Package 'PJFM'

November 6, 2024

Type Package

Title Variational Inference for High-Dimensional Joint Frailty Model

Version 0.1.0

Maintainer Jiehuan Sun <jiehuan.sun@gmail.com>

Description Joint frailty models have been widely used to study the associations between recurrent events and a survival outcome. However, existing joint frailty models only consider one or a few recurrent events
and cannot deal with high-dimensional recurrent events. This package can be used to fit our recently developed penalized joint frailty model that can handle high-dimensional recurrent events. Specifically, an adaptive lasso penalty is imposed on the parameters for the effects of the recurrent events on the survival outcome, which allows for variable selection.
Also, our algorithm is computationally efficient, which is based on the Gaussian variational approximation method.

Depends R (>= 3.6.0)

Imports Rcpp (>= 1.0.0), survival(>= 3.2), statmod(>= 1.4), pracma(>= 2.2), Matrix(>= 1.3)

LinkingTo Rcpp, RcppArmadillo, RcppEnsmallen

Suggests splines

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

NeedsCompilation yes

Author Jiehuan Sun [aut, cre]

Repository CRAN

Date/Publication 2024-11-06 21:00:06 UTC

Contents

`control_list` *control_list*

Description

This list contains a list of parameters specifying the joint frailty model.

Details

- ID_name: the variable name indicating the patient ID in both recurrent events data and survival data.
- item_name: the variable name indicating the types of recurrent events in the recurrent events data.
- time_name: the variable name indicating the occurrence time in the recurrent events data.
- fix_cov: a set of variables names indicating the covariates of fixed-effects in the recurrent events submodel. If NULL, not baseline covariates are included.
- random_cov: a set of variables names indicating the covariates of random-effects in the recurrent events submodel. If NULL, not baseline covariates are included.
- recur_fix_time_fun: a function specifying the time-related basis functions (fixed-effects) in the recurrent events submodel.
- recur_ran_time_fun: a function specifying the time-related basis functions (random-effects) in the recurrent events submodel. If this is an intercept only function, then only a random intercept is included (i.e. a joint frailty model).
- surv_fix_time_fun: a log-hazard function for the survival submodel.
- surv_time_name the variable name for the survival time in the survival data.
- surv_status_name the variable name for the censoring indicator in the survival data.
- surv_cov a set of variables names specifying the baseline covariates in the survival submodel.
- n_points an integer indicating the numebr of nodes being used in the Gaussian quadrature.

Author(s)

Jiehuan Sun <`jiehuan.sun@gmail.com`>

PJFM_fit *The function to fit PJFM.*

Description

The function is used to fit PJFM.

Usage

```
PJFM_fit(
  RecurData = NULL,
  SurvData = NULL,
  control_list = NULL,
  EventName = NULL,
  nlam = 50,
  ridge = 0,
  pmax = 10,
  min_ratio = 0.01,
  maxiter = 100,
  eps = 1e-04
)
```

Arguments

RecurData	a data frame containing the recurrent events data (see RecurData).
SurvData	a data frame containing the survival data (see SurvData).
control_list	a list of parameters specifying the joint frailty model (see control_list).
EventName	a vector indicating which set of recurrent events to be analyzed. If NULL, all recurrent events in RecurData will be used.
nlam	number of tuning parameters.
ridge	ridge penalty.
pmax	the maximum of biomarkers being selected. The algorithm will stop early if the maximum has been reached.
min_ratio	the ratio between the largest possible penalty and the smallest penalty to tune.
maxiter	the maximum number of iterations.
eps	threshold for convergence.

Value

return a list with the following objects.

object_name	indicates whether this is a PJFM or JFM object. If JFM object, then some recurrent events were selected and the returned model is the refitted model with only selected recurrent events, but no penalty; otherwise, PJFM object is returned.
fit	fitted models with estimated parameters in both submodels.
hess	Hessian matrix; only available for JFM object.

References

Jiehuan Sun. "Dynamic Prediction with Penalized Joint Frailty Model of High-Dimensional Recurrent Event Data and a Survival Outcome".

Examples

```
require(splines)
data(PJFMdata)

up_limit = ceiling(max(SurvData$ftime))
bs_fun <- function(t=NULL){
    bs(t, knots = NULL, degree = 2, intercept = TRUE, Boundary.knots= c(0,up_limit))
}

recur_fix_time_fun = bs_fun
recur_ran_time_fun <- function(x=NULL){
    xx = cbind(1, matrix(x, ncol = 1))
    colnames(xx) = c(”intercept”,”year_1”)
    xx[,1,drop=FALSE]
    #xx
}

surv_fix_time_fun = bs_fun

control_list = list(
    ID_name = ”ID”, item_name = ”feature_id”,
    time_name = ”time”, fix_cov = ”x”, random_cov = NULL,
    recur_fix_time_fun = recur_fix_time_fun,
    recur_ran_time_fun = recur_ran_time_fun,
    surv_fix_time_fun = surv_fix_time_fun,
    surv_time_name = ”ftime”,  surv_status_name = ”fstat”,
    surv_cov = ”x”, n_points = 5
)


## this step takes about a few minute.
## analyze the first 10 recurrent events
res = PJFM_fit(RecurData=RecurData, SurvData=SurvData,
               control_list=control_list, EventName=1:10)

## get summary table
summary_table = PJFM_summary(res)
```

PJFM_prediction *The function to calculate predicted probabilities*

Description

The function is used to calculate predicted probabilities.

Usage

```
PJFM_prediction(
  res = NULL,
  RecurData_test = NULL,
  SurvData_test = NULL,
  control_list = NULL,
  t_break = 1,
  tau = 0.5
)
```

Arguments

`res` a model fit returned by PJFM_fit; the prediction only works the returned model fit is JFM, but not PJFM.

`RecurData_test` a data frame containing the recurrent events data on the test dataset (see `RecurData`).

`SurvData_test` a data frame containing the survival data on the test dataset (see `SurvData`).

`control_list` a list of parameters specifying the joint frailty model (see `control_list`).

`t_break` the landmark time point

`tau` the prediction window (i.e., (t_break, t_break+tau]).

Value

return a data frame, which contains all the variables in SurvData_test as well as t_break, tau, and risk. The column risk indicates the predicted probability of event in the given prediction window.

References

Jiehuan Sun. "Dynamic Prediction with Penalized Joint Frailty Model of High-Dimensional Recurrent Event Data and a Survival Outcome".

Examples

```
require(splines)
data(PJFMdata)

up_limit = ceiling(max(SurvData$ftime))
bs_fun <- function(t=NULL){
    bs(t, knots = NULL, degree = 2, intercept = TRUE, Boundary.knots= c(0,up_limit))
}

recur_fix_time_fun = bs_fun
recur_ran_time_fun <- function(x=NULL){
    xx = cbind(1, matrix(x, ncol = 1))
    colnames(xx) = c(”intercept”,”year_1”)
    xx[,1,drop=FALSE]
    #xx
}

surv_fix_time_fun = bs_fun

control_list = list(
    ID_name = "ID", item_name = "feature_id",
    time_name = "time", fix_cov = "x", random_cov = NULL,
    recur_fix_time_fun = recur_fix_time_fun,
    recur_ran_time_fun = recur_ran_time_fun,
    surv_fix_time_fun = surv_fix_time_fun,
    surv_time_name = "ftime",  surv_status_name = "fstat",
    surv_cov = "x", n_points = 5
)


train_id = 1:200
test_id = 200:300

SurvData_test = SurvData[is.element(SurvData$ID, test_id), ]
RecurData_test = RecurData[is.element(RecurData$ID, test_id), ]

SurvData = SurvData[is.element(SurvData$ID, train_id), ]
RecurData = RecurData[is.element(RecurData$ID, train_id), ]

## this step takes a few minutes.
## analyze the first 10 recurrent events
res = PJFM_fit(RecurData=RecurData, SurvData=SurvData,
               control_list=control_list, EventName=1:10)


## get prediction probabilities
pred_scores = PJFM_prediction(res=res,RecurData_test=RecurData_test,
                              SurvData_test=SurvData_test,control_list=control_list,
                              t_break = 1, tau = 0.5)
```

`PJFM_summary` *The function to get summary table of PJFM fit.*

Description

The function is used to get summary table of PJFM fit.

Usage

```
PJFM_summary(res = NULL)
```

Arguments

`res`	a model fit returned by PJFM_fit; SE estimates are only available for JFM, but not PJFM.

Value

return a data frame, which contains parameter estimates in both submodels.

References

Jiehuan Sun. "Dynamic Prediction with Penalized Joint Frailty Model of High-Dimensional Recurrent Event Data and a Survival Outcome".

`RecurData` *Simulated Recurrent Events Data*

Description

This dataset contains recurrent events data.

Usage

```
data(PJFMdata)
```

Format

A data frame with 57582 rows and 3 variables

Details

- ID: patient ID
- feature_id: types of recurrent events
- time: occurrence time

Author(s)

Jiehuan Sun `<jiehuan.sun@gmail.com>`

SurvData *Simulated Survival Data*

Description

This dataset contains survival outcome.

Usage

```
data(PJFMdata)
```

Format

A data frame with 300 rows and 4 variables

Details

- ID: patient ID
- fstat: censoring indicator
- ftime: survival time
- x: baseline covariates

Author(s)

Jiehuan Sun <jiehuan.sun@gmail.com>

Index