

# 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

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

<b>Index</b>	<b>15</b>
--------------	-----------

---

ACGaussian	<i>Albert and Chib algorithm with Gaussian distribution for latent variable</i>
------------	---

---

### Description

Implementation of Albert and Chib (1993) algorithm as described by Sorensen et. al (1995) for using in categorical data analysis

### Details

Internal use

### References

ALBERT, J.H.; CHIB, S. (1993) Bayesian analysis of binary and polychotomus response data. Journal American Statistical Association, Washington, v.88, n.422, p.669-679.

SORENSEN, D.A.; ANDERSEN S.; GIANOLA D.; KORSGAARD I.(1995) Bayesian inference in threshold models using Gibbs sampling, Genetics Selection Evolution, Les Ulis, v. 27, n. 3, p. 229-249.

### See Also

[Bayesthresh](#)

---

ACt	<i>Albert and Chib algorithm with t-Student distribution for latent variable</i>
-----	--

---

### Description

Algorithm described by Silva and Bueno Filho (2010) for using in categorical data analysis adapted from Albert and Chib (1993).

### Details

Internal use

**References**

ALBERT, J.H.; CHIB, S. (1993) Bayesian analysis of binary and polychotomus response data. Journal American Statistical Association, Washington, v.88, n.422, p.669-679.

SILVA, W.J.; BUENO FILHO. (2010) An algorithm for thresholds models using cumulative Gaussian or cummulative Student's "t" distributions. Brazilian Journal of Biometry, Sao Paulo, v. 28, n. 3, p.59-83.

**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**

SORENSEN, D.; GIANOLA, D. Likelihood, bayesian and MCMC methods in quantitative genetics. United States of America: Springer, 2004. 740 p.

JEFFREYS, H. Theory of probability. Oxford: Clarendon Press, 1961. 470 p.

**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  $\sim$  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.

<code>data</code>	an optional data frame containing the variables named in <code>formula</code> . By default the variables are taken from the environment from which <code>Bayesthresh</code> is called.
<code>subset, na.action</code>	further model specification arguments as in <code>lm</code> ; see there for details.
<code>A</code>	Matrix of variance-covariance of random effects.
<code>algor</code>	is a list that contains the name of the algorithm to be used. By default the algorithm is the NC with function link Gaussian
<code>Write</code>	the <code>Write</code> is a function that by default is FALSE. If TRUE, the function save the iterations of the sampling process in the matrix
<code>priors</code>	<code>priors</code> is a list that contains the parameters of the priors used to estimate the variance components of random effects
<code>burn, jump, ef.iter</code>	are of the arguments of iteration. By default the <code>burn</code> , <code>jump</code> and <code>ef.iter</code> (effective iterations) are 50, 2 e 4000 respectively
<code>model</code>	logical scalar. If FALSE the model frame in slot <code>frame</code> 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)
```

---

```
coef.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

```
# 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)

coef(dex1)
coef(dex1, HPDinterval=TRUE)

# Not run
dex2 <- Bayesthresh(cor ~ (1|Consumer) + Sacarose, burn = 0,
jump = 1, ef.iter = 10, data=sensory)
coef(dex2)
```

---

compVar	<i>Variance component estimates</i>
---------	-------------------------------------

---

**Description**

Summaries for variance components for the random effects in the model

**Usage**

```
compVar(object)
```

**Arguments**

object            object class "Bayesthresh"

**Examples**

```
### 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)
```

---

MCGaussian	<i>Kizilkaya et. al. (2003) algorithm with Gaussian distribution for latent variable</i>
------------	--

---

**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.

**Details**

Internal use

**References**

COWLES, M. K. (1996) Accelerating Monte Carlo Markov chain convergence for cumulative link generalized linear models, *Statistics and Computing*, Netherlands: Springer, v.6, n. 2 p. 101-111.

KIZILKAYA, K.; CARNIER, P.; ALBERA, A.; BITTANTE, G.; TEMPELMAN, R. J. (2003) Cumulative t-link threshold models for genetic analysis of calving ease scores, *Genetics Selection Evolution, Les Ulis*, v. 35, p. 489-512.

**See Also**

[Bayesthresh](#)

---

MCMCsample

*MCMC sample*

---

**Description**

Returns the chains of the MCMC process

**Usage**

MCMCsample(object)

**Arguments**

object is an object of class "Bayesthresh"

**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)
MCMCsample(dex1)
```



---

MCt	<i>Kizilkaya et. al. (2003) algorithm using Student's "t" distribution as link function</i>
-----	---

---

### 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

COWLES, M. K. Accelerating Monte Carlo Markov chain convergence for cumulative link generalized linear models, *Statistics and Computing*, Netherlands: Springer, v.6, n. 2 p. 101-111, June 1996.

KIZILKAYA, K.; CARNIER, P.; ALBERA, A.; BITTANTE, G.; TEMPELMAN, R. J. Cumulative t-link threshold models for genetic analysis of calving ease scores, *Genetics Selection Evolution*, Les Ulis, v. 35, p. 489-512, Mar. 2003.

### See Also

[Bayesthresh](#)

---

NCGaussian	<i>Nandram and Chen (1996) algorithm with Gaussian distribution for latent variable</i>
------------	---

---

### 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**

NANDRAM, B.; CHEN, M. H. (1996) Reparameterizing the generalized linear model to accelerate gibbs sampler convergence, *Journal of Statistical Computation and Simulation*, Abingdon, v. 54, p.129-144.

SILVA, J.W. (2008) Algorithms for thresholds models using Normal and Student-t cumulative distributions. Thesis (Doctorate in Statistics and Agricultural Experimentation), University Federal of Lavras, p. 99.

**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**

NANDRAM, B.; CHEN, M. H. (1996) Reparameterizing the generalized linear model to accelerate gibbs sampler convergence, *Journal of Statistical Computation and Simulation*, Abingdon, v. 54, p.129-144.

SILVA, J.W. (2008) Algorithms for thresholds models using Normal and Student-t cumulative distributions. Thesis (Doctorate in Statistics and Agricultural Experimentation), University Federal of Lavras, p. 99.

**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,
     xlim = NULL, ylim = 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 x limits of the plot
xlim	the y 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 <a href="#">plot</a>

## 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)
```

```

Sacarose <- factor(sensory$sacarose)

#### Model
# Not run
dex1 <- Bayesthresh(cor ~ (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 Bayesthresh model object

## Usage

```

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

```

## Arguments

<code>object</code>	Object of class "Bayesthresh"
<code>...</code>	No usage

## Examples

```

# 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	<i>Extract the random effects of the model</i>
----------------	--

---

**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**

```
# 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	<i>Sensory analysis of banana</i>
---------	-----------------------------------

---

**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**

SILVA, W.J.; BUENO FILHO. (2010) An algorithm for thresholds models using cumulative Gaussian or cummulative Student's "t" distributions. Brazilian Journal of Biometry, Sao Paulo, v. 28, n. 3, p.59-83.

---

summary.Bayesthresh    *Summary*

---

**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](#)

MCMCsample, [8](#)

MCT, [9](#)

NCGaussian, [9](#)

NCT, [10](#)

plot, [11](#)

plot.random.effects, [11](#)

predict.Bayesthresh, [12](#)

random.effects, [13](#)

sensory, [13](#)

summary.Bayesthresh, [14](#)