

Package ‘DDPGPSurv’

June 24, 2018

Type Package

Title DDP-GP Survival Analysis

Version 1.0

Date 2018-06-20

Author William Hua <whua4@jhu.edu>, Yanxun Xu <yanxun.xu@jhu.edu>

Maintainer William Hua <whua4@jhu.edu>

Description A nonparametric Bayesian approach to survival analysis. The functions perform inference via MCMC simulations from the posterior distributions for a Dependent Dirichlet Process-Gaussian Process prior. To maximize computational efficiency, some of the computations are performed in 'Rcpp'.

License CC0

Imports Rcpp (>= 0.12.11), MASS, survival, mc2d, mvnfast

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.0.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-06-24 12:50:51 UTC

R topics documented:

DDPGPSurv-package	2
DDPGP_Dens	4
DDPGP_Haz	5
DDPGP_meansurvival	7
DDPGP_Surv	8
mcmc_DDPGP	10
simulate_data	12

Index	13
--------------	-----------

Description

A nonparametric Bayesian approach to survival analysis. The functions perform inference via MCMC simulations from the posterior distributions for a Dependent Dirichlet Process-Gaussian Process prior. To maximize computational efficiency, some of the computations are performed in 'Rcpp'.

Details

The Dependent Dirichlet Process-Gaussian Process model is summarized below. The Dependent Dirichlet Process has a Gaussian Process as its base measure.

$$y_i | \mu_h(x_i), \sigma^2 = \sum_{h=1}^{\infty} w_h N(\mu_h(x_i), \sigma^2)$$

$$\mu_h(x) \sim GP(x\beta_h, C(\cdot, \cdot))$$

$$C(x_i, x_j) = \sigma_0^2 \exp\{-\|\frac{x_i - x_j}{l^2}\|^2\} + I_{\{i=j\}} J^2$$

Here, the mean of the Gaussian Process is modeled using linear regression with regression coefficients β_h and J is a small diagonal perturbation to the squared exponential covariance function (set to 0.01).

To complete the model specification, independent hyperpriors are assumed:

$$\beta_h \sim N(\beta_0, \Sigma_0)$$

$$\sigma^{-2} \sim \text{Gamma}(\delta_1, \delta_2)$$

We use a hyperparameter M to control the weights (w_h) in the model:

$$M \sim \text{Gamma}(\lambda_1, \lambda_2)$$

Author(s)

William Hua <whua4@jhu.edu>, Yanxun Xu <yanxun.xu@jhu.edu>

Maintainer: William Hua <whua4@jhu.edu>

Examples

```
#####
#Source dependent packages
library(MASS)
library(mc2d)
library(mvnfast)
library(survival)
#Simulate Data using built-in data generation
```

```

##
seed=1
set.seed(seed)
Npat=5
data <- simulate_data(Npat)

#####
#Run MCMC
#####
#Inputs for mcmc
response <- log(data$OS)
covariate <- cbind(scale(data$Age),data$AUC,data$CR)
censor_status <- data$death
mcmc_settings<-NULL
mcmc_settings$nskip<-10
mcmc_settings$nburn<-50
mcmc_settings$ndisplay<-100
mcmc_settings$nsave<-20
mcmc_settings$sigma_jump<-c(0,4,2.5,4,2.6)
#####
#Run MCMC function
mcmc_Gibbs<-mcmc_DDPGP(response,covariate,censor_status,mcmc_settings)

#####
#Plotting Density/Survival/Hazard Estimation
#####
#Parameters in Plotting Estimation for Functions
range=seq(2,8,1)
example_AUC <- 5
example_CR <- 1
example_Age <- 0
new_pat<-cbind(example_Age,example_AUC,example_CR)
if_plot=1
#####
#Plot DDP-GP Density Estimation
a=DDPGP_Dens(mcmc_Gibbs,new_pat,range, if_plot)

#Plot DDP-GP Survival Estimation
b=DDPGP_Surv(mcmc_Gibbs,new_pat,range, if_plot)

#Plot DDP-GP Hazard Estimation
c=DDPGP_Haz(mcmc_Gibbs,new_pat,range, if_plot)
#####
#Plotting Mean Survival Estimation
#####
#Parameters in Plotting Mean Survival Estimation
range_AUC <- seq(2.6, 7, 0.1)
new_pat_1<-cbind(example_Age,range_AUC,example_CR)
if_plot=1
DDPGP_mean<-DDPGP_meansurvival(mcmc_Gibbs,new_pat_1,if_plot,cov_col=2)

```

DDPGP_Dens

*Posterior density function estimation using MCMC results***Description**

Posterior density function estimation using MCMC results

Usage

```
DDPGP_Dens(mcmc, new_pat, range, if_plot = 0, quantiles = c(0.025, 0.975),
           color = "green")
```

Arguments

mcmc	A list-The MCMC output from mcmc_Gibbs or FiniteDP
new_pat	A matrix consisting of the covariates of new sample points. Each row represents one sample point. It can also be vector consisting of the covariates of a new sample point.
range	A vector consisting of the values to be evaluated in the plot for density estimation.
if_plot	A logical variable indicating whether the density estimation should be plotted. 1 means estimation should be plotted; 0 means otherwise. Default value is 0, i.e. not plotted.
quantiles	A two-dimensional vector determining the quantiles for the confidence bounds to be plotted. The first value is the lower quantile and the second value is the upper quantile. Default is (0.025,0.975)
color	A string determining the color of the plot.

Value

A list composed of the estimation for the density across the given range. The Density_estimation is the mean across all iterations, and the Density_lower_quant and Density_upper_quant are the density estimations for the specified quantiles. Each row represents the estimation for each new sample point.

Examples

```
#####
#Source dependent packages
library(MASS)
library(mc2d)
library(mvnfast)
library(survival)
#Simulate Data using built-in data generation
##
seed=1
set.seed(seed)
```


new_pat	A matrix consisting of the covariates of new sample points. Each row represents one sample point. It can also be vector consisting of the covariates of a new sample point.
range	A vector consisting of the values to be evaluated in the plot for hazard estimation.
if_plot	A logical variable indicating whether the hazard estimation should be plotted. 1 means estimation should be plotted; 0 means otherwise. Default value is 0, i.e. not plotted.
quantiles	A two-dimensional vector determining the quantiles for the confidence bounds to be plotted. The first value is the lower quantile and the second value is the upper quantile. Default is (0.025,0.975)
color	A string determining the color of the plot.

Value

A list composed of the estimation for the hazard across the given range. The Hazard_estimation is the mean across all iterations, and the Hazard_lower_quant and Hazard_upper_quant are the hazard function estimations for the specified quantiles.

Examples

```
#####
#Source dependent packages
library(MASS)
library(mc2d)
library(mvnfast)
library(survival)
#Simulate Data using built-in data generation
##
seed=1
set.seed(seed)
Npat=5
data <- simulate_data(Npat)

#####
#Run MCMC
#####
#Inputs for mcmc
response <- log(data$OS)
covariate <- cbind(scale(data$Age),data$AUC,data$CR)
censor_status <- data$death
mcmc_settings<-NULL
mcmc_settings$nskip<-10
mcmc_settings$nburn<-50
mcmc_settings$ndisplay<-100
mcmc_settings$nsave<-20
mcmc_settings$sigma_jump<-c(0,4,2.5,4,2.6)
#####
#Run MCMC function
mcmc_Gibbs<-mcmc_DDPGP(response,covariate,censor_status,mcmc_settings)
```

```
#####
#Plotting Hazard Estimation
#####
#Parameters in Plotting Estimation for Functions
range=seq(2,8,1)
example_AUC <- 5
example_CR <- 1
example_Age <- 1
new_pat<-cbind(example_Age,example_AUC,example_CR)
if_plot=1
#####
#Plot DDP-GP Hazard Estimation
c=DDPGP_Haz(mcmc_Gibbs,new_pat,range, if_plot)
```

DDPGP_meansurvival *Posterior mean survival estimation using Gibbs MCMC Outputs*

Description

Posterior mean survival estimation using Gibbs MCMC Outputs

Usage

```
DDPGP_meansurvival(mcmc, new_pat, if_plot = 0, quantiles = c(0.025, 0.975),
  cov_col = 1)
```

Arguments

mcmc	A list-The MCMC output from mcmc_Gibbs.
new_pat	A matrix consisting of the covariates of new sample points. Each row represents one sample point. It can also be vector consisting of the covariates of a new sample point.
if_plot	A logical variable indicating whether the mean survival estimation should be plotted. 1 means estimation should be plotted; 0 means otherwise. Default value is 0, i.e. not plotted.
quantiles	A two-dimensional vector determining the quantiles for the confidence bounds to be plotted. The first value is the lower quantile and the second value is the upper quantile. Default is (0.025,0.975)
cov_col	An integer defining which covariate column in new_pat to plot on the x-axis for the mean survival figure.

Value

A list composed of the estimation for the mean survival across the values for the covariate specified. The mean_survival is the average of the mean survival calculation for all iterations, and lower_quant and upper_quant is that for the quantiles specified. optimal is the optimal covariate value (which is specified in cov_col). meansurvival_all is the mean survival for all iterations.

Examples

```
#####
#Source dependent packages
library(MASS)
library(mc2d)
library(mvnfast)
library(survival)
#Simulate Data using built-in data generation
##
seed=1
set.seed(seed)
Npat=5
data <- simulate_data(Npat)

#####
#Run MCMC
#####
#Inputs for mcmc
response <- log(data$OS)
covariate <- cbind(scale(data$Age),data$AUC,data$CR)
censor_status <- data$death
mcmc_settings<-NULL
mcmc_settings$nskip<-10
mcmc_settings$nburn<-50
mcmc_settings$ndisplay<-100
mcmc_settings$nsave<-20
mcmc_settings$sigma_jump<-c(0,4,2.5,4,2.6)
#####
#Run MCMC function
mcmc_Gibbs<-mcmc_DDPGP(response,covariate,censor_status,mcmc_settings)
#####
#Plotting Mean Survival Estimation
#####
#Parameters in Plotting Mean Survival Estimation
example_CR <- 1
example_Age <- 1
range_AUC <- seq(2.6, 7, 0.1)
new_pat_1<-cbind(example_Age,range_AUC,example_CR)
if_plot=1
DPGP_mean<-DDPGP_meansurvival(mcmc_Gibbs,new_pat_1,if_plot,cov_col=2)
```

Description

Posterior survival function estimation using Gibbs MCMC results

Usage

```
DDPGP_Surv(mcmc, new_pat, range, if_plot = 0, quantiles = c(0.025, 0.975),
           color = "green")
```

Arguments

mcmc	A list-The MCMC output from mcmc_Gibbs
new_pat	A matrix consisting of the covariates of new sample points. Each row represents one sample point. It can also be vector consisting of the covariates of a new sample point.
range	A vector consisting of the values to be evaluated in the plot for survival estimation.
if_plot	A logical variable indicating whether the survival estimation should be plotted. 1 means estimation should be plotted; 0 means otherwise. Default value is 0, i.e. not plotted.
quantiles	A two-dimensional vector determining the quantiles for the confidence bounds to be plotted. The first value is the lower quantile and the second value is the upper quantile. Default is (0.025,0.975)
color	A string determining the color of the plot.

Value

A list composed of the estimation for the survival across the given range. The Survival_estimation is the mean across all iterations, and the Survival_lower_quant and Survival_upper_quant are the survival function estimations for the specified quantiles.

Examples

```
#####
#Source dependent packages
library(MASS)
library(mc2d)
library(mvtnfast)
library(survival)
#Simulate Data using built-in data generation
##
seed=1
set.seed(seed)
Npat=5
data <- simulate_data(Npat)

#####
#Run MCMC
#####
#Inputs for mcmc
response <- log(data$OS)
covariate <- cbind(scale(data$Age),data$AUC,data$CR)
censor_status <- data$death
mcmc_settings<-NULL
```

```

mcmc_settings$nskip<-10
mcmc_settings$nburn<-50
mcmc_settings$ndisplay<-100
mcmc_settings$nsave<-20
mcmc_settings$sigma_jump<-c(0,4,2.5,4,2.6)
#####
#Run MCMC function
mcmc_Gibbs<-mcmc_DDPGP(response,covariate,censor_status,mcmc_settings)
#####
#Plotting Survival Estimation
#####
#Parameters in Plotting Estimation for Functions
range=seq(2,8,1)
example_AUC <- 5
example_CR <- 1
example_Age <- 1
new_pat<-cbind(example_Age,example_AUC,example_CR)
if_plot=1
#####
#Plot DDP-GP Survival Estimation
b=DDPGP_Surv(mcmc_Gibbs,new_pat,range, if_plot)

```

mcmc_DDPGP

Gibbs Sampling MCMC function

Description

Gibbs Sampling MCMC function

Usage

```

mcmc_DDPGP(response, covariate, censor_status, mcmc_settings, lambda1 = 1,
  lambda2 = 1, delta1 = 4, delta2 = 3, mu0 = 0, tau2 = 10)

```

Arguments

response	An n-dimensional vector consisting of the verall survival time of the patients in the dataset (n is the number of patients).
covariate	An nxp dimensional matrix consisting of the relevant covariates for each patient (p is the number of covariates)
censor_status	An n-dimensional vector with the censoring status of the response (0 for censored, 1 for uncensored)
mcmc_settings	A list for MCMC setup. nskip is the number of iterations to skip between saved iterations. nburn is the number of total iterations to burn. ndisplay is number of iterations per which the display message will appear. nsave is the total number of iterations to save. The total number of iterations (including those not saved) will be mcmc_settings\$nburn+mcmc_settings\$nskip*mcmc_settings\$nsave. mcmc_settings\$sigma_jump

is a $(p+2)$ -dimensional vector, where p is the number of covariates. These are the jump sizes in the M-H algorithm. The first value in this vector refers to the jump size of the 1's covariate column (which is typically 0). The next p values are the jump sizes of the covariates in the order of the covariate vector. The last value in this vector is the jump size for the variance term (σ_0) in the covariance matrix. See example.

lambda1	A hyper-parameter in prior for the M ($M \sim \text{Gamma}(\text{lambda1}, \text{lambda2})$). Default value is 1.
lambda2	A hyper-parameter in prior for the M ($M \sim \text{Gamma}(\text{lambda1}, \text{lambda2})$). Default value is 1.
delta1	A hyper-parameter in prior for the σ^2 ($1/(\sigma^2) \sim \text{Gamma}(\text{delta1}, \text{delta2})$). Default value is 4.
delta2	A hyper-parameter in prior for the σ^2 ($1/(\sigma^2) \sim \text{Gamma}(\text{delta1}, \text{delta2})$). Default value is 3.
mu0	A hyper-parameter for the mean in the normal prior for the Metropolis Hastings algorithm for the scale parameter (l) in the covariance matrix. Default value is 0.
tau2	A hyper-parameter for the variance in the normal prior for the Metropolis Hastings algorithm for the scale parameter (l) in the covariance matrix. Default value is 10.

Value

A list composed of the mcmc outputs. These outputs can be inputted into the other functions in this package.

Examples

```
#####
#Source dependent packages
library(MASS)
library(mc2d)
library(mvnfast)
library(survival)
#Simulate Data using built-in data generation
##
seed=1
set.seed(seed)
Npat=5
data <- simulate_data(Npat)

#####
#Run MCMC
#####
#Inputs for mcmc
response <- log(data$OS)
covariate <- cbind(scale(data$Age), data$AUC, data$CR)
censor_status <- data$death
mcmc_settings<-NULL
```

```

mcmc_settings$nskip<-10
mcmc_settings$nburn<-50
mcmc_settings$ndisplay<-100
mcmc_settings$nsave<-20
mcmc_settings$sigma_jump<-c(0,4,2.5,4,2.6)
#####
#Run MCMC function
mcmc_Gibbs<-mcmc_DDPGP(response,covariate,censor_status,mcmc_settings)
#####
#Plotting Density/Survival/Hazard Estimation
#####

```

simulate_data

Simulate data

Description

Simulate data

Usage

```
simulate_data(Npat, method = 1, mixture = 1, censor = 0)
```

Arguments

Npat	A positive integer indicating the number of patients to be simulated.
method	A binary value (either 0 or 1). If method is set to 0, the survival times are simulated from a Weibull distribution. If method is set to 1, the survival times are simulated from a Lognormal distribution. Default value is 1.
mixture	A binary value (either 0 or 1). If method is set to 0, the survival times are simulated from a mixture of two distributions. If method is set to 1, the survival times are simulated from single distribution. Default value is 1.
censor	A binary value (either 0 or 1). If censor is set to 1, survival times of patients will be censored (at roughly 25 If censor is set to 0, the survival times of patients will not be censored. Default value is 0.

Value

Returns a list composed of the patients' generated attributes. The covariates are Age, AUC, and CR.

Examples

```

#Simulate Data for 10 patients
Npat<- 10
data<-simulate_data(Npat)

```

Index

DDPGP_Dens, [4](#)
DDPGP_Haz, [5](#)
DDPGP_meansurvival, [7](#)
DDPGP_Surv, [8](#)
DDPGPSurv (DDPGPSurv-package), [2](#)
DDPGPSurv-package, [2](#)

mcmc_DDPGP, [10](#)

simulate_data, [12](#)