Package 'tdROC'

December 8, 2023

Type Package

Title tdROC: Nonparametric Estimation of Time-Dependent ROC, Brier Score, and Survival Differnce from Right Censored Time-to-Event Data with or without Competing Risks

Version 2.0

Maintainer Xiaoyang Li <xli35@mdanderson.org>

Author Xiaoyang Li <xli35@mdanderson.org>
Zhe Yin <zhe.yin@uth.tmc.edu>
Liang Li <LLi15@mdanderson.org>

Description The tdROC package facilitates the estimation of time-dependent ROC (Receiver Operating Characteristic) curves and the Area Under the time-dependent ROC Curve (AUC) in the context of survival data, accommodating scenarios with right censored data and the option to account for competing risks. In addition to the ROC/AUC estimation, the package also estimates time-dependent Brier score and survival difference. Confidence intervals of various estimated quantities can be obtained from bootstrap. The package also offers plotting functions for visualizing time-dependent ROC curves.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R(>= 3.6.0)

Imports survival (>= 3.4),
graphics (>= 4.2.2),
stats (>= 4.2.2),
Rcpp (>= 1.0.10),
magrittr (>= 2.0.3)

LinkingTo Rcpp (>= 1.0.10)

NeedsCompilation yes

RoxygenNote 7.2.3

References Li L, Greene T, Hu B. A simple method to estimate the time-dependent receiver operating characteristic curve and the area under the curve with right censored data. Stat Methods Med Res. 2018;27(8):2264-2278. doi:10.1177/0962280216680239
Wu C, Li L. Quantifying and estimating the predictive accuracy for censored time-to-event data with competing risks. Stat Med. 2018;37(21):3106-3124. doi:10.1002/sim.7806

R topics documented:

`mayo` *Example data: Mayo Data*

Description

This example dataset is included for illustration. The Mayo PBC data is publicly available and has been used in many statistical researches (e.g., Zheng and Heagerty 2005). This example dataset is a subset of the full PBC data.

Usage

```
data(mayo)
```

Format

A data frame with 312 observations and 4 variables:

`time`: event time or censoring time

`censor` : censoring indicator.

`mayoscore4` and `mayoscore5`: derived from 4 and 5 covariates respectively.

References

Heagerty, P. J., & Zheng, Y. (2005). Survival model predictive accuracy and ROC curves. Biometrics, 61(1), 92-105.

Examples

```
data(mayo)
head(mayo)
```

Melano *Example data: Malignant Melanoma Data*

Description

This example dataset is included for illustration. The Melano data is publicly available. In 1962-1977, 205 patients with malignant melanoma (skin cancer) had a radical operation performed at an academic medical center. At the end of the follow-up, 57 died from cancer, 14 died from other causes, and the other 134 patients were alive (censored). This example dataset illustrates the prediction accuracy of competing risk outcomes with baseline age and tumor thickness.

Usage

```
data(Melano)
```

Format

A data frame with 312 observations and 4 variables: time (event time/censoring), time), censor (censoring mayoscore4, mayoscore5. The two scores are derived from 4 and 5 covariates respectively.

References

Andersen, P. K. , & Skovgaard, L. T. (2010). Regression with linear predictors. New York, NY: Springer.

Examples

```
data(Melano)
head(Melano)
```

plot_tdROC *Plot the time-dependent ROC curve*

Description

This function reads in object returned by `tdROC()` and plot ROC curve for it.

Usage

```
plot_tdROC(
  x,
  lwd = 2,
  xlab = "1-specificity",
  ylab = "sensitivity",
  xlim = c(0, 1),
  ylim = c(0, 1),
  main = "ROC curve",
  col = "black",
  abline = T,
  ...
)
```

Arguments

x	the object returned by tdROC().
lwd	user-specified line width. Default is 2.
xlab	user-specified label for x-axis. Default is "1-specificity".
ylab	user-specified label for y-axis. Default is "sensitivity".
xlim	user-specified limit for x axis. Default is c(0,1).
ylim	user-specified limit for y axis. Default is c(0,1).
main	user-specified title for the plot. Default is "ROC curve"
col	user-specified color for ROC curve. Defualt is "black".
abline	user-specified reference diagnol line. Default is True.
...	for future methods

Value

Returns a plot of ROC curve. If the tdROC object comes with bootstrap result, then the ROC curve will be plotted with confidence interval.

Examples

```
library(survival)
data(mayo)
dat <- mayo[, c("time", "censor", "mayoscore5")]
fm <- tdROC(
  X = dat$mayoscore5, Y = dat$time, delta = dat$censor,
  tau = 365 * 6, span = 0.1, nboot = 0, alpha = 0.05, n.grid = 1000, cut.off = 5:9
)
# plot the object "fm" from tdROC()
plot_tdROC(fm)
```

---

plot_tdROC_cr *Plot the time-dependent ROC curve with competing risk*

Description

This function reads in object returned by tdROC.cr() and plot ROC curve for it.

Usage

```
plot_tdROC_cr(
  x,
  lwd = 2,
  xlab = "1-specificity",
  ylab = "sensitivity",
  xlim = c(0, 1),
  ylim = c(0, 1),
  col = c("red", "blue"),
  main = "ROC curve",
  abline = T,
  ...
)
```

Arguments

x	the object returned by tdROC.cr().
lwd	user-specified line width. Default is 2.
xlab	user-specified label for x-axis. Default is "1-specificity".
ylab	user-specified label for y-axis. Default is "sensitivity".
xlim	user-specified limit for x axis. Default is c(0,1).
ylim	user-specified limit for y axis. Default is c(0,1).
col	user-specified color for ROC curve. Defualt is "c("red", "blue")" for the primary event and competing event.
main	user-specified title for the plot. Default is "ROC curve"
abline	user-specified reference diagnol line. Default is True.
...	for future methods

Value

Returns several plots of ROC curve. For competing risk data, there are two definitions of controls introduced by Zheng et al, which was listed below

$$\text{Definition A:Case}k : T \leq \tau, \delta = k; \text{Control}_A : (T > \tau) \cup (T \leq \tau \cap \delta \neq k)$$

$$\text{Definition B:Case}k : T \leq \tau, \delta = k; \text{Control}_B : (T > \tau)$$

For more details about above two definitions, please read details of function tdROC.cr. If the tdROC.cr object comes without bootstrap result, the ROC curve for above two definitions will be plotted together and indicated by the specified col. If the tdROC.cr object with bootstrap result, one more ROC curve with confidence interval will be plotted for each definition.

References

Zheng Y, Cai T, Jin Y, Feng Z. Evaluating prognostic accuracy of biomarkers under competing risk. Biometrics. 2012;68(2):388-396. doi:10.1111/j.1541-0420.2011.01671.x

Examples

```
library(survival)
data(Melano)
tdROC.cr_res <- tdROC.cr(
  X = Melano$thick, Y = Melano$time,
  delta = Melano$status, tau = 1800, nboot = 100
)
plot_tdROC_cr(tdROC.cr_res)
```

tdROC *Estimate time-dependent prediction accuracy measures, including the ROC, AUC, Brier score, and survival difference, with right-censored survival data.*

Description

This is a core function of the 'tdROC' package. It uses the nonparametric weights proposed by Li (Li et al., 2015) to estimate a number of time-dependent prediction accuracy measures for right-censored survival outcomes, including ROC curve, AUC, Brier score, and survival difference. For each measure, the variance can be estimated through bootstrap resampling.

Usage

```
tdROC(
  X,
  Y,
  delta,
  tau,
  span = 0.1,
  h = NULL,
  type = "uniform",
  n.grid = 1000,
  X.min = NULL,
  X.max = NULL,
  cut.off = NULL,
  nboot = 0,
  alpha = 0.05,
  epsilon = NULL,
  method = "both",
  output = "both"
)
```

Arguments

X	a numeric vector of risk score in the same length as Y and delta, such as biomarker or predicted probability. A higher value indicates higher risk of the event. The calibration results (Brier score, survival difference) are applicable only when the risk score has the predicted probability interpretation.
Y	a numeric vector of time to event in the same length as X and delta.
delta	a vector of binary indicator of event (1) or censoring (0) in the same length as X and Y.
tau	a scalar, the prediction horizon at which the prediction is evaluated.
span	a numeric value, the proportion of neighbour observations used in nearest neighbor method, default to 0.1.
h	a numeric value, the bandwidth of kernel weights, the defualt is NULL. If not specified, the function will use the value of span to calculate kernel weights. In case both span and h are specified, the function will use h.

type	a character value, indicating the type of kernel function used to calculate kernel weights. The default is "uniform" kernel. Other options are "Epanechnikov" and "normal". It will only be used when the bandwidth h is specified.
n.grid	a positive integer, the number of grid points used when calculating the ROC curve. The default is 1000.
X.min	the lower boundary of grid cut-off points for biomarker X. The default is the minimum of X.
X.max	the upper boundary of grid cut-off points for biomarker X. The default is the maximum of X.
cut.off	a vector of X cut-off values at which sensitivity and specificity will be calculated.
nboot	the number of bootstrap replications to be used for variance estimation. The default is nboot = 0, corresponding to no variance estimation.
alpha	It is (1 - level of confidence interval)/2, default is 0.05. It is used only when nboot > 0.
epsilon	The precision parameter used in an approximation to the weight calculation when the sample size is large. If a weight corresponding to a specific risk score is already calculated, then the weights corresponding to adjacent risk scores, within the distance specified by epsilon, will be the same under the approximation. This approximation avoids repeated calculation of weights that are almost the same, and hence increases the speed of computation in this situation. The default is NULL, which means no approximation is used. A higher value indicates less precision.
method	It is used to specify which method you would like to use to estimate AUC, default to "both". Other options are "integral" and "empirical".
output	It is used to specify which kind of output you want, default to "both". Other options are "AUC", including AUC, sensitivity, and specificity are included, and "calibration" including Brier Score and survival difference.

Details

This function takes the risk score value X, the time-to-event data Y and censoring indicator delta as input to estimate a number of time-dependent prediction accuracy measures for right-censored survival outcomes, including ROC curve, AUC, Brier score, and survival difference. The confidence intervals of above quantities will be estimated by bootstrap.

This function offer two options to estimate AUC. The first one make use of estimated sensitivity and specificity to calculate the AUC via trapezoidal integration by setting a series of cutoff point. The output will also include corresponding sensitivity and specificity for our plot function. The other one estimate AUC by the empirical estimator of the proportion of concordance pairs with proposed weight estimator (Li et al, 2015). These two methods will generate quite similar estimates. The option can be set by argument method.

We also include Brier Score and survival difference to evaluate the calibration metrics. Their definitions are included below. They can be estimated with the proposed conditional probability weight (Wu and Li, 2018). Both of them are measures to assess the accuracy of probabilistic predictions X. The calibration result makes sense only when the risk score X is a predicted probability, and should be ignored otherwise.

$$\text{Brier Score} = E[1(T \leq \tau, \delta = 1) - X]^2$$

$$\text{Survival difference} = E[1(T \leq \tau, \delta = 1) - X]$$

As mentioned in arguments, we introduced a small precision parameter `epsilon` to speed up the computation when the sample size is large. For each subject with a risk score, X_i, we assess whether there exists a previously processed grid point, $X_{grid,m}$ where $1 \leq m \leq j$, within the proximity of X_i such that $|X_i - X_{grid,m}| < \epsilon$. In the absence of such a point, we designate X_i as a new grid point, $X_{grid,j+1}$, and store the corresponding `survfit` object for subsequent weight estimation and mark it as a processed grid point. Conversely, if a previously processed grid point is found, we directly utilize the stored `survfit` object associated with it for weight calculation. Given that the most time-consuming step in our estimation process is the `survfit` computation, this method significantly reduces computing time without incurring notable bias especially when dealing with large sample sizes.

Value

Returns a list of the following items:

`main_res`: a list of `AUC.integral` estimated by trapezoidal integration, `AUC.empirical` estimated by empirical estimator of the proportion of concordance pairs. and a data frame `ROC` with dimension `(2+n.grid) x 3` with columns `cut.off`, `sens`, and `spec`.

`calibration_res`: brier score and survival difference estimated based on the formula similar to Wu and Li (2018). When the risk score `X` is a biomarker value instead of a predicted cumulative incidence probability, the brier score and survival difference cannot be calculated. In this case, please disregard the calibration results.

`boot_res`: a list of bootstrap results, including `bAUC`, `bAUC2`, `bBS`, `bSurvDiff`, `bROC`. For `bAUC`, `bAUC2`, `bBS`, `bSurvDiff`, each one is a list including corresponding mean, standard deviation, and confidence interval. `bROC` is a data frame with colomns `sens.mean`, `sens.sd`, `sens.lower`, `sens.upper`, `spec.mean`, `spec.sd`, `spec.lower`, `spec.upper`

Examples

```
library(survival)
data(mayo)
dat <- mayo[, c("time", "censor", "mayoscore5")]
fm <- tdROC(
  X = dat$mayoscore5, Y = dat$time, delta = dat$censor,
  tau = 365 * 6, span = 0.1, nboot = 0, alpha = 0.05,
  n.grid = 1000, cut.off = 5:9
)
# In the following example, We use biomarker mayoscore5 to estimate predicted probability
# tipycally a monotone transformation function such as expit() is used to transform biomarker
# with range out of range into estimated probability between 0 and 1
expit <- function(x){ 1/(1+exp(-x)) }

tdROC(
  X = expit(dat$mayoscore5), Y = dat$time, delta = dat$censor,
  tau = 365 * 6, span = 0.1, nboot = 0, alpha = 0.05,
  n.grid = 1000, cut.off = 5:9
)

tdROC(
  X = expit(dat$mayoscore5), Y = dat$time, delta = dat$censor,
  tau = 365 * 6, span = 0.1, nboot = 0, alpha = 0.05,
  n.grid = 1000, cut.off = 5:9, epsilon = 0.05
)
```

tdROC.cr *Estimate time-dependent prediction accuracy measures, including the ROC, AUC, Brier score, and survival probability difference, with competing risk data.*

Description

This is a core function of the 'tdROC' package. It uses the nonparametric weights proposed by Wu (Wu and Li, 2018) to estimate a number of time-dependent prediction accuracy measures for right-censored survival outcomes, including ROC curve, AUC, Brier score, and survival difference, with competing risk data. For each measure, the variance can be estimated through bootstrap resampling.

Usage

```
tdROC.cr(
  X,
  Y,
  delta,
  tau,
  span = 0.1,
  h = NULL,
  type = "uniform",
  epsilon = 0.01,
  cut.off = NULL,
  n.grid = 1000,
  nboot = 0,
  alpha = 0.05,
  method = "both",
  output = "both"
)
```

Arguments

X	a numeric vector of risk score in the same length as Y and delta, such as biomarker or predicted probability. A higher value indicates higher risk of the event. The calibration results (Brier score, survival difference) are applicable only when the risk score has the predicted probability interpretation.
Y	a numeric vector of time to event in the same length as X and delta.
delta	a vector of numeric indicator of event type in the same length as X and Y. The primary event should be coded as 1, the competing event should be coded as 2, and censoring should be coded as 0.
tau	a scalar, the prediction horizon at which the prediction is evaluated.
span	a numeric value, the proportion of neighbour observations used in nearest neighbor method. The default is 0.1.
h	a numeric value, the bandwidth of kernel weights, the defualt is NULL. If not specified, the function will use the value of span to calculate kernel weights. In case both span and h are specified, the function will use h.
type	a character value, indicating the type of kernel function used to calculate kernel weights. The default is "uniform" kernel. Other options are "Epanechnikov" and "normal". It will only be used when the bandwidth h is specified.

epsilon	The precision parameter used in an approximation to the weight calculation when the sample size is large. If a weight corresponding to a specific risk score is already calculated, then the weights corresponding to adjacent risk scores, within the distance specified by epsilon, will be the same under the approximation. This approximation avoids repeated calculation of weights that are almost the same, and hence increases the speed of computation in this situation. The default is NULL, which means no approximation is used. A higher value indicates less precision.
cut.off	a vector of X cut-off values at which sensitivity and specificity will be calculated.
n.grid	a positive integer, the number of grid points used when calculating the ROC curve. The default is 1000.
nboot	the number of bootstrap replications to be used for variance estimation. The default is nboot = 0, corresponding to no variance estimation.
alpha	It is (1 - level of confidence interval)/2, default is 0.05. It is used only when nboot > 0.
method	It is used to specify which method you would like to use to estimate AUC, default to "both". Other options are "integral" and "empirical".
output	It is used to specify which kind of output you want, default to "both". Other options are "AUC", including AUC, sensitivity, and specificity are included, and "calibration" including Brier Score and survival difference.

Details

This function takes the risk score value X, the time-to-event data Y and censoring indicator delta as input to estimate a number of time-dependent prediction accuracy measures for survival outcomes, including ROC curve, AUC, Brier score, and survival difference, with competing risk. The confidence intervals of above quantities are estimated by bootstrap.

For competing risk data, there are two definition of controls introduced by Zheng et al, which are listed below

$$\text{Definition A:Case}k : T \leq \tau, \delta = k; \text{Control}_A : (T > \tau) \cup (T \leq \tau \cap \delta \neq k)$$

$$\text{Definition B:Case}k : T \leq \tau, \delta = k; \text{Control}_B : (T > \tau)$$

Based on the definition A, both the event-free subjects and subjects who experience other competing events were included as controls. While definition B include only event-free subjects. This function offers two options to estimate AUC. The first one make use of estimated sensitivity and specificity to calculate the AUC via trapezoidal integration by setting a series of cutoff point. For the two different definition, we separately calculate the sensitivity, specificity and AUC. The output will also include the sensitivities and specificities for our plot function. The other one estimates AUC by the empirical estimator of the proportion of concordance pairs with proposed weight estimator (Wu and Li, 2018). These two methods generate quite similar estimates. The option can be set by the argument method.

In addition to the above prediction measures, we include Brier Score and survival difference to evaluate the calibration metrics. Their definitions are included below. They can be estimated with the proposed conditional probability weight (Wu and Li, 2018). Both of them are measures to assess the accuracy of probabilistic predictions X. The calibration result makes sense only when the risk score X is a predicted probability, and should be ignored otherwise.

$$\text{Brier Score} = E[1(T \leq \tau, \delta = 1) - X]^2$$

$$\text{Survival difference} = E[1(T \leq \tau, \delta = 1) - X]$$

This function uses the same approximation as `tdROC` with the argument `epsilon`

Value

Returns a list of the following items:

`main_res`: a list of `AUC.A.integral` estimated by trapezoidal integration for definition A, `AUC.A.empirical` estimated by empirical estimator for definition A, `AUC.B.integral` estimated by trapezoidal integration for definition B, `AUC.B.empirical` estimated by empirical estimator for definition B, and a data frame `ROC` with dimension `(2+n.grid) x 4` with columns `cut.off`, `sens`, `specA` and `specB`.

`calibration_res`: brier score and survival difference estimated based on the formula similar to Wu and Li (2018). When the risk score `X` is a biomarker value instead of a predicted cumulative incidence probability, the brier score and survival difference cannot be calculated. In this case, please disregard the calibration results.

`boot_res`: a list of bootstrap results, including `bAUC.A.integral`, `bAUC.A.empirical`, `bAUC.B.integral`, `bAUC.B.empirical`, `bBS`, `bSurvDiff`, `bROC`. For `bAUC.A.integral`, `bAUC.A.empirical`, `bAUC.B.integral`, `bAUC.B.empirical`, `bBS`, `bSurvDiff`, each one is a list including corresponding mean, standard deviation, confidence interval. `bROC` is a data frame with colomns `sens.mean`, `sens.sd`, `sens.lower`, `sens.upper`, `specA.mean`, `specA.sd`, `specA.lower`, `specA.upper`, `specB.mean`, `specB.sd`, `specB.lower`, `specB.upper`

References

Zheng Y, Cai T, Jin Y, Feng Z. Evaluating prognostic accuracy of biomarkers under competing risk. Biometrics. 2012;68(2):388-396. doi:10.1111/j.1541-0420.2011.01671.x

Examples

```
library(survival)
data(Melano)
expit <- function(x){ 1/(1+exp(-x)) }
tdROC.cr(X = expit(Melano$thick) , Y = Melano$time, delta = Melano$status, tau = 1800, nboot = 100)
```

Index