Package 'SurvivalPath'

July 3, 2022

Type Package

Title Construction and Visualization of Survival Path Tree using Time-Series Survival Data

Version 1.3.2

Description Facilitates building personalized survival path models. The function survivalpath() return tree structure results, which can be used to draw easily beautiful and ready-to-publish survival path tree. See Shen L, et al (2018) <doi:10.1038/s41467-018-04633-7> .

Depends R (>= 3.6)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports ggplot2, ggtree, treeio, dplyr, rms, survival, survminer, survivalROC, Hmisc

Maintainer Zhang Tao <nyy1920@163.com>

URL https://github.com/zhangt369/SurvivalPath

NeedsCompilation no

Author Shen Lujun [aut, dtc], Zhang Tao [cre, cph]

Repository CRAN

Date/Publication 2022-07-03 18:30:05 UTC

R topics documented:

`compareTreatmentPlans` *Compare and Draw the KM curve of specified treatment plan or exposure in selected nodes*

Description

Based on the survival tree, specify the node of interest and the treatment methods, draw survival curves to evaluate the impact of treatments or exposure.

Usage

```
compareTreatmentPlans(
df,
treepoints,
mytree,
source,
treatment
)
```

Arguments

`df`	"data" in the return result of the `survivalpath()` function
`treepoints`	list object;Specify the node for drawing the KM curve, which is displayed in the survival path graphs
`mytree`	"tree" in the return result of the `survivalpath()` function
`source`	Data.frame of time slice data, which could be returned by `timedivision()`
`treatment`	Factor variable in the source data.frame. This argument is to specify the intervention or exposure that of interest at a specific node.

Details

The function creates survival curves of specified treatment plan or exposure in selected nodes. The results should be interpreted with caution as the effect of covariates have not been adjusted.

Value

No return value.

See Also

survminer

Examples

```
library(dplyr)
data("DTSDHCC")
id = DTSDHCC$ID[!duplicated(DTSDHCC$ID)]
set.seed(123)
id = sample(id,500)
miniDTSDHCC <- DTSDHCC[DTSDHCC$ID %in% id,]
dataset = timedivision(miniDTSDHCC,"ID","Date",period = 90,left_interval = 0.5,right_interval=0.5)
resu <- generatorDTSD(dataset,periodindex="time_slice",IDindex="ID" ,timeindex="OStime_day",
 statusindex="Status_of_death",variable =c( "Age", "Amount.of.Hepatic.Lesions",
 "Largest.Diameter.of.Hepatic.Lesions",
 "New.Lesion","Vascular.Invasion" ,"Local.Lymph.Node.Metastasis",
 "Distant.Metastasis" , "Child_pugh_score" ,"AFP"),predict.time=365*1)
result <- survivalpath(resu,time_slices =9)

mytree <- result$tree

library(ggplot2)
library(ggtree)
ggtree(mytree, color="black",linetype=1,size=1.2,ladderize = TRUE )+
 theme_tree2() +
 geom_text2(aes(label=label),hjust=0.6, vjust=-0.6 ,size=3.0)+
 geom_text2(aes(label=paste(node,size,mytree@data$survival,mytree@data$survivalrate,sep = "/")),
 hjust=0.6, vjust=-1.85 ,size=3.0)+
 #geom_point2(aes(shape=isTip, color=isTip), size=mytree1@data$os/40)+
 geom_point2(aes(shape=isTip, color=isTip), size=mytree@data$size%/%200+1,show.legend=FALSE)+
 #guides(color=guide_legend(title="node name/sample number/Median survival time/Survival rate")) +
 labs(size= "Nitrogen",
      x = "TimePoints",
      y = "Survival",
      subtitle = "node_name/sample number/Median survival time/Survival rate",
      title = "Survival Tree") +
 theme(legend.title=element_blank(),legend.position = c(0.1,0.9))

#Comparing the efficacy of treatment methods by drawing survival curves
treepoints = c(14,20)
compareTreatmentPlans(result$data, treepoints,mytree,dataset,"Resection")
```

DTSDHCC *SYSUCC Hepatocellular Carcinoma Dynamic Time Series Data*

Description

Time series dataset of 2360 patients with intermediate stage hepatocellular carcinoma (HCC), with each time point observation included data of 12 clinical variables and 2 survival outcome variables.

Format

A dataframe with 11684 observations and 14 variables. The data of each patient at each time point is sorted into a separate row. The variable "ID" refers to individual patient identification numbers.

The variable "Date" refers to the time point of each observation. A total of 12 clinical variables were arranged sequentially, including 1 demographic variable ("Age"), 8 observational variables ("Amount of Hepatic Lesions", "Largest Diameter of Hepatic Lesions", "New Lesion", "Vascular Invasion", "Local Lymph Node Metastasis", "Distant Metastasis", "Child_pugh_score" and "AFP"), 3 treatment variables ("TargetedTherapy", "Embolization", "Resection") and 2 outcome variables ("Status_of_death","OStime_day"). The missing values of the original dataset have been filled using Random forest regression method.

Author(s)

Lujun Shen,Tao Zhang

`evaluate` *Performance Evaluation of Survival Path Model*

Description

According to the survival path, using Harrell's concordance index C-index to evaluate the discriminative ability of the survival path in each specified time slice for prognosis.

Usage

```
evaluate(
survivalpath,
minnodesize
)
```

Arguments

`survivalpath`	The output of the `survivalpath()` function
`minnodesize`	The minimal sample size of specific node for inclusion of performance evaluation.

Details

For patients in each specific time slice, the class of survival path is regarded as a factor when computing the C-index value.

Value

The evaluate function returns an object, which includes timeslice, Indexofmodes and Cindex

`timeslice`	time slice for which performance is evaluated.
`Indexofnodes`	Nodes that is used for performance evaluation in the specified time slice.
`Cindex`	Harrell's concordance index (C-index) value for specified time slices.

EvolutionAfterTreatment *Display node transition with specified treatment plan or exposure*

Description

Calculate the number of subjects (proportion) assigned to different sub-nodes after specified treatment plan or exposure in certain node.

Usage

```
EvolutionAfterTreatment(
df,
treepoint,
mytree,
source,
treatment
)
```

Arguments

df	"data" in the return result of the survivalpath() function
treepoint	list object;Specify the node for drawing the KM curve, the node number is displayed in the survival path tree graph.
mytree	"tree" in the return result of the survivalpath() function
source	Data.frame of time slice data, which could be returned by timedivision()
treatment	Factor variable in the source data.frame. This argument is to specify the intervention or exposure that of interest at a specific node.

Value

A data.frame object, whose rows and columns represents the number of subjects in the sub-nodes (in the next time slice) and treatment plan, respectively.

Examples

```
library(dplyr)
data("DTSDHCC")
id = DTSDHCC$ID[!duplicated(DTSDHCC$ID)]
set.seed(123)
id = sample(id,500)
miniDTSDHCC <- DTSDHCC[DTSDHCC$ID %in% id,]
dataset = timedivision(miniDTSDHCC,"ID","Date",period = 90,left_interval = 0.5,right_interval=0.5)
resu <- generatorDTSD(dataset,periodindex="time_slice",IDindex="ID" ,timeindex="OStime_day",
 statusindex="Status_of_death",variable =c( "Age", "Amount.of.Hepatic.Lesions",
 "Largest.Diameter.of.Hepatic.Lesions",
 "New.Lesion","Vascular.Invasion" ,"Local.Lymph.Node.Metastasis",
```

```
 "Distant.Metastasis" , "Child_pugh_score" ,"AFP"),predict.time=365*1)
result <- survivalpath(resu,time_slices =9)
mytree <- result$tree

#Draw the survival Path model
library(ggplot2)
library(ggtree)
ggtree(mytree, color="black",linetype=1,size=1.2,ladderize = TRUE )+
 theme_tree2() +
 geom_text2(aes(label=label),hjust=0.6, vjust=-0.6 ,size=3.0)+
 geom_text2(aes(label=paste(node,size,mytree@data$survival,mytree@data$survivalrate,sep = "/")),
 hjust=0.6, vjust=-1.85 ,size=3.0)+
 #geom_point2(aes(shape=isTip, color=isTip), size=mytree1@data$os/40)+
 geom_point2(aes(shape=isTip, color=isTip), size=mytree@data$size%/%200+1,show.legend=FALSE)+
 #guides(color=guide_legend(title="node name/sample number/Median survival time/Survival rate")) +
 labs(size= "Nitrogen",
      x = "TimePoints",
      y = "Survival",
      subtitle = "node_name/sample number/Median survival time/Survival rate",
      title = "Survival Tree") +
 theme(legend.title=element_blank(),legend.position = c(0.1,0.9))

treepoint=15
A = EvolutionAfterTreatment(result$data, treepoint,mytree,dataset,"Resection")
mytable <- xtabs(~ `Resection`+treepoint, data=A)
prop.table(mytable,1)
```

---

generatorDTSD *Instantiate the an object of class Dynamic Time Series Data (DTSD)*

Description

Generate DTSD class objects using a dataframe. The dataframe should include unique identification number for each subject, multiple rows arranged data (contain risk factors, survival time and outcomes) representing observations at different time slices/time points.

Usage

```
generatorDTSD(dataset,
periodindex,
IDindex,
timeindex,
statusindex,
variable,
ifclassifydata=TRUE,
predict.time=365,
isfill=TRUE
)
```

Arguments

dataset	A dataframe of time-series observations, containing identification numbers of each subject, index of time slice, value of risk factors, survival time, and survival outcomes.
periodindex	Time slice indicator, represent index of time slice of specific observation, This variable is normally coded by integers, e.g. 0, 1, 2...
IDindex	Variable name representing patient identification number.
timeindex	Variable name representing follow up time for censored data for each specific observation.
statusindex	The status indicator representing the patient's outcome status. For Overall survival, the status is normally coded by the policy 0=alive, 1=dead.
variable	List object containing the risk factors required for modeling.
ifclassifydata	A logical value, which is optional. Judgment on whether to classify risk factors automatically. When `ifclassifydata` is `TRUE` (default is TRUE), survivalROC method is used to find cutoff to dichotomize risk factors.
predict.time	Optional, Time of event assessment for identifying the best cutoff using survivalROC. When `ifclassifydata` is `TRUE`, `predict.time` is used in combination.
isfill	Logical value, used to confirm whether to fill in missing data. If it is True, then fill.

Details

This function return a DTSD class object for conducting survivalpath function. This function facilitate enabling automatic binary classification of continuous variables. When continuous variables need to be classified, survivalROC uses survival data at the predict.time to calculate cutoffs. The cutoff will be used for construction of survival path at all time slices.

Value

return a DTSD class object for survivalpath() function.

time	`time` list object; Event time or censoring time for subjects.Each element of the list represents, the event time or censoring time starting from each observation
status	`status` list object; Indicator of status,normally use 0/1 coding. If death or event,1,otherwise,0. Each element of the list represents, the subject's outcome/event.
tsdata	`tsdata` list object; Each element of `tsdata` contains the risk factors listed in `variable`. Each element of the list represents the data frame of each time slice, normally arranged in ascending order
tsid	`tsid` list object; patient identification number. Each element of the list represents,the identification number of patient at each time slice
length	`time`,`status`,`tsdata` ,`tsid` are the same length `length`.
ts_size	List object, representing sample size at each time slice.
cutoff	List object, representing cut-off values for each variable used for modeling.

Examples

```
library(dplyr)
data("DTSDHCC")
id = DTSDHCC$ID[!duplicated(DTSDHCC$ID)]
set.seed(123)
id = sample(id,500)
miniDTSDHCC <- DTSDHCC[DTSDHCC$ID %in% id,]
dataset = timedivision(miniDTSDHCC,"ID","Date",period = 90,left_interval = 0.5,right_interval=0.5)
resu <- generatorDTSD(dataset,periodindex="time_slice",IDindex="ID" ,timeindex="OStime_day",
 statusindex="Status_of_death",variable =c( "Age", "Amount.of.Hepatic.Lesions",
 "Largest.Diameter.of.Hepatic.Lesions",
 "New.Lesion","Vascular.Invasion" ,"Local.Lymph.Node.Metastasis",
 "Distant.Metastasis" , "Child_pugh_score" ,"AFP"),predict.time=365*1)
```

---

matchsubgroup *Screen and collect data of subjects that meet the given conditions*

Description

Based on screening criteria, for each specific subject, the data of the observation that meet the conditions for the first time and the data of subsequent observations in following time slices will be collected. The data from initial time slice that meet the given conditions to last time slice were then compiled into a new time-slice dataset, with an aim to create personalized survival path map.

Usage

```
matchsubgroup(
DTSD,
varname,
varvalue
)
```

Arguments

DTSD	Object of class DTSD
varname	list object;The variable used to screen subjects, and the variables need to be contained in the time-slice data.
varvalue	list object;Subjects whose varname variable value equal to varvalue will be selected

Details

According to the input time, status, variables, subject ID, etc., the data of eligible subjects is screened through specified given conditions. The subject whose variable data of the first and subsequent time slices are sequentially screened. Once an observation meet the given condition, data of that observation and the observations in following time slices will be for the subject will be

collected. Data of all subject that meet the criteria will be compiled into a new time-slice dataset. Based on the new dataset, the function returns a new DTSD object was got. The final returned result contains four list objects: time, state, timeslicedata, subject ID (tspatientid).

Value

Returns a DTSD object.

Author(s)

Shen Lujun and ZhangTao

Examples

```
library(dplyr)
data("DTSDHCC")
id = DTSDHCC$ID[!duplicated(DTSDHCC$ID)]
set.seed(123)
id = sample(id,120)
miniDTSDHCC <- DTSDHCC[DTSDHCC$ID %in% id,]
dataset = timedivision(miniDTSDHCC,"ID","Date",period = 90,left_interval = 0.5,right_interval=0.5)
resu <- generatorDTSD(dataset,periodindex="time_slice",IDindex="ID" ,timeindex="OStime_day",
 statusindex="Status_of_death",variable =c( "Age", "Amount.of.Hepatic.Lesions",
 "Largest.Diameter.of.Hepatic.Lesions",
 "New.Lesion","Vascular.Invasion" ,"Local.Lymph.Node.Metastasis",
 "Distant.Metastasis" , "Child_pugh_score" ,"AFP"),predict.time=365*1)

varname=list('Amount.of.Hepatic.Lesions')
varvalue=list(1)
df <- matchsubgroup(resu,varname=varname ,varvalue=varvalue)

result <- survivalpath(df,time_slices =4)
```

---

plotKM *Compare and Draw the KM curves of any given nodes*

Description

According to the survival path tree, draw the KM curves of the using any nodes on the survival tree

Usage

```
plotKM(
df,
treepoints,
mytree,
risk.table=TRUE
)
```

Arguments

df	"data" in the returned result of the survivalpath() function
treepoints	list object;Specify the node for drawing the KM curve, which is in the survival path tree
mytree	"tree" in the returned result of the survivalpath() function
risk.table	Logical value. Allowed values include:TRUE or FALSE specifying whether to show the risk table. Default is FALSE.

Details

Plot survival curves for patients contained in nodes in the survival path tree.

Value

No return value.

See Also

survminer

Examples

```
library(dplyr)
data("DTSDHCC")
id = DTSDHCC$ID[!duplicated(DTSDHCC$ID)]
set.seed(123)
id = sample(id,500)
miniDTSDHCC <- DTSDHCC[DTSDHCC$ID %in% id,]
dataset = timedivision(miniDTSDHCC,"ID","Date",period = 90,left_interval = 0.5,right_interval=0.5)
resu <- generatorDTSD(dataset,periodindex="time_slice",IDindex="ID" ,timeindex="OStime_day",
 statusindex="Status_of_death",variable =c( "Age", "Amount.of.Hepatic.Lesions",
 "Largest.Diameter.of.Hepatic.Lesions",
 "New.Lesion","Vascular.Invasion" ,"Local.Lymph.Node.Metastasis",
 "Distant.Metastasis" , "Child_pugh_score" ,"AFP"),predict.time=365*1)
result <- survivalpath(resu,time_slices =9)

mytree <- result$tree

library(ggplot2)
library(ggtree)
ggtree(mytree, color="black",linetype=1,size=1.2,ladderize = TRUE )+
 theme_tree2() +
 geom_text2(aes(label=label),hjust=0.6, vjust=-0.6 ,size=3.0)+
 geom_text2(aes(label=paste(node,size,mytree@data$survival,mytree@data$survivalrate,sep = "/")),
 hjust=0.6, vjust=-1.85 ,size=3.0)+
 #geom_point2(aes(shape=isTip, color=isTip), size=mytree1@data$os/40)+
 geom_point2(aes(shape=isTip, color=isTip), size=mytree@data$size%/%200+1,show.legend=FALSE)+
 #guides(color=guide_legend(title="node name/sample number/Median survival time/Survival rate")) +
 labs(size= "Nitrogen",
      x = "TimePoints",
```

```
      y = "Survival",
      subtitle = "node_name/sample number/Median survival time/Survival rate",
      title = "Survival Tree") +
 theme(legend.title=element_blank(),legend.position = c(0.1,0.9))

#plot KM curve
treepoints = c(14,20)
plotKM(result$data, treepoints,mytree,risk.table=T)
```

`survivalpath` *Build Survival Path Model Using Dynamic Time Series Data (DTSD) object*

Description

Survival Path Mapping for Dynamic Prediction of Cancer Patients Using Time-Series Survival Data.This is the core function that build survival path tree model based on Akaike information criterion (AIC) and self-designed arguments.

Usage

```
survivalpath(
DTSD,
time_slices,
treatments=NULL,
num_categories=2,
p.value=0.05,
minsample = 15,
degreeofcorrelation=0.7,
rates=365
)
```

Arguments

`DTSD`	A DTSD class object. See function `generatorDTSD()` for details.
`time_slices`	numeric, define the total number of time slices (starting from the front) needed to be included in the survival path model
`treatments`	A list object, with default value of NULL. This argument is used to specify the intervention measures/exposure taken by the observation at different time slices. The treatment or exposure variables specified will not be utilized in construction of the survival path model
`num_categories`	Numeric, the default value is 2. The maximum number of branches that each node can divide
`p.value`	`p.value` for hypothesis testing; variables with p value less than p.value in univariate analysis are significant candidate variables and will undergo further feature selection

`minsample` Minimum sample size for branching

`degreeofcorrelation`
default 0.7;When the correlation between variables is greater than this value, the variables are considered to have collinearity. The pair of variables that exceed the correlation coefficient will automatically compare their Akaike information criterion (AIC) values when each of two serve as the only predictor for outcome; the variable with the smaller AIC value will be removed.

`rates` Numeric value. Calculate the rate of the outcome for the nodes in the survival path model at the time point of the argument `rates`

Details

After the pre-processing of data, under a user-defined parameters on covariates, significance level, minimum bifurcation sample size and number of time slices for analysis, survival paths can be computed using the main function, which can be visualized as a tree diagram.

Value

The survivalpath function returns an object, which includes data, tree and df.

`data` `data` describes the grouping variables and values for each observation at different time slices.

`tree` A `treedata` object `tree`,which facilitate creation of tree diagram and mapping of patients' personalized survival path

`df` A Data.frame object containing the node numbers corresponding to each observation at different time slices in survival path tree model tree. The dataframe added three new columns, the parent_node correspond to the upper node that the observation belongs to, which indicate the group of participants for modeling and feature selection; the sub_node indicates the node that the corresponding observation represent after subdivision from the parent_node, the information of sub_node is used for model evaluation and comparison. The variable_value indicate the reason for transfer from the parent_node to the sub_node.

`maxpath` The longest path length in the survival path model.

Note

The idea of developing the SurvivalPath R package stems from our previous exploratory work, in which we attempted to achieve dynamic prognosis prediction by establishing survival paths based on the time-series data of patients with hepatocellular carcinoma (HCC). The survival path approach we proposed provide a potential solution for dynamic prognosis prediction and management of cancer patients by constructing survival path maps using returned key prognostic factors after analysis of structured time-series survival data. More importantly, the survival path model could be easily understood and utilized by clinicians when compared to black-box models. The SurvivalPath R package is a newly developed tool to facilitate fast building of survival path models, with an aim of promoting standardization of this methodology. In this package we optimized the feature selection process. Oneto one collinearity analysis was embedded (as an argument) to screen out noncollinear candidate variables before formal feature selection in the main function to reduces the confounding impact of potential collinearity on feature selection in the Cox model. In addition, the SurvivalPath

R package is now compatible with continuous variable. The classifydata function enabling automatic binary classification of continuous variables and their entry into the model. This methodology is still young, and we welcome efforts from all the world to improve it.

Author(s)

Lujun Shen and Tao Zhang

References

Lujun Shen. (2018) *Dynamically prognosticating patients with hepatocellular carcinoma through survival paths mapping based on time-series data*, https://www.nature.com/articles/s41467-018-04633-7.pdf
Nat Commun. 2018 Jun 8;9(1):2230. doi: 10.1038/s41467-018-04633-7. PMID: 29884785; PMCID: PMC5993743.

Examples

```
library(dplyr)
data("DTSDHCC")
#Randomly select a proportion of cases for demo
id = DTSDHCC$ID[!duplicated(DTSDHCC$ID)]
set.seed(123)
id = sample(id,500)
miniDTSDHCC <- DTSDHCC[DTSDHCC$ID %in% id,]
#Convert multiple rows time series data into time-slices data
dataset = timedivision(miniDTSDHCC,"ID","Date",period = 90,left_interval = 0.5,right_interval=0.5)
#Create DTSD object using time-slices data
resu <- generatorDTSD(dataset,periodindex="time_slice",IDindex="ID" ,timeindex="OStime_day",
 statusindex="Status_of_death",variable =c( "Age", "Amount.of.Hepatic.Lesions",
 "Largest.Diameter.of.Hepatic.Lesions",
 "New.Lesion","Vascular.Invasion" ,"Local.Lymph.Node.Metastasis",
 "Distant.Metastasis" , "Child_pugh_score" ,"AFP"),predict.time=365*1)
#Construction of survival path using this function, takes minutes
result <- survivalpath(resu,time_slices =9)

#Draw Suvival Path Tree
library(ggplot2)
library(ggtree)
mytree <- result$tree

ggtree(mytree, color="black",linetype=1,size=1.2,ladderize = TRUE )+
 theme_tree2() +
 geom_text2(aes(label=label),hjust=0.6, vjust=-0.6 ,size=3.0)+
 geom_text2(aes(label=paste(node,size,mytree@data$survival,mytree@data$survivalrate,sep = "/")),
 hjust=0.6, vjust=-1.85 ,size=3.0)+
 #geom_point2(aes(shape=isTip, color=isTip), size=mytree1@data$os/40)+
 geom_point2(aes(shape=isTip, color=isTip), size=mytree@data$size%/%200+1,show.legend=FALSE)+
 #guides(color=guide_legend(title="node name/sample number/Median survival time/Survival rate")) +
 labs(size= "Nitrogen",
      x = "TimePoints",
      y = "Survival",
```

```
      subtitle = "node_name/sample number/Median survival time/Survival rate",
      title = "Survival Tree") +
 theme(legend.title=element_blank(),legend.position = c(0.1,0.9))
```

`timedivision` *Convert Multiple Rows Arranged Time-Series Data into Time-Slices Data*

Description

Data preprocessing process essential for building survival path model. For each subject with observations at different time point, screen out specific observations at each specific time slice by setting associated parameters, includes period, left_interval and right_interval.

Usage

```
timedivision(dataset,
ID,
time,
period=30,
left_interval = 0.5,
right_interval = 0.5
)
```

Arguments

`dataset`	A multiple rows arranged time-series dataset, containing identification numbers, follow-up time points, risk factors, survival time, and survival status.
`ID`	Character string, representing`ID` corresponding to each row of data in the dataset, which should be unique for each subject.
`time`	Date format, which indicates time point of each observation.
`period`	Numeric, utilized to customize follow-up sampling `period`;normally counting in days.
`left_interval`	Numeric, preferentially fall into the interval of (0,1). For a specific sampling in time slice `T`, the earliest sampling in the time interval [`left_interval`*period, right_interval*period] is considered as the sampling data of the specific time slice `T`.
`right_interval`	same as above.

Details

This function is used to facilitate automatic generation of time-slice data. The date of observations for each subject should be arranged in ascending order. The researchers can skip this process if they intend to prepare time-slice data manually or using customized codes. It's important to note that this function only support data sampling of the "earliest" observation of interval in each time slice. If no observation fall into the interval of time slice T, then sampling of observation in time slice T+1 for that subject will be terminated.

Value

data.frame;observations of different time slices for each ID.The new data.frame returned added a new column "time_slice", which indicates the time slice of each observation included.

Author(s)

Lujun Shen and Tao Zhang

Examples

```
library(dplyr)
data("DTSDHCC")
id = DTSDHCC$ID[!duplicated(DTSDHCC$ID)]
set.seed(123)
id = sample(id,500)
miniDTSDHCC <- DTSDHCC[DTSDHCC$ID %in% id,]
dataset = timedivision(miniDTSDHCC,"ID","Date",period = 90,left_interval = 0.5,right_interval=0.5)
resu <- generatorDTSD(dataset,periodindex="time_slice",IDindex="ID" ,timeindex="OStime_day",
 statusindex="Status_of_death",variable =c( "Age", "Amount.of.Hepatic.Lesions",
 "Largest.Diameter.of.Hepatic.Lesions",
 "New.Lesion","Vascular.Invasion" ,"Local.Lymph.Node.Metastasis",
 "Distant.Metastasis" , "Child_pugh_score" ,"AFP"),predict.time=365*1)
```

Index