Package ‘Bayesthresh’

February 19, 2015

Version 2.0.1
Date 2012-11-14
Title Bayesian thresholds mixed-effects models for categorical data
Author Fabio Mathias Correa <fmcorrea@uesc.br> and Julio Silvio de Sousa Bueno Filho <juliobuenof@gmail.com>
Maintainer Fabio Mathias Correa <fmcorrea@uesc.br>
Description This package fits a linear mixed model for ordinal categorical responses using Bayesian inference via Monte Carlo Markov Chains. Default is Nandran & Chen algorithm using Gaussian link function and saving just the summaries of the chains. Among the options, package allow for two other options of algorithms, for using Student’s “t” link function and for saving the full chains.
Depends R(>= 2.15.2), lme4 (>= 0.999375-35), MASS, VGAM, mvtnorm, matrixcalc, coda
License GPL (>= 2)
Repository CRAN
Date/Publication 2013-03-26 12:05:55
NeedsCompilation yes

R topics documented:

  ACGaussian .............................................................. 2
  ACt ................................................................. 2
  Bayes.factor .......................................................... 3
  Bayesthresh ........................................................... 4
  coef.Bayesthresh ....................................................... 6
  compVar .............................................................. 7
  MCGaussian ........................................................... 7
  MCMCsample .......................................................... 8
  MCt ................................................................. 9
  NCGaussian ............................................................ 9
ACt

NCt ................................................................. 10
plot.random.effects ........................................... 11
predict.Bayesthresh .......................................... 12
random.effects ................................................. 13
sensory ............................................................ 13
summary.Bayesthresh ......................................... 14

Index 15

ACGaussian

Albert and Chib algorithm with Gaussian distribution for latent variable

Description


Details

Internal use

References


See Also

Bayesthresh

ACt

Albert and Chib algorithm with t-Student distribution for latent variable

Description


Details

Internal use
Bayes.factor

References


See Also

Bayesthresh

Bayes.factor  Bayes factor of the two models

Description

Compares two models by evaluating their Bayes factor

Usage

Bayes.factor(model1, model2, inter=TRUE)

Arguments

model1  object of the class model "Bayesthresh"
model2  object of the class model "Bayesthresh"
inter   If TRUE, print to scale for interpretation of the Bayes factor

Details

At each step during the Markov chains, the marginal likelihood for a model is evaluated, conditioning on actual values for the parameters in that step. Bayes factor is then estimated by the ratios of the arithmetic means of marginal likelihoods from both models. Details of the implementation can be found in Sorensen and Gianola (2004). For a discussion of the possible interpretation of Bayes factors, see Jeffreys(1961)

References


Bayesthresh

Examples

data(sensory)

Consumer <- factor(sensory$consumer)
Sacarose <- factor(sensory$sacarose)

# Not run

#### Model 1

# Model with Gaussian link

dex1 <- Bayesthresh(flavor ~ (1|Consumer) + Sacarose, burn = 0, jump = 1,
  ef.iter = 10, data=sensory)
summary(dex1)

#### Model 2

# Model with t-Student link

dex2 <- Bayesthresh(flavor ~ (1|Consumer) + Sacarose, burn = 0, jump = 1,
  ef.iter = 10, algor=list(algorithm="NC", link="t"),data=sensory)
summary(dex2)
Bayes.factor(dex1,dex2)

Bayesthresh

Bayesian thresholds mixed-effects models for categorical data

Description

This package fits a linear mixed model for ordinal categorical responses using Bayesian inference via Monte Carlo Markov Chains. Default is Nandran & Chen algorithm using Gaussian link function and saving just the summaries of the chains. Among the options, package allow for two other options of algorithms, for using Student’s ‘t’ link function and for saving the full chains.

Usage

Bayesthresh(formula, data, subset, na.action, A=NULL,
  algor = list(algorithm="NC", link="Gaussian"),
  Write=FALSE, priors = list(ru = 10, su = 2, dre = 20, dse = 5),
  burn = 50, jump = 2, ef.iter = 4000, model=TRUE)

Arguments

formula a two-sided linear formula object describing the fixed-effects part of the model, with the response on the left of a ~ operator and the terms, separated by + operators, on the right. The vertical bar character "|" separates an expression for a model matrix and a grouping factor.
Bayesthresh

data

an optional data frame containing the variables named in formula. By default the variables are taken from the environment from which Bayesthres is called.

subset, na.action

further model specification arguments as in lm; see there for details.

A

Matrix of variance-covariance of random effects.

algor

is a list that contains the name of the algorithm to be used. By default the algorithm is the NC with function link Gaussian

Write

the Write is a function that by default is FALSE. If TRUE, the function save the iterations of the sampling process in the matrix

priors

priors is a list that contains the parameters of the priors used to estimate the variance components of random effects

burn, jump, ef.iter

are of the arguments of iteration. By default the burn, jump and ef.iter (effective iterations) are 50, 2 e 4000 respectively

model

logical scalar. If FALSE the model frame in slot frame is truncated to zero rows.

Details

subset an optional expression indicating the subset of the rows of data that should be used in the fit. This can be a logical vector, or a numeric vector indicating which observation numbers are to be included, or a character vector of the row names to be included. All observations are included by default.

na.action a function that indicates what should happen when the data contain NAs. The default action (na.fail) prints an error message and terminate if there are any incomplete observations.

algor the are three options algorithms, AC, MC and NC, with link function Gaussian and t-Student distribution. The object algor, by default is list(algorithm="NC", link="Gaussian").

Write if Write=TRUE, the chain of iterations is saved in the file output.txt. The convergence process can be analyzed by the library coda

priors The object defines the priors for the variance components of the AC and MC algorithms. For the NC algorithm can be also defined a prior of residual variance. Objects ru (shape parameter) and su (scale parameter) are the parameters of inverse gamma for the variance components. The NC algorithm allows to change parameters of the residual variance. dre (shape parameter) and dse (scale parameter) define the prior of the residual variance. By default, algorithms AC and MC have a residual variance equal to 1.

Examples

# Not run
data(sensory)

Consumer <- factor(sensory$consumer) # Random effect
Sacarose <- factor(sensory$sacarose) # Fixed effect

## Model
# Not run
dex1 <- Bayesthresh(cor ~ (1|Consumer) + Sacarose,
burn = 0, jump = 1, ef.iter = 10, data=sensory)
summary(dex1)

cbind.Bayesthresh

##  Coefficients for the fixed effects model

Description

Extract the fixed effects of model. Coefficients correspond to columns of space spanned by the
columns of design matrix for fixed effects

Usage

## S3 method for class 'Bayesthresh'
coef(object, HPDinterval=FALSE, prob=0.95,...)

Arguments

- **object**: is an object of class "Bayesthresh"
- **HPDinterval**: If HPDinterval=TRUE, return the credibility interval for fixed effects model. By
default is FALSE. See package coda
- **prob**: A numeric scalar in the interval (0,1) giving the target probability content of
the intervals. The nominal probability content of the intervals is the multiple of
1/nrow(obj) nearest to prob
- ... no usage

Examples

```r
# Not run
data(sensory)
Consumer <- factor(sensory$consumer)
Sacarose <- factor(sensory$sacarose)

### Model
# Not run
dex1 <- Bayesthresh(cor ~ (1|Consumer) + Sacarose, burn = 0,
jump = 1, ef.iter = 10, data=sensory, Write=TRUE)

cbind(dex1)
cbind(dex1, HPDinterval=TRUE)

# Not run
dex2 <- Bayesthresh(cor ~ (1|Consumer) + Sacarose, burn = 0,
jump = 1, ef.iter = 10, data=sensory)
cbind(dex2)
```
Description
Summaries for variance components for the random effects in the model

Usage

```r
compVar(object)
```

Arguments

- `object` object class "Bayesthresh"

Examples

```r
### Not run
data(sensory)

Consumer <- factor(sensory$consumer)
Sacarose <- factor(sensory$sacarose)

### Model
dex1 <- Bayesthresh(cor ~ (1|Consumer) + Sacarose, burn = 0,
jump = 1, ef.iter = 10, data=sensory)
compVar(dex1)
```

Description
Implementation follows Kizilkaya et. al. (2003), that presented an algorithm adapted from Cowles (1996) for mixed models with Gaussian distribution of the latent variable.
References


See Also

Bayesthresh

Description

Returns the chains of the MCMC process

Usage

MCMCsampling(object)

Arguments

object is an object of class "Bayesthres"

Examples

# Not run
data(sensory)

Consumer <- factor(sensory$consumer)
Sacarose <- factor(sensory$sacarose)

#### Model
# Not run
dex1 <- Bayesthresh(cor ~ (1|Consumer) + Sacarose, burn = 0, Write=TRUE,
jump = 1, ef.iter = 10, data=sensory)
MCMCsampling(dex1)
**MCt**

*Kizilkaya et. al. (2003) algorithm using Student’s "t" distribution as link function*

---

**Description**

The implementation follows Kizilkaya et. al. (2003) which presented a modification on Cowles (1996) algorithm that allows for using a Student’s "t" distribution as link function for the latent variable.

**Details**

Internal use

**References**


**See Also**

Bayesthresh

---

**NCGaussian**

*Nandram and Chen (1996) algorithm with Gaussian distribution for latent variable*

---

**Description**

This brings an implementation of Nandram and Chen (1996) algorithm adapted by Silva (2008) for mixed models with inverse Gaussian distribution used as a link function for the latent variable.

**Details**

Internal use
References


See Also

Bayesthresh

---

Nct

*Nandram and Chen (1996) algorithm with t-Student distribution for latent variable*

Description

Implementation of Nandram and Chen (1996) algorithm using inverse Student’s "t" distribution as link function for latent variable (adapted by Silva 2008).

Details

Internal use

References


See Also

Bayesthresh
plot.random.effects

Plot random effects of model

Description
Plot the estimates of random effects with confidence intervals

Usage
## S3 method for class 'random.effects'
plot(x, by = NULL, interval = "confidence", level = 0.95, xlab = NULL, ylab = NULL, xline = NULL, yline = NULL, pch = 19, col.points = 'red', col.seg = gray(0.5), ...)

Arguments
- x: "random.effects" object
- by: defines the element to be plotted
- interval: Define the interval to be used. See details
- level: the confidence level required
- xlab: a title for the x axis
- ylab: a title for the y axis
- ylim: the y limits of the plot
- xlim: the x limits of the plot
- pch: either an integer specifying a symbol or a single character to be used as the default in plotting points
- col.points: a specification for the default plotting color points
- col.seg: a specification for the default plotting color segments
- ...: Arguments to be passed to method, such as plot

Details
- interval: If interval='hpd' is plotted the posterior mean of the random effects with credibility interval. The credibility interval for this case is obtained from the sample using the function HPDinterval of the package coda. If interval='confidence', the credibility interval is obtained from normal distribution quantile (qnorm((1-level)/2, mean, sd). By default is interval='confidence'

Examples
# Not run
data(sensory)
Consumer <- factor(sensory$consumer)
predict.Bayesthresh

Sacarose <- factor(sensory$sacarose)

#### Model
# Not run
dex1 <- Bayesthresh(flavor ~ (1|Consumer) + Sacarose, Write=TRUE, burn = 10, jump = 2, ef.iter = 10, data=sensory)

rand <- random.effects(dex1, HPDinterval=TRUE)
plot(rand, interval='hpd')
plot(rand)

---

predict.Bayesthresh Extract the predict values

Description

Predict values based on the Bayesthres model object

Usage

```r
## S3 method for class 'Bayesthresh'
predict(object, ...)
```

Arguments

- `object` Object of class "Bayesthresh"
- `...` No usage

Examples

```r
# Not run
data(sensory)

Consumer <- factor(sensory$consumer)
Sacarose <- factor(sensory$sacarose)

#### Model
# Not run
dex1 <- Bayesthresh(flavor ~ (1|Consumer) + Sacarose, burn = 0, jump = 1, ef.iter = 10, data=sensory)
predict(dex1)
```
random.effects

Extract the random effects of the model

Description

Extract the random effects of the model of the class Bayesthres

Usage

random.effects(object, HPDinterval=FALSE, prob=0.95)

Arguments

- **object**: is an object of class “Bayesthres”
- **HPDinterval**: If HPDinterval=TRUE, return the credibility interval for fixed effects model. By default is FALSE. See package coda
- **prob**: A numeric scalar in the interval (0,1) giving the target probability content of the intervals. The nominal probability content of the intervals is the multiple of 1/nrow(obj) nearest to prob

Examples

```r
# Not run
data(sensory)

Consumer <- factor(sensory$consumer)
Sacarose <- factor(sensory$sacarose)

### Model
# Not run
dex1 <- Bayesthresh(cor ~ (1|Consumer) + Sacarose, burn = 0, Write=TRUE,
                   jump = 1, ef.iter = 10, data=sensory)
random.effects(dex1)
random.effects(dex1, HPDinterval=TRUE)
```

sensory

Sensory analysis of banana

Description

An data set for illustrated of the use package Bayesthresh

Usage

data(sensory)
sensory
Details

Data used by Silva and Bueno-Filho (2010) for illustrated the using of the algorithm ACG and ACt.

References


summary.Bayesthresh

Description

Summary of Bayesthresh

Usage

## S3 method for class 'Bayesthresh'
summary(object, ...)

Arguments

- object: object class of "Bayesthresh"
- ...: no usage
Index

*Topic methods
  predict.Bayesthresh, 12
  random.effects, 13
  sensory, 13
  summary.Bayesthresh, 14

*Topic models
  Act, 2
  compVar, 7
  predict.Bayesthresh, 12
  random.effects, 13
  sensory, 13
  summary.Bayesthresh, 14

*Topic plot.random.effects
  plot.random.effects, 11

*Topic random.effects
  plot.random.effects, 11

*Topic variance components
  compVar, 7

ACGaussian, 2
Act, 2

Bayes.factor, 3
Bayesthresh, 2, 3, 4, 8–10

coef.Bayesthresh, 6
compVar, 7

lm, 5

MCGaussian, 7
MCMCsamp, 8
Mct, 9

NCGaussian, 9
Nct, 10

plot, 11
plot.random.effects, 11
predict.Bayesthresh, 12
random.effects, 13
sensory, 13
summary.Bayesthresh, 14