Package 'brea'

August 30, 2025

Version 0.4.2

Date 2025-08-30

Title Bayesian Recurrent Events Analysis

Description Functions to produce MCMC samples for posterior inference in semiparametric Bayesian discrete time competing risks recurrent events models and multistate models.

Maintainer Adam J King <king@cpp.edu>

License GPL-3

Suggests knitr, rmarkdown, survival, coda

VignetteBuilder knitr

NeedsCompilation no

Author Adam J King [aut, cre]

Repository CRAN

Date/Publication 2025-08-30 16:20:02 UTC

Contents

brea_mcmc 1
brea_mcmc_ms 4

Index **9**

`brea_mcmc` *Bayesian Discrete Survival Inference*

Description

This function performs Metropolis-Hastings exploration of the posterior distribution for Bayesian discrete time survival models. The models may have competing risks, semiparametric covariate effects (like arbitrary baseline hazards), and random effects shared across repeated events and correlated across competing risks.

Usage

```
brea_mcmc(x, y, priors = NULL, S = 1000, B = 100, n = NULL, K = NULL,
store_re = FALSE, block_mh = TRUE)
```

Arguments

x	a matrix of positive whole numbers or a dataframe with positive whole or factor columns specifying the values of the (discretized) predictors at each person-time point; the (i,j) entry of x is the value of predictor j at discrete person-time observation i
y	a nonnegative whole number vector or matrix indicating event occurrence at each person-time point; with only one event type, the i^th entry of the vector y is 0 for no event occurrence at observation i and 1 for event occurrence (or potentially greater than 1 if that observation represents replicated observations; see entry for n); with more than one event type, the (i,j) entry of the matrix y counts the number of events of type j occurring at discrete person-time observation i
priors	a list with one element for each predictor variable (column of x) specifying the prior type to use for that predictor; see Details for more information
S	the number of MCMC iterations to perform post-burn-in (post-burn-in iterations are stored in the object returned by the brea_mcmc function)
B	the number of burn-in MCMC iterations to perform (burn-in iterations are not stored)
n	a vector of positive whole numbers with length equal to the number of person-time observations; the i^th entry of n is the number of replicated observations that observation i represents (defaults to 1 for each observation if NULL is supplied)
K	a positive whole number vector whose m^th entry is the number of distinct values that the m^th discretized predictor may assume; if NULL is supplied, this vector will be automatically determined from x
store_re	if TRUE, the random effects are stored from each post-burn-in MCMC iteration, and if FALSE they are not stored
block_mh	if TRUE, a block Metropolis-Hastings algorithm will be used to update parameters for quantitative covariates using the gmrf prior type; otherwise, single-site random walk Metropolis updates will be used

Details

The data provided to the brea_mcmc function is specified at the person-time level: there is one row in x and y for each discrete time point each person or thing was at risk for event occurrence. All predictors in x must be encoded as factors or their corresponding integer codes. The underlying type of predictor is specified in the priors argument, which is a list with one element for each predictor variable which specifies both the type of that predictor and the prior distribution to use. The allowed predictor types are:

- "cat" for categorical variables. The first element of the prior specification list is the string "cat", and the second element is a positive conditional prior standard deviation parameter.

- "gmrf" for underlying continuous predictors; continuous predictors should be discretized with cut() before being included into x; Gaussian Markov random field (GMRF) priors are then used to smooth the effects of adjacent categories of the discretized continuous predictor. The first element of the prior specification list is the string "gmrf", the second element is a prior degrees of freedom for the scaled inverse chi-squared prior for the random walk increment variance, and the third element is a prior scale for the scaled inverse chi-squared.

- "re" for variables (like subject id numbers) that represent random effects. The first element of the prior specification list is the string "re", the second element is a prior degrees of freedom for an inverse Wishart prior for the random effects covariance, and the third element is a prior scale matrix for the random effects covariance.

- "zero" for predictors that are not used in the current MCMC run. This is provided as a convenient way to exclude predictors from certain runs. The first and only element of the prior specification list is the string "zero".

Value

A list providing the values from each of the S post-burn-in MCMC iterations for the intercept, the other linear predictor parameters, standard deviations of the GMRF random walk increments, and covariances of the random effects:

b_0_s	an R by S matrix (where R is the number of competing risks) whose (r,s) entry is the intercept for risk r at MCMC iteration s
b_m_s	a list whose m^th element is the R by K[m] by S array whose (r,k,s) entry is the value of the linear predictor contribution of value k for predictor m on risk r at MCMC iteration s
s_m_s	a list whose m^th element is, in the case of a GMRF prior, the R by S matrix whose (r,s) entry is the random walk standard deviation for predictor m and competing risk r at MCMC iteration s, or, in the case of a random effects priors, the R by R by S array whose (,,s) entry is the random effects covariance matrix for predictor m at MCMC iteration s
b_m_a	a list whose m^th element is the length K[m] vector giving the number of accepted Metropolis proposals for each value of predictor m across all S post-burn-in MCMC iterations

Author(s)

Adam King

Examples

```
# leukemia remission times data from Table 1 of "Regression Models and
# Life-Tables" (Cox, 1972)

# times of event occurrence or right censoring:
time <- c(6,6,6,6,7,9,10,10,11,13,16,17,19,20,22,23,25,32,32,34,35,
          1,1,2,2,3,4,4,5,5,8,8,8,8,11,11,12,12,15,17,22,23)

# event/censoring indicator (1 for event occurrence, 0 for right censoring):
```

```
event <- c(0,1,1,1,1,0,0,1,0,1,1,0,0,0,1,1,0,0,0,0,0,
           1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)

# treatment group indicator (1 for treatment, 2 for control):
treat <- c(rep(1,21),rep(2,21))

# total number of person-time observations among all subjects:
N <- sum(time)

# create and fill in person-time predictor and event variables:
x <- matrix(0,N,2)  # first column is time, second column is treatment
y <- numeric(N)
next_row <- 1  # next available row in the person-time variables
for (i in seq_along(time)) {
  rows <- next_row:(next_row + time[i] - 1)  # observations for subject i
  x[rows,1] <- seq_len(time[i])  # time is integers 1,2,...,time[i]
  x[rows,2] <- rep(treat[i],time[i])  # treatment group is constant
  y[rows] <- c(rep(0,time[i] - 1),event[i])  # outcome is 0's until event
  next_row <- next_row + time[i]  # increment the next available row pointer
}

# group the time-at-risk variable into 3-week intervals:
x[,1] <- cut(x[,1],seq(0,36,3),labels=FALSE)

# use GMRF prior for time, and categorical prior for treatment group:
priors <- list(list("gmrf",3,.01),list("cat",4))

# run the default of 100 burn-in iterations followed by 1,000 stored iterations:
fit <- brea_mcmc(x,y,priors)

# examine the structure of the returned posterior samples and acceptance counts:
str(fit)

# posterior samples of the treatment and control group parameters:
b_treatment <- fit$b_m_s[[2]][1,1,]
b_control <- fit$b_m_s[[2]][1,2,]

# posterior sample of treatment effect on linear predictor scale:
d <- b_control - b_treatment

# posterior mean, median, and sd of treatment effect on linear predictor scale:
mean(d); median(d); sd(d)

# posterior mean and median hazard ratios:
mean(exp(d)); median(exp(d))
```

---

brea_mcmc_ms *Bayesian Discrete Multistate Inference*

Description

This function performs Metropolis-Hastings exploration of the posterior distribution for Bayesian discrete time multistate models. The models may have multiple states, competing risks, semiparametric covariate effects (like arbitrary baseline hazards), and random effects shared across repeated events and correlated across states and competing risks.

Usage

```
brea_mcmc_ms(x, y, priors = NULL, S = 1000, B = 100, n = NULL, K = NULL, store_re = FALSE)
```

Arguments

x	a list with one element for each state in the multistate model; each list element is a matrix of positive whole numbers or a dataframe with positive whole or factor columns specifying the values of the (discretized) predictors at each person-time point; the (i,j) entry of x[[st]] is the value of predictor j at discrete person-time observation i of state st
y	a list with one element for each state in the multistate model; each list element is a nonnegative whole number vector or matrix indicating event occurrence at each person-time point; with only one event type, the i^th entry of the vector y[[st]] is 0 for no event occurrence at observation i and 1 for event occurrence (or potentially greater than 1 if that observation represents replicated observations; see entry for n); with more than one event type, the (i,j) entry of the matrix y[[st]] counts the number of events of type j occurring at discrete person-time observation i
priors	a list with one element for each state in the multistate model; each element of priors is a list with one element for each predictor variable (column of x[[st]]) specifying the prior type to use for that predictor; see Details for more information
S	the number of MCMC iterations to perform post-burn-in (post-burn-in iterations are stored in the object returned by the brea_mcmc function)
B	the number of burn-in MCMC iterations to perform (burn-in iterations are not stored)
n	a list with one element for each state in the multistate model; each list element is a vector of positive whole numbers with length equal to the number of person-time observations for the corresponding state; the i^th entry of n[[st]] is the number of replicated observations that observation i represents (defaults to 1 for each observation if NULL is supplied)
K	a list with one element for each state in the multistate model; each list element is a positive whole number vector whose m^th entry is the number of distinct values that the m^th discretized predictor for that state may assume; if NULL is supplied, this vector will be automatically determined from x
store_re	if TRUE, the random effects are stored from each post-burn-in MCMC iteration, and if FALSE they are not stored

Details

The data provided to the brea_mcmc function is specified at the person-time level: for each state st, there is one row in x[[st]] and y[[st]] for each discrete time point each person or thing was at risk for event occurrence in that state. All predictors in x must be encoded as factors or their corresponding integer codes. The underlying type of predictor is specified in the priors argument, where priors[[st]] is a list with one element for each predictor variable which specifies both the type of that predictor and the prior distribution to use. The allowed predictor types are:

- "cat" for categorical variables. The first element of the prior specification list is the string "cat", and the second element is a positive conditional prior standard deviation parameter.
- "gmrf" for underlying continuous predictors; continuous predictors should be discretized with cut() before being included into x; Gaussian Markov random field (GMRF) priors are then used to smooth the effects of adjacent categories of the discretized continuous predictor. The first element of the prior specification list is the string "gmrf", the second element is a prior degrees of freedom for the scaled inverse chi-squared prior for the random walk increment variance, and the third element is a prior scale for the scaled inverse chi-squared.
- "re" for variables (like subject id numbers) that represent random effects. The first element of the prior specification list is the string "re", the second element is a prior degrees of freedom for an inverse Wishart prior for the random effects covariance, and the third element is a prior scale matrix for the random effects covariance.
- "zero" for predictors that are not used in the current MCMC run. This is provided as a convenient way to exclude predictors from certain runs. The first and only element of the prior specification list is the string "zero".

Value

A list providing the values from each of the S post-burn-in MCMC iterations for the intercepts, the other linear predictor parameters, standard deviations of the GMRF random walk increments, and covariances of the random effects:

b_0_s	a list with one element for each state st where b_0_s[[st]] is an R[st] by S matrix (where R[st] is the number of competing risks) whose (r,s) entry is the intercept for risk r at MCMC iteration s
b_m_s	a list with one element for each state st where b_m_s[[st]] is a list whose m^th element is the R[st] by K[[st]][m] by S array whose (r,k,s) entry is the value of the linear predictor contribution of value k for predictor m on risk r at MCMC iteration s
s_m_s	a list with one element for each state st where s_m_s[[st]] is a list whose m^th element is, in the case of a GMRF prior, the R[st] by S matrix whose (r,s) entry is the random walk standard deviation for predictor m and competing risk r at MCMC iteration s, or, in the case of a random effects priors, the sum(R) by sum(R) by S array whose (,,s) entry is the random effects covariance matrix for predictor m at MCMC iteration s
b_m_a	a list with one element for each state st where b_m_a[[st]] is a list whose m^th element is the length K[[st]][m] vector giving the number of accepted Metropolis proposals for each value of predictor m across all S post-burn-in MCMC iterations

Author(s)

Adam King

Examples

```
# leukemia remission times data from Table 1 of "Regression Models and
# Life-Tables" (Cox, 1972)

# times of event occurrence or right censoring:
time <- c(6,6,6,6,7,9,10,10,11,13,16,17,19,20,22,23,25,32,32,34,35,
          1,1,2,2,3,4,4,5,5,8,8,8,8,11,11,12,12,15,17,22,23)

# event/censoring indicator (1 for event occurrence, 0 for right censoring):
event <- c(0,1,1,1,1,0,0,1,0,1,1,0,0,0,1,1,0,0,0,0,0,
           1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)

# treatment group indicator (1 for treatment, 2 for control):
treat <- c(rep(1,21),rep(2,21))

# total number of person-time observations among all subjects:
N <- sum(time)

# create and fill in person-time predictor and event variables:
x <- matrix(0,N,2)  # first column is time, second column is treatment
y <- numeric(N)
next_row <- 1  # next available row in the person-time variables
for (i in seq_along(time)) {
  rows <- next_row:(next_row + time[i] - 1)  # observations for subject i
  x[rows,1] <- seq_len(time[i])  # time is integers 1,2,...,time[i]
  x[rows,2] <- rep(treat[i],time[i])  # treatment group is constant
  y[rows] <- c(rep(0,time[i] - 1),event[i])  # outcome is 0's until event
  next_row <- next_row + time[i]  # increment the next available row pointer
}

# group the time-at-risk variable into 3-week intervals:
x[,1] <- cut(x[,1],seq(0,36,3),labels=FALSE)

# use GMRF prior for time, and categorical prior for treatment group:
priors <- list(list("gmrf",3,.01),list("cat",4))

# run the default of 100 burn-in iterations followed by 1,000 stored iterations:
fit <- brea_mcmc_ms(list(x),list(y),list(priors))

# examine the structure of the returned posterior samples and acceptance counts:
str(fit)

# posterior samples of the treatment and control group parameters:
b_treatment <- fit$b_m_s[[1]][[2]][1,1,]
b_control <- fit$b_m_s[[1]][[2]][1,2,]

# posterior sample of treatment effect on linear predictor scale:
d <- b_control - b_treatment
```

```
# posterior mean, median, and sd of treatment effect on linear predictor scale:
mean(d); median(d); sd(d)

# posterior mean and median hazard ratios:
mean(exp(d)); median(exp(d))
```

Index

∗ **regression**
brea_mcmc, 1
brea_mcmc_ms, 4
∗ **survival**
brea_mcmc, 1
brea_mcmc_ms, 4

brea_mcmc, 1
brea_mcmc_ms, 4