

# Package ‘PMLE4SCR’

June 8, 2026

**Type** Package

**Title** Pseudo Maximum Likelihood Estimation for Semi-Competing Risks  
Data

**Version** 0.1.0

**Description** Implements two-stage pseudo maximum likelihood estimation (PMLE) for copula-based regression models with semi-competing risks data. The marginal distributions are modeled by semiparametric transformation regression models, and the dependence between bivariate event times is specified by a parametric copula function. See Arachchige, Chen and Zhou (2025) <[doi:10.1007/s10985-024-09640-z](https://doi.org/10.1007/s10985-024-09640-z)> for details.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Imports** trust, dplyr, rlang, VineCopula

**Suggests** knitr, rmarkdown, kableExtra, SemiCompRisks

**RoxygenNote** 7.3.3

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Qian M. Zhou [aut],  
Md. Ismail Hossain [aut, cre]

**Maintainer** Md. Ismail Hossain <[mr2618@msstate.edu](mailto:mr2618@msstate.edu)>

**Repository** CRAN

**Date/Publication** 2026-06-08 18:20:19 UTC

## Contents

fitSPT . . . . .	2
MLE4SCR . . . . .	3
PMLE4SCR . . . . .	5
predictFitSPT . . . . .	7
<b>Index</b>	<b>9</b>

---

fitSPT

*Fit a semi-parametric transformation (SPT) regression model*


---

### Description

Fit a semiparametric transformation regression model, where the unspecified baseline function is estimated via nonparametric maximum likelihood estimation (NPMLE) method.

### Usage

```
fitSPT(data, time, status, formula = ~1, Gfun = "PH")
```

### Arguments

data	a data frame containing variables names specified in time, status, and formula.
time	a character string specifying the variable name in the data for the event time.
status	a character string specifying the variable name in the data for the censoring indicator (taking value 1 if the time is observed, and taking value 0 when it is censored).
formula	an object of class <code>formula</code> : a symbolic description of the model to be fitted.
Gfun	a character string specifying the link function, Gfun = "PH" (proportional hazards as default) or Gfun = "PO" (proportional odds) in the SPT model.

### Value

a list of the following components:

beta	a data frame containing the estimates, model-based standard errors (SE), and robust SEs for regression coefficients.
dLambda	a data frame containing the estimates, model-based SEs, and robust SEs for the jumps of the baseline function at observed time points.
Lambda	a data frame containing the estimates, model-based SEs, and robust SEs for the baseline function at observed time points.
varcov	a list of two elements: model, the model-based variance-covariance matrix for all regression and baseline function parameter estimators, and robust, the robust variance-covariance matrix for parameter estimators.
Psi.theta	a data frame containing the estimated asymptotic expansion of the NPMLE estimator.
call	a list containing the specified values of the input arguments.
convergence	a logical value indicating whether the maximization of the log-likelihood converges or not.
niter	an interger which is the number of iterations for the maximization of the log-likelihood.

**Examples**

```

data(BMT, package = "SemiCompRisks")
BMT$g <- factor(BMT$g, levels = c(2, 3, 1),
               labels = c("AML-low", "AML-high", "ALL"))
fitsPT(BMT, time = "T1", status = "delta1",
       formula = ~ g, Gfun = "PH")$beta

```

MLE4SCR

*Simultaneous maximum likelihood estimation (MLE) of copula-based regression models for semi-competing risks (SCR) data*

**Description**

Perform MLE to simultaneously estimate the parameters associated with a copula-based regression models.

**Usage**

```

MLE4SCR(
  data,
  time,
  death,
  status_time,
  status_death,
  T.fmla = ~1,
  D.fmla = ~1,
  Gfun = list(T = "PH", D = "PH"),
  copula.family,
  copula.control = list(link = NULL, formula = ~1),
  initial = NULL
)

```

**Arguments**

<code>data</code>	a data frame containing variables names specified in <code>time</code> , <code>death</code> , <code>status_time</code> , <code>status_death</code> , <code>T.fmla</code> , <code>D.fmla</code> , and <code>formula</code> of <code>copula.control</code> .
<code>time</code>	a character string specifying the variable name in the data for the time to a non-terminal event.
<code>death</code>	a character string specifying the variable name in the data for the time to a terminal event.
<code>status_time</code>	a character string specifying the variable name in the data for the censoring indicator for the non-terminal event (take value 1 if the non-terminal event time is observed, and taking value 0 if it is censored).

<code>status_death</code>	a character string specifying the variable name in the data for the censoring indicator for the terminal event (take value 1 if the terminal event time is observed, and taking value 0 if it is censored).
<code>T.fmla</code>	an object of class <code>formula</code> : a symbolic description of the regression model to be fitted for the marginal distribution of the non-terminal event time.
<code>D.fmla</code>	an object of class <code>formula</code> : a symbolic description of the regression model to be fitted for the marginal distribution of the terminal event time.
<code>Gfun</code>	a list of two components T and D, both character strings specifying the link function in the semiparametric transformation (SPT) regression model for the non-terminal and terminal event times, respectively: "PH" (proportional hazards as the default value) or "PO" (proportional odds).
<code>copula.family</code>	a character string: "Clayton", "Gumbel", "Frank", or "Gaussian" specifying the copula family to be used for the dependence between the bivariate event times.
<code>copula.control</code>	a list of two components: <code>link</code> (a character string: "identity", "log", or "log-1", specifying the link function for the regression model of the copula parameter; if <code>link = NULL</code> (default), the link function will be the default function for the specified copula family: "log" for Clayton, "identity" for Frank, "log-1" for Gumbel, and "tanh" for Gaussian), and <code>formula</code> (an object of class <code>formula</code> : a symbolic description of the regression model to be fitted for the copula parameter under the specified link function; if <code>formula = ~ 1</code> as default, the copula parameter is a constant)
<code>initial</code>	a numerical value or a vector of numerical values for the initial values of the copula parameter or the regression coefficients for the copula parameter. If <code>initial = NULL</code> (default), the initial value is set to be the one corresponding to Kendall's tau being 0.5.

## Value

a list of the following components:

`gamma`: a data frame containing the estimates and robust standard errors (SE) of the copula parameter or regression coefficients for the copula parameter.

`gamma.cov`: a matrix containing the variance or variance-covariance matrix of the copula parameter estimators.

`betaT`: a data frame containing the estimates and robust SEs of the regression coefficients for the marginal distribution of the non-terminal event time.

`dLambdaT`: a data frame containing the estimates and robust SEs of the jumps of the baseline function for the marginal distribution of the non-terminal event time.

`thetaT.cov`: a matrix containing the variance-covariance matrix of the nonterminal marginal parameter estimators.

`betaD`: a data frame containing the estimates and robust SEs of the regression coefficients for the marginal distribution of the terminal event time.

`dLambdaD`: a data frame containing the estimates and robust SEs of the jumps of the baseline function for the marginal distribution of the terminal event time.

`thetaD.cov`: a matrix containing the variance-covariance matrix of the terminal marginal parameter estimators.

**call:** a list containing the specified values of input arguments `time`, `death`, `status_time`, `status_death`, `T.fmla`, `D.fmla`, `copula.family`, and the following two components:

**copula.link:** a list containing three R functions: `"h.fun"` (the link function used for the copula parameter), `"dot.h.fun"` (the first-order derivative of `"h.fun"`), and `"ddot.h.fun"` (the second-order derivative of `"h.fun"`).

**copula.fmla:** the specified value of formula of the input argument `copula.control`.

**Par2Tau:** a list containing two R functions: `"tau.alpha"` (transformation from the copula parameter to Kendall's tau), and `"Dtau.alpha"` (the first-order derivative of `"tau.alpha"` function)

## Examples

```
data(BMT, package = "SemiCompRisks")
BMT$g <- factor(BMT$g, levels = c(2, 3, 1),
               labels = c("AML-low", "AML-high", "ALL"))
myfit <- MLE4SCR(BMT, time = "T2", death = "T1",
                status_time = "delta2", status_death = "delta1",
                T.fmla = ~ g, D.fmla = ~ g,
                copula.family = "Clayton",
                copula.control = list(link = NULL, formula = ~ g),
                initial = NULL)

myfit$gamma
myfit$betaT
```

---

PMLE4SCR

*Two-stage pseudo maximum likelihood estimation (PMLE) of copula-based regression models for semi-competing risks (SCR) data*

---

## Description

Perform two-stage PMLE to estimate the parameters associated with a copula-based regression models.

## Usage

```
PMLE4SCR(
  data,
  time,
  death,
  status_time,
  status_death,
  T.fmla = ~1,
  D.fmla = ~1,
  Gfun = list(T = "PH", D = "PH"),
  copula.family,
```

```

  copula.control = list(link = NULL, formula = ~1),
  initial = NULL
)

```

### Arguments

<code>data</code>	a data frame containing variables names specified in <code>time</code> , <code>death</code> , <code>status_time</code> , <code>status_death</code> , <code>T.fmla</code> , <code>D.fmla</code> , and <code>formula</code> of <code>copula.control</code> .
<code>time</code>	a character string specifying the variable name in the data for the time to a non-terminal event.
<code>death</code>	a character string specifying the variable name in the data for the time to a terminal event.
<code>status_time</code>	a character string specifying the variable name in the data for the censoring indicator for the non-terminal event (take value 1 if the non-terminal event time is observed, and taking value 0 if it is censored).
<code>status_death</code>	a character string specifying the variable name in the data for the censoring indicator for the terminal event (take value 1 if the terminal event time is observed, and taking value 0 if it is censored).
<code>T.fmla</code>	an object of class <code>formula</code> : a symbolic description of the regression model to be fitted for the marginal distribution of the non-terminal event time.
<code>D.fmla</code>	an object of class <code>formula</code> : a symbolic description of the regression model to be fitted for the marginal distribution of the terminal event time.
<code>Gfun</code>	a list of two components T and D, both character strings specifying the link function in the semiparametric transformation (SPT) regression model for the non-terminal and terminal event times, respectively: "PH" (proportional hazards as the default value) or "PO" (proportional odds).
<code>copula.family</code>	a character string: "Clayton", "Gumbel", "Frank", or "Gaussian" specifying the copula family to be used for the dependence between the bivariate event times.
<code>copula.control</code>	a list of two components: <code>link</code> (a character string: "identity", "log", or "log-1", specifying the link function for the regression model of the copula parameter; if <code>link = NULL</code> (default), the link function will be the default function for the specified copula family: "log" for Clayton, "identity" for Frank, "log-1" for Gumbel, and "tanh" for Gaussian), and <code>formula</code> (an object of class <code>formula</code> : a symbolic description of the regression model to be fitted for the copula parameter under the specified link function; if <code>formula = ~ 1</code> (default), the copula parameter is a constant)
<code>initial</code>	a numerical value or a vector of numerical values for the initial values of the copula parameter or the regression coefficients for the copula parameter. If <code>initial = NULL</code> (default), the initial value is set to be the one corresponding to Kendall's tau being 0.5.

### Value

a list of the following components:

**gamma:** a data frame containing the estimates and robust standard errors (SE) of the copula parameter or regression coefficients for the copula parameter.

- `gamma.cov`: a matrix containing the variance or variance-covariance matrix of the copula parameter estimators.
- `betaT`: a data frame containing the estimates and robust SEs of the regression coefficients for the marginal distribution of the non-terminal event time.
- `dLambdaT`: a data frame containing the estimates and robust SEs of the jumps of the baseline function for the marginal distribution of the non-terminal event time.
- `thetaT.cov`: a matrix containing the variance-covariance matrix of the nonterminal marginal parameter estimators.
- `betaD`: a data frame containing the estimates and robust SEs of the regression coefficients for the marginal distribution of the terminal event time.
- `dLambdaD`: a data frame containing the estimates and robust SEs of the jumps of the baseline function for the marginal distribution of the terminal event time.
- `thetaD.cov`: a matrix containing the variance-covariance matrix of the terminal marginal parameter estimators.
- `call`: a list containing the specified values of input arguments `time`, `death`, `status_time`, `status_death`, `T.fmla`, `D.fmla`, `copula.family`, and the following two components:
- `copula.link`: a list containing three R functions: "h.fun" (the link function used for the copula parameter), "dot.h.fun" (the first-order derivative of "h.fun"), and "ddot.h.fun" (the second-order derivative of "h.fun").
  - `copula.fmla`: the specified value of formula of the input argument `copula.control`.
- `Par2Tau`: a list containing two R functions: "tau.alpha" (transformation from the copula parameter to Kendall's tau), and "Dtau.alpha" (the first-order derivative of "tau.alpha" function)

## Examples

```
data(BMT, package = "SemiCompRisks")
BMT$g = factor(BMT$g, levels = c(2, 3, 1),
              labels = c("AML-low", "AML-high", "ALL"))
myfit = PMLE4SCR(BMT, time = "T2", death = "T1",
                status_time = "delta2", status_death = "delta1",
                T.fmla = ~ g, D.fmla = ~ g,
                copula.family = "Clayton",
                copula.control = list(link = NULL, formula = ~ g),
                initial = NULL)

myfit$gamma
myfit$betaT
```

**Description**

Computes survival probability estimates for new data using a fitted `fitSPT` object.

**Usage**

```
predictFitSPT(obj, newdata)
```

**Arguments**

`obj` an object of `fitSPT`.  
`newdata` a data frame containing the same variables in the formula of `fitSPT`.

**Value**

a list of the following components:

`surv` a data frame containing the estimate, model-based standard error (SE), and robust SE for the survival probabilities for a given data.

`Psi.surv` a data frame containing the estimated asymptotic expansion of the estimator of the survival probabilities.

**Examples**

```
data(BMT, package = "SemiCompRisks")
BMT$g <- factor(BMT$g, levels = c(2, 3, 1),
               labels = c("AML-low", "AML-high", "ALL"))
fit <- fitSPT(BMT, time = "T1", status = "delta1",
             formula = ~ g, Gfun = "PH")
predictFitSPT(fit, BMT)$surv
```

# Index

fitSPT, [2](#), [8](#)

formula, [2](#), [4](#), [6](#)

MLE4SCR, [3](#)

PMLE4SCR, [5](#)

predictFitSPT, [7](#)