

Package ‘eHOF’

July 22, 2025

Version 1.16

Encoding UTF-8

Title Extended HOF (Huisman-Olff-Fresco) Models

Maintainer Florian Jansen <florian.jansen@uni-rostock.de>

Depends R (>= 2.5.0), mgcv, lattice

Suggests vegdata, vegan, knitr, testthat

LazyData yes

VignetteBuilder knitr

Description Extended and enhanced hierarchical logistic regression models (called Huisman-Olff-Fresco in biology, see Huisman et al. 1993 Journal of Vegetation Science <[doi:10.1111/jvs.12050](https://doi.org/10.1111/jvs.12050)>) models. Response curves along one-dimensional gradients including no response, monotone, plateau, unimodal and bimodal models.

License GPL (>= 3)

RoxygenNote 7.3.2

NeedsCompilation no

Author Florian Jansen [aut, cre],
Jari Oksanen [aut]

Repository CRAN

Date/Publication 2024-12-14 23:30:02 UTC

Contents

acre	2
autolayout	2
betaresp	3
eHOF	4
mtf	4
Para	5
Para_deriv	6
plot.HOF	7

Index	10
--------------	-----------

acre

*Vegetation plots from arable fields in North-Eastern Germany***Description**

Species and Environmental Data from arable fields in North-Eastern Germany. The data contains 100 selected plots (rows) and 6 species (columns).

Usage

```
data(acre)
```

Format

Two data frames, one data frame called acre with 100 vegetation plots (rows). Species are named with 7 letter shortletters according to the taxonomic reference list GermanSL. Plant performance is given as presence-absence. The second data frame, named acre.env consists of plot information about date of survey, survey area, total plant cover and measured soil pH.

Details

Soil pH was measured in 1-mol KCl buffer solution.

References

Manthey, M. (2003). Vegetations\''okologie der \'Acker und Ackerbrachen Mecklenburg-Vorpommerns. 209 p., Berlin [u.a.]: Cramer.

Examples

```
data(acre)
hist(acre.env$PH_KCL)
## Not run:
vignette('eHOF', 'eHOF')

## End(Not run)
```

autolayout

*Automatically Specifying Complex Plot Arrangements***Description**

Automatic version of function [layout](#) for given number of subplots. Defined matrix schemes for [layout](#) for up to 32 subfigures.

Usage

```
autolayout(N, byrow = TRUE, ...)
```

Arguments

N	number of subfigures
byrow	see matrix
...	additional arguments for layout or matrix .

Value

Number of figures, N, see [layout](#).

Examples

```
for(i in sample(1:32, 1)) {  
  autolayout(i, byrow=TRUE)  
  layout.show(i)  
}
```

betaresp	<i>Beta response models</i>
----------	-----------------------------

Description

Generating beta response models. Implemented to compare the hierarchical logistic regression models with beta response models.

Usage

```
betaresp(x, p1, p2, alpha, gamma, hi)
```

Arguments

x	Gradient values
p1, p2	response endpoints
alpha, gamma	shape parameters of the response
hi	maximum height of response

Author(s)

Jari Oksanen

Examples

```
x <- seq(0,3, len=201)
m <- betaresp(x, 2, 0.8, 0.1, 0.5, 0.8)
plot(m)
```

eHOF

Extended HOF (Huisman-Olff-Fresco) Models

Description

Extended and enhanced hierarchical logistic regression models (called Huisman-Olff-Fresco in biology, see Huisman et al. 1993 Journal of Vegetation Science <doi:10.1111/jvs.12050>) models. Response curves along one-dimensional gradients including no response, monotone, plateau, unimodal and bimodal models.

References

Jansen, F., Dengler, J (2011) Plant names in vegetation databases - a neglected source of bias, Journal of vegetation science, 21(6), 1179-1186. <http://dx.doi.org/10.1111/j.1654-1103.2010.01209.x>

mtf

Species Data and Altitude from Mt. Field, Tasmania

Description

Species and Environmental Data from Mt. Field, Tasmania. The acre data frame has 100 selected sites (rows) and 6 species (columns). Data frame mtf.env has only one variable: Altitude above sea level (in meters) for each site. The data are a subset of well drained sites from a more extensive data set.

Usage

```
data(mtf)
```

Format

The species data frame contains the following species:

EPACSERP a numeric vector

CYATPETI a numeric vector

NOTHCUNN a numeric vector

POA.GUNN a numeric vector

BAUERUBI a numeric vector

References

Minchin, P.R. (1989). Montane vegetation of the Mt. Field massif, Tasmania: a test of some hypotheses about properties of community patterns. *Vegetatio* 83, 97.110.

Examples

```
data(mtf)
```

Para

Curve parameters of eHOF models

Description

Derive common shape parameters from the different model types. Calculate a set of parameters (see values below) from eHOF models.

Usage

```
Para(resp, ...)

## S3 method for class 'HOF'
Para(resp, model, newdata = NULL, ...)
```

Arguments

resp	response model results, see [HOF()]
...	further arguments passed to or from other methods, e.g. for [pick.model()]
model	response model type. If not specified, the output of [pick.model()] will be used
newdata	vector of gradient values to use

Details

For models VI and VII Para will give you two expectancy values for the ranges left and right of the pessimum between the model optima. If you want to have the overall expectancy value, use:
`gradient <- seq(min(Para(resp)$range), max(Para(resp)$range), length.out=10000) weighted.mean(gradient, predict(resp, newdata=gradient))`

Value

* species Name or abbreviat of the species. * abund sum Abundance sum, i.e. sum of all response values divided by M. * range Range of x values. * model Model type, if not specified the result of [pick.model]. * para Model parameters (a to d). * M Maximum response value, specified in the HOF function call. * mini Location of the minimum, i.e. the gradient value, where the response is lowest, for model VI and VII the lowest response between the two optima. * pess Lowest estimated response value. * top Highest estimated response value(s). * opt Location of the optimum, i.e. the gradient value, where the species response is highest. NA for model I and an optimum interval for

model type III. * expect Expectancy value, i.e. average x value under the model curve). * max slope Highest slope, i.e. maximum of the first derivation of the curve. * centralBorder Following Heegard, the central borders are calculated as the gradient values, where the response reaches " $\exp(-1/2)$ " of the top. * outerBorder Following Heegard, the outer borders of the species niche are calculated as the gradient values, where the response reaches $\exp(-2)$ of the top. * raw mean Average of measured x values.

Author(s)

Florian Jansen

References

Heegard, E. (2002) The outer border and central border for species-environmental relationships estimated by non-parametric generalised additive models. *Ecological Modelling*, 157, 131-139.

Damgaard, C. (2006) Modelling ecological presence-absence data along an environmental gradient: threshold levels of the environment. *Environ Ecol Stat* 13:229-236.

Para_deriv

Curve derivatives of eHOF models

Description

Slope and inflection points for the eHOF model types.

Usage

```
Para_deriv(
  resp,
  newdata = NULL,
  model = NULL,
  p,
  type = c("slope", "inflection"),
  ...
)
```

Arguments

resp	response model results, see [HOF()].
newdata	new gradient vector, if NULL a vector with 10000 values within gradient range will be used.
model	response model type. If not specified, the output of [pick.model()] will be used.
p	Vector of model parameters. If not specified, resp\$models[[model]]\$par will be used.
type	Output type: slope or inflection points of model curves.
...	further arguments passed to or from other methods, e.g. for [pick.model()].

Value

For slope a vector of slopes for all newdata points. Inflection points are the extremes of the second derivative.

Author(s)

Florian Jansen

References

Peper, J., Jansen, F., Pietzsch, D., Manthey, M. 2011. Patterns of plant species turnover along grazing gradients. *Journal of Vegetation Science* 22: 457–466.

plot.HOF

Plot Hierarchical Logistic Regression Models

Description

Plot single or multiple HOF models with or without model parameters.

Usage

```
## S3 method for class 'HOF'
plot(
  x,
  marginal = c("bar", "rug", "hist", "points", "n"),
  boxp = TRUE,
  las.h = 1,
  y1,
  main,
  model,
  test = "AICc",
  modeltypes,
  onlybest = TRUE,
  penal,
  gam = FALSE,
  selectMethod,
  silent = FALSE,
  para = FALSE,
  gam.se = FALSE,
  color,
  newdata = NULL,
  lwd = 1,
  leg = TRUE,
  add = FALSE,
  xlabel,
  ...
)
```

Arguments

<code>x</code>	an object from <code>HOF(spec, ...)</code> .
<code>marginal</code>	type of marginal representation for occurrences/absences.
<code>boxp</code>	plotting of horizontal boxplots
<code>las.h</code>	orientation of axes labels (0 = vertical, 1 = horizontal)
<code>yl</code>	range of y axis, useful for rare species. Must be given as fraction of M (between 0 and 1).
<code>main</code>	optional plot title
<code>model</code>	specific HOF model used, if not selected automatically.
<code>test</code>	See <code>pick.model</code>
<code>modeltypes</code>	vector of suggested model types
<code>onlybest</code>	plot only the best model according to chosen Information criterion. If set to FALSE all calculated models will be plotted, but the best model with a thicker line.
<code>penal</code>	penalty term for model types, default is the number of model parameter
<code>gam</code>	plot Generalized additive response
<code>selectMethod</code>	See <code>pick.model</code>
<code>silent</code>	Print messages
<code>para</code>	should model parameters (optima, raw.mean, niche,...) be plotted.
<code>gam.se</code>	plotting of two times standard error of <code>predict.gam</code> as confidence interval
<code>color</code>	model line color, vector of length seven
<code>newdata</code>	curves are plotted for original x-values. Otherwise you have to provide a vector with new gradient values.
<code>lwd</code>	line width of model curve(s)
<code>leg</code>	legend for model type (and parameters)
<code>add</code>	add to existing plot
<code>xlabel</code>	x axis label
<code>...</code>	further arguments passed to or from other methods.

Details

Plottype layout will give a normal plot for a single species, or if the HOF object contains several species, the graphics display will be divided by [autolayout](#). Multiple species can also be plotted by a lattice xyplot and plotted with `plot.HOF` for every species. The third option (`plottype='all'`) plots all selected species on the same graph which might be useful to evaluate e.g. the species within one vegetation plot, see examples.

A rug adds a rug representation (1-d plot) of the data to the plot. A rug plot is a compact way of illustrating the marginal distributions of x. Positions of the data points along x and y are denoted by tick marks, reminiscent of the tassels on a rug. Rug marks are overlaid onto the axis. A `dit='bar'` plot will display the original response values. For binary data this will be identical to rug.

Author(s)

Florian Jansen

References

de la Cruz Rot M (2005) Improving the Presentation of Results of Logistic Regression with R. Bulletin of the Ecological Society of America 86: 41-48

See Also

[HOF()]

Examples

```
data(acre)
sel <- c('MATREC', 'RUMECT', 'SILENOC', 'APHAARV', 'MYOSARV', 'DESUSOP', 'ARTE#VU')
mo <- HOF(acre[match(sel, names(acre))], acre.env$PH_KCL, M=1, bootstrap=NULL)
par(mar=c(2,2,1,.1))
plot(mo, para=TRUE)

#' An example for plottype='all' to show species responses for the species within
#' the most acidic and the most calcareous vegetation plot.
## Not run:
  allSpeciesFromAnAcidicPlot <- acre['57',] > 0
  mods.acidic <- HOF(acre[,allSpeciesFromAnAcidicPlot],acre.env$PH_KCL,M=1,bootstrap=NULL)
  allSpeciesFromAnCalcareousPlot <- acre['87',] > 0
  mods.calc <- HOF(acre[,allSpeciesFromAnCalcareousPlot],acre.env$PH_KCL,M=1,bootstrap=NULL)

  autolayout(2)
  plot(mods.acidic, plottype='all', main='Plot with low pH')
  abline(v=acre.env$PH_KCL[acre.env$RELEVE_NR == '57'])
  names(mods.acidic)

  plot(mods.calc, plottype='all', main='Plot with high pH')
  abline(v=acre.env$PH_KCL[acre.env$RELEVE_NR == '87'])
  names(mods.calc)

## End(Not run)
```

Index

* datasets

acre, [2](#)

mtf, [4](#)

* models

Para, [5](#)

Para_deriv, [6](#)

* model

plot.HOF, [7](#)

acre, [2](#)

autolayout, [2](#), [8](#)

betaresp, [3](#)

betaresponse (betaresp), [3](#)

eHOF, [4](#)

layout, [2](#), [3](#)

matrix, [3](#)

mtf, [4](#)

Para, [5](#)

Para_deriv, [6](#)

plot.betaresp (betaresp), [3](#)

plot.HOF, [7](#)

print.betaresp (betaresp), [3](#)