Package 'gammSlice'

January 3, 2019

Version 2.0-2

Date 2019-01-03

Title Generalized Additive Mixed Model Analysis via Slice Sampling

Maintainer Matt P. Wand <matt.wand@uts.edu.au>

Depends KernSmooth, lattice, mgcv

Description Uses a slice sampling-based Markov chain Monte Carlo to
conduct Bayesian fitting and inference for generalized additive
mixed models. Generalized linear mixed models and generalized
additive models are also handled as special cases of generalized
additive mixed models. The methodology and software is described
in Pham, T.H. and Wand, M.P. (2018). Australian and New Zealand
Journal of Statistics, 60, 279-330 <DOI:10.1111/ANZS.12241>.

License GPL (>= 2)

NeedsCompilation yes

Author Tung H. Pham [aut],
Matt P. Wand [aut, cre]

Repository CRAN

Date/Publication 2019-01-03 08:30:03 UTC

R topics documented:

gSlc — *Generalized additive mixed model analysis via slice sampling*

Description

Use slice sampling-based Markov chain Monte Carlo to fit a generalized additive mixed model.

Usage

```
gSlc(formula, data = NULL, random = NULL, family, control = gSlc.control())
```

Arguments

formula	Formula describing the generalized additive mixed model.
data	Data frame containing the input data. This argument is optional.
random	List describing random effects structure.This argument is optional.
family	Distribution family of the response variable. The options are "binomial" and "poisson".
control	Control options specified by gSlc.control.

Details

A Bayesian generalized additive mixed model is fitted to the input data according to specified formula. Such models are special cases of the general design generalized linear mixed models of Zhao, Staudenmayer, Coull and Wand (2003). Markov chain Monte Carlo, with slice sampling for the fixed and random effects, is used to obtain samples from the posterior distributions of the model parameters. Full details of the sampling scheme are in the appendix of Pham and Wand (2018).

Value

An object of class ″gScl″. The functions summary() and plot() are used to obtain a summary and plot of the fits. The object is a list with the following components:

nu	Matrix containing Markov chain Monte Carlo samples of the entire nu=(beta,u) vector. The rows correspond to Markov chain Monte Carlo replicates and the columns correspond to entries of the nu=(beta,u) vector.
beta	Matrix containing Markov chain Monte Carlo samples of the beta vector corresponding to the linear components of the model. The rows correspond to Markov chain Monte Carlo replicates and the columns correspond to entries of the beta vector.
sigmaSquared	Matrix containing Markov chain Monte Carlo samples of the entire sigma squared vector. The rows correspond to Markov chain Monte Carlo replicates and the columns correspond to entries of the sigmaSquared vector.
y	Response data vector.
XlinPreds	Matrix containing predictors that are purely linear components of the model.

linPredNames	Names of XlinPreds.
XsplPreds	Matrix containing predictors that are penalised spline components of the model.
splPredNames	Names of XsplPreds.
Zspl	Horizontal concatenation of each of the spline basis "Z" matrices used for smooth function components.
ncZspl	Vector giving the numbers of columns in the horizontal partition of Zspl corresponding to each smooth function component.
range.x.list	List containing values of the range.x input to the internal ZOSull() function.
intKnots.list	List containing values of the intKnots input to the internal ZOSull() function.
family	Character string indicating the family of the fitted model; either "binomial" or "poisson".
modelType	Charater string indicating the type of model fitted.

Author(s)

Tung Pham <tungstats@gmail.com> and Matt Wand <matt.wand@uts.edu.au>

References

Neal, R.M. (2003).
Slice sampling (with discussion).
The Annals of Statistics, **31**, 705-767.

Pham, T. and Wand, M.P. (2018).
Generalized additive mixed model analysis via gammSlice.
Australian and New Zealand Journal of Statistics, **60**, 279-300.

Zhao, Y., Staudenmayer, J., Coull, B.A. and Wand, M.P. (2003).
General design Bayesian generalized linear mixed models.
Statistical Science, **21**, 35-51.

See Also

gSlc.control, plot.gSlc, summary.gSlc

Examples

```
## Not run:
# Example 1 of Pham & Wand (2018):

set.seed(39402)
m <- 100 ; n <- 2
beta0True <- 0.5 ; betaxTrue <- 1.7
sigsqTrue <- 0.8 ; idnum <- rep(1:m,each=n)
x <- runif(m*n)
U <- rep(rnorm(m,0,sqrt(sigsqTrue)),each=n)
mu <- 1/(1+exp(-(beta0True+betaxTrue*x+U)))
y <- rbinom((m*n),1,mu)
```

```
fit1 <- gSlc(y ~ x,random = list(idnum = ~1),family = "binomial")
summary(fit1)

## End(Not run)

## Not run:
# Example 2 of Pham & Wand (2018):

set.seed(53902)
n <- 400 ; x <- runif(n)
fTrue <- function(x) return(cos(4*pi*x) + 2*x - 1)
mu <- exp(fTrue(x)) ; y <- rpois(n,mu)
fit2 <- gSlc(y~s(x),family="poisson")
summary(fit2)
plot(fit2)

## End(Not run)

## Not run:
# Example 3 of Pham & Wand (2018):

set.seed(981127)
n <- 500 ; betax1True <- 0.5;  x1 <- sample(c(0,1),n,replace=TRUE)
x2 <- runif(n) ; fTrue <- function(x) return(sin(2*pi*x))
mu <- 1/(1+exp(-(betax1True*x1+fTrue(x2)))) ; y <- rpois(n,mu)
y <- rbinom(n,1,mu)
fit3 <- gSlc(y ~ x1 + s(x2),family="binomial")
summary(fit3)
plot(fit3)

## End(Not run)

## Not run:
# Example 4 of Pham & Wand (2018):

set.seed(2966703)
m <- 100 ; n <- 10;  x1 <- runif(m*n);   x2 <- runif(m*n)
idnum <- rep(1:m,each=n) ; sigsqTrue <- 1
U <- rep(rnorm(m,0,sqrt(sigsqTrue)),each=n)
mu <- exp(U + cos(4*pi*x1) + 2*x1 + sin(2*pi*x2^2)) ; y <- rpois(m*n,mu)
fit4 <- gSlc(y ~ s(x1) + s(x2),random = list(idnum=~1),family = "poisson")
summary(fit4)
plot(fit4)

## End(Not run)
```

gSlc.control   *Controlling generalized additive mixed model fitting via slice sampling*

Description

Function for optional use in calls to gSlc() to control Markov chain Monte Carlo sample sizes values and other specifications for slice sampling-based fitting of generalized additive mixed models.

Usage

```
gSlc.control(nBurn=5000,nKept=5000,nThin=5,fixedEffPriorVar=1e10,
             sdPriorScale=1e5,numBasis=NULL,preTransfData=TRUE,msgCode=1)
```

Arguments

nBurn   The length of the Markov chain Monte Carlo burnin. The first nBurnin Markov
                     chain Monte Carlo samples are discarded. The default value of nBurnin is 5000.

nKept   The number of kept Markov chain Monte Carlo samples after the burnin period.
                     The default value of nKept is 5000.

nThin   Thinning factor applied to the retained Markov chain Monte Carlo samples. Set-
                     ting nThin to be an intege greater than 1 results in every nThinth value in the
                     post-burnin samples being retained. The final Markov chain Monte Carlo sam-
                     ple size is an integer close to nIter divided by nIter. The default value of
                     nThin is 5.

fixedEffPriorVar
                     The variance in the independent zero mean Normal priors of the fixed effect
                     parameters after the data of each predictor have been transformed to the interval
                     [0,1]. The default value of fixedEffPriorVar is 1e10.

sdPriorScale   The scale parameter in the Half Cauchy priors on standard deviation parameters
                     after the data of each predictor have been transformed to the interval [0,1]. The
                     default value of sdPriorScale is 1e5.

numBasis   Vector of positive integers specifying the number of spline basis functions to be
                     used for each smooth function component.

preTransfData   Boolean flag:
                     TRUE = pre-transform each of the predictors to unit interval for Bayesian anal-
                     ysis with the priors specified by fixedEffPriorVar and sdPriorScale (the
                     default),
                     FALSE = do not perform any pre-transformation of the predictors.

msgCode   A code for specification of the nature of messages printed concerning progress
                     of the Markov chain Monte Carlo sampling:
                     0 = no messages printed,
                     1 = percentages 1,2,...,10 and then 20,30,...,100
                     (the default),
                     2 = percentages 1,2,...,100,
                     3 = percentages 10,20,...,100.

Author(s)

Tung Pham <tungstats@gmail.com> and Matt Wand <matt.wand@uts.edu.au>.

References

Pham, T. and Wand, M.P. (2018). Generalized additive mixed model analysis via gammSlice. *Australian and New Zealand Journal of Statistics*, **60**, 279-300.

Zhao, Y., Staudenmayer, J., Coull, B.A. and Wand, M.P. (2003). General design Bayesian generalized linear mixed models. *Statistical Science*, **21**, 35-51.

See Also

gSlc

Examples

```
## Not run:
library(gammSlice)
set.seed(39402) ; m <- 100 ; n <- 2
beta0True <- 0.5 ; betaxTrue <- 1.7 ; sigsqTrue <- 0.8
idnum <- rep(1:m,each=n) ; x <- runif(m*n)
U <- rep(rnorm(m,0,sqrt(sigsqTrue)),each=n)
mu <- 1/(1+exp(-(beta0True+betaxTrue*x+U)))
y <- rbinom((m*n),1,mu)
fit <- gSlc(y ~ x,random = list(idnum = ~1),family = "binomial")
summary(fit)

# Illustration of user-specified priors:

fitMyPriors <- gSlc(y ~ x,random = list(idnum = ~1),
                    family = "binomial",
                    control = gSlc.control(fixedEffPriorVar=1e13,
                                           sdPriorScale=1e3))
summary(fitMyPriors)

# Illustration of specification of larger Markov chain Monte Carlo samples:

fitBigMCMC <- gSlc(y ~ x,random = list(idnum = ~1),family = "binomial",
                   control = gSlc.control(nBurn=10000,nKept=8000,nThin=10))
summary(fitBigMCMC)

## End(Not run)
```

---

indonRespir   *Eespiratory infection in Indonesian children*

Description

Indonesian Children's Health Study of respiratory infections for a cohort of 275 Indonesian children. The data are longitudinal with each child having between 1 and 6 repeated measurements.

Usage

```
data(indonRespir)
```

Format

A data frame with 1200 observations on the following 12 variables:

`idnum` child identification number.

`respirInfec` indicator of presence of resipiratory infection.

`age` age of the child in years.

`vitAdefic` indicator of Vitamin A deficiency:
 1 = the child had Vitamin A deficiency,
 0 = the child did not have Vitamin A deficiency.

`female` indicator of child being female:
 1 = the child is female,
 0 = the child is male.

`height` height of the child in centimeters.

`stunted` indicator of the child being "short for his/her age":
 1 = the child is "short for his/her age",
 0 = the child is not "short for his/her age"

`visit2` indicator that the child had exactly 2 clinical visits:
 1 = the exact number of clinical visits was 2,
 0 = the exact number of clinical visits was not 2.

`visit3` indicator that the child had exactly 3 clinical visits:
 1 = the exact number of clinical visits was 3,
 0 = the exact number of clinical visits was not 3.

`visit4` indicator that the child had exactly 4 clinical visits:
 1 = the exact number of clinical visits was 4,
 0 = the exact number of clinical visits was not 4.

`visit5` indicator that the child had exactly 5 clinical visits:
 1 = the exact number of clinical visits was 5,
 0 = the exact number of clinical visits was not 5.

`visit6` indicator that the child had exactly 6 clinical visits:
 1 = the exact number of clinical visits was 6,
 0 = the exact number of clinical visits was not 6.

Source

Sommer, A. (1982). *Nutritional Blindness*. New York: Oxford University Press.

References

Diggle, P., Heagerty, P., Liang, K.-L. and Zeger, S. (2002). *Analysis of Longitudinal Data (Second Edition)*. Oxford: Oxford University Press.

Examples

```
library(gammSlice) ; data(indonRespir)
plot(indonRespir$age,jitter(indonRespir$respirInfec))
```

---

plot.gSlc   *Plot smooth function components of* gSlc() *fits*

Description

Smooth function components of generalized additive mixed model fits obtained via gSlc are plotted.

Usage

```
## S3 method for class 'gSlc'
plot(x,gridSize=401,colour = TRUE,responseScale = FALSE,
                rug = TRUE,rugColour="dodgerblue",curveColour = "darkgreen",
                varBandPolygon = TRUE,varBandColour = "palegreen",
                xlab = NULL,ylab = NULL,bty = "l",cex.axis = 1,
                cex.lab = 1,...)
```

Arguments

x	gSlc() fit object.
gridSize	Number of grid points used in graphical display of smooth function fits.
colour	Boolean flag: TRUE = produce colour plots FALSE = produce black and white plots.
responseScale	Boolean flag: TRUE = the smooth function fits are plotted on the response scale FALSE = the smooth function fits are plotted on the link scale (the default).
rug	Boolean flag: TRUE = add rug graphics to the base of each smooth function plot showing the predictor data (the default), FALSE = do not add rug graphs.
rugColour	colour of the rug graphics. The default value is "dodgerblue".
curveColour	colour of the curves in the smooth function display. The default value is "dark-green".
varBandPolygon	Boolean flag: TRUE = display the variability band as a polygon (the default), FALSE = display the variability band using dashed curves.
varBandColour	colour of the variability band polygon in the smooth function display. The default value is "palegreen".
xlab	optional argument: character string vector for horizontal labels for smooth function plots.

ylab	optional argument: character string vector for vertical labels for smooth function plots.
bty	character string which specifies the type of box which is drawn about plots. See `help(par)` for details. The default value is "l".
cex.axis	positive number specifying the factor by which numbers along the axes are expanded.
cex.lab	positive number specifying the factor by which characters in the axis labels are expanded.
...	place-holder for other graphic parameters.

Details

For each smooth function component of the generalized additive mixed model specified in the call to gSlc the pointwise posterior mean is plotted along with a shaded polygon corresponding to pointwise 95% credible sets.

Author(s)

Tung Pham <tungstats@gmail.com> and Matt Wand <matt.wand@uts.edu.au>.

References

Pham, T. and Wand, M.P. (2018). Generalized additive mixed model analysis via gammSlice. *Australian and New Zealand Journal of Statistics*, **60**, 279-300.

See Also

gSlc, summary.gSlc

Examples

```
library(gammSlice)
set.seed(53902)
n <- 400 ; x <- runif(n)
fTrue <- function(x) return(cos(4*pi*x) + 2*x - 1)
mu <- exp(fTrue(x)) ; y <- rpois(n,mu)
fit <- gSlc(y~s(x),family="poisson",control=gSlc.control(nBurn=200,nKept=200,nThin=1,msgCode=0))
plot(fit)
plot(fit,responseScale=TRUE,rug=FALSE)
points(x,y,col="dodgerblue")
```

---

summary.gSlc   *Summary of the generalized additive mixed model fit produced by* gSlc

---

Description

A graphical table showing, for key model parameters, the Markov chain Monte Carlo samples, diagnostic plots and numerical summaries.

Usage

```
## S3 method for class 'gSlc'
summary(object,colour=TRUE,paletteNumber=1,...)
```

Arguments

object	A gSlc() fit object.
colour	Boolean flag: TRUE = produce a colour graphical table, FALSE = produce a black and white graphical table .
paletteNumber	If colour = TRUE then there there are two possible colour palettes. These are determined by whether paletteNumber is set to 1 or 2. The default is paletteNumber=1.
...	place-holder for additional arguments.

Details

The columns of the graphical table are:

1. parameter name,
2. trace plot of the Markov chain Monte Carlo sample,
3. plot of Markov chain Monte Carlo sample against its lag 1 sample,
4. sample autocorrelation function,
5. kernel density estimate of the posterior density function,
6. posterior mean and 95% credible interval.

Author(s)

Tung Pham <tungstats@gmail.com> and Matt Wand <matt.wand@uts.edu.au>.

References

Pham, T.H. and Wand, M.P. (2018). Generalized additive mixed model analysis via gammSlice. *Australian and New Zealand Journal of Statistics*, **60**, 279-300.

See Also

gSlc, plot.gSlc

Examples

```
library(gammSlice)
set.seed(39402) ; m <- 100 ; n <- 2
beta0True <- 0.5 ; betaxTrue <- 1.7 ; sigsqTrue <- 0.8
idnum <- rep(1:m,each=n) ; x <- runif(m*n)
U <- rep(rnorm(m,0,sqrt(sigsqTrue)),each=n)
mu <- 1/(1+exp(-(beta0True+betaxTrue*x+U)))
y <- rbinom((m*n),1,mu)
fit1 <- gSlc(y ~ x,random = list(idnum = ~1),family = "binomial",
            control = gSlc.control(nBurn=150,nKept=100,nThin=1))
summary(fit1)
summary(fit1,paletteNumber = 2)
summary(fit1,colour = FALSE)

## Not run:
# Re-fit with higher Markov chain Monte Carlo sample:

fit2 <- gSlc(y ~ x,random = list(idnum = ~1),family = "binomial")
summary(fit2)
summary(fit2,paletteNumber = 2)
summary(fit2,colour = FALSE)

## End(Not run)
```

---

toenail   *Toenail infection clinical trial*

Description

Data from a clinical trial in which two anti-fungal treatments for toenail infection are compared.

Usage

```
data(toenail)
```

Format

A data frame with 1908 observations on the following 5 variables:

idnum patient identification number.

onycholysis indicator concerning the severity of onycholysis:
    1 = moderate or severe onycholysis,
    0 = no or mild onycholysis.

terb indicator of whether the treatement was terbinafine:
    1 = treatment was terbinafine,
    0 = treatment was itraconazole.

months time in months since the the start of the trial when clinical visit took place.

visit visit number.

References

De Backer, M., De Vroey, C., Lesaffre, E., Scheys, I. and De Keyser, P. (1998). Twelve weeks of continuous oral therapy for toenail onychomycosis caused by dermatophytes: a double-blind comparative trial of terbinafine 250 mg/day versus itraconazole 200 mg/day. *Journal of the American Academy of Dermatology*, **38**, S57-S63.

Examples

```
library(gammSlice) ; data(toenail)
plot(jitter(toenail$terb),jitter(toenail$onycholysis))
```

Index