Package 'maxnet'

October 13, 2022

Type Package

Title Fitting 'Maxent' Species Distribution Models with 'glmnet'

Version 0.1.4

Date 2021-07-08

Author Steven Phillips

Maintainer Steven Phillips <mrmaxent@gmail.com>

Imports glmnet

Description

Procedures to fit species distributions models from occurrence records and environmental variables, using 'glmnet' for model fitting. Model structure is the same as for the 'Maxent' Java package, version 3.4.0, with the same feature types and regularization options. See the 'Maxent' website <http://biodiversityinformatics.amnh.org/open_source/maxent> for more details.

License MIT + file LICENSE

URL https://github.com/mrmaxent/maxnet

RoxygenNote 5.0.1

LazyData true

NeedsCompilation no

Repository CRAN

Date/Publication 2021-07-09 14:10:02 UTC

R topics documented:

maxnet-package *Maxent over glmnet*

Description

Maxent species distribution modeling using glmnet for model fitting

Details

Package:	maxnet
Type:	Package
Date:	2013-06-06
License:	To be determined

Create Maxent models for species distributions from presence and background data, using the glmnet package to do the model fitting. By default, feature sets and regularization are the same as the Maxent java application.

Author(s)

Steven Phillips <phillips@research.att.com>

References

Phillips & Dudik Fithian & Hastie Glmnet

bradypus *Occurrence records and background data for the brown-throated three-toed sloth, Bradypus variegatus*

Description

A dataset containing environmental data at 116 Bradypus variegatus occurrence points and 1000 background points in South and Central America. Occurrence data are from Anderson and Handley (2001); see Phillips et al. (2006) for descriptions of the predictor variables.

Usage

```
bradypus
```

Format

An object of class `data.frame` with 1116 rows and 15 columns.

References

Anderson, R. P. and Handley, Jr., C. O. (2001). A new species of three-toed sloth (Mammalia: Xenarthra) from Panama, with a review of the genus Bradypus. Proceedings of the Biological Society of Washington 114, 1-33.

Phillips, S. J. et al. (2006). Maximum entropy modeling of species geographic distributions. Ecological Modelling 190, 231-259

`hinge` *Maxent feature classes*

Description

Create and evaluate Maxent's feature classes

Usage

```
hinge(x, nknots = 50)
thresholds(x, nknots=50)
categorical(x)
```

Arguments

`x`	a predictor: a factor for categorical, otherwise numeric.
`nknots`	number of knots.

Value

These functions are typically called by `model.matrix` rather than directly by a user. `hinge`, `threshold` and `categorical` return a matrix with a column for each feature of the specified type. `hinge` creates `2*nknots-2` hinge features, half with `min=min(x)` and half with `max=max(x)`, and knots evenly spaced between `min(x)` and `max(x)`. A hinge feature `h(min,knot)` or `h(knot,max)` is 0 if the predictor is below the first argument, 1 if the predictor is above the second argument, and linearly interpolated inbetween. A threshold feature is 1 if the predictor is above the knot, 0 otherwise. A categorical feature is 1 if the predictor matches the category and 0 otherwise.

Author(s)

Steven Phillips

Examples

```
library(maxnet)
data(bradypus)
hinge(bradypus$tmp6190_ann,nknots=10)
categorical(bradypus$ecoreg)
```

maxnet *Maxent over glmnet*

Description

Maxent species distribution modeling using glmnet for model fitting

Usage

```
maxnet(p, data, f = maxnet.formula(p, data), regmult = 1,
   regfun = maxnet.default.regularization, addsamplestobackground=T, ...)
maxnet.default.regularization(p, m)
maxnet.formula(p, data, classes="default")

## S3 method for class 'maxnet'
predict(object, newdata, clamp=T, type=c("link","exponential","cloglog","logistic"), ...)
```

Arguments

p	a vector of 1 (for presence) or 0 (for background).
data	a matrix or data frame of predictor variables.
f	a formula to determine the features to be used.
regmult	a constant to adjust regularization.
regfun	a function to compute regularization constant for each feature.
addsamplestobackground	if T, add to the background any presence sample that is not already there.
object	an object of class "maxnet", i.e., a fitted model.
newdata	values of predictor variables to predict to.
m	a matrix of feature values.
clamp	if true, predictors and features are restricted to the range seen during model training.
type	type of response required.
classes	continuous feature classes desired, either "default" or any subset of "lqpht" (for example, "lh").
...	not used.

Details

Using `lp` for the linear predictor and `entropy` for the entropy of the exponential model over the background data, the values plotted on the y-axis are:

`lp` if `type` is "link".

`exp(lp)` if `type` is "exponential".

`1-exp(-exp(entropy+lp))` if `type` is "cloglog".

`1/(1+exp(-entropy-lp))` if `type` is "logistic".

Value

Maxnet returns an object of class `maxnet`, which is a list consisting of a glmnet model with the following elements added:

`betas`	nonzero coefficients of the fitted model
`alpha`	constant offset making the exponential model sum to one over the background data
`entropy`	entropy of the exponential model
`penalty.factor`	the regularization constants used for each feature
`featuremins`	minimum of each feature, to be used for clamping
`featuremaxs`	maximum of each feature, to be used for clamping
`varmin`	minimum of each predictor, to be used for clamping
`varmax`	maximum of each predictor, to be used for clamping
`samplemeans`	mean of each predictor over samples (majority for factors)
`levels`	levels of each predictor that is a factor

Author(s)

Steven Phillips

Examples

```
library(maxnet)
data(bradypus)
p <- bradypus$presence
data <- bradypus[,-1]
mod <- maxnet(p, data)
plot(mod, type="cloglog")
mod <- maxnet(p, data, maxnet.formula(p, data, classes="lq"))
plot(mod, "tmp6190_ann")
```

`plot.maxnet` *Response plots for maxnet models*

Description

Create response plots for each predictor in a maxnet model

Usage

```
## S3 method for class 'maxnet'
plot(x, vars = names(x$samplemeans), common.scale = T,
   type = c("link", "exponential", "cloglog", "logistic"), ylab = NULL, ...)

response.plot(mod, v, type, mm=mod$samplemeans, min=mod$varmin[v], max=mod$varmax[v],
  levels=unlist(mod$levels[v]), plot=T, xlab=v, ylab=tools::toTitleCase(type), ...)
```

Arguments

x	an object of class maxnet, i.e., a fitted model.
vars	vector of predictors for which response plots are desired.
common.scale	if true, all plots use the same scale on the y-axis.
type	type of response to plot on y-axis.
xlab	label for x-axis.
ylab	label for y-axis.
mod	a fitted model, must be of type maxnet if default values used for other arguments.
v	name of variable to be plotted.
mm	sample means (or majorities for factors) for predictors; predictors other than v are given these values.
min	minimum value of v; determines range of x-axis
max	maximum value of v; determines range of x-axis
levels	if v is a factor, determines levels to be plotted
plot	if false, don't draw the plot
...	passed to plot or barplot

Value

If plot is false, return a vector of y values, one for each factor or 100 ranging from min - 0.1*(max-min) to max + 0.1*(max-min).

Author(s)

Steven Phillips

Index