoronto.ca>

Repository CRAN

Date/Publication 2023-11-24 14:30:02 UTC

Contents

bird_colors	2
jnt_cat	2
jnt_cont	4
lizard_home_range	6
microlophus	6
tree_Furnariidae	7
Index	8

bird_colors

Climatic drivers of coloration in Furnariidae species

Description

Data for redness and brightness of different plumage patches in Furnariidae species and associated climatic data

Usage

```
data(bird_colors)
```

Format

An object of class data.frame

References

Marcondes et al. (2021) The American Naturalist 197:592-606 ([PubMed](#))

Seeholzer et al. (2017) Evolution 71:702:715 ([PubMed](#))

jnt_cat

Calculation and visualization of regions of non-significance to assess the influence of categorical moderators

Description

Produces a plot showing regions of non-significance defined by predictor values for which no significant differences in the dependent variable are found between categories.

Usage

```
jnt_cat(
  X,
  Y,
  m,
  data,
  alpha.sig = 0.05,
  plot.full = FALSE,
  correlation = NULL,
  cols = c("black", "black"),
  pch = c(16, 1),
  cex = 1,
  xlab = X,
  ylab = Y,
```

```

lty = c(1, 2),
line.col = c("black", "black"),
lwd = c(1, 1),
legend = TRUE
)

```

Arguments

X	A character string defining the name of the covariate (e.g., size in an allometry analysis). Must be the same as the name of the variable in the dataset (see argument “data”).
Y	A character string defining the name of the dependent variable. Must be the same as the name of the variable in the dataset (see argument “data”).
m	A character string defining the name of a categorical moderator (e.g., males and females, herbivorous and carnivorous, etc). Must be the same as the name of the variable in the dataset (see argument “data”). The variable must have two levels.
data	A dataframe containing the variables in the model.
alpha.sig	A value representing the significance value (alpha) to be considered.
plot.full	A logical. It indicates whether the plot should show the JN non-significance regions even if they don’t or just partially overlap the data. The default option is ‘FALSE’, meaning that the plot limits will depend only on the range of predictor values
correlation	an optional corStruct object describing the within-group correlation structure. See the documentation of corClasses for a description of the available corStruct classes. If a grouping variable is to be used, it must be specified in the form argument to the corStruct constructor. Defaults to NULL, corresponding to un-correlated errors.
cols	A vector of strings defining the colors of the symbols to be used in the plot. By default, c(‘black’, ‘black’) is used, which combines with the default in the argument ‘pch’ to present two groups of datapoints as open and close.
pch	A vector of strings defining the symbols to be used to represent distinct groups in the plot. Use same symbol codes as in the argument ‘pch’ in the R base function ‘plot’. By default, c(16, 1) is used, which combines with the default in the argument ‘cols’ to present two groups of datapoints as open and close.
cex	number indicating the amount by which plotting symbols should be scaled relative to the default (1).
xlab	A title for the X axis. Defaults to the name of the predictor variable in the data.
ylab	A title for the Y axis. Defaults to the name of the dependent variable in the data.
lty	A vector defining the line type of the regression lines for each category. Defaults to c(1,2).
line.col	A vector of strings defining the line colors of the regression lines for each category. Defaults to c(‘black’, ‘black’).
lwd	A vector defining the line width of the regression lines for each category. Defaults to c(1,1).
legend	A logical indicating whether a legend should appear on top of the plot. Defaults to ‘TRUE’.

Value

List with four elements: (1) results from the linear model, (2) lower and (3) upper limits of non-significance in the predictor, and (4) a graphical output.

References

Toyama, K. S. (2023). JNplots: an R package to visualize outputs from the Johnson-Neyman technique for categorical and continuous moderators, including options for phylogenetic regressions. bioRxiv, 2023-05.

Examples

```
data(microlophus)
jnt_cat(X='svl', Y='hl', m='species', data=microlophus,
xlab='log(SVL)', ylab='log(head length)')
```

jnt_cont

Calculation and visualization of regions of non-significance to assess the influence of continuous moderators

Description

Produces a plot showing how changes in the moderator affect the slope and significance of the relationship between the dependent variable and the predictor.

Usage

```
jnt_cont(
  X,
  Y,
  m,
  data,
  alpha.sig = 0.05,
  correlation = NULL,
  res = 100,
  xlab = X,
  ylab = Y,
  col.gradient = TRUE,
  sig_color = "lightblue",
  nonsig_color = "grey",
  max_col_grad = "red",
  min_col_grad = "blue",
  legend = TRUE
)
```

Arguments

X	A character string defining the name of the covariate (e.g., size in an allometry analysis). Must be the same as the name of the variable in the dataset (see argument “data”).
Y	A character string defining the name of the dependent variable. Must be the same as the name of the variable in the dataset (see argument “data”).
m	A character string defining the name of a continuous moderator. Must be the same as the name of the variable in the dataset (see argument “data”). The variable must be continuous.
data	A dataframe containing the variables in the model.
alpha.sig	A value representing the significance value (alpha) to be considered.
correlation	an optional corStruct object describing the within-group correlation structure. See the documentation of corClasses for a description of the available corStruct classes. If a grouping variable is to be used, it must be specified in the form argument to the corStruct constructor. Defaults to NULL, corresponding to uncorrelated errors.
res	A numerical value that aids in the plotting of regions of (non)significance. Default=100, higher numbers increase the number of fitted regression lines plotted (N=res-1).
xlab	A title for the X axis. Defaults to the name of the predictor variable in the data.
ylab	A title for the Y axis. Defaults to the name of the dependent variable in the data.
col.gradient	A logical indicating whether the significant regression lines should be plotted with a gradient of colors representing moderator values. Defaults to 'TRUE'.
sig_color	If col.gradient = FALSE, a character string indicating the color of the significant regression lines. Defaults to 'lightblue'.
nonsig_color	If col.gradient = FALSE, a character string indicating the color of the non-significant regression lines. Defaults to 'grey'.
max_col_grad	If col.gradient = TRUE, a character string indicating the maximum color of the gradient.
min_col_grad	If col.gradient = TRUE, a character string indicating the minimum color of the gradient.
legend	A logical indicating whether a legend should appear on top of the plot. Defaults to 'TRUE'.

Value

List with six elements: (1) results from the linear model, (2) lower and (3) upper limits of (non)significance in the moderator, (4) lower and (5) upper data limit in the data, and (6) a graphical output.

References

Toyama, K. S. (2023). JNplots: an R package to visualize outputs from the Johnson-Neyman technique for categorical and continuous moderators, including options for phylogenetic regressions. *bioRxiv*, 2023-05.

Examples

```
#### non-phylogenetic model ####
data(lizard_home_range)
jnt_cont(X='PHR95_overlap_z', Y='hrsize95', m='degree_z',
data=lizard_home_range, xlab = 'home range overlap 95',
ylab='home range size 95')

#### phylogenetic model ####
jnt_cont(X='bio12', Y='back_bright', m='bio1', data=bird_colors,
correlation=corPagel(1, tree_Furnariidae),xlab='precipitation (mm)',
ylab='back brightness (scaled)',res=200)
```

lizard_home_range	<i>Determinants of home range size in sleepy lizards</i>
-------------------	--

Description

Dataset includes data on the determinants of home range size in the lizard *Tiliqua rugosa*

Usage

```
data(lizard_home_range)
```

Format

An object of class data.frame

References

Payne et al. (2021) Ecological Monographs 92:e1519 ([doi:10.1002/ecm.1519](#))

microlophus	<i>Morphology of Microlophus lizards (Squamata: Tropiduridae)</i>
-------------	---

Description

Data for body size and head morphology of *Microlophus* lizards

Usage

```
data(microlophus)
```

Format

An object of class data.frame

References

Toyama et al. (2018) Zoology 129:45-53 ([PubMed](#))

tree_Furnariidae	<i>Phylogeny of Furnariidae</i>
------------------	---------------------------------

Description

Phylogeny of Furnariidae

Usage

```
data(tree_Furnariidae)
```

Format

An object of class phylo

References

Harvey et al. (2020) Science 370:1343-1348 ([PubMed](#))

Index

* **datasets**

- bird_colors, [2](#)
- lizard_home_range, [6](#)
- microlophus, [6](#)

* **phylogeny**

- tree_Furnariidae, [7](#)

bird_colors, [2](#)

corClasses, [3](#), [5](#)

corStruct, [3](#), [5](#)

jnt_cat, [2](#)

jnt_cont, [4](#)

lizard_home_range, [6](#)

microlophus, [6](#)

tree_Furnariidae, [7](#)