# Package: maxnet (via r-universe)

September 4, 2024

**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** utils, stats, glmnet, dplyr

**Suggests** stars, terra

**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

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.3

**LazyData** true

**Encoding** UTF-8

**Repository** https://atsa-es.r-universe.dev

**RemoteUrl** https://github.com/BigelowLab/maxnet

**RemoteRef** HEAD

**RemoteSha** 787504c9b9250aa3ebbf721164fb075bda9125bd

# Contents



---

`maxnet-package` *Maxent over 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.

## Author(s)

Steve Phillips

## References

Phillips & Dudik, Fithian & Hastie, glmnet

## See Also

Useful links:

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

---

`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

A data frame with 1116 observation of 14 variables with a presence/absence flag

- presence numeric 1 = presence, 0 = absence
- cld6190_ann integer
- dtr6190_ann integer
- ecoreg factor
- frs6190_ann integer
- h_dem integer
- pre6190_ann integer
- pre6190_l1 integer
- pre6190_l10 integer
- pre6190_l4 integer
- pre6190_l7 integer
- tmn6190_ann integer
- tmp6190_ann integer
- tmx6190_ann integer
- vap6190_ann integer

## 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

---

# categorical *Maxent feature classes*

---

## Description

Create and evaluate Maxent's feature classes.

These functions are typically called by `model.matrix` rather than directly by a user.

`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.

## Usage

```
categorical(x)

hinge(x, nknots = 50)

thresholds(x, nknots = 50)
```

## Arguments

| | |
|---|---|
| x | a predictor: a factor for categorical, otherwise numeric |
| nknots | number of knots |

## Value

hinge, threshold and categorical return a matrix with a column for each feature of the specified type.

## Author(s)

Steve Phillips

## Examples

```
## Not run:
  library(maxnet)
  data(bradypus)
  hinge(bradypus$tmp6190_ann,nknots=10)
  categorical(bradypus$ecoreg)

## End(Not run)
```

---

# maxnet *Maxent over glmnet*

---

## Description

Maxent species distribution modeling using glmnet for model fitting

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"

## 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")
```

## Arguments

| | |
|---|---|
| p | numeric, a vector of 1 (for presence) or 0 (for background) |
| data | a matrix or data frame of predictor variables |
| f | formula, determines the features to be used |
| regmult | numeric, a constant to adjust regularization |
| regfun | function, computes regularization constant for each feature |
| addsamplestobackground | logical, if TRUE then add to the background any presence sample that is not already there |
| ... | not used |
| m | a matrix of feature values |
| classes | charcater, continuous feature classes desired, either "default" or any subset of "lqpht" (for example, "lh") |

## 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)

Steve Phillips

## Examples

```
## Not run:
  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")

## End(Not run)
```

---

`plot.maxnet` *Create response plots for user selected predictors in a maxnet model*

---

## Description

Create response plots for user selected predictors in a maxnet model

## Usage

```
## S3 method for class 'maxnet'
plot(
  x,
  vars = names(x$samplemeans),
  common.scale = TRUE,
  type = c("link", "exponential", "cloglog", "logistic"),
  ylab = NULL,
  plot = TRUE,
  mar = c(5, 5, 4, 2),
  N = 100,
  ...
)
```

## Arguments

| | |
|---|---|
| `x` | an object of class maxnet, i.e., a fitted model. |
| `vars` | character, vector of predictors for which response plots are desired. |
| `common.scale` | logical, if true, all plots use the same scale on the y-axis. |
| `type` | character, type of response to plot on y-axis. |
| `ylab` | character, label for y-axis |

| | |
|---|---|
| plot | logical, if TRUE render a plot, if FALSE return a list of data frames with variable and response columns |
| mar | numeric, 4 element value for margins (lines, in order of bottom, left, top, right) See par for details. |
| N | numeric, the number of intervals over which to sample the response |
| ... | other arguments passed to plot or barplot |

## Value

if plot is FALSE then return a list of data frames that contain variable and response columns otherwise NULL invisibly

## See Also

response.plot

---

# predict.maxnet *Predict using a maxnet model*

---

## Description

Prediction can be on a spatial raster or vector space.

## Usage

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

## Arguments

| | |
|---|---|
| object | an object of class "maxnet", i.e., a fitted model. |
| newdata | values of predictor variables to predict to, possibly matrix, data.frame, SpatRaster or stars object. |
| clamp | logical, f true, predictors and features are restricted to the range seen during model training. |
| type | character, type of response required. Using lp for the linear predictor and entropy for the entropy of the exponential model over the background data, the values returned are determined by the value of type. |

- "link"yields lp

- "exponential"yields `exp(lp)`
- "cloglog"yields `1-exp(-exp(entropy+lp))`
- "logistic"yields `1/(1+exp(-entropy-lp))`

`...` not used

## Value

vector with predicted values (one per input row), `SpatRaster` or `stars` object of predicted values

---

`response.plot` *Compute and plot a single response variable*

---

## Description

Compute and plot a single response variable

## Usage

```
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),
  N = 100,
  ...
)
```

## Arguments

| | |
|---|---|
| `mod` | a fitted model, must be of type maxnet if default values used for other arguments. |
| `v` | charvacter, name of variable to be plotted. |
| `type` | character, type of response to plot on y-axis. |
| `mm` | numeric, sample means (or majorities for factors) for predictors; predictors other than v are given these values. |
| `min` | numeric, minimum value of v; determines range of x-axis |
| `max` | numeric, maximum value of v; determines range of x-axis |
| `levels` | vector, if v is a factor, determines levels to be plotted |
| `plot` | logical, if `TRUE` render the plot (or barplot) if `FALSE` then compute the response and return a data.frame |

| | |
|---|---|
| xlab | character, label for x-axis |
| ylab | character, label for y-axis |
| N | numeric, the number of intervals over which to sample the response |
| ... | other argument passed to plot or barplot |

## Value

if plot is FALSE a data frame that contains variable and response columns otherwise NULL invisibly

## See Also

plot.maxnet

# Index