

Package ‘SpeciesMix’

October 29, 2016

Title Fit Mixtures of Archetype Species

Version 0.3.4

Author Piers K Dunstan, Scott D Foster and Ross Darnell

Description Fitting Mixtures to Species distributions using BFGS and analytical derivatives.

Depends R(>= 2.15.0),MASS,stats,numDeriv

SystemRequirements C++11

Maintainer Piers Dunstan <Piers.Dunstan@csiro.au>

License GPL-2

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-10-29 10:48:16

R topics documented:

artificial.data	1
clusterSelect	3
logLike.pars	4
predict.archetype	5
print.archetype	6
SpeciesMix	7
SpeciesMixtures	8
Index	9

artificial.data *Create an artificial data set*

Description

artificial.data creates an artificial data set of multispecies presence absence data based on a binomial response

Usage

```
artificial.data(formula,data,theta,S,dist="bernoulli")
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class):a symbolic description of the model to be fitted
data	a data frame containing the variables in the formula
theta	coefficients for each species archetype. Matrix of G x number of parameters. Each row is a different species archetype.
S	number of species to create
dist	The distribution to sample from, "bernoulli" gives presence/absence (default), "negbin" gives abundances. For negbin, intercepts are chosen at random from -5 to 5, theta held at 1.

Details

each species is a random realisation of the archetype species

Value

a list with elements pa (matrix of presence absence data) and group (membership of each species to each group)

Authors

Piers Dunstan and Scott Foster

Examples

```
## for bernoulli
theta <- matrix(c(-0.9,-0.6,0.5,1,-0.9,1,0.9,-0.9),4,2,byrow=TRUE)
dat <- data.frame(y=rep(1,100),x=runif(100,0,2.5))
dat1 <- artificial.data(y~1+x,dat,theta,20)
## for negbin
theta <- matrix(c(-0.9,-0.6,0.5,1,-0.9,1,0.9,-0.9),4,2,byrow=TRUE)
dat <- data.frame(y=rep(1,100),x=runif(100,0,2.5))
dat1 <- artificial.data(y~1+x,dat,theta,20,dist="negbin")
```

clusterSelect

Fit SpeciesMix for a range of values for G

Description

clusterSelect fits models with varying values of G to determine the appropriate number of archetype species.

Usage

```
#clusterSelect(sp.form, sp.data, covar.data, G=1:10, \n
#em.prefit=TRUE, em.steps=4 ,em.refit=3, \n
#est.var=FALSE, trace=TRUE)
```

Arguments

sp.form	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted
sp.data	a data frame containing the species information. The frame is arranged so that each row is a site and each column is a species. Species names should be included as column names otherwise numbers from 1:S are assigned.
covar.data	a data frame containing the covariate data for each site. Names of columns must match that given in
G	Vector containing the range of archetype species to fit.
em.prefit	obtain initial parameter estimates from EM
em.steps	number of EM steps to do if using em.prefit
em.refit	refits model so that the global maxima can be found using EM.
est.var	calculate the variance covariance matrix for each group
trace	the trace of the EM steps

Details

fits multiple fitMix models across the range of values for G. Most of the arguments are passed directly to fitMix

Value

aic	vector containing the aic value for each value of G
bic	bic
fm	a list containing all output from each value of G.

Authors

Piers Dunstan and Scott Foster

Examples

```
G <- 4
S <- 20
theta <- matrix(c(-0.9,-0.6,0.5,1,-0.9,1,0.9,-0.9),4,2,byrow=TRUE)
dat <- data.frame(y=rep(1,100),x=runif(100,0,2.5),z=rnorm(100,10,2))
dat1 <- artificial.data(y~1+x,dat,theta,S)
dat <- dat[,2:3]
clusters <- clusterSelect(obs~1+x,dat1$pa,dat,G=2:5,em.refit=2)
```

logLike.pars

*logLike.pars***Description**

logLike.pars return loglike for give parameters. pi is on normal scale with G groups

Usage

```
logLike.pars(pi,coef,sp.form,sp.data,covar.data)
```

Arguments

pi	vector of pi returned from SpeciesMix
coef	matrix of coefficients returned from Species Mix
sp.form	an object of class "formula" (or one that can be coerced to that class):a symbolic description of the model to be fitted
sp.data	a data frame containing the species information. The frame is arranged so that each row is a site and each column is a species. Species names should be included as column names otherwise numbers from 1:S are assigned.
covar.data	a data frame containng the covariate data for each site. Names of columns must match that given in

Details

To Come

Value

logl loglikelihood

Authors

Piers Dunstan and Scott Foster

Examples

```
G <- 4
S <- 50
theta <- matrix(c(-9, 35, -32, 0, 0.7, 0, -16, 23, -8.2, -3, -0.6, 0.8), 4, 3, byrow=TRUE)
dat <- data.frame(y=rep(1, 200), x=runif(200, 0, 2.5), z=rnorm(200, 10, 2))
dat <- data.frame(dat, x.sq=dat$x^2)
dat1 <- artificial.data(y~1+x+x.sq, dat, theta, S)
fm4 <- SpeciesMix(obs~1+x+x.sq, dat1$pa, dat, G=4, em.prefit=TRUE, em.refit=1, est.var=FALSE)
logLik.pars(fm4$pi, fm4$coef, obs~1+x+x.sq, dat1$pa, dat)
```

predict.archetype	<i>Predict presence of archetypes from fitted model</i>
-------------------	---

Description

predict.archetype

Arguments

object	a model object returned from SpeciesMix
new.obs	covariate data at new locations for predictions. Must be a data frame with same row names as in the model formula
...	further arguments passed to or from other methods.

Details

predicts probability of occurrence at new location with standard errors calculated using delta method

Value

fit	predicted probability of presence
se.fit	standard error of fit

Method

predict(object, new.obs, ...)

Authors

Piers Dunstan and Scott Foster

Examples

```
G <-4
S <- 50
theta <- matrix(c(-9,35,-32,0,0.7,0,-16,23,-8.2,-3,-0.6,0.8),4,3,byrow=TRUE)
dat <- data.frame(y=rep(1,200),x=runif(200,0,2.5),z=rnorm(200,10,2))
dat <- data.frame(dat,x.sq=dat$x^2)
dat1 <- artificial.data(y~1+x+x.sq,dat,theta,S)
fm4 <- SpeciesMix(obs~1+x+x.sq,dat1$pa,dat,G=4,em.prefit=TRUE,em.refit=1,est.var=TRUE)
new.obs <- data.frame(x=seq(0,2.5,0.01),x.sq=seq(0,2.5,0.01)^2)
predict(fm4,new.obs)
```

```
print.archetype      Print archetypes from fitted model
```

Description

```
print.archetype
```

Arguments

```
x          a model object returned from SpeciesMix
...        further arguments passed to or from other methods.
```

Details

Prints the fitted model returned from SpeciesMix, giving the coefficients, standard errors and posterior probabilities

Method

```
print(x,...)
```

Authors

Piers Dunstan and Scott Foster

Examples

```
G <-4
S <- 50
theta <- matrix(c(-9,35,-32,0,0.7,0,-16,23,-8.2,-3,-0.6,0.8),4,3,byrow=TRUE)
dat <- data.frame(y=rep(1,200),x=runif(200,0,2.5),z=rnorm(200,10,2))
dat <- data.frame(dat,x.sq=dat$x^2)
dat1 <- artificial.data(y~1+x+x.sq,dat,theta,S)
fm4 <- SpeciesMix(obs~1+x+x.sq,dat1$pa,dat,G=4,em.prefit=TRUE,em.refit=1,est.var=TRUE)
print(fm4)
```

SpeciesMix

*Fit finite mixture models to species distributions***Description**

SpeciesMix is used to fit mixtures of glms to multivariate species data. The function uses BFGS to optimise the mixture likelihood. There is the option to use EM get appropriate starting parameters.

Usage

```
#SpeciesMix(sp.form,sp.data,covar.data,G=2, \n
#pars=NA, em.prefit=TRUE,em.steps=3, em.refit = 1, \n
#dist="bernoulli" , est.var = FALSE,residuals=FALSE,trace=TRUE)
```

Arguments

sp.form	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The response variable is arbitrary, all the responses are in sp.data.
sp.data	a data frame containing the species information. The frame is arranged so that each row is a site and each column is a species. Species names should be included as column names otherwise numbers from 1:S are assigned.
covar.data	a data frame containng the covariate data for each site. Names of columns must match that given in sp.form
G	number of archetype species to fit
pars	vector of parameters order is additive logisic transform of pi, glm coefficients by column (ie all intercepts first, etc)
em.prefit	find initial parameter values using EM
em.steps	number of EM steps to preform
em.refit	number of times to refit using EM
dist	a choice of "bernoulli" (presence absence data; default), "negbin" (abundance data with negative binomial) or "tweedie" (biomass data with tweedie distribution)
est.var	estimate the variance covariance matrix (TRUE/FALSE)
residuals	claculate species wise residuals
trace	the trace

Details

SpeciesMix is a wrapper for fitmix.cpp that allows for easier data input. The data frames are merged into the appropriate format for the use in fitmix.cpp. Minima is found using vmmmin (BFGS) and the gradients are calculated using CPPAD (auto differentiation)

Value

logl	loglikelihood
aic	AIC
bic	BIC
tau	posterior probabilities giving group membership. These values indicate which archetype each species belongs to.
pi	prior probabilities of archetype membership.
coef	coefficients for each species archetype
covar	variance co-variance matrix at the maxima. Calculated if est.var is TRUE
hessian	Hessian matrix at maximum likelihood. Calculated if est.var is TRUE
gradient	derivative of parameters at the maxima. Calculated if est.var is TRUE

Authors

Piers Dunstan and Scott Foster

Examples

```
G <-4
S <- 50
theta <- matrix(c(-9,35,-32,0,0.7,0,-16,23,-8.2,-3,-0.6,0.8),4,3,byrow=TRUE)
dat <- data.frame(y=rep(1,200),x=runif(200,0,2.5),z=rnorm(200,10,2))
dat <- data.frame(dat,x.sq=dat$x^2)
dat1 <- artificial.data(y~1+x+x.sq,dat,theta,S)
fm4 <- SpeciesMix(obs~1+x+x.sq,dat1$pa,dat,G=4,em.prefit=TRUE,em.refit=1,est.var=TRUE)
```

SpeciesMixtures

Package Description

Description

SpeciesMix is a package to fit finite mixture models to multispecies distributions. It implements the methods described in Dunstan et al. (in Review)

Index

*Topic **misc**

- artificial.data, 1
- clusterSelect, 3
- logLike.pars, 4
- predict.archetype, 5
- print.archetype, 6
- SpeciesMix, 7
- SpeciesMixtures, 8

artificial.data, 1

clusterSelect, 3

logLike.pars, 4

predict.archetype, 5

print.archetype, 6

SpeciesMix, 7

SpeciesMixtures, 8