Package ‘smoothROCtime’

November 14, 2018

Type Package

Title Smooth Time-Dependent ROC Curve Estimation

Version 0.1.0

Author Susana Diaz-Coto <U0266718@uniovi.es>

Maintainer Susana Diaz-Coto <U0266718@uniovi.es>

Imports ks

Suggests KMsurv,lattice, survival

Description Computes smooth estimations for the Cumulative/Dynamic and Incident/Dynamic ROC curves, in presence of right censorship, based on the bivariate kernel density estimation of the joint distribution function of the Marker and Time-to-event variables.

License GPL

LazyData TRUE

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-11-14 10:40:03 UTC

R topics documented:

smoothROCtime-package *Smooth Time-Dependent ROC Curve Estimation*

Description

Computes smooth estimations for the Cumulative/Dynamic and Incident/Dynamic ROC curves, in presence of right censorship, based on the bivariate kernel density estimation of the joint distribution function of the Marker and Time-to-event variables.

Details

- funcen: Bivariate kernel density estimation of the joint density function of the $(marker, time-to-event)$ variable.
- stRoc: Smooth estimations for Cumulative/Dynamic and Incident/Dynamic ROC curves.
- plot.sROCt: Plots of Cumulative/Dynamic and Incident/Dyanmic ROC curve estimations.

Author(s)

Susana Diaz-Coto <UO266718@uniovi.es>

Maintainer: Susana Diaz-Coto <UO266718@uniovi.es>

References

P. Martinez-Camblor and J. C. Pardo-Fernandez. Smooth time-dependent receiver operating characteristic curve estimators. Statistical Methods in Medical Research, 27(3):651-674, 2018. https://doi.org/10.1177/0962280217740786.

P. Martinez-Camblor, G. F-Bayon, and S. Perez-Fernandez. Cumulative/dynamic ROC curve estimation. JOURNAL of Statistical Computation and Simulation, 86(17):3582-3594, 2016. https://doi.org/10.1080/00949655.2016.1175442.

T. Duong. Bandwidth matrices for multivariate kernel density estimation. Ph.D. Thesis, University of Western, Australia, 2004.

L. Li, T. Green, and B. Hu. A simple method to estimate the time-dependent receiver operating characteristic curve and the area under the curve with right censored data. Statistical Methods in Medical Research, 27(8), 2016. https://doi.org/10.1177/0962280216680239.

See Also

CRAN package ks is used in this package.

funcen — *Bivariate kernel density estimation under random censoring*

Description

Computes the kernel density estimation of the bivariate vector $(marker, time-to-event)$ with the time-to-event variable subject to right censorship, according to the procedure exposed in https://doi.org/10.1177/0962280217740786.

Usage

```
funcen(data, H, bw, adj, ...)
```

Arguments

`data`	matrix with three columns: time-to-event, censoring status (0=censored/1=uncensored) and marker.
`H`	2x2 bandwidth matrix when it is specified in an explicit way.
`bw`	method for computing the bandwidth matrix. Most of the methods included in the `kde` function can be used: Hpi, Hpi.diag, Hlscv, Hlscv.diag, Hbcv, Hbcv.diag, Hscv, Hscv.diag, Hucv and Hucv.diag. Other considered methods are naive.pdf (diag(N^-1/5, N^-1/5)^2) and naive.cdf (diag(N^-1/3, N^-1/3)^2), where N is the sample size.
`adj`	adjusment parameter for calculating the bandwidth matrix. Default value 1.
`...`	`kde` function arguments can also be used for specifying the way in which the kernel density function estimation should be computed.

Details

The matrix of bandwidths can be defined by using H=matrix() or automatically selected by the method indicated in bw.

Given the matrix of bandwidths, H, the argument adj modifies it and the final computed matrix is adj^2 H.

If H is missing, the naive.pdf method is used for obtaining the kernel density estimation.

Function `funcen` generates, from the original set of data, a collection of pseudodata through an iterative weights allocation process, with two main goals: keep the information from the censored observations represented in the sample and prepare data so they can be used as incoming parameters in the `kde` function included in the **ks** package. A weighted kernel density estimation is therefore finally computed.

There should be at least two uncensored observations for computing the density estimation.

Omitted parameters are considered to be the default ones in the `kde` function.

Value

An object of class kde is returned. It is a list where the most relevant values are:

- `x` matrix containing the pseudodata values. It has two columns: marker and time-to-event.
- `eval.points` list of points where the bivariate kernel estimation is calculated.
- `estimate` values of the density estimation.
- `H` bandwidth matrix.
- `names` variable names.
- `w` weights calculated by the function and allocated to pseudodata.

References

P. Martinez-Camblor and J. C. Pardo-Fernandez. Smooth time-dependent receiver operating characteristic curve estimators. Statistical Methods in Medical Research, 27(3):651-674, 2018. https://doi.org/10.1177/0962280217740786.

T. Duong. Bandwidth matrices for multivariate kernel density estimation. Ph.D. Thesis, University of Western, Australia, 2004. http://www.mvstat.net/tduong.

Examples

```
library(smoothROCtime)
require(KMsurv)
require(lattice)

data(kidtran)

# Preparing data: a logarithmic transformationof the time-to-event variable is made
  DT <- cbind(log(kidtran$time),kidtran$delta,kidtran$age)
  n  <-length(log(kidtran$time))

# Directly definition of the bandwidth matrix
  H  <- diag((c(sd(kidtran$age),sd(log(kidtran$time)))*n^(-0.2))^2)

# Kernel density function estimation
  density <- funcen(data=DT,H=H)

# Plot graphics
  wireframe(density$estimate, row.values=density$eval.points[[1]],
            column.values=density$eval.points[[1]],zlab="density")
  contour(x=density$eval.points[[1]],
          y=density$eval.points[[2]],
          z=density$estimate,
          ylim=c(6,10))
```

`plot.sROCt` *Plots of time-dependent ROC curve estimations*

Description

Plots of both Cumulative and Incident/Dynamic ROC curve estimations, provided by function `stRoc`.

Usage

```
## S3 method for class 'sROCt'
plot(x, tcr, xlab, ylab, type = "l", lwd = 5, ...)
```

Arguments

`x`	object of class sROCt generated with `stRoc` function and containing the estimations of the time-dependent ROC curves for one single point or a vector of points.
`tcr`	type of time-dependent ROC curve estimation that will be plotted: • "C" for **C**umulative/Dynamic, • "I" for **I**ncident/Dynamic, • "B" for **B**oth time-dependent ROC curve estimations.
`xlab`	a tittle for the x axis. The default value is "False - Positive Rate".
`ylab`	a tittle for the y axis. The default value is "True - Negative Rate".
`type`	what type of plot is going to be drawn. The default value is "l" and a line will be plotted.
`lwd`	line width. As a default value "5" is taken.
`...`	plot function arguments can also be used for customizing the plot.

Details

Parameter `tcr` is mandatory with no default values. If a "B" is indicated and the sROCt object placed as `x` parameter contains only one type of time-dependent ROC curve estimation, an error message will be returned. Another error message will appear in case of placing either "C" or "I" when the sROCt object does not contain the suitable ROC curve estimation.

When one single type of ROC curve estimation is chosen, one graphic will be drawn for each point of time in the sROCt object, having as many independent plots as number of points of time. Graphic parameters like axis labels or line width will be the same for all the plots.

In case of choosing both time-dependent ROC curve estimations, they will be plotted in a single graphic for each point of time in sROCt object. As before, we will have as many independent plots as points of time and the graphic parameters will be the same in all plots.

Examples

```
library(smoothROCtime)
require(survival)

# Monoclonal Gammapothy of Undetermined Significance dataset
  data(mgus)

# Time-to-event
  time   <- ifelse(is.na(mgus$pctime), mgus$futime,mgus$pctime)

# Status
  status <- ifelse(is.na(mgus$pctime), 0, 1)

# Preparing data
  DT <-as.data.frame(cbind(log(time), status, mgus$alb))
  colnames(DT) <- c("futime", "pcm", "alb")
  dta <-  na.omit(cbind(DT$futime, DT$pcm, -DT$alb))

# Point of Time
  t10 <- log(10*365.25) # ten years in logarithm scale

# Cumulative/Dynamic and Incident dynamic ROC curve estimations at t=10 years
  rcu <- stRoc(data=dta, t=t10, tcr="B", meth = "1", verbose=TRUE)

# Plots of both ROC curve estimations
  plot(rcu, tcr="B", frame=FALSE)
```

---

stRoc *Smooth Time-dependent ROC curve estimations*

Description

Provides smooth estimations of **Cumulative/Dynamic (C/D)** and **Incident/Dynamic (I/D)** ROC curves in presence of rigth censorship and the corresponding Areas Under the Curves (AUCs), at a single point of time or a vector of points.

- The function computes two different procedures to obtain smooth estimations of the **C/D ROC curve**. Both are based on the kernel density estimation of the joint distribution function of the marker and time-to-event variables, provided by `funcen` function. The first method, to which we will refere as **smooth method**, is carried out according to the methodology proposed in https://doi.org/10.1177/0962280217740786. The second one uses this estimation of the joint density function of the variables marker and time-to-event for computing the weights or probabilities allocated to censored observations (undefined individuals) in https://doi.org/10.1080/00949655.2016.1175442 and https://doi.org/10.1177/0962280216680239. It will be referred as **p-kernel** method.
- In case of the **I/D ROC curve**, a smooth approximation procedure (**smooth method**) is computed based as well on the kernel density estimation of the joint distribution function of the marker and time-to-event variables proposed in https://doi.org/10.1177/0962280217740786

Usage

```
stRoc(data, t, H, bw, adj, tcr, meth, ...)
```

Arguments

`data`	matrix of data values with three columns: time-to-event, censoring status (0=censored/1=uncensored) and marker.
`t`	point of time or vector of points where the time-dependent ROC curve is estimated.
`H`	2x2 bandwidth matrix.
`bw`	procedure for computing the bandwidth matrix. Most of the methods included at the kde function can be used: Hpi, Hpi.diag, Hlscv, Hlscv.diag, Hbcv, Hbcv.diag, Hscv, Hscv.diag, Hucv and Hucv.diag. Other considered methods are naive.pdf (diag(N^-1/5, N^-1/5)^2) and naive.cdf (diag(N^-1/3, N^-1/3)^2), where N is the sample size.
`adj`	adjusment parameter for calculating the bandwidth matrix. Default value 1.
`tcr`	type of time-dependent ROC curve estimation that will be estimated: • “C” for **C**umulative/Dynamic, • “I” for **I**ncident/Dynamic, • “B” for **B**oth time-dependent ROC curve estimations.
`meth`	method for computing the estimation of the C/D ROC curve.The suitable values are: • “1” for the smooth method, • “2” for the p-kernel method. As default value the smooth method is taken.
`...`	`kde` function arguments can be used for estimating the bivariate kernel density function.

Details

Function `funcen` is called from each execution of function `stRoc`, in order to compute the kernel density estimation of the joint distribution of the (Marker, Time-to-event) variable, therefore, the input parameters in `funcen` are input parameters as well in `stRoc` and the same considerations apply.

The matrix of bandwidths can be defined by using H=matrix() or automatically selected by the method indicated in `bw`.

Given the matrix of bandwidths, H, the argument `adj` modifies it and the final matrix is adj^2 H.

If `H` is missing, the naive.pdf method is used.

If `tcr` is missing the C/D ROC curve estimation will be computed with the method indicated in `meth`.

If no value has been placed in `meth` the smooth method will be used. The I/D ROC curve estimation will be always computed with the smooth method.

Value

An object of class `sROCt` is returned. It is a list with the following values:

- `th` considered thresholds for the marker.
- `FP` false-positive rate calculated at each point in `th`.
- `TP` true-positive rate estimated at each point in `th`.
- `p` points where the time-dependent ROC curve is evaluated.
- `R` time-dependent ROC curve values computed at `p`.
- `t` time/s at which each time-dependent ROC curve estimation is computed. Each point ot time will appear as many times as the length of the vector of points `p`.
- `auc` area under the corresponding time-dependent ROC curve estimation. As in the previous case, each value appears as many times as the length of the vector of points `p`.
- `tcr` type of time-dependent ROC curve estimation computed,
  - "C" - **C**umulative/Dynamic.
  - "I" - **I**ncident/Dynamic.

  For each computed time-dependent ROC curve estimation this value is repeated as many times as the length of `p`.
- `Pi` probabilities calculated for the individuals in the sample if the p-kernel method has been used for the estimation of the C/D ROC curve. This element is a matrix with the following columns:
  - `time` - single point of time at which the estimation each the C/D ROC curve has been computed.
  - `obvt` - observed times for the individuals in the sample.
  - `p` - estimations of the probabilities computed and allocated to each subject.

References

P. Martinez-Camblor and J. C. Pardo-Fernandez. Smooth time-dependent receiver operating characteristic curve estimators. Statistical Methods in Medical Research, 27(3):651-674, 2018.https://doi.org/10.1177/0962280217740786.

P. Martinez-Camblor, G. F-Bay?n, and S. P?rez-Fern?ndez. Cumulative/dynamic ROC curve estimation. JOURNAL of Statistical Computation and Simulation, 86(17):3582-3594, 2016. https://doi.org/10.1080/00949655.2016.1175442.

L. Li, T. Green, and B. Hu. A simple method to estimate the time-dependent receiver operating characteristic curve and the area under the curve with right censored data. Statistical Methods in Medical Research, 27(8), 2016. https://doi.org/10.1177/0962280216680239.

T. Duong. Bandwidth matrices for multivariate kernel density estimation. Ph.D. Thesis, University of Western, Australia, 2004. http://www.mvstat.net/tduong.

Examples

```
library(smoothROCtime)
require(KMsurv)

data(kidtran)

# Preparing data: a logarithmic transformation of the time-to-event variable is made
  DT <- cbind(log(kidtran$time),kidtran$delta,kidtran$age)

# Point of Time
  t5 <- log(5*365.25) # five years in logarithm scale

# Cumulative/dynamic ROC curve estimation
  rcd <- stRoc(data=DT, t=t5, bw="Hpi", tcr="C", meth=2)

# Plot graphic
  plot(rcd$p, rcd$ROC, type="l", lwd=5, main="C/D ROC",xlab="FPR",ylab="TPR")
  lines(c(0,1),c(0,1),lty=2,col="gray")
```

Index