Package ‘JMH’

April 2, 2026

Type Package

Title Joint Model of Heterogeneous Repeated Measures and Survival Data

Version 1.0.4

Date 2026-03-29

Maintainer Shanpeng Li <`lishanpeng0913@ucla.edu`>

Description Maximum likelihood estimation for the semi-parametric joint modeling of competing risks and longitudinal data in the presence of heterogeneous within-subject variability, proposed by Li and colleagues (2023) <doi:10.48550/arXiv.2506.12741>.
The proposed method models the within-subject variability of the biomarker and associates it with the risk of the competing risks event. The time-to-event data is modeled using a (cause-specific) Cox proportional hazards regression model with time-fixed covariates.
The longitudinal outcome is modeled using a mixed-effects location and scale model. The association is captured by shared random effects. The model is estimated using an Expectation Maximization algorithm. This is the final release of the 'JMH' package. Active development has been moved to the 'FastJM' package, which provides improved functionality and ongoing support. Users are strongly encouraged to transition to 'FastJM'.

License GPL (>= 3)

NeedsCompilation yes

Imports Rcpp (>= 1.0.7), parallel, dplyr, stats, caret, pec

LinkingTo Rcpp, RcppEigen

Depends R (>= 3.5.0), survival, nlme, utils, MASS, statmod, magrittr

RoxygenNote 7.3.2

Suggests testthat (>= 3.0.0), spelling

Language en-US

LazyData true

Encoding UTF-8

Author Shanpeng Li [aut, cre],
Gang Li [ctb]

Repository CRAN

Date/Publication 2026-04-02 15:50:02 UTC

Contents

`AUCJMMLSM` *Time-dependent AUC for joint models*

Description

Time-dependent AUC for joint models

Usage

```
AUCJMMLSM(
  seed = 100,
  object,
  landmark.time = NULL,
  horizon.time = NULL,
  obs.time = NULL,
  method = c("Laplace", "GH"),
  quadpoint = NULL,
  maxiter = 1000,
  n.cv = 3,
  survinitial = TRUE,
  opt = "nlminb",
  initial.para = FALSE,
  LOCF = FALSE,
  LOCFcovariate = NULL,
  clongdata = NULL,
  metric = c("AUC", "Cindex"),
  ...
)
```

Arguments

`seed`	a numeric value of seed to be specified for cross validation.
`object`	object of class 'JMMLSM'.
`landmark.time`	a numeric value of time for which dynamic prediction starts..
`horizon.time`	a numeric vector of future times for which predicted probabilities are to be computed.
`obs.time`	a character string of specifying a longitudinal time variable.
`method`	estimation method for predicted probabilities. If `Laplace`, then the empirical empirical estimates of random effects is used. If `GH`, then the pseudo-adaptive Gauss-Hermite quadrature is used.
`quadpoint`	the number of pseudo-adaptive Gauss-Hermite quadrature points if `method = "GH"`.
`maxiter`	the maximum number of iterations of the EM algorithm that the function will perform. Default is 10000.
`n.cv`	number of folds for cross validation. Default is 3.
`survinitial`	Fit a Cox model to obtain initial values of the parameter estimates. Default is TRUE.
`opt`	Optimization method to fit a linear mixed effects model, either nlminb (default) or optim.
`initial.para`	Initial guess of parameters for cross validation. Default is FALSE.
`LOCF`	a logical value to indicate whether the last-observation-carried-forward approach applies to prediction. If `TRUE`, then `LOCFcovariate` and `clongdata` must be specified to indicate which time-dependent survival covariates are included for dynamic prediction. Default is FALSE.
`LOCFcovariate`	a vector of string with time-dependent survival covariates if `LOCF = TRUE`. Default is NULL.
`clongdata`	a long format data frame where time-dependent survival covariates are incorporated. Default is NULL.
`metric`	a string to indicate which metric is used.
`...`	Further arguments passed to or from other methods.

Value

a list of matrices with conditional probabilities for subjects.

Author(s)

Shanpeng Li <`lishanpeng0913@ucla.edu`>

See Also

`JMMLSM`, `survfitJMMLSM`

`cdata` *Simulated competing risks data*

Description

The `cdata` data frame has 200 rows and 6 columns.

Usage

```
data(cdata)
```

Format

This data frame contains the following columns:

`ID` patient identifier.

`survtime` event time.

`cmprsk` event indicator. `0` denotes censoring, `1` risk 1, and `2` risk 2.

`var1` treatment indicator. `0` denotes the placebo group and `1` the treatment group.

`var2` continuous variable.

`var3` continuous variable.

`ConcordanceJMMLSM` *Concordance for joint models*

Description

Concordance for joint models

Usage

```
ConcordanceJMMLSM(
  seed = 100,
  object,
  opt = "nlminb",
  n.cv = 3,
  maxiter = 10000,
  initial.optimizer = "BFGS",
  initial.para = TRUE,
  ...
)
```

Arguments

`seed`	a numeric value of seed to be specified for cross validation.
`object`	object of class 'JMMLSM'.
`opt`	Optimization method to fit a linear mixed effects model, either nlminb (default) or optim.
`n.cv`	number of folds for cross validation. Default is 3.
`maxiter`	the maximum number of iterations of the EM algorithm that the function will perform. Default is 10000.
`initial.optimizer`	Method for numerical optimization to be used. Default is `BFGS`.
`initial.para`	Initial guess of parameters for cross validation. Default is FALSE.
`...`	Further arguments passed to or from other methods.

Value

a list of matrices with conditional probabilities for subjects.

Author(s)

Shanpeng Li <`lishanpeng0913@ucla.edu`>

See Also

`JMMLSM`

`JMMLSM` *Joint Modeling for Continuous outcomes*

Description

Joint modeling of longitudinal continuous data and competing risks

Usage

```
JMMLSM(
  cdata,
  ydata,
  long.formula,
  surv.formula,
  variance.formula,
  random,
  maxiter = 1000,
  epsilon = 1e-04,
  quadpoint = NULL,
  print.para = FALSE,
```

```
  survinitial = TRUE,
  initial.para = NULL,
  method = "adaptive",
  opt = "nlminb",
  initial.optimizer = "BFGS"
)
```

Arguments

`cdata`	a survival data frame with competing risks or single failure. Each subject has one data entry.
`ydata`	a longitudinal data frame in long format.
`long.formula`	a formula object with the response variable and fixed effects covariates to be included in the longitudinal sub-model.
`surv.formula`	a formula object with the survival time, event indicator, and the covariates to be included in the survival sub-model.
`variance.formula`	an one-sided formula object with the fixed effects covariates to model the variance of longitudinal sub-model.
`random`	a one-sided formula object describing the random effects part of the longitudinal sub-model. For example, fitting a random intercept model takes the form ~ 1|ID. Alternatively. Fitting a random intercept and slope model takes the form ~ x1 + ... + xn|ID.
`maxiter`	the maximum number of iterations of the EM algorithm that the function will perform. Default is 10000.
`epsilon`	Tolerance parameter. Default is 0.0001.
`quadpoint`	the number of Gauss-Hermite quadrature points to be chosen for numerical integration. Default is 15 which produces stable estimates in most dataframes.
`print.para`	Print detailed information of each iteration. Default is FALSE, i.e., not to print the iteration details.
`survinitial`	Fit a Cox model to obtain initial values of the parameter estimates. Default is TRUE.
`initial.para`	a list of initialized parameters for EM iteration. Default is NULL.
`method`	Method for proceeding numerical integration in the E-step. Default is adaptive.
`opt`	Optimization method to fit a linear mixed effects model, either nlminb (default) or optim.
`initial.optimizer`	Method for numerical optimization to be used. Default is `BFGS`.

Value

Object of class `JMMLSM` with elements

`ydata`	the input longitudinal dataset for fitting a joint model. It has been re-ordered in accordance with descending observation times in `cdata`.

`cdata`	the input survival dataset for fitting a joint model. It has been re-ordered in accordance with descending observation times.
`PropEventType`	a frequency table of number of events.
`beta`	the vector of fixed effects for the mean trajectory in the mixed effects location and scale model.
`tau`	the vector of fixed effects for the within-subject variability in the mixed effects location and scale model.
`gamma1`	the vector of fixed effects for type 1 failure for the survival model.
`gamma2`	the vector of fixed effects for type 2 failure for the survival model. Valid only if `CompetingRisk = TRUE`.
`alpha1`	the vector of association parameter(s) for the mean trajectory for type 1 failure.
`alpha2`	the vector of association parameter(s) for the mean trajectory for type 2 failure. Valid only if `CompetingRisk = TRUE`.
`vee1`	the vector of association parameter(s) for the within-subject variability for type 1 failure.
`vee2`	the vector of association parameter(s) for the within-subject variability for type 2 failure. Valid only if `CompetingRisk = TRUE`.
`H01`	the matrix that collects baseline hazards evaluated at each uncensored event time for type 1 failure. The first column denotes uncensored event times, the second column the number of events, and the third columns the hazards obtained by Breslow estimator.
`H02`	the matrix that collects baseline hazards evaluated at each uncensored event time for type 2 failure. The data structure is the same as `H01`. Valid only if `CompetingRisk = TRUE`.
`Sig`	the variance-covariance matrix of the random effects.
`iter`	the total number of iterations until convergence.
`convergence`	convergence identifier: 1 corresponds to successful convergence, whereas 0 to a problem (i.e., when 0, usually more iterations are required).
`vcov`	the variance-covariance matrix of all the fixed effects for both models.
`sebeta`	the standard error of `beta`.
`setau`	the standard error of `tau`.
`segamma1`	the standard error of `gamma1`.
`segamma2`	the standard error of `gamma2`. Valid only if `CompetingRisk = TRUE`.
`sealpha1`	the standard error of `alpha1`.
`sealpha2`	the standard error of `alpha2`. Valid only if `CompetingRisk = TRUE`.
`sevee1`	the standard error of `vee1`.
`sevee2`	the standard error of `vee2`. Valid only if `CompetingRisk = TRUE`.
`seSig`	the vector of standard errors of covariance of random effects.
`loglike`	the log-likelihood value.
`EFuntheta`	a list with the expected values of all the functions of random effects.

`CompetingRisk`	logical value; TRUE if a competing event are accounted for.
`quadpoint`	the number of Gauss Hermite quadrature points used for numerical integration.
`LongitudinalSubmodelmean`	the component of the `long.formula`.
`LongitudinalSubmodelvariance`	the component of the `variance.formula`.
`SurvivalSubmodel`	the component of the `surv.formula`.
`random`	the component of the `random`.
`call`	the matched call.

Examples

```
require(JMH)
data(ydata)
data(cdata)
## fit a joint model

fit <- JMMLSM(cdata = cdata, ydata = ydata,
              long.formula = Y ~ Z1 + Z2 + Z3 + time,
              surv.formula = Surv(survtime, cmprsk) ~ var1 + var2 + var3,
              variance.formula = ~ Z1 + Z2 + Z3 + time,
              quadpoint = 6, random = ~ 1|ID, print.para = FALSE)

## make dynamic prediction of two subjects
cnewdata <- cdata[cdata$ID %in% c(122, 152), ]
ynewdata <- ydata[ydata$ID %in% c(122, 152), ]
survfit <- survfitJMMLSM(fit, seed = 100, ynewdata = ynewdata, cnewdata = cnewdata,
                         u = seq(5.2, 7.2, by = 0.5), Last.time = "survtime",
                         obs.time = "time", method = "GH")
oldpar <- par(mfrow = c(2, 2), mar = c(5, 4, 4, 4))
plot(survfit, include.y = TRUE)
par(oldpar)

## Overall, time-independent concordance index (C-index)
Concord <- ConcordanceJMMLSM(seed = 100, fit, n.cv = 3)
summary(Concord)
```

`MAEQJMMLSM` *A metric of prediction accuracy of joint model by comparing the predicted risk with the empirical risks stratified on different predicted risk group.*

Description

A metric of prediction accuracy of joint model by comparing the predicted risk with the empirical risks stratified on different predicted risk group.

Usage

```
MAEQJMMLSM(
  seed = 100,
  object,
  landmark.time = NULL,
  horizon.time = NULL,
  obs.time = NULL,
  method = c("Laplace", "GH"),
  quadpoint = NULL,
  maxiter = 1000,
  survinitial = TRUE,
  n.cv = 3,
  quantile.width = 0.25,
  opt = "nlminb",
  initial.para = FALSE,
  LOCF = FALSE,
  LOCFcovariate = NULL,
  clongdata = NULL,
  ...
)
```

Arguments

`seed`	a numeric value of seed to be specified for cross validation.
`object`	object of class 'JMMLSM'.
`landmark.time`	a numeric value of time for which dynamic prediction starts..
`horizon.time`	a numeric vector of future times for which predicted probabilities are to be computed.
`obs.time`	a character string of specifying a longitudinal time variable.
`method`	estimation method for predicted probabilities. If `Laplace`, then the empirical empirical estimates of random effects is used. If `GH`, then the standard Gauss-Hermite quadrature is used.
`quadpoint`	the number of standard Gauss-Hermite quadrature points if `method = "GH"`.
`maxiter`	the maximum number of iterations of the EM algorithm that the function will perform. Default is 10000.
`survinitial`	Fit a Cox model to obtain initial values of the parameter estimates. Default is TRUE.
`n.cv`	number of folds for cross validation. Default is 3.
`quantile.width`	a numeric value of width of quantile to be specified. Default is 0.25.
`opt`	Optimization method to fit a linear mixed effects model, either nlminb (default) or optim.
`initial.para`	Initial guess of parameters for cross validation. Default is FALSE.
`LOCF`	a logical value to indicate whether the last-observation-carried-forward approach applies to prediction. If `TRUE`, then `LOCFcovariate` and `clongdata` must be

	specified to indicate which time-dependent survival covariates are included for dynamic prediction. Default is FALSE.
`LOCFcovariate`	a vector of string with time-dependent survival covariates if `LOCF = TRUE`. Default is NULL.
`clongdata`	a long format data frame where time-dependent survival covariates are incorporated. Default is NULL.
`...`	Further arguments passed to or from other methods.

Value

a list of matrices with conditional probabilities for subjects.

Author(s)

Shanpeng Li <`lishanpeng0913@ucla.edu`>

See Also

`JMMLSM`, `survfitJMMLSM`

`PEJMMLSM` *A metric of prediction accuracy of joint model by comparing the predicted risk with the counting process.*

Description

A metric of prediction accuracy of joint model by comparing the predicted risk with the counting process.

Usage

```
PEJMMLSM(
  seed = 100,
  object,
  landmark.time = NULL,
  horizon.time = NULL,
  obs.time = NULL,
  method = c("Laplace", "GH"),
  quadpoint = NULL,
  maxiter = 1000,
  n.cv = 3,
  survinitial = TRUE,
  opt = "nlminb",
  initial.para = FALSE,
  LOCF = FALSE,
  LOCFcovariate = NULL,
  clongdata = NULL,
  ...
)
```

Arguments

`seed`	a numeric value of seed to be specified for cross validation.
`object`	object of class 'JMMLSM'.
`landmark.time`	a numeric value of time for which dynamic prediction starts..
`horizon.time`	a numeric vector of future times for which predicted probabilities are to be computed.
`obs.time`	a character string of specifying a longitudinal time variable.
`method`	estimation method for predicted probabilities. If `Laplace`, then the empirical empirical estimates of random effects is used. If `GH`, then the pseudo-adaptive Gauss-Hermite quadrature is used.
`quadpoint`	the number of pseudo-adaptive Gauss-Hermite quadrature points if `method = "GH"`.
`maxiter`	the maximum number of iterations of the EM algorithm that the function will perform. Default is 10000.
`n.cv`	number of folds for cross validation. Default is 3.
`survinitial`	Fit a Cox model to obtain initial values of the parameter estimates. Default is TRUE.
`opt`	Optimization method to fit a linear mixed effects model, either nlminb (default) or optim.
`initial.para`	Initial guess of parameters for cross validation. Default is FALSE.
`LOCF`	a logical value to indicate whether the last-observation-carried-forward approach applies to prediction. If `TRUE`, then `LOCFcovariate` and `clongdata` must be specified to indicate which time-dependent survival covariates are included for dynamic prediction. Default is FALSE.
`LOCFcovariate`	a vector of string with time-dependent survival covariates if `LOCF = TRUE`. Default is NULL.
`clongdata`	a long format data frame where time-dependent survival covariates are incorporated. Default is NULL.
`...`	Further arguments passed to or from other methods.

Value

a list of matrices with conditional probabilities for subjects.

Author(s)

Shanpeng Li <`lishanpeng0913@ucla.edu`>

See Also

`JMMLSM`, `survfitJMMLSM`

`plot.survfitJMMLSM` *Plot conditional probabilities for new subjects*

Description

Plot conditional probabilities for new subjects. If `CompetingRisk = FALSE`, print the survival probabilities. Otherwise, print the cumulative incidence probabilities for each failure type.

Usage

```
## S3 method for class 'survfitJMMLSM'
plot(
  x,
  include.y = FALSE,
  xlab = NULL,
  ylab = NULL,
  xlim = NULL,
  ylim.long = NULL,
  ylim.surv = NULL,
  ...
)
```

Arguments

`x`	x of class `survfitJMMLSM`.
`include.y`	include longitudinal responses of this subject versus time. Default is FALSE.
`xlab`	X axis label.
`ylab`	Y axis label.
`xlim`	X axis support.
`ylim.long`	Y axis support for the longitudinal outcome.
`ylim.surv`	Y axis support for the event / survival probability.
`...`	further arguments passed to or from other methods.

Value

plots of conditional probabilities over different pre-specified time points for subjects. If single failure type, then survival probabilities will be returned. Otherwise, cumulative incidence probabilities for each failure type will be returned.

Author(s)

Shanpeng Li <`lishanpeng0913@ucla.edu`>

See Also

`survfitJMMLSM`

`print.JMMLSM` *Print JMMLSM*

Description

Print contents of JMMLSM object.

Usage

```
## S3 method for class 'JMMLSM'
print(x, digits = 4, ...)
```

Arguments

`x`	Object of class 'JMMLSM'.
`digits`	number of digits of decimal to be printed.
`...`	Further arguments passed to or from other methods.

Value

a summary of data, joint model, log likelihood, and parameter estimates.

Author(s)

Shanpeng Li

See Also

`JMMLSM`

`print.survfitJMMLSM` *Print survfitJMMLSM*

Description

Print survfitJMMLSM

Usage

```
## S3 method for class 'survfitJMMLSM'
print(x, ...)
```

Arguments

`x`	x of class 'survfitJMMLSM'.
`...`	Further arguments passed to or from other methods.

Value

a list of matrices with conditional probabilities for subjects.

Author(s)

Shanpeng Li <`lishanpeng0913@ucla.edu`>

See Also

`JMMLSM`, `survfitJMMLSM`

`summary` *Summaries of evaluation metrics for joint models*

Description

Produce result summaries of a joint model fit.

Usage

```
## S3 method for class 'AUCJMMLSM'
summary(object, digits = 4, ...)

## S3 method for class 'ConcordanceJMMLSM'
summary(object, digits = 4, ...)

## S3 method for class 'JMMLSM'
summary(object, process = c("longitudinal", "survival"), digits = 4, ...)

## S3 method for class 'MAEQJMMLSM'
summary(object, digits = 3, ...)

## S3 method for class 'PEJMMLSM'
summary(object, error = c("MAE", "Brier"), ...)
```

Arguments

`object`	object of class ’PEJMMLSM’.
`digits`	number of decimal points to be rounded.
`...`	Further arguments passed to or from other methods.
`process`	for which model (i.e., longitudinal model or survival model) to extract the estimated coefficients.
`error`	a character string that specifies the loss function.

Value

a list of matrices with conditional probabilities for subjects.

a list of matrices with conditional probabilities for subjects.

A table to summarize the model results.

a list of matrices with conditional probabilities for subjects.

a list of matrices with conditional probabilities for subjects.

Author(s)

Shanpeng Li <`lishanpeng0913@ucla.edu`>

See Also

`JMMLSM`, `survfitJMMLSM`

`JMMLSM`, `survfitJMMLSM`

`JMMLSM`

`JMMLSM`, `survfitJMMLSM`

`JMMLSM`, `survfitJMMLSM`

`survfitJMMLSM` *Prediction in Joint Models*

Description

This function computes the conditional probability of surviving later times than the last observed time for which a longitudinal measurement was available.

Usage

```
survfitJMMLSM(
  object,
  seed = 100,
  ynewdata = NULL,
  cnewdata = NULL,
  u = NULL,
  Last.time = NULL,
  obs.time = NULL,
  LOCF = FALSE,
  LOCFcovariate = NULL,
  clongdata = NULL,
  method = c("Laplace", "GH"),
  quadpoint = NULL,
  ...
)
```

Arguments

`object`	an object inheriting from class `JMMLSM`.
`seed`	a random seed number to proceed non-parametric bootstrap. Default is 100.
`ynewdata`	a data frame that contains the longitudinal and covariate information for the subjects for which prediction of survival probabilities is required.
`cnewdata`	a data frame that contains the survival and covariate information for the subjects for which prediction of survival probabilities is required.
`u`	a numeric vector of times for which prediction survival probabilities are to be computed.
`Last.time`	a numeric vector or character string. This specifies the known time at which each of the subjects in cnewdata was known to be alive. If NULL, then this is automatically taken as the survival time of each subject. If a numeric vector, then it is assumed to be greater than or equals to the last available longitudinal time point for each subject. If a character string, then it should be a variable in cnewdata.
`obs.time`	a character string of specifying a longitudinal time variable in ynewdata.
`LOCF`	a logical value to indicate whether the last-observation-carried-forward approach applies to prediction. If `TRUE`, then `LOCFcovariate` and `clongdata` must be specified to indicate which time-dependent survival covariates are included for dynamic prediction. Default is FALSE.
`LOCFcovariate`	a vector of string with time-dependent survival covariates if `LOCF = TRUE`. Default is NULL.
`clongdata`	a long format data frame where time-dependent survival covariates are incorporated. Default is NULL.
`method`	a character string specifying the type of probability approximation; if `Laplace`, then a first order estimator is computed. If `GH`, then the standard Gauss-Hermite quadrature is used instead.
`quadpoint`	number of quadrature points used for estimating conditional probabilities when `method = "GH"`. Default is NULL. If `method = "GH"`, then 15 is used.
`...`	further arguments passed to or from other methods.

Value

a list of matrices with conditional probabilities for subjects.

Author(s)

Shanpeng Li `<lishanpeng0913@ucla.edu>`

See Also

`JMMLSM`

`vcov` *Variance-covariance matrix of the estimated parameters for joint models*

Description

Extract variance-covariance matrix for joint models.

Usage

```
## S3 method for class 'JMMLSM'
vcov(object, ...)
```

Arguments

`object`	an object inheriting from class `JMMLSM`.
`...`	further arguments passed to or from other methods.

Value

a matrix of variance covariance of all parameter estimates.

Author(s)

Shanpeng Li `<lishanpeng0913@ucla.edu>`

See Also

`JMMLSM`

`ydata` *Simulated longitudinal data*

Description

The `ydata` data frame has 1353 rows and 6 columns.

Usage

```
data(ydata)
```

Format

This data frame contains the following columns:

`ID`	patient identifier.
`Y`	response variable.
`time`	visit time.
`Z1`	treatment indicator. `0` denotes the placebo group and `1` the treatment group.
`Z2`	continuous variable..
`Z3`	continuous variable..

Index