Package ‘DPWeibull’

March 22, 2018

Type Package

Title Dirichlet Process Weibull Mixture Model for Survival Data

Version 1.2

Date 2018-03-21

Author Yushu Shi, Michael Martens

Maintainer Yushu Shi <shiyushu2006@gmail.com>

Depends Rcpp (>= 0.12.4), truncdist, DPpackage, matrixStats

Description Use Dirichlet process Weibull mixture model and dependent Dirichlet process Weibull mixture model for survival data with and without competing risks. Dirichlet process Weibull mixture model is used for data without covariates and dependent Dirichlet process model is used for regression data. The package is designed to handle exact/right-censored/ interval-censored observations without competing risks and exact/right-censored observations for data with competing risks. Inside each cluster of Dirichlet process, we assume a multiplicative effect of covariates as in Cox model and Fine and Gray model. In addition, we provide a wrapper for DPdensity() function from the R package 'DPpackage'. This wrapper automatically uses Low Information Omnibus prior and can model one and two dimensional data with Dirichlet mixture of Gaussian distributions.

License GPL (>= 2)

LinkingTo Rcpp

NeedsCompilation yes

RoxygenNote 5.0.1

Repository CRAN

Date/Publication 2018-03-22 08:29:52 UTC

R topics documented:

`continue` *Resume MCMC run*

Description

Self-defined S3 method that resumes MCMC run from an object of class dpm ,ddp, dpmcomp or ddpcomp.

Usage

```
continue(previous,iteration=1000,...)
```

Arguments

`previous`	An object of class dpm, ddp, dpmcomp or ddpcomp
`iteration`	The number of iterations to continue sampling. The default is 1000.
`...`	Arguments to be passed to method

Value

An object of the class of previous.

continue.ddp — *Resume MCMC run*

Description

Resume MCMC run from an object of class ddp.

Usage

```
## S3 method for class 'ddp'
continue(previous,...)
```

Arguments

`previous`	An object of class ddp
`...`	Arguments to be passed to method

Value

An object of class ddp.

continue.ddpcomp — *Resume MCMC run*

Description

Resume MCMC run from an object of class ddpcomp.

Usage

```
## S3 method for class 'ddpcomp'
continue(previous,...)
```

Arguments

`previous`	An object of class ddpcomp
`...`	Arguments to be passed to method

Value

An object of class ddpcomp.

continue.dpm *Resume MCMC run*

Description

Resume MCMC run from an object of class dpm.

Usage

```
## S3 method for class 'dpm'
continue(previous,...)
```

Arguments

previous	An object of class dpm
...	Arguments to be passed to method

Value

An object of class dpm.

continue.dpmcomp *Resume MCMC run*

Description

Resume MCMC run from an object of class dpmcomp.

Usage

```
## S3 method for class 'dpmcomp'
continue(previous,...)
```

Arguments

previous	An object of class dpmcomp
...	Arguments to be passed to method

Value

An object of class dpmcomp.

`dpweib` *Dirichlet process mixture/Dependent Dirichlet process model for survival/competing risks data*

Description

Use Dirichlet process mixture/dependent Dirichlet process Weibull model for survival data with/without competing risks. When regression covariates are present, the model is a dependent Dirichlet process model. For competing risks data we only consider two potential causes of events and the user can combine events of secondary interests. In competing risks regression model, the estimates provided focus on the primary cause (cause 1), and the user can switch the event indicator to get the estimates for the secondary cause.

Usage

```
dpweib(formula,data, high.pct = NULL, predtime = NULL, comp = FALSE,
alpha = 0.05, simultaneous = FALSE, burnin = 5000, iteration = 5000,
alpha00 = 1.354028, alpha0 = 0.03501257, lambda00 = 7.181247,
alphaalpha = 0.2, alphalambda = 0.1, a = 1, b = 1, gamma0 = 1,
gamma1 = 1, thin = 10, betasl = 2.5, addgroup = 2)
```

Arguments

`formula` A formula written in regular $y \sim x_1 + x_2 + \ldots + x_p$ regression format. The y can be a Surv object of the type "right" (this is not suitable for competing risks data), or a two-column matrix with the first column being the event time, second column being the exact observation indicator, (This is not suitable for interval censored data.) or a three-column matrix with the first column for the left end point of the observation time, second column for the right end point of the observation time and the third column for the no-right-censoring indicator. (If the exact observation time is observed, the first column is equal to the second column and the third column is 1. If it is an interval-censored observation, the first column is less than the second column and the third column is 1. If it is a right-censored observation, then the value of the second column is ignored and the third column is 0.) The regression covaraites can be continuous or factors. Since the model is flexible enough, interaction terms are not necessary.

`data` an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which dpweib is called.

`high.pct` The estimated high percentile (95th) percentile of the data-generating distribution of the average population given by the user. If the user does not provide this value, we will look into the data. If there is no censoring, we take the 95th percentile of the observed data. If censoring takes less than 15% of the total observations, we use the maximum of the observed time. If the censoring takes more than 15%, we suggest a scaling parameter by first finding the time t corresponding to the observed survival rate at the end of study from the plot of the median of the components (survmedian) generated by our LIO prior on a 0 to 10 scale, then set the scaling parameter to be the largest observation time multiplied by 10/t.

- `predtime` A vector given by the user to specify the time points where the inferences will be made. If the user does not provide it, we will take 40 time points located evenly from the beginning to the high.pct.
- `comp` A logical value indicating whether this is competing risks data or not. The default is FALSE.
- `alpha` $1-\alpha$ is the probability for constructing credible intervals. The default α is 0.05.
- `simultaneous` A logical value indicating whether to provide simultaneous credible intervals. The default is FALSE.
- `burnin` Number of burnin iterations. The default is 5000.
- `iteration` Number of iterations. The default is 5000.
- `alpha00` Parameter for the base distribution of λ in non-competing risks data model and λ_1, λ_2 in competing risks data model. The default is 1.354028.
- `alpha0` Parameter for the base distribution of λ in non-competing risks data model and λ_1, λ_2 in competing risks data model. The default is 0.03501257.
- `lambda00` Parameter for the base distribution of λ in non-competing risks data model and λ_1, λ_2 in competing risks data model. The default is 7.181247.
- `alphaalpha` Parameter for the base distribution of α in non-competing risks data model and α_1, α_2 in competing risks data model. The default is 0.2.
- `alphalambda` Parameter for the base distribution of α in non-competing risks data model and α_1, α_2 in competing risks data model. THe default value is 0.1.
- `a` Parameter for the gamma prior of the concentration parameter of DP. The default is 1.
- `b` Parameter for the gamma prior of the concentration parameter of DP. The default is 1.
- `gamma0` Parameter for the base distribution of p in competing risks data model. The default value is 1.
- `gamma1` Parameter for the base distribution of p in competing risks data model. The default value is 1.
- `thin` Thinning. The default value is 10.
- `betasl` Parameter for the base distribution of the regression coefficients β in non-competing risks data model and β_1 and β_2 in competing risks data model. The default value is 2.5.
- `addgroup` Number of new parameters proposed for each cluster assignment. The default is 2 (suggested by Neal).

Details

For no regression, no competing risks data, the function dpweib implements dirichlet process Weibull mixture model. The basic form of model is the following.

$$\begin{aligned} y_i|\alpha_i,\lambda_i &\sim Weib(t_i|\alpha_i,\lambda_i), \quad i=1,...,n \\ (\alpha_i,\lambda_i)|G &\sim G, \quad i=1,...,n \\ G &\sim DP(G_0,\nu) \\ G_0 &= Ga(\lambda|\alpha_0,\lambda_0)I_{(f(\lambda),\infty)}(\alpha)Ga(\alpha_\alpha,\lambda_\alpha) \\ \lambda_0 &\sim Ga(\alpha_{00},\lambda_{00}) \\ \nu &\sim Ga(a,b) \end{aligned}$$

where$f(\lambda) = max(0, \log\{\log(20)/\lambda\}/\log(25))$.

For regression data without competing risks, the method is a mixture of Cox model.

$$\begin{aligned} y_i|\alpha_i,\lambda_i,\boldsymbol{\beta_i},\mathbf{Z_i} &\sim Weib(y_i|\alpha_i,\lambda_i\exp(\mathbf{Z_i^T}\boldsymbol{\beta_i})), \quad i=1,...,n \\ (\alpha_i,\lambda_i,\boldsymbol{\beta_i})|G &\sim G, \quad i=1,...,n \\ G &\sim DP(G_0,\nu) \\ G_0 &= Ga(\lambda|\alpha_0,\lambda_0)I_{(f(\lambda),u)}(\alpha)Ga(\alpha_\alpha,\lambda_\alpha)q(\boldsymbol{\beta}) \\ \lambda_0 &\sim Ga(\alpha_{00},\lambda_{00}) \\ \nu &\sim Ga(a,b) \end{aligned}$$

The density function corresponding to this Weibull notation is $p(y_i|\alpha_i,\lambda_i) = \lambda_i\alpha_i y_i^{\alpha_i-1}e^{-\lambda_i y_i^{\alpha_i}}, \quad y_i > 0, \quad \alpha_i > 0, \quad \lambda_i > 0$. $[x] = Ga(\alpha,\lambda)$ denotes that the density function of x is $\frac{\lambda^\alpha}{\Gamma(\alpha)}x^{\alpha-1}e^{-\lambda x}$, $\alpha > 0$, $\lambda > 0$, $x > 0$. $q(\beta)$ is the base distribution for regression coefficients.The details of the choice of base distribution is described in our coming paper.

In competing risks data, the likelihood for each individual can be written as

$$L = \{f_1(t_i)\}^{I(c_i=1)}\{f_2(t_i)\}^{I(c_i=2)}\{1-F_1(t_i)-F_2(t_i)\}^{I(c_i=0)},$$

where $f_1(\cdot)$ and $f_2(\cdot)$ are the subdistribution density functions for cause 1 and 2 and survival function for the ith observation can be expressed as $1-F_1(t_i)-F_2(t_i)$. In order to model it, we introduce a parameter p, which is the cumulative incidence function of primary cause at ∞, $p = F_1(\infty)$. The likelihood can be written as

$$L = \{pd_1(t_i)\}^{I(c_i=1)}\{(1-p)d_2(t_i)\}^{I(c_i=2)}\{1-pD_1(t_i)-(1-p)D_2(t_i)\}^{I(c_i=0)}.$$

Here the D_1, D_2, d_1, d_2 are the normalized baseline cumulative incidence functions and cause-specific density functions and are modeled with Weibull mixtures as above, while p is the normalizing parameter for the baseline distribution. When regression covariates are present in a competing risks data, we modify the above likelihood with respect to the value of covaraites, such that

$$F_1(t|\mathbf{Z},\boldsymbol{\beta_1},p) = 1-(1-pD_{01}(t))^{\exp(\mathbf{Z^T}\boldsymbol{\beta_1})}.$$

The cause-specific density function for cause 1 is

$$f_1(t|\mathbf{Z},\boldsymbol{\beta_1},p) = \exp(\mathbf{Z^T}\boldsymbol{\beta_1})[1-pD_{01}(t)]^{\exp(\mathbf{Z^T}\boldsymbol{\beta_1})-1}pd_{01}(t).$$

The model for the secondary cause is defined as

$$F_2(t|\mathbf{Z}, \boldsymbol{\beta_1}, \boldsymbol{\beta_2}, p) = (1-p)^{\exp(\mathbf{Z^T}\boldsymbol{\beta_1})}(1-(1-D_{02}(t))^{\exp(\mathbf{Z^T}\boldsymbol{\beta_2})}),$$

which leads to the cause-specific subdensity function for cause 2 as

$$f_2(t|\mathbf{Z}, \boldsymbol{\beta_2}, p) = (1-p)^{\exp(\mathbf{Z^T}\boldsymbol{\beta_1})}(1-D_{02}(t))^{\exp(\mathbf{Z^T}\boldsymbol{\beta_2})-1}\exp(\mathbf{Z^T}\boldsymbol{\beta_2})d_{02}(t).$$

Value

This function can generate 4 different kinds of output based on the data set given. They all share,

`c`	a vector, the cluster assignment in the last iteration, useful for the resumption of MCMC iteration
`nm`	a vector, the number of observations in each cluster from the last iteration, useful for the resumption of MCMC iteration
`emptybasket`	only useful for the resumption of MCMC iteration
`allbaskets`	only useful for the resumption of MCMC iteration
`ngrp`	a vector, the number of clusters in each iteration, useful for the resumption of MCMC iteration
`predtime`	the time points where the inferences are made
`high.pct`	the scaling parameter of observations used in the model
`usertime`	a logic value, whether user provides time for estimation or not

$1-\alpha$ is the probability for constructing credible intervals.

`simultaneous`	Whether give simultaneous credible intervals.

For non-competing risks data, dpweib can generate two classes of output, dpm and ddp, for data with and without covariates separately. They both have

`alpharec`	a matrix, saved samples of αs, the rows correspond to the iterations saved, the columns correspond to the observations
`lambdarec`	a matrix, saved samples of λs, the rows correspond to the iterations saved, the columns correspond to the observations
`lambda0rec`	a matrix, saved samples of λ_0s, the rows correspond to the iterations saved, the columns correspond to the observations
`lambdascaled`	a matrix, saved samples of λs under 0 to 10 scale, the rows correspond to the iterations saved, the columns correspond to the observations, only useful for the resumption of MCMC iteration
`tl`	the left end point
`tr`	the right end point
`pi`	right censoring indicator
`delta`	exact observation indicator

For dpm output, it has

`S`	a matrix, the estimated survival function for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations
`Spred`	a vector, the estimated survival function at specified time points
`Spredu`	a vector, the estimated pointwise upper credible interval for survival function at specified time points
`Spredl`	a vector, the estimated pointwise lower credible interval for survival function at specified time points
`d`	a matrix, the estimated density function for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations
`dpred`	a vector, the estimated density function at specified time points
`dpredu`	a vector, the estimated pointwise upper credible interval for density function at specified time points
`dpredl`	a vector, the estimated pointwise lower credible interval for density function at specified time points
`h`	a matrix, the estimated hazard function for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations
`hpred`	a vector, the estimated hazard function at specified time points
`hpredu`	a vector, the estimated pointwise upper credible interval for hazard function at specified time points
`hpredl`	a vector, the estimated pointwise lower credible interval for hazard function at specified time points

When simultaneous is specified TRUE, the function also provides

`Sbandu`	a vector, the estimated simultaneous upper credible interval for survival function at specified time points
`Sbandl`	a vector, the estimated simultaneous lower credible interval for survival function at specified time points
`dbandu`	a vector, the estimated simultaneous upper credible interval for density function at specified time points
`dbandl`	a vector, the estimated simultaneous lower credible interval for density function at specified time points
`hbandu`	a vector, the estimated simultaneous upper credible interval for hazard function at specified time points
`hbandl`	a vector, the estimated simultaneous lower credible interval for hazard function at specified time points

For ddp output, it also has

`betarec`	a matrix, saved samples of βs, which is consist of horizontal-merged blocks. One block corresponds to one observation. Inside each block, the rows correspond to the iterations saved, the columns correspond to the covariates.
`x`	the covariate matrix
`xmean`	a vector, the mean for each covariate(including created binary dummy covariates)

xsd	a vector, the standized deviation for each covariate, if the covariate is binary, then it is set to be 0.5.(including created binary dummy covariates)
xscale	The matrix used to scale log hazard ratio
loghr	a matrix, the estimated log hazard ratio for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations
loghr.est	a vector, the estimated log hazard ratio at specified time points
loghru	a vector, the estimated pointwise upper credible interval for log hazard ratio at specified time points
loghrl	a vector, the estimated pointwise lower credible interval for log hazard ratio at specified time points
indicator	a vector, whether a covariate is binary
covnames	a vector, the names of covariates

When simultaneous is specified TRUE, the function also provides

loghrbandu	a vector, the estimated simultaneous upper credible interval for log hazard ratio at specified time points
loghrbandl	a vector, the estimated simultaneous lower credible interval for log hazard ratio at specified time points

For competing risks data, dpweib can generate two classes of output, dpmcomp and ddpcomp, for data with and without covariate separately. They both have

alpharec1	a matrix, saved samples of α_1s, the rows correspond to the iterations saved, the columns correspond to the observations
lambdarec1	a matrix, saved samples of λ_1s, the rows correspond to the iterations saved, the columns correspond to the observations
lambda0rec1	a matrix, saved samples of λ_{01}s, the rows correspond to the iterations saved, the columns correspond to the observations
lambdascaled1	a matrix, saved samples of λ_1s under 0 to 10 scale, the rows correspond to the iterations saved, the columns correspond to the observations, only useful for the resumption of MCMC iteration
alpharec2	a matrix, saved samples of α_2s, the rows correspond to the iterations saved, the columns correspond to the observations
lambdarec2	a matrix, saved samples of λ_2s, the rows correspond to the iterations saved, the columns correspond to the observations
lambda0rec2	a matrix, saved samples of λ_{02}s, the rows correspond to the iterations saved, the columns correspond to the observations
lambdascaled2	a matrix, saved samples of λ_2s under 0 to 10 scale, the rows correspond to the iterations saved, the columns correspond to the observations, only useful for the resumption of MCMC iteration
prec	a matrix, saved samples of p, the rows correspond to the iterations saved, the columns correspond to the observations
t	the observed time

`event` the event indicator

For dpmcomp output, it has

- `CIF1` a matrix, the estimated cumulative incidence function for cause 1 for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations
- `CIF1.est` a vector, the estimated cumulative incidence function of cause 1 at specified time points
- `CIF1u` a vector, the estimated pointwise upper credible interval for cumulative incidence function of cause 1 at specified time points
- `CIF1l` a vector, the estimated pointwise lower credible interval for cumulative incidence function of cause 1 at specified time points
- `d1` a matrix, the estimated subdistribution density function for cause 1 for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations
- `d1.est` a vector, the estimated subdistribution density function of cause 1 at specified time points
- `d1u` a vector, the estimated pointwise upper credible interval for subdistribution density function of cause 1 at specified time points
- `d1l` a vector, the estimated pointwise lower credible interval for subdistribution density function of cause 1 at specified time points
- `h1` a matrix, the estimated subdistribution hazard function for cause 1 at specified time points, the columns correspond to time points, the rows correspond to saved iterations
- `h1.est` a vector, the estimated subdistribution hazard function of cause 1 at specified time points
- `h1u` a vector, the estimated pointwise upper credible interval for subdistribution hazard function of cause 1 at specified time points
- `h1l` a vector, the estimated pointwise lower credible interval for subdistribution hazard function of cause 1 at specified time points
- `CIF2` a matrix, the estimated cumulative incidence function for cause 2 for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations
- `CIF2.est` a vector, the estimated cumulative incidence function of cause 2 at specified time points
- `CIF2u` a vector, the estimated pointwise upper credible interval for cumulative incidence function of cause 2 at specified time points
- `CIF2l` a vector, the estimated pointwise lower credible interval for cumulative incidence function of cause 2 at specified time points
- `d2` a matrix, the estimated subdistribution density function for cause 2 for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations

`d2.est` a vector, the estimated subdistribution density function of cause 2 at specified time points

`d2u` a vector, the estimated pointwise upper credible interval for subdistribution density function of cause 2 at specified time points

`d2l` a vector, the estimated pointwise lower credible interval for subdistribution density function of cause 2 at specified time points

`h2` a matrix, the estimated subdistribution hazard function for cause 2 for each saved iteration, the columns correspond to time points, the rows correspond to saved iterations

`h2.est` a vector, the estimated subdistribution hazard function of cause 2 at specified time points

`h2u` a vector, the estimated pointwise upper credible interval for subdistribution hazard function of cause 2 at specified time points

`h2l` a vector, the estimated pointwise lower credible interval for subdistribution hazard function of cause 2 at specified time points

When simultaneous is specified TRUE, the function also provides

`CIF1bandu` a vector, the estimated simultaneous upper credible interval for cumulative incidence function of cause 1 at specified time points

`CIF1bandl` a vector, the estimated simultaneous lower credible interval for cumulative incidence function of cause 1 at specified time points

`d1bandu` a vector, the estimated simultaneous upper credible interval for subdistribution density function of cause 1 at specified time points

`d1bandl` a vector, the estimated simultaneous lower credible interval for subdistribution density function of cause 1 at specified time points

`h1bandu` a vector, the estimated simultaneous upper credible interval for subdistribution hazard function of cause 1 at specified time points

`h1bandl` a vector, the estimated simultaneous lower credible interval for subdistribution hazard function of cause 1 at specified time points

`CIF2bandu` a vector, the estimated simultaneous upper credible interval for cumulative incidence function of cause 2 at specified time points

`CIF2bandl` a vector, the estimated simultaneous lower credible interval for cumulative incidence function of cause 2 at specified time points

`d2bandu` a vector, the estimated simultaneous upper credible interval for subdistribution density function of cause 2 at specified time points

`d2bandl` a vector, the estimated simultaneous lower credible interval for subdistribution density function of cause 2 at specified time points

`h2bandu` a vector, the estimated simultaneous upper credible interval for subdistribution hazard function of cause 2 at specified time points

`h2bandl` a vector, the estimated simultaneous lower credible interval for subdistribution hazard function of cause 2 at specified time points

For ddpcomp output, it also has

betarec1	a matrix, saved samples of β_1s, which is consist of horizontal-merged blocks. One block corresponds to one observation. Inside each block, the rows correspond to the iterations saved, the columns correspond to the covariates.
betarec2	a matrix, saved samples of β_2s, which is consist of horizontal-merged blocks. One block corresponds to one observation. Inside each block, the rows correspond to the iterations saved, the columns correspond to the covariates.
xmean	a vector, the mean for each covariate(including created dummy covariates)
xsd	a vector, the standized deviation for each covariate, if the covariate is binary, then it is set to be 0.5(including created dummy covariates).
x	the covariate matrix
xscale	The matrix used to scale log hazard ratio
covnames	a vector, the names of covariates
loghr.est	the estimated log subdistribution hazard ratio at specified time points for cause 1
loghru	the estimated pointwise upper credible interval for log subdistribution hazard ratio at specified time points for cause 1
loghrl	the estimated pointwise lower credible interval for log subdistribution hazard ratio at specified time points for cause 1
indicator	a vector, whether a covariate is binary

When simultaneous is specified TRUE, the function also provides

loghrbandu	a vector, the estimated simultaneous upper credible interval for log subdistribution hazard ratio at specified time points
loghrbandl	a vector, the estimated simultaneous lower credible interval for log subdistribution hazard ratio at specified time points

Source

Gilks,W.R. and Best,N.G. and Tan,K.K.C. (1995) Adaptive rejection Metropolis sampling within Gibbs sampling, *Applied Statistics*, 455-472

Neal,R.M (2000) Markov chain sampling methods for Dirichlet process mixture models,*Journal of computational and graphical statistics*, **9**, Num 2, 249-265

Kottas,A. (2006) Nonparametric Bayesian survival analysis using mixtures of Weibull distributions, *Journal of Statistical Planning and Inference*, **136**, Num 3, 578-596

Shi,Y. and Martens,M. and Banerjee,A. and Laud,P. (2017) Low Information Omnibus Priors for Dirichlet Process Mixture Models(Manuscript)

Shi,Y. and Laud,P. and Neuner,J (2017) A Dependent Dirichlet Process Model for Time-to-event Data (Manuscript)

Shi,Y. and Laud,P. and Neuner,J (2017) A Dependent Dirichlet Process Model for Competing Risks Data with an Application to a Breast Cancer Study (Manuscript)

Examples

```
## Not run:
library("DPWeibull")
#example for no regression data without competing risks
p<-0.8
mu1<-0
mu2<-1.2
sigma1<-sqrt(0.25)
sigma2<-sqrt(0.02)
npts<-200
index<-rbinom(npts,1,p)
x.orig<-ifelse(index,rlnorm(npts,mu1,sigma1),rlnorm(npts,mu2,sigma2))
tl<-x.orig
tr<-x.orig
event<-rep(1,npts)
y<-cbind(tl,tr,event)
result<-dpweib(y~1)
summary(result)
plot(result)

##################################################################

# example for regression data without competing risks
library("DPWeibull")
rweib<-function(n,alpha,lambda)
{
(-log(runif(n))/lambda)^(1/alpha)
}
npts<-400
x1<-rnorm(npts)
x2<-rnorm(npts)
x<-cbind(x1,x2)
lambdagen<-exp(0.2*x1+0.2*x2)
x.orig<-1:npts
for(i in 1:npts){
x.orig[i]<-rweib(1,3,lambdagen[i])
}
tl<-x.orig
tr<-x.orig
event<-rep(1,npts)
xpred<-matrix(c(0,0),ncol=2)
time<-(1:40)/20
y<-cbind(tl,tr,event)
data<-data.frame(y=y,x1=x1,x2=x2)
result<-dpweib(y~x1+x2, data, quantile(x.orig,0.95),predtime=time, comp=FALSE)
plot(result,,simultaneous=TRUE)
summary(result)
predresult<-predict(result,xpred)
plot(predresult)
summary(predresult)

#######################################################################
```

```
#competing risks model without covariates
library("DPWeibull")
npts<-200
p<-0.8
u1<-runif(npts)
c<-ifelse(u1<p,1,2)
r1<-1
r2<-1
a<-0
b<-2.2
u2<-runif(npts)
t<-ifelse(c==1,-log(1-u2)/r1,-log(1-u2)/r2)
u3<-runif(npts,a,b)
c<-ifelse(u3<t,0,c)
t<-ifelse(c==0,u3,t)
times<-(1:50)/50*2.5
y<-cbind(t,c)
result<-dpweib(y~1,predtime=times,burnin=2000,iteration=2000)
summary(result)
plot(result)
result<-continue(result,simultaneous=TRUE,burnin=2000,iteration=2000)
summary(result)
plot(result)

############################################################################

#competing risks data with factor covariates
library("DPWeibull")
npts<-400
x1<-rbinom(npts,1,0.5)
x2<-rbinom(npts,1,0.5)
x<-cbind(x1,x2)
beta1<-c(-1,1)
beta2<-c(2,-2)
alphagen1<-2
alphagen2<-0.7
lambdagen1<-exp(-6)
lambdagen2<-exp(-2)
inverseweib<-function(u,alpha,lambda){
  (-log(1-u)/lambda)^(1/alpha)
}
zbeta1<-x %*% beta1
zbeta2<-x %*% beta2
p<-0.8
p1<-1-(1-p)^exp(zbeta1)
u2<-runif(npts)
G1<-1/p*(1-(1-u2*p1)^exp(-zbeta1))
G2<-1-(1-u2)^exp(-zbeta2)
t1<-inverseweib(G1,alphagen1,lambdagen1)
t2<-inverseweib(G2,alphagen2,lambdagen2)
u1<-runif(npts)
c<-ifelse(u1<p1,1,2)
```

```
t<-ifelse(c==1,t1,t2)
cens<-runif(npts,0,200)
event<-ifelse(t<cens,c,0)
time<-ifelse(c==0,cens,t)
x1<-ifelse(x1==1,"good","bad")
x2<-ifelse(x2==1,"good","bad")
x2pred<-factor(rep(c("good","bad"),times=2))
x1pred<-factor(c("good","good","bad","bad"))
xpred<-cbind(x1pred,x2pred)
y<-cbind(time,event)
result<-dpweib(y~factor(x1)+factor(x2),predtime=(1:50)*2)
summary(result)
plot(result)
newresult<-predict(result,xpred)
summary(newresult)
plot(newresult)

## End(Not run)
```

`normwrapper` *A wrapper for the DPdensity function using LIO prior*

Description

A wrapper for the DPdensity function from the R package DPpackage. This wrapper automatically uses the Low Information Omnibus prior in generating a posterior density sample for a Dirichlet mixture of Gaussians distributions.

Usage

```
normwrapper(y,y50=NULL,y95=NULL,mcmc=list(nburn = 1000,
nsave = 1000, nskip = 10, ndisplay = 100),ngrid=1000,grid=NULL)
```

Arguments

`y`	A vector or matrix giving the data from which the density estimate is to be computed.
`y50`	A vector of specified medians of coordinates the data generating distribution.
`y95`	A vector of specified 95th percentiles of coordinates of the data generating distribution.
`mcmc`	A list giving the MCMC parameters. The list must include the following integers: nburn giving the number of burn-in scans (the default value is 1000), nskip giving the thinning interval (the default value is 10), nsave giving the total number of scans (the default value is 1000), and ndisplay giving the number of saved scans to be displayed on screen (the function reports on the screen when every ndisplay iterations have been carried out. The default value is 100).
`ngrid`	Number of grid points where the density estimate is evaluated. This is only used if the dimension of y is lower or equal to 2. The default value is 1000.

grid — Matrix of dimension ngrid*nvar of grid points where the density estimate is evaluated. This is only used if the dimension of y is lower or equal to 2. The default value is NULL, under which the grid is chosen according to the range of the data.

Details

This generic function fits a Dirichlet process mixture of Gaussians model for density estimation (Escobar and West, 1995). Using estimates for the medians **m** and 95th percentiles **c** of coordinates of the data's distribution, the original data is rescaled as $\boldsymbol{z}_i = 2\ Diag(\boldsymbol{c})^{-1}(\boldsymbol{y_i} - \boldsymbol{m})$. Then, we fit the Gaussian DPM model

$$
\begin{aligned}
\mathbf{z}_i|\boldsymbol{\mu}_i, \mathbf{T}_i &\sim No(\boldsymbol{\mu}_i, \mathbf{T}_i),\\
(\boldsymbol{\mu}_i, \mathbf{T}_i)|G &\sim G,\\
G|G_0 &\sim DP(\nu, G_0),\\
G_0|\lambda, \boldsymbol{\Psi} &= NoWi(\mathbf{m}_\mu, \lambda, k_T, \boldsymbol{\Psi}),\\
\lambda &\sim Ga(a_\lambda, b_\lambda),\\
\boldsymbol{\Psi} &\sim Wi(k_\psi, \mathbf{W}_\psi),\\
\nu &\sim Ga(a_\nu, b_\nu),
\end{aligned}
$$

where $No(\boldsymbol{m}, \boldsymbol{U})$ denotes a normal distribution with mean **m** and precision matrix **U**, $Ga(a, b)$ denotes a Gamma distribution with shape parameter a and rate parameter b, $Wi(k, \boldsymbol{W})$ denotes a Wishart distribution with degrees of freedom k and rate matrix **W** (expectation $k\boldsymbol{W}^{-1}$), and $NoWi(\boldsymbol{m}, \lambda, k, \boldsymbol{\Psi})$ denotes a Normal-Wishart distribution with location **m**, precision factor λ, degrees of freedom k, and rate matrix **W**. The LIO prior specifies the hyperparameters as $\boldsymbol{m}_\mu = \mathbf{0}, k_T = p + 2, a_\lambda = 3/2, b_\lambda = v^2/2, k_\psi = p$, and $\boldsymbol{W}_\psi = p\boldsymbol{I}$, where $v^2 = 100p(n-1)/[n\ \chi^2_{p,0.99}]$ and $p = dim\ \boldsymbol{y}_i$. Details of this choice of prior are provided in our forthcoming paper (Shi et al., 2017).

Value

An object of class "DPdensity", whose elements include:

y — Data set used for estimation

mcmc — List of MCMC parameters

save.state$randsave — Matrix containing MCMC samples of cluster parameters

grid1 — First coordinates of points at which the density is estimated

grid2 — Second coordinates of points at which the density is estimated

dens — Density estimates at points in grid

fun1 — Marginal density estimates at points in grid1

fun2 — Marginal density estimates at points in grid2

Since Jara's DPdensity function generates the MCMC samples, the DPdensity object contains many other elements that we have left unmodified. Interpretation of these elements might differ from that in the original function due to the rescaling process performed in the wrapper.

Source

Escobar,M.D. and West,M. (1995) Bayesian density estimation and inference using mixtures, *Journal of the American Statistical Association* **90** Num 430, 577–588

Jara,A. and Hanson,T.E. and Quintana,F.A. and Muller,P. and Rosner,G.L.(2011) DPpackage: Bayesian semi-and nonparametric modeling in R, *Journal of Statistical Software* **40** Num 5, 1

Shi,Y. and Martens,M. and Banerjee,A. and Laud,P. (2017) Low Information Omnibus Priors for Dirichlet Process Mixture Models(Manuscript)

Examples

```
## Not run:
library(DPWeibull)
# Scalar data from gamma(2,1)

n <-  200
y <- rgamma(n,2,1)
# Specify percentiles
fit <- normwrapper(y=y,y50=1,y95=4)

plot(fit$dens~fit$grid1,xlim=c(0,8),type="l")
curve(dgamma(x,2),xlim=c(0,8),lty=2,add=TRUE)
rug(y)

#############################################################################

# Bivariate t / normal mixture
library(mvtnorm)

df1 <- Inf
mu1 <- c(2,0)
T1 <- 3*solve(matrix(c(1,1,1,4),nrow=2))
df2 <- 5
mu2 <- c(0,0)
T2 <- (df2-2)/df2*matrix(c(1,0,0,1),c(2,2))

n <- 400
ratio <- 0.5
n1 <- rbinom(1,n,ratio)
n2 <- n-n1
Y1 <- rmvt(n1,df=df1,sigma=solve(T1),delta=mu1)
Y2 <- rmvt(n2,df=df2,sigma=solve(T2),delta=mu2)
Y <- rbind(Y1,Y2)

# MCMC settings
nburn = 1000
nsave = 1000
nskip = 0
ndisplay = 1000
mcmc = list(nburn=nburn,nsave=nsave,nskip=nskip,ndisplay=ndisplay)

ngrid <- 1000
```

```
mesh1 <- 7/100;
mesh2 <- 6/100
grid <- cbind(seq(-3,4,mesh1),seq(-3,3,mesh2))

# Use sample percentiles
fit <- normwrapper(y=Y,mcmc=mcmc,ngrid=ngrid,grid=grid)

image(x=fit$grid1,y=fit$grid2,z=fit$dens,col=terrain.colors(12))
contour(x=fit$grid1,y=fit$grid2,z=fit$dens,add=TRUE)

y50 <- c(0,0)
y95 <- c(3,3)

# Specify percentiles
fit2 <- normwrapper(y=Y,y50=y50,y95=y95,mcmc=mcmc,ngrid=ngrid,grid=grid)

image(x=fit2$grid1,y=fit2$grid2,z=fit2$dens,col=terrain.colors(12))
contour(x=fit2$grid1,y=fit2$grid2,z=fit2$dens,add=TRUE)

##############################################################################

# Air quality data, a real data example
set.seed(13)
data("airquality")
Y <- cbind(airquality$Ozone,airquality$Solar.R)
Y <- Y[!is.na(rowSums(Y)),]
n <- nrow(Y)
p <- 2

# MCMC settings
nburn <- 1000
nsave <- 1000
nskip <- 0
ndisplay <- 1000
mcmc <- list(nburn=nburn,nsave=nsave,nskip=nskip,ndisplay=ndisplay)
ngrid <- 10000
grid <- NULL

# Use sample percentiles
fit <- normwrapper(y=Y,mcmc=mcmc,ngrid=ngrid,grid=grid)

# Scatter plot
plot(Y[,2]~Y[,1],cex = 0.5,xlab = "Ozone (parts per billion)",
     ylab = "Solar radiation (Langleys)")

# Contour plot of bivariate density
image(x=fit$grid1,y=fit$grid2,z=fit$dens,col=terrain.colors(12),
      xlab = "Ozone (parts per billion)",
      ylab = "Solar radiation (Langleys)",main = "Density estimate")
contour(x=fit$grid1,y=fit$grid2,z=fit$dens,add=TRUE)

# Marginal density plots
plot(fit$fun1~fit$grid1,type="l",xlab = "Ozone (parts per billion)",
```

```
     ylab = "Density",main = "Marginal density estimate")
plot(fit$fun2~fit$grid2,type="l",xlab = "Solar radiation (Langleys)",
     ylab = "Density",main = "Marginal density estimate")

## End(Not run)
```

`plot.ddp` *plot estimated log hazard ratio functions from an object of class ddp.*

Description

plot estimated log hazard ratio functions with credible intervals from an object of class ddp.

Usage

```
## S3 method for class 'ddp'
plot(x,simultaneous=FALSE, exp=FALSE, ...)
```

Arguments

`x`	Output an object of class ddp
`simultaneous`	Plot simultaneous credible intervals or not. The default is FALSE.
`exp`	Plot hazard ratio (TRUE) or log hazard ratio (FALSE). The default is FALSE.
`...`	Arguments to be passed to method

Value

plot estimated log hazard ratio functions from an object of class ddp.

`plot.ddpcomp` *plot estimated log subdistribution hazard ratio functions for cause 1 from an object of class ddpcomp.*

Description

plot estimated log subdistribution hazard ratio functions with credible intervals for cause 1 from an object of class ddpcomp.

Usage

```
## S3 method for class 'ddpcomp'
plot(x,simultaneous=FALSE,exp=FALSE,...)
```

Arguments

`x`	Output an object of class ddpcomp
`simultaneous`	Plot simultaneous credible intervals or not. The default is FALSE.
`exp`	Plot hazard ratio (TRUE) or log hazard ratio (FALSE). The default is FALSE.
`...`	Arguments to be passed to method

Value

plot estimated log subdistribution hazard ratio functions for cause 1 from an object of class ddpcomp.

---

plot.dpm — *plot estimated survival/density/hazard functions from an object of class dpm.*

Description

plot estimated survival/density/hazard functions with credible intervals from an object of class dpm.

Usage

```
## S3 method for class 'dpm'
plot(x,simultaneous=FALSE,...)
```

Arguments

`x`	Output an object of class dpm
`simultaneous`	Plot simultaneous credible intervals or not. The default is FALSE.
`...`	Arguments to be passed to method

Value

plot estimated survival/density/hazard functions from an object of class dpm.

`plot.dpmcomp` *plot estimated cumulative incidence/ subdistribution density/ subdistribution hazard functions from an object of class dpmcomp.*

Description

plot estimated cumulative incidence/ subdistribution density/ subdistribution hazard functions with credible intervals from an object of class dpmcomp.

Usage

```
## S3 method for class 'dpmcomp'
plot(x,simultaneous=FALSE,...)
```

Arguments

`x`	Output an object of class dpmcomp
`simultaneous`	Plot simultaneous credible intervals or not. The default is FALSE.
`...`	Arguments to be passed to method

Value

plot estimated cumulative incidence/ subdistribution density/ subdistribution hazard functions from an object of class dpm.

`plot.predddp` *plot estimated survival/density/hazard functions from an object of class predddp.*

Description

plot estimated survival/density/hazard functions with credible intervals from an object of class predddp.

Usage

```
## S3 method for class 'predddp'
plot(x,...)
```

Arguments

`x`	Output an object of class predddp
`...`	Arguments to be passed to method

Value

plot estimated survival/density/hazard functions from an object of class predddp.

`plot.predddpcomp` *plot estimated cumulative incidence function/ subdistribution density/ subdistribution hazard functions of cause 1 from an object of class predddpcomp.*

Description

plot estimated cumulative incidence function/ subdistribution density/ subdistribution hazard functions with credible intervals of cause 1 from an object of class predddpcomp.

Usage

```
## S3 method for class 'predddpcomp'
plot(x,...)
```

Arguments

`x`	Output an object of class predddpcomp
`...`	Arguments to be passed to method

Value

plot estimated cumulative incidence function/ subdistribution density/ subdistribution hazard functions of cause 1 from an object of class predddpcomp.

`predict.ddp` *generate predictions for dependent Dirichlet process Weibull model data without competing risks.*

Description

generate predictions for dependent Dirichlet process Weibull model data without competing risks.

Usage

```
## S3 method for class 'ddp'
predict(object,xpred,alpha=0.05,tpred=NULL,...)
```

Arguments

`object`	Output from dpweib, must be ddpcomp class
`xpred`	The new covariates for predictions
`tpred`	The time points where the predictions are made. If is not given by the user, it will use the time points where the log hazard ratios are calculated in dpweib function.
`alpha`	$1-\alpha$ is the probability for constructing credible intervals. The default α is 0.05.
`...`	Arguments to be passed to method

Value

tpred	The time points where the predictions are made.
alpha	$1 - \alpha$ is the probability for constructing credible intervals.
Spred	A matrix, the estimated survival for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
Spredu	A matrix, the estimated upper pointwise credible interval of the survival functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
Spredl	A matrix, the estimated lower pointwise credible interval of the survival functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
dpred	A matrix, the estimated density for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
dpredu	A matrix, the estimated upper pointwise credible interval of the density functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
dpredl	A matrix, the estimated lower pointwise credible interval of the density functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
hpred	A matrix, the estimated hazard for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
hpredu	A matrix, the estimated upper pointwise credible interval of the hazard functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
hpredl	A matrix, the estimated lower pointwise credible interval of the hazard functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.

---

predict.ddpcomp — *generate predictions for dependent Dirichlet process Weibull model data with competing risks.*

Description

generate predictions for dependent Dirichlet process Weibull model data with competing risks.

Usage

```
## S3 method for class 'ddpcomp'
predict(object,xpred,alpha=0.05,tpred=NULL,...)
```

Arguments

`object`	Output from dpweib, must be ddpcomp class
`xpred`	The new covariates for predictions
`alpha`	$1-\alpha$ is the probability for constructing credible intervals. The default α is 0.05.
`tpred`	The time points where the predictions are made. If is not given by the user, it will use the time points where the log hazard ratios are calculated in dpweib function.
`...`	Arguments to be passed to method

Value

`tpred`	The time points where the predictions are made.
`alpha`	$1-\alpha$ is the probability for constructing credible intervals.
`Fpred`	A matrix, the estimated cumulative incidence functions of cause 1 for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`Fpredu`	A matrix, the estimated upper pointwise credible interval of the cumulative incidence functions of cause 1 for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`Fpredl`	A matrix, the estimated lower pointwise credible interval of the cumulative incidence functions of cause 1 for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`dpred`	A matrix, the estimated subdistribution density functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`dpredu`	A matrix, the estimated upper pointwise credible interval of the subdistribution density functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`dpredl`	A matrix, the estimated lower pointwise credible interval of the subdistribution density functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`hpred`	A matrix, the estimated subdistribution hazard functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`hpredu`	A matrix, the estimated upper pointwise credible interval of the subdistribution hazard functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.
`hpredl`	A matrix, the estimated lower pointwise credible interval of the subdistribution hazard functions for new covariates. Each row corresponds to a covariate configuration. Each column corresponds to a time point.

summary.ddp — *generate summary of the dpweib output with ddp class.*

Description

generate estimated log hazard ratio and corresponding credible intervals of the dpweib output with ddp class at 4 time points (1/4, 1/2, 3/4 and 1 of high.pct).

Usage

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

Arguments

`object`	Output from dpweib of ddp class
`...`	Arguments to be passed to method

Value

an object of class summary.ddp

summary.ddpcomp — *generate summary of the dpweib output with ddpcomp class.*

Description

generate estimated log subdistribution hazard ratio and the corresponding credible intervals of the dpweib output with ddpcomp class at 4 time points (1/4, 1/2, 3/4 and 1 of high.pct).

Usage

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

Arguments

`object`	Output from dpweib of ddpcomp class
`...`	Arguments to be passed to method

Value

an object of class summary.ddpcomp

summary.dpm — *generate summary of the dpweib output with dpm class.*

Description

generate estimated survival and corresponding credible intervals of the dpweib output with dpm class at 4 time points(1/4, 1/2, 3/4 and 1 of high.pct).

Usage

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

Arguments

`object`	Output from dpweib of class dpm
`...`	Arguments to be passed to method

Value

an object of class summary.dpm

summary.dpmcomp — *generate summary of the dpweib output with dpmcomp class.*

Description

generate estimated cumulative incidence functions and corresponding credible intervals of the dpweib output with dpmcomp class at 4 time points (1/4, 1/2, 3/4 and 1 of high.pct).

Usage

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

Arguments

`object`	Output from dpweib of class dpmcomp
`...`	Arguments to be passed to method

Value

an object of class summary.dpmcomp

summary.predddp — *generate summary of the predict output with predddp class.*

Description

generate estimated survival and corresponding credible intervals at 4 time points (1/4, 1/2, 3/4 and 1 of high.pct) for each covaraite configuration.

Usage

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

Arguments

`object`	Output from dpweib of predddp class
`...`	Arguments to be passed to method

Value

an object of class summary.predddp

summary.predddpcomp — *generate summary of the predict output with predddpcomp class.*

Description

generate estimated cumulative incidence functions and corresponding credible intervals at 4 time points (1/4, 1/2, 3/4 and 1 of high.pct) for each covaraite configuration.

Usage

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

Arguments

`object`	Output from dpweib of predddpcomp class
`...`	Arguments to be passed to method

Value

an object of class summary.predddpcomp

survmedian *The median of the surivival functions generated by LIO prior*

Description

This data set gives the median of 20000 random survival functions generated by our LIO prior on a 0 to 10 scale. This data set is primarily used for determining scale parameter when heavy end-of-study censoring is present.

Source

Shi,Y. and Martens,M. and Banerjee,A. and Laud,P. (2017) Low Information Omnibus Priors for Dirichlet Process Mixture Models(Manuscript)

Index