

Package ‘SLCARE’

January 14, 2024

Title Semiparametric Latent Class Analysis of Recurrent Events

Version 1.1.0

Description

Efficient R package for latent class analysis of recurrent events, based on the semiparametric multiplicative intensity model by Zhao et al. (2022) <doi:10.1111/rssb.12499>. SLCARE returns estimates for non-functional model parameters along with the associated variance estimates and p-values. Visualization tools are provided to depict the estimated functional model parameters and related functional quantities of interest. SLCARE also delivers a model checking plot to help assess the adequacy of the fitted model.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.3

Depends R (>= 2.10)

LazyData true

Imports dplyr, ggplot2, magrittr, methods, nnet, reda, reReg, rlang,
stats, tidyr, utils

NeedsCompilation no

Author Qi Yu [aut, cre],
Limin Peng [aut]

Maintainer Qi Yu <qi.yu2@emory.edu>

Repository CRAN

Date/Publication 2024-01-14 07:50:05 UTC

R topics documented:

plot.SLCARE	2
predict.SLCARE	3
print.SLCARE	3
SimData	4
SLCARE	5
summary.SLCARE	7
Index	8

`plot.SLCARE`*Produce Plots for SLCARE*

Description

Generate cumulative baseline intensity function, estimated mean function, model checking plot for SLCARE.

Usage

```
## S3 method for class 'SLCARE'  
plot(x, type = c("ModelChecking", "mu0", "EstMeans"), ...)
```

Arguments

<code>x</code>	an object of class SLCARE.
<code>type</code>	the type of the plot.
<code>...</code>	other arguments.

Details

SLCARE provides visualization tools to depict the estimated functional model parameters and related functional quantities of interest. These tools include:

mu0: estimated cumulative baseline intensity function.

EstMeans: estimated mean function plot - The crude estimates for the class-specific mean functions of recurrent events

SLCARE also provides a tool to help assess the adequacy of the fitted model:

ModelChecking: model checking plot - A plot shows the comparison of the observed recurrent events versus the expected number of recurrent events. A major departure from the identity line may suggest a lack-of-fit of the assumed models.

Value

A ggplot object.

See Also

[SLCARE\(\)](#)

predict.SLCARE	<i>Predict Results for SLCARE</i>
----------------	-----------------------------------

Description

Calculate the posterior predicted number of recurrent events.

Usage

```
## S3 method for class 'SLCARE'
predict(object, integer = FALSE, ...)
```

Arguments

object	an object of class SLCARE.
integer	logicals. If FALSE (the default value), the function returns to float numbers.
...	other arguments.

See Also

[SLCARE\(\)](#)

print.SLCARE	<i>Print Results for SLCARE</i>
--------------	---------------------------------

Description

Print point estimates (Est), bootstrap standard error estimates (SE), initial estimates for the estimation algorithm (Init), convergence criterion (ConvergeLoss), latent class membership probability (ClassProb), predicted number of recurrent events (PostPredict), relative entropy of the fitted model (Entropy), p-value (pvalue) for SLCARE.

Usage

```
## S3 method for class 'SLCARE'
print(
  x,
  type = c("Est", "SE", "Init", "ConvergeLoss", "ClassProb", "PostPredict", "Entropy",
    "PValue"),
  ...
)
```

Arguments

`x` an object of class SLCARE.
`type` the type of the output.
`...` other arguments.

See Also

[SLCARE\(\)](#)

SimData

Simulated dataset for demonstration

Description

A simulated dataset perturbed from a real dataset with the following variables:

Usage

```
data(SimData)
```

Format

A data frame with 476 rows and 6 variables.

id subjects identification

start start time of the interval for the recurrent event.

stop ending time of the interval for the recurrent event; when time origin is 0 this variable also marks the recurrence or terminal/censoring time.

event recurrent event indicator; 1 if a recurrent event is observed.

x1 a binary baseline covariate.

x2 a continuous baseline covariate.

Description

Fit semiparametric latent class model for recurrent event.

Usage

```
SLCARE(  
  formula = "x1 + x2",  
  alpha = NULL,  
  beta = NULL,  
  data = data,  
  id_col = "id",  
  start_col = "start",  
  stop_col = "stop",  
  event_col = "event",  
  K = NULL,  
  gamma = 0,  
  max_epochs = 500,  
  conv_threshold = 0.01,  
  boot = NULL  
)
```

Arguments

formula	A string specifying the variables of interest to be involved in the regression, with the format of "x1 + x2".
alpha	initial values for alpha for estimation procedure (multinomial logistic regression model). This should be NULL or a numeric matrix. 'NULL' refers to obtain initial value with the proposed method.
beta	initial value for beta for estimation procedure (recurrent event model). This should be NULL or a numeric matrix. 'NULL' refers to obtain initial value with the proposed method.
data	a long-format Data-frame.
id_col	the unique identifier per subject.
start_col	start time of the interval for the recurrent event.
stop_col	ending time of the interval for the recurrent event.
event_col	The status indicator; 1 if a recurrent event is observed.
K	number of latent classes.
gamma	individual frailty. 0 represents the frailty equals 1 and k represents the frailty follows gamma(k,k).
max_epochs	maximum iteration epochs for the estimation procedure.

conv_threshold converge threshold for the estimation procedure.
boot a numeric value specifies the number of bootstraps for variance estimation.
When boot = NULL, variance estimation will not be performed.

Details

Model:

Suppose the recurrent events process is observed with the intensity function proposed in Zhao et al. (2022):

$$\lambda_i(t) = \sum_{k=1}^K I(\xi_i = k) \times \lambda_0(t) \times W_i \times \eta_{0,k} \times \exp(\tilde{Z}_i^\top \tilde{\beta}_{0,k})$$

where K is the number of latent classes in the whole population, ξ_i denotes the unobserved latent class membership, $\lambda_0(t)$ is an unspecified, continuous, nonnegative baseline intensity function shared by all latent classes, C_i is the subject specific censoring time, \tilde{Z}_i is the time-independent covariates, W_i is a positive subject-specific latent variable independent of $(\xi_i, \tilde{Z}_i, C_i)$.

The distribution of the latent class membership ξ_i is modeled by a logistic regression model:

$$P(\xi_i = k | \tilde{Z}_i) = p_k(\alpha_0, \tilde{Z}_i) \doteq \frac{\exp(\tilde{Z}_i^\top \alpha_{0,k})}{\sum_{k=1}^K \exp(\tilde{Z}_i^\top \alpha_{0,k})}, \quad k = 1, \dots, K$$

SLCARE is build for introducing a robust and flexible algorithm to carry out Zhao et al. (2022)'s latent class analysis method for recurrent event data described above. The detailed discussion of the proposed estimation algorithms can be found in the paper "SLCARE: An R package for Semiparametric Latent Class Analysis of Recurrent Events" (in preparation).

Initial Values:

The proposed estimating algorithm needs an input of initial values for $\hat{\beta}$ and $\hat{\alpha}$. SLCARE allows users to specify the initial values for the estimation algorithm by their own choice. SLCARE also provide an automated initializer which obtains the initial values using a combination of K-means clustering, multinomial regression and Wang et al. (2001)'s multiplicative intensity model. The detailed discussion of the proposed estimation algorithms can be found in the paper "SLCARE: An R package for Semiparametric Latent Class Analysis of Recurrent Events" (in preparation).

Specify the number of latent classes and individual frailty:

SLCARE allows the frailty distribution to be $W = 1$ or W follows a distribution that is parameterized as Gamma(k,k). These choices of frailty distributions cover a variety of density forms. Suggested by Zhao et al. (2022), users can choose the distribution of individual frailty and the number of latent classes based on the model entropy provided by SLCARE. An example of model selection can be found in the paper "SLCARE: An R package for Semiparametric Latent Class Analysis of Recurrent Events" (in preparation).

Examples

```
data(SimData)

# fit SLCARE with K = 2, formula = "x1 + x2" and default settings for other arguments
model1 <- SLCARE(data = SimData, K = 2, formula = "x1 + x2")
```

```
# summary results
summary(model1, digits = 3)

# generate model checking plot
plot(model1, type = "ModelChecking")

# plot estimated cumulative baseline intensity function
plot(model1, type = "mu0")

# generate estimated mean function plot
plot(model1, type = "EstMeans")

# check class membership probabilities of the 6th - 10th subjects in SimData
print(model1, type = "ClassProb")[6:10,]

# check the predicted number of recurrent events of the 6th - 10th subjects in SimData
print(model1, type = "PostPredict")[6:10,]

# check the change in parameter estimates in the last iteration
print(model1, type = "ConvergeLoss")
```

summary.SLCARE

Summary Results for SLCARE

Description

Summary results for SLCARE object including regression coefficients, corresponding standard error estimates and relative entropy of the fitted model.

Usage

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

Arguments

object	an object of class SLCARE.
digits	minimal number of significant digits.
...	other arguments.

See Also

[SLCARE\(\)](#)

Index

plot.SLCARE, [2](#)
predict.SLCARE, [3](#)
print.SLCARE, [3](#)

SimData, [4](#)
SLCARE, [5](#)
SLCARE(), [2-4](#), [7](#)
summary.SLCARE, [7](#)