Package 'ROCt'

March 8, 2016

Type Package

Title Time-dependent ROC curve estimators and expected utility functions

Version 0.9.3

Date 2015-11-04

Author Y. Foucher, E. Dantan, P. Tessier, F. Le Borgne, and M. Lorent

Maintainer Y. Foucher <Yohann.Foucher@univ-nantes.fr>

Description This package contains functions in order to estimate diagnostic and prognostic capacities of continuous markers. More precisely, one function concerns the estimation of time-dependent ROC (ROCt) curve, as proposed by Heagerty, Lumley and Pepe (Biometrics, 2000). One function concerns the adaptation of the ROCt theory for studying the capacity of a marker to predict the excess of mortality of a specific population compared to the general population (same age, gender and calendar year). This last part is based on additive relative survival models and the work of Pohar-Perme (Biometrics, 2011). We also propose two functions for cut-off estimation in medical decision making by maximizing time-dependent expected utility function. Finally, we propose confounder-adjusted estimators of ROC and ROCt curves by using the Inverse Probability Weighting (IPW) approach. For the confounder-adjusted ROC curve (without censoring), we also proposed the implementation of the estimator based on placement values (Pepe and Cai, Biometrics, 2004).

License GPL (>=2)

LazyLoad yes

Depends splines, date, survival, relsurv, timereg

URL www.r-project.org, www.divat.fr

R topics documented:

ROCt-package 2
adjusted.ROC 3
adjusted.ROCt 4
AUC 6
crude.ROCt 7
dataDIVAT 9
dataKTFS 10
EUt1 11

EUt2 13
fr.ratetable 16
net.ROCt 17
rein.ratetable 19

Index **21**

`ROCt-package` *Time-dependent ROC curves estimation and other estimators to evaluate the diagnostic or prognostic capacities of a marker and its utility for medical decision making.*

Description

Compute ROC curve and expected utility for medical decision making, with and without censoring.

Details

Package:	ROCt
Type:	Package
Version:	0.9.3
Date:	2015-11-04
License:	GPL (>=2)
LazyLoad:	yes

Several functions are available:

crude.ROCt	This function allows the estimation of a crude time-dependent ROC curve, respecting the definition proposed by Heagerty et al. (2000).
net.ROCt	This function allows the estimation of net time-dependent ROC curve, i.e. when the only cause of death is due to the disease.
EUt	The expected utility theory allows the estimation of optimal cut-of for medical decision making.
AUC	This function computes the area under ROC curve using the trapezoidal rule based on two vectors of sensitivities and specificities.
adjusted.ROC	This function allows for the estimation of ROC curve by taking into account possible confounding factors (IPW or placement values estimators).
adjusted.ROCt	This function allows for the estimation of time-dependent ROC curve by taking into account possible confounding factors (IPW estimator).

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

References

Heagerty PJ., Lumley T., Pepe MS. (2000) Time-dependent ROC Curves for Censored Survival Data and a Diagnostic Marker. Biometrics, 56, 337-344.

Lorent M., Giral M., Foucher Y. (2013) Net time-dependent ROC curves: a solution for evaluating the accuracy of a marker to predict disease-related mortality. Statistics in Medicine, 33, 2379-89.

Tessier P. et al. (2015) Threshold definition of a prognostic marker in stratified medicine: An expected utility function for time-to-event data. Manuscript submitted.

Le Borgne F. et al. (2015) Adjusted time-dependent ROC curves. Manuscript submitted.

See Also

URL: www.divat.fr

`adjusted.ROC` *Confounders-adjusted ROC curves without right censored data*

Description

This function allows for the estimation of ROC curve by taking into account possible confounding factors (IPW or placement values estimators).

Usage

```
adjusted.ROC(status, variable, confounders,  database, precision, estimator)
```

Arguments

`status`	A character string with the name of the variable in `database` which represents the disease status indicator (for instance: 0=healthy, 1=diseased).
`variable`	A character string with the name of the variable in `database` which represents the diagnostic/prognostic variable under interest.
`confounders`	An object of class "formula". More precisely only the right part with an expression of the form `~ model`, where `model` is the linear predictor of the logistic regressions performed for each cut-off value. The user can use `~1` to obtain the crude estimation.
`database`	An object of the class `data.frame` containing the variables previously detailed.
`precision`	A numeric vector with values between 0 and 1. The values represent the x-axis (1-specificity) of the ROC graph for which the user want to obtain the corresponding sensitivities. 0 and 1 are not allowed.
`estimator`	Two possible estimators can be used: "ipw" and "pv". IPW is based on the Inverse Probability Weigthing theory as proposed by Le Borgne et al. (2015). The IPW estimator is selected by default. The user can also use the placement values estimator as proposed by Pepe and Cai (Biometrics, 2004).

Details

This function computes confounder-adjusted ROC curve for uncensored data. We adapted the usual estimator by considering the probability for a patient to have a prognostic variable higher than a given cut-off value, given the possible confounding factors. This estimator is obtain by using "ipw" in the option `estimator`. The user can also use the estimator first proposed by Pepe and Cai (Biometrics, 2004) which is based on placement values.

Value

table	This data frame presents the sensitivities and specificities.
auc	The area under the ROC curve.

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

References

Blanche P, Dartigues J, Jacqmin-Gadda H. (2013) Review and comparison of roc curve estimators for a time-dependent outcome with marker-dependent censoring. Biometrical Journal, 55, 687-704.

Pepe MS and Cai T. (2004) The analysis of placement values for evaluating discriminatory measures. Biometrics, 60(2), 528-35.

Le Borgne F, Giraudeau B, Giral M, Foucher Y. (2015) Adjusted time-dependent ROC curves. Submitted.

Examples

```
# import and attach the data example

data(dataDIVAT)

# The ROC curve to evaluate the crude capacities of the recipient age for the
# prognosis of post kidney transplant mortality (we ignore the censoring process)

roc1 <- adjusted.ROC(status="death", variable="ageR", confounders=~1,
 database=dataDIVAT, precision=seq(0.1,0.9, by=0.1) )

# The confounder-adjusted ROC curve to evaluate the capacities
# of the recipient age for the prognosis of post kidney transplant
# mortality by taking into account the donor age and the recipient
# gender (we ignore the censoring process).

roc2 <- adjusted.ROC(status="death", variable="ageR", confounders=~bs(ageD, df=3) +
  sexeR, database=dataDIVAT, precision=seq(0.1,0.9, by=0.1))

# The corresponding ROC graph

plot(1-roc1$table$sp, roc1$table$se, ylim=c(0,1), xlim=c(0,1), ylab="sensitivity",
  xlab="1-specificity", type="l", lty=1, col=1, lwd=2)

lines(1-roc2$table$sp, roc2$table$se, col=2, lwd=2, lty=2)

abline(c(0,0), c(1,1), lty=2)

legend("bottomright", lty=1:2, lwd=2, col=1:2, c(
  paste("Crude estimation, (AUC=", round(roc1$auc, 2), ")", sep=""),
  paste("Adjusted estimation, (AUC=", round(roc2$auc, 2), ")", sep="") ) )
```

`adjusted.ROCt` *Confounders-adjusted time-dependent ROC curves with right censored data*

Description

This function allows for the estimation of time-dependent ROC curve by taking into account possible confounding factors (IPW estimator).

Usage

```
adjusted.ROCt(times, failures, variable, confounders, database,
 pro.time, precision)
```

Arguments

`times`	A character string with the name of the variable in `database` which represents the follow up times.
`failures`	A character string with the name of the variable in `database` which represents the event indicator (0=right censored, 1=event).
`variable`	A character string with the name of the variable in `database` which represents the prognostic variable under interest. This variable is collected at the baseline.
`confounders`	An object of class "formula". More precisely only the right part with an expression of the form `~ model`, where `model` is the linear predictor of the logistic regressions performed for each cut-off value. The user can use `~1` to obtain the crude estimation.
`database`	An object of the class `data.frame` containing the variables previously detailed.
`pro.time`	The value of prognostic time represents the maximum delay for which the capacity of the variable is evaluated. The same unit than the one used in the argument `times`.
`precision`	The quintiles (between 0 and 1) of the prognostic variable used for computing each point of the time dependent ROC curve. 0 (min) and 1 (max) are not allowed.

Details

This function computes confounder-adjusted time-dependent ROC curve with right-censored data. We adapted the naive IPCW estimator as explained by Blanche, Dartigues and Jacqmin-Gadda (2013) by considering the probability for a patient to have a prognostic variable higher than a given cut-off value, given the possible confounding factors.

Value

`table`	This data frame presents the sensitivities and specificities associated with the cut-off values.
`auc`	The area under the time-dependent ROC curve for a prognostic up to `pro.time`.

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

References

Blanche P, Dartigues J, Jacqmin-Gadda H. (2013) Review and comparison of roc curve estimators for a time-dependent outcome with marker-dependent censoring. Biometrical Journal, 55, 687-704.

Le Borgne F, Giraudeau B, Giral M, Foucher Y. (2015) Adjusted time-dependent ROC curves. Submitted.

Examples

```
# import and attach the data example

data(dataDIVAT)

# The time-dependent ROC curve to evaluate the crude capacities of the recipient
# age for the prognosis of post kidney transplant mortality up to 2500 days.

roc1 <- adjusted.ROCt(times="death.time", failures="death", variable="ageR",
 confounders=~1, database=dataDIVAT, pro.time=2500, precision=seq(0.1,0.9, by=0.1))

# The confounder-adjusted time-dependent ROC curve to evaluate the
# capacities of the recipient age for the prognosis of post kidney
# transplant mortality up to 2500 days by taking into account the
# donor age and the recipient gender.

roc2 <- adjusted.ROCt(times="death.time", failures="death", variable="ageR",
 confounders=~bs(ageD, df=3) + sexeR, database=dataDIVAT, pro.time=2500,
 precision=seq(0.1,0.9, by=0.1))

# The corresponding ROC graph

plot(1-roc1$table$sp, roc1$table$se, ylim=c(0,1), xlim=c(0,1), ylab="sensitivity",
  xlab="1-specificity", type="l", lty=1, col=1, lwd=2)

lines(1-roc2$table$sp, roc2$table$se, col=2, lwd=2, lty=2)

abline(c(0,0), c(1,1), lty=2)

legend("bottomright", lty=1:2, lwd=2, col=1:2, c(
paste("Crude estimation, (AUC=", round(roc1$auc, 2), ")", sep=""),
paste("Adjusted estimation, (AUC=", round(roc2$auc, 2), ")", sep="") ) )
```

AUC *Area under ROC curve from sensitivities and specificities*

Description

This function computes the area under ROC curve by using the trapezoidal rule.

Usage

```
AUC(sens, spec)
```

Arguments

sens	A numeric vector with the sensitivities
spec	A numeric vector with the sensitivities

Details

This function computes the area under ROC curve using the trapezoidal rule. The value of the area is directly returned.

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

Examples

```
se.temp <- c(0, 0.5, 0.5, 1)
sp.temp <- c(1, 0.5, 0.5, 0)
AUC(se.temp, sp.temp)
```

---

crude.ROCt *Time-dependent ROC curves with right censored data*

Description

This function performs the characteristics of a time-dependent ROC curve based on k-nearest neighbor's (knn) estimator or only based on the Kaplan and Meier estimator.

Usage

```
crude.ROCt(times, failures, variable, pro.time, cut.off, estimator, prop)
```

Arguments

times	A numeric vector with the follow up times.
failures	A numeric vector with the event indicator (0=right censored, 1=event).
variable	A numeric vector with the prognostic variable. This variable is collected at the baseline.
pro.time	The value of prognostic time represents the maximum delay for which the capacity of the variable is evaluated. The same unit than the one used in the argument time.
cut.off	The cut-off values of the variable used to define the possible binary tests.
estimator	Three possible estimators can be used: 'kaplan-meier', 'akritas' or 'naive'. The naive estimator is selected by default.
prop	This is the unilateral proportion of the nearest neighbors. The estimation will be based on 2*prop (both right and left proportions) of the total sample size. This parameter will only be used if estimator='akritas'.

Details

This function computes time-dependent ROC curve with right-censored data. It can use Akritas approach (nearest neighbor's estimation) for ensuring monotone increasing ROC curve, instead of the simple Kaplan-Meier estimator. This Akritas approach may be avoid if the sample size is large because of computing time. Both estimators were defined by Heagerty, Lumley and Pepe (Biometrics, 2000). A third alternative is the use of the naive estimator as explained by Blanche, Dartigues and Jacqmin-Gadda (2013). This estimator is less time-consuming compared to the Akritas approach.

Value

`table`	This data frame presents the sensitivities and specificities associated with the cut-off values. One can observe NA if the value cannot be computed.
`auc`	The area under the time-dependent ROC curve for a prognostic up to prognostic time.
`missing`	Number of deleted observations due to missing data.

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

References

Heagerty PJ., Lumley T., Pepe MS. (2000) Time-dependent ROC Curves for Censored Survival Data and a Diagnostic Marker. Biometrics, 56, 337-344.

Akritas MG. (1994) Nearest neighbor estimation of a bivariate distribution under random censoring. Annals of Statistics, 22, 1299-1327.

Uno H. et al. (2007) Evaluating prediction rules for t-year survivors with censored regression models. Journal of the American Statistical Association, 102, 527-37.

Blanche P, Dartigues J, Jacqmin-Gadda H. (2013) Review and comparison of roc curve estimators for a time-dependent outcome with marker-dependent censoring. Biometrical Journal, 55, 687-704.

Examples

```
# import and attach the data example

data(dataDIVAT)

# A subgroup analysis to reduce the time needed for this exemple

dataDIVAT <- dataDIVAT[1:400,]

# cut-off values definition (choose more values in prectice)

age.cut <- quantile(dataDIVAT$ageR, probs=seq(0.1, 0.9, by=0.1))

# the ROC curve (with the naive estimator) to predict the all-cause
# mortality up to the 3000 days

roc1 <- crude.ROCt(times=dataDIVAT$death.time,
 failures=dataDIVAT$death, variable=dataDIVAT$ageR,
 pro.time=3000, cut.off=age.cut, estimator="naive")
```

```
# the sensibilities and specificities associated with the cut off values

roc1$table

# the ROC curve (Kaplan-Meier estimator without the knn correction)
# to predict the all-cause mortality up to the 3000 days

# the ROC graph

plot(1-roc1$table$sp, roc1$table$se, ylim=c(0,1), xlim=c(0,1), ylab="sensitivity",
 xlab="1-specificity", type="l", lty=1, col=2, lwd=2)

abline(c(0,0), c(1,1), lty=2)

legend("bottomright", paste("Naive, (AUC=", round(roc1$auc, 2), ")", sep=""),
lty=1, lwd=2, col=2)

# the AUC

roc1$auc

AUC(sens=roc1$table$se, spec=roc1$table$sp)
```

`dataDIVAT` *A sample of the DIVAT cohort in orter to study the mortality of kidney transplant recipients.*

Description

A data frame with 4267 French kidney transplant recipients. The time-to-event is the time between the transplantation and the recipient death. This time can be right-censored. A vector of covariates is also collected at the transplantation.

Usage

```
data(dataDIVAT)
```

Format

A data frame with 4267 observations (raws) with the 8 following variables (columns).

`ageR` This numeric vector represents the age of the recipient (in years)

`sexeR` This numeric vector represents the gender of the recipient (1=men, 0=female)

`year.tx` This numeric vector represents the year of the transplantation

`ante.diab` This numeric vector represents the diabetes statute (1=yes, 0=no)

`pra` This numeric vector represents the pre-graft immunization using the panel reactive antibody (1=detectable, 0=undetectable)

`ageD` This numeric vector represents the age of the donor (in years)

`death.time` This numeric vector represents the follow up time in days (until death or censoring)

`death` This numeric vector represents the death indicator at the follow-up end (1=death, 0=alive)

Details

The immunology and nephrology department of the Nantes University hospital constituted a data bank with the monitoring of medical records for kidney and/or pancreas transplant recipients. The associated software is called DIVAT. Here is a sample of 4267 patients. The time-to-event is the time between the transplantation and the death of the recipient. This time can be right-censored. A vector of covariates, all measured at the transplantation, is also collected for each patient.

Source

URL: http://www.divat.fr.

Examples

```
data(dataDIVAT)

### a short summary of the recipient age at the transplantation
summary(dataDIVAT$ageR)

### Kaplan and Meier estimation of the recipient survival
plot(survfit(Surv(death.time/365.25, death) ~ 1, data = dataDIVAT),
 xlab="Post transplantation time (in years)", ylab="Patient survival",
 mark.time=FALSE)
```

---

dataKTFS *A sample of the DIVAT cohort in orter to study the prognostic capacities of the Kidney Transplant Failure Score (KTFS).*

Description

A data frame with 2169 French kidney transplant recipients. The time-to-event is the time between the transplantation and the return in dialysis. This time can be right-censored, especially at the time of the patient death with a functioning graft. The KTFS is a score proposed by Foucher et al. (2010) to stratify the recipients according to their risk of return in dialysis.

Usage

```
data(dataKTFS)
```

Format

A data frame with 2169 observations (raws) with the 3 following variables (columns).

`time` This numeric vector represents the follow up time in years (until return in dialyis or censoring)

`failure` This numeric vector represents the graft failure indicator at the follow-up end (1=return, 0=censoring)

`score` This numeric vector represents the KTFS values.

Details

The immunology and nephrology department of the Nantes University hospital constituted a data bank with the monitoring of medical records for kidney and/or pancreas transplant recipients. The associated software is called DIVAT. Here is the training sample of 2169 patients used for the construction of the KTFS, a scoring system proposed by Foucher et al. (2010) in order to stratify patients according to their risk of return in dialysis. The KTFS is based on 8 parameters collected during the first year post transplantation. Patients with a KTFS value higher than 4.17 were considered at high-risk.

Source

URL: http://www.divat.fr.

References

Foucher Y. al. (2010) A clinical scoring system highly predictive of long-term kidney graft survival. Kidney International, 78, 1288-94.

Examples

```
data(dataKTFS)

### a short summary of the recipient age at the transplantation
summary(dataKTFS$score)

### Kaplan and Meier estimation of the recipient survival
plot(survfit(Surv(time, failure) ~ I(score>4.17), data = dataKTFS),
 xlab="Post transplantation time (in years)", ylab="Patient survival",
 mark.time=FALSE, col=c(2,1), lty=c(2,1))
legend("bottomleft", c("Recipients in the high-risk group",
 "Recipients in the low-risk group"), col=1:2, lty=1:2)
```

EUt1 — *Optimal cut-off estimation of a prognostic marker to stratify the medical decision between two treatments A and B, when only the group B is observed.*

Description

This function allows the estimation of an optimal cut-off for medical decision making between two treatments A and B from a prognostic marker by maximizing the expected utility in a time-dependent context. Only the times-to-event of the group B are observed.

Usage

```
EUt1(times, failures, variable, pro.time, u.A0, u.A1, u.B0, u.B1, n.boot,
 rmst.change)
```

Arguments

times	A numeric vector with the follow up times for the patients receiving the treatment B.
failures	A numeric vector with the event indicator for the patients receiving the treatment B (0=right censoring, 1=event).
variable	A numeric vector with the observed values of the marker/variable under interest X for the patients receiving the treatment B. This variable is collected at the baseline (times=0). By convention, we assume that patients with X>k will preferentially receive A, k being the optimal cut-off. In contrast, patients with X<k will receive preferentially B.
pro.time	The prognostic time for which the prognostic capacities of the marker and the patient outcomes are considered in the same unit than the one used in the argument times.
u.A0	A value of the utility of a patient receiving the treatment A before the event occurrence. This value should respect the 0-1 scale (from death to perfect health).
u.A1	A value of the utility of a patient receiving the treatment A after the event occurrence. This value should respect the 0-1 scale.
u.B0	A value of the utility of a patient receiving the treatment B before the event occurrence. This value should respect the 0-1 scale.
u.B1	A value of the utility of a patient receiving the treatment B after the event occurrence. This value should respect the 0-1 scale.
n.boot	Number of bootstrap iterations to compute the 95% confidence interval of the optimal cut-off. The default value is NULL: no confidence interval is estimated.
rmst.change	A numeric vector with the expected relative change in the Restricted Mean Survival Time (RMST) by using the treatment A instead of the treatment B among patients with X>k.

Details

This function computes the expected time-dependent utility. The data may be right-censored. The user has to define the utilities (qualities of life related to the possible health states). A positive test is defined by a marker value higher than the cut-off (X>k): by convention the the patient will receive the treatment A. For example, the user has a cohort of patients receiving the treatment B. She(he) assumes that an alternative treatment A would be more convenient for patients with high-values of the marker X. She(he) aims to compute the optimal cut-off value for a future stratified medical decision rule: treatment A for patients with X>k and treatment B for patients with X<k. The user has to enter the observed cohort of patients with the treatment B. Additional to the assumptions related to health-state utilities, the user have to specify in rmst.change the expected relative change in terms of RMST between the two treatments. For instance, if the life expectancy of a patient with treatment B over the next 8 years (value entered in pro.time) is 6.70 years, and assuming that the treatment A increases this life expectancy during the next 8 years by 1.33 years, the expected relative change in RMST is 0.20 (=1.33/6.7.) (Royston and Parmar, 2011)

Value

estimation	This is a single value if n.boot=NULL, which corresponds to the estimated cut-off that maximizes the time-dependent expected utility of the medical decision. If this value corresponds to the minimum of the marker, all the patients should be treated with A. If this value corresponds to the maximum of the marker, all the patients should conserve the treatment B. When n.boot is not null, two

additional values are returned: `CIinf` is the lower bound of the 95% confidence interval and `CIsup` is the upper bound of the 95% confidence interval.

`max.eu` This value corresponds to the maximum expected utility associated with the `estimation`.

`table` This data frame is composed by 8 columns representing respectively the cut-off values, the time-dependent expected utilities (`utility`), the proportions of patients with a marker value higher (`pA`) and lower (`pB`) than the cut-off value, the numbers of QALYs for patients with a marker value higher (`qA`) and lower (`qB`) than the cut-off value, the RMST for patients with a marker value higher (`eA`) and lower (`eB`) than the cut-off value.

`delta.rmst` This value represents the expected RMST for patients with a marker higher than the `estimation` (treated with A) minus the observed RMST for patients with a marker higher than the `estimation`(treated with B).

`delta.qaly` This value represents the number of QALYs for patients with a marker higher than the `estimation` (treated with A) minus the observed number of QALYs for patients with a marker higher than the `estimation` (treated with B).

`missing` Number of deleted observations due to missing data.

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

References

Foucher et al. (2010) A clinical scoring system highly predictive of long-term kidney graft survival. Kidney International, 78:1288-94.

Royston and Parmar. (2011) The use of restricted mean survival time to estimate the treatment effect in randomized clinical trials when the proportional hazards assumption is in doubt. Statistics in Medicine, 30(19):2409-21.

Dantan et al. (2015) Threshold definition of a prognostic marker in stratified medicine: An expected utility function for time-to-event data. Manuscript submitted.

Examples

```
# import and attach the data example

data(dataKTFS)

dataKTFS$score <- round(dataKTFS$score, 2)

# the expected utility function for a prognostic up to 8 years

EUt.obj <- EUt1(dataKTFS$time, dataKTFS$failure, dataKTFS$score, pro.time=8,
 u.A0=0.81*0.95, u.A1=0.53, u.B0=0.81, u.B1=0.53, rmst.change=0.2)

plot(EUt.obj$table$cut.off, EUt.obj$table$utility, type="l",
 xlab="Cut-off values", ylab="Expected utility", col=1, lty=1)

segments(EUt.obj$estimation, 0, EUt.obj$estimation, EUt.obj$max.eu, lty=3)
segments(0, EUt.obj$max.eu, EUt.obj$estimation, EUt.obj$max.eu, lty=3)

text(EUt.obj$estimation-0.2, 6.22,
```

```
 paste("Optimal cut-off=", round(EUt.obj$estimation,2)), srt=90, cex=0.8)
 text(min(dataKTFS$score)+1.4, EUt.obj$max.eu-0.006,
 paste("Expected utility=", round(EUt.obj$max.eu, 2)), cex=0.8)

# the optimal cut-off: patients with an higher value should receive the treatment A

EUt.obj$estimation
```

EUt2 *Optimal cut-off estimation of a prognostic marker to stratify the medical decision between two treatments A and B, when both groups A and B are observed.*

Description

This function allows the estimation of an optimal cut-off for medical decision making between two treatments A and B from a prognostic marker by maximizing the expected utility in a time-dependent context.

Usage

```
EUt2(times, failures, variable, treatment, pro.time, u.A0, u.A1, u.B0, u.B1,
 n.boot)
```

Arguments

times	A numeric vector with the follow up times.
failures	A numeric vector with the event indicator (0=right censoring, 1=event).
variable	A numeric vector with the observed values of the marker/variable under interest X. This variable is collected at the baseline (times=0). By convention, we assume that patients with X>k will preferentially receive A, k being the optimal cut-off. In contrast, patients with X<k will preferentially receive B.
treatment	A character vector with the observed treatment. Only character strings "A" and "B" are allowed.
pro.time	The prognostic time for which the capacities of the marker and the patient outcomes are considered in the same unit than the one used in the argument times.
u.A0	A value of the utility of a patient receiving the treatment A before the event occurrence. This value should respect the 0-1 scale (from death to perfect health).
u.A1	A value of the utility of a patient receiving the treatment A after the event occurrence. This value should respect the 0-1 scale.
u.B0	A value of the utility of a patient receiving the treatment B before the event occurrence. This value should respect the 0-1 scale.
u.B1	A value of the utility of a patient receiving the treatment B after the event occurrence. This value should respect the 0-1 scale.
n.boot	Number of bootstrap iterations to compute the 95% confidence interval of the optimal cut-off. The default value is NULL: no confidence interval is estimated.

Details

This function computes the expected time-dependent utility for the cut-off values defined by the user. The data may be right-censored. The user has to define the utilities (qualities of life related to the possible health-states after the marker-based decision). A positive test is defined by a marker value higher than the cut-off (X>k): by convention the patient will receive preferentially the treatment A. A negative test is defined by a marker value lower than or equal to the cut-off (X>k): by convention the patients will receive preferentially the treatment B. For example, the user has data from a clinical trial in which treatments A and B independently to the value of the marker X. She(he) assumes that an alternative treatment A will be more convenient for patients with high value of the marker X. She(he) aims to compute the optimal cot-off value for a future stratified medical decision rule: treatment A for patients with X>k and treatment B for patients with X<k.

Value

estimation	his is a single value if n.boot=NULL, which corresponds to the estimated cut-off that maximizes the time-dependent expected utility of the medical decision. If this value corresponds to the minimum of the marker, all the patients should be treated with A. If this value corresponds to the maximum of the marker, all the patients should conserve the treatment B. When n.boot is not null, two additional values are returned: CIinf is the lower bound of the 95% confidence interval and CIsup is the upper bound of the 95% confidence interval.
max.eu	This value corresponds to the maximum expected utility associated with the estimation.
table	This data frame is composed by 8 columns representing respectively the cut-off values (cut.off), the time-dependent expected utilities (utility), the proportions of patients with a marker value higher (pA) and lower (pB) than the cut-off value, the numbers of QALYs for patients with a marker value higher (qA) and lower (qB) than the cut-off value, the RMST for patients with a marker value higher (eA) and lower (eB) than the cut-off value.
delta.rmst	This is a vector with two values. The first value, entitled high.level, represents the RMST for patients with a marker higher than the estimation and treated with A minus the RMST for patients with a marker higher than the estimation and treated with B. The second value, entitled low.level, represents the RMST for patients with a marker lower than or equal to the estimation and treated by B minus the RMST for patients with a marker lower than or equal to the estimation and treated with A.
delta.qaly	This is a vector with two values. The fist value, entitled high.level, represents the number of QALYs for patients with a marker higher than the estimation and treated by A minus the number of QALYs for patients with a marker higher than the estimation and treated with B. The second value, entitled low.level, represents the number of QALYs for patients with a marker lower than or equal to the estimation and treated with B minus the number of QALYs for patients with a marker lower than or equal to the estimation and treated with A.
missing	Number of deleted observations due to missing data.

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

References

Schlichting et al. (1983) Prognostic factors in cirrhosis identified by Cox's regression model. Hepatol Baltim Md, 3(6):889-95.

Royston and Parmar. (2011) The use of restricted mean survival time to estimate the treatment effect in randomized clinical trials when the proportional hazards assumption is in doubt. Statistics in Medicine, 30(19):2409-21.

Dantan et al. (2015) Threshold definition of a prognostic marker in stratified medicine: An expected utility function for time-to-event data. Manuscript submitted.

Examples

```
# import and attach the data example

data(csl)
csl <- csl[order(csl$id, csl$time),]
csl$ordre <-0
for (i in unique(csl$id)) {csl$ordre[csl$id==i] <- 1:sum(csl$id==i)}

csl$ttt[csl$treat==0]<-"A"
csl$ttt[csl$treat==1]<-"B"

csl0 <- csl[csl$ordre==1,]
csl0<-csl0[,c(1,4,5,14,9)]

# the expected utility function for a prognostic up to 8 years

EUt.obj <- EUt2(csl0$eventT, csl0$dc, csl0$prot.base, treatment= csl0$ttt,
 pro.time=8, u.A0=0.75*0.95, u.A1=0, u.B0=0.75, u.B1=0)

plot(EUt.obj$table$cut.off, EUt.obj$table$utility, type="l",
 xlab="Cut-off values", ylab="Expected utility",col=1, lty=1)

segments(EUt.obj$estimation, 0, EUt.obj$estimation, EUt.obj$max.eu, lty=3)
segments(0, EUt.obj$max.eu, EUt.obj$estimation, EUt.obj$max.eu, lty=3)

text(EUt.obj$estimation-2, 3.38,
 paste("Optimal cut-off=",round(EUt.obj$estimation,2)), srt=90, cex=0.8)
 text(min(csl0$prot.base)+40, EUt.obj$max.eu-0.005,
 paste("Expected utility=",round(EUt.obj$max.eu,2)), cex=0.8)

# the optimal cut-off: patients with an higher value should receive the treatment A

EUt.obj$estimation
```

`fr.ratetable` *Expected mortality rates of the general French population.*

Description

An object of class ratetable for the expected mortality of the French population. It is an array with three dimensions: age, sex and year.

Usage

```
data(fr.ratetable)
```

Format

The format is "ratetable". The attributes are:

`dim` A numeric vector with the length of each dimension.

`dimnames` A list of vectors with the names of each variable of the three dimensions.

`dimid` A character vector with the identification of the dimensions: `age`, `year` and `sex`.

`factor` A vector of indicators=1 if the corresponding dimension do not vary according to the `time`. Only the `sex` is associated to 1.

`cutpoints` A list of the thresholds to identify the mortality rates according to the time-varying dimensions (NULL for `sex`).

`class` The class of the object: `ratetable`.

Details

The organization of a ratetable object is described in details by Therneau (1999) and Pohar (2006).

Source

The original data and updates can be downloaded from the Human Life-Table Database (HMD, The Human Mortality Database). URL: http://www.mortality.org/

References

Therneau T., Offord J. Expected Survival Based on Hazard Rates (Update), Technical Report, Section of Biostatistics, Mayo Clinic 63, 1999.

Pohara M., Stare J. Relative survival analysis in R. Computer methods and programs in biomedicine, 81: 272-278, 2006.

Examples

```
data(fr.ratetable)

is.ratetable(fr.ratetable)
```

---

net.ROCt *Net time-dependent ROC curves with right censored data*

Description

This function performs the characteristics of a net time-dependent ROC curve based on k-nearest neighbor's (knn) estimator or only based on the Pohar-Perme estimator (Pohar, 2012).

Usage

```
net.ROCt(times, failures, variable, p.age, p.sex, p.year,
 rate.table, pro.time, cut.off, knn=FALSE,
 prop=NULL)
```

Arguments

times	A numeric vector with the follow up times.
failures	A numeric vector with the event indicator (0=right censored, 1=event).
variable	A numeric vector with the prognostic variable. This variable is collected at the baseline.
p.age	A numeric vector with the age of the individuals at the baseline (in days).
p.sex	A character vector with the gender the individuals ('male' or 'female').
p.year	A numeric vector with the calendar year at the baseline (number of days from the January 1, 1960).
rate.table	A ratetable object with the expected mortality rates by age, sex, and cohort year. The same units used in p.age, p.sex, p.year
pro.time	The value of prognostic time represents the maximum delay for which the capacity of the variable is evaluated. The same unit than the one used in the argument time.
cut.off	The cut-off values of the variable used to define the possible binary tests.
knn	A logical value indicating whether k-nearest neighbor's estimator should be used.
prop	This is the proportion of the nearest neighbors. The estimation will be based on 2*prop (both right and left proportions) of the total sample size.

Details

This function computes net time-dependent ROC curve with right-censored data using estimator defined by Pohar-Perm et al. (2011) and the k-nearest neighbor's (knn) estimator. The aim is to evaluate the capacity of a variable (measured at the baseline) to predict the excess of mortality of a studied population compared to the general population mortality. Using the knn estimator ensures a monotone and increasing ROC curve, but the computation time may be long. This approach may thus be avoided if the sample size is large because of computing time.

Value

table	This data frame presents the sensitivities and specificities associated with the cut-off values. One can observe NA if the value cannot be computed.
auc	The area under the time-dependent ROC curve for a prognostic up to prognostic time.
missing	Number of deleted observations due to missing data.
warning	This message indicates possible difficulties in the computation of the net ROC curve, for instance if the net survival was not lower or equal1 to 1 for particular cut-off values or times.

Author(s)

Y. Foucher <Yohann.Foucher@univ-nantes.fr>

References

Pohar M., Stare J., Esteve J. (2012) On Estimation in Relative Survival. Biometrics, 68, 113-120.

Lorent M., Giral M., Foucher Y. (2013) Net time-dependent ROC curves: a solution for evaluating the accuracy of a marker to predict disease-related mortality. Statistics in Medicine, 33, 2379-89.

Examples

```
# import the observed data

data(dataDIVAT)

# A subgroup analysis to reduce the time needed for this example

dataDIVAT <- dataDIVAT[1:400,]

# import the expected mortality rates

data(fr.ratetable)

# the values of recipient age used for computing the sensibilities and
# specificities (choose more values in practice)

age.cut <- quantile(dataDIVAT$ageR, probs=seq(0.1, 0.9, by=0.1))

# recoding of the variables for matching with the ratetable

dataDIVAT$sex <- "male"
dataDIVAT$sex[dataDIVAT$sexeR==0] <- "female"
dataDIVAT$year <- mdy.date(month=01, day=01, year=dataDIVAT$year.tx,
 nineteen = TRUE, fillday = FALSE, fillmonth = FALSE)
dataDIVAT$age <- dataDIVAT$ageR*365

# the ROC curve (without correction by the knn estimator) to
# reduce the time for computing this example. In prectice, the
# correction should by used in case of non-montone results.

roc1 <- net.ROCt(times=dataDIVAT$death.time,
 failures=dataDIVAT$death,  variable=dataDIVAT$ageR,
 p.age=dataDIVAT$age, p.sex=dataDIVAT$sex, p.year=dataDIVAT$year,
 rate.table=fr.ratetable, pro.time=3000, cut.off=age.cut, knn=FALSE)

# the sensibilities and specificities associated with the cut off values

roc1$table

# the traditional ROC graph

plot(c(1,1-roc1$table$sp,0), c(1,roc1$table$se,0), ylim=c(0,1), xlim=c(0,1),
 ylab="sensitivity", xlab="1-specificity", type="l", lty=1, col=2, lwd=2)

abline(c(0,0), c(1,1), lty=2)

legend("bottomright", paste("Without knn, (AUC=",
 round(roc1$auc, 2), ")", sep=""),lty=1, lwd=2, col=2)
```

`rein.ratetable` *Expected mortality of French patients with End Stage Kidney Disease (ESKD) in dialysis and registrited previously on waiting list for renal transplantation*

Description

An object of the class ratetable which contains the expected mortality of French patients with End Stage Kidney Disease (ESKD) in dialysis and registered previously on waiting list for renal transplantation. It is an array with two dimensions: the time since the registry on waiting list and the patient profile at this baseline.

Usage

```
data(rein.ratetable)
```

Format

The format is "ratetable". The attributes are:

`dim` A numeric vector with the length of each dimension.

`dimnames` A list of vectors with the names of each variable of both dimensions (see details for modalities of `profile`).

`dimid` A character vector with the identification of the dimensions: `time`, `profile`.

`factor` A vector of indicators=1 if the corresponding dimension do not vary according to the `time`. The `profile` is associated to 1.

`cutpoints` A list of the thresholds to identify the mortality rates according to the time-varying dimensions (NULL for `profile`).

`class` The class of the object: `ratetable`.

Details

The organization of a ratetable object is described in details by Therneau (1999) and Pohar (2006). We have estimated these data by using the renal epidemiology and information network (REIN). A competing risk model was performed in order to deal with the loss of follow-up to the transplantation. The `profile` represents the concatenation of the significant binary variables associated with the mortality on waiting list. In order to obtain the same variable in your database, we have to concatenate the following values in the same order. These values are collected at the registration date:

a. 1 if the patient is male and 0 if female.
b. 1 if the time between the first dialysis and the registration is higher than 1 year and 0 otherwise.
c. 1 if hemodialysis and 0 if peritoneal dialysis.
d. 1 if the patient is diabetic and 0 otherwise.
e. 1 if the patient had history of cardiac failure diabetic and 0 otherwise.
f. 1 if the patient had history of vascular failure diabetic and 0 otherwise.
g. 1 if the patient had history of cancer and 0 otherwise.
h. 1 if the age of the patients is in between 40 and 60 years and 0 otherwise.
i. 1 if the age of the patients is higher than 60 years and 0 otherwise.
j. 1 if the calendar year at registration is strictly higher than 2007 and 0

Source

URL: http://www.soc-nephrologie.org/REIN/

References

Pohar M., Stare J. (2006) Relative survival analysis in R. Computer methods and programs in biomedicine, 81, 272-278.

Examples

```
data(rein.ratetable)

is.ratetable(rein.ratetable)
```

Index

∗Topic **AUC**
AUC, 6
∗Topic **Akritas**
crude.ROCt, 7
net.ROCt, 17
∗Topic **Cut-off**
EUt1, 11
EUt2, 13
∗Topic **Divat**
dataDIVAT, 9
dataKTFS, 10
∗Topic **France**
fr.ratetable, 16
∗Topic **IPW**
adjusted.ROC, 3
adjusted.ROCt, 4
∗Topic **Net survival**
net.ROCt, 17
∗Topic **ROC**
adjusted.ROC, 3
adjusted.ROCt, 4
crude.ROCt, 7
net.ROCt, 17
ROCt-package, 2
∗Topic **Time-dependent**
adjusted.ROCt, 4
crude.ROCt, 7
EUt1, 11
EUt2, 13
ROCt-package, 2
∗Topic **Utility**
EUt1, 11
EUt2, 13
ROCt-package, 2
∗Topic **datasets**
dataDIVAT, 9
dataKTFS, 10
fr.ratetable, 16
rein.ratetable, 19
∗Topic **ratetable**
fr.ratetable, 16
rein.ratetable, 19

adjusted.ROC, 3
adjusted.ROCt, 4
AUC, 6

crude.ROCt, 7

dataDIVAT, 9
dataKTFS, 10

EUt1, 11
EUt2, 13

fr.ratetable, 16

net.ROCt, 17

rein.ratetable, 19
ROCt-package, 2