Package 'SurrogateOutcome'

August 29, 2025

Type Package

Title Estimation of the Proportion of Treatment Effect Explained by Surrogate Outcome Information

Version 1.2

Description
Provides functions to estimate the proportion of treatment effect on a censored primary outcome that is explained by the treatment effect on a censored surrogate outcome/event. All methods are described in detail in Parast, et al (2020) ``Assessing the Value of a Censored Surrogate Outcome" <doi:10.1007/s10985-019-09473-1> and Wang et al (2025) ``Model-free Approach to Evaluate a Censored Intermediate Outcome as a Surrogate for Overall Survival" (Under Review) <doi:10.48550/arXiv.2412.14129>. A tutorial for this package can be found at <https://www.laylaparast.com/surrogateoutcome>.

License GPL

Imports stats, survival

NeedsCompilation no

Author Layla Parast [aut, cre], Xuan Wang [aut]

Maintainer Layla Parast <parast@austin.utexas.edu>

Contents

censor.weight 2
cumsum2 3
delta.estimate 3
delta.estimate.RMST 5
delta.q.event.RMST 6
delta.q.event.semi.RMST 8
delta.t.RMST 9
ExampleData 10
helper.si 11
IV.event 12
Kern.FUN 14
new.q 15
pred.smooth.surv 15
R.opt.event 16
R.q.event 18
R.t.estimate 21

resam 24
VTM 24
WEIGHT 25
WEIGHT.p 25

Index **27**

`censor.weight` *Calculates censoring probability for weighting*

Description

Helper function; calculates censoring probability needed for inverse probability of censoring weighting

Usage

```
censor.weight(data.x, data.delta, t, weight = NULL)
```

Arguments

`data.x`	numeric vector, the observed event time: X = min(T, C) where T is the time of the primary outcome, C is the censoring time
`data.delta`	numeric vector of 0/1, the censoring indicator: D = I(T<C) where T is the time of the primary outcome, C is the censoring time
`t`	number, the time of interest
`weight`	a numeric vector or matrix of weights used for perturbation-resampling, default is null.

Details

Computes the Kaplan Meier estimate of survival for the censoring random variable at the specified time

Value

Kaplan Meier estimate of survival for censoring at time t

Author(s)

Layla Parast

Examples

```
data(ExampleData)
censor.weight(data.x = ExampleData$x1, data.delta = ExampleData$delta1, t=5)
```

cumsum2 *Helper function*

Description

Helper function; should not be called directly by user.

Usage

```
cumsum2(mydat)
```

Arguments

mydat	mydat

Value

out	matrix

Author(s)

Layla Parast

delta.estimate *Estimates the treatment effect at time t, defined as the difference in the restricted mean survival time*

Description

Estimates the treatment effect at time t, defined as the difference in the restricted mean survival time.

Usage

```
delta.estimate(xone, xzero, deltaone, deltazero, t, std = FALSE, conf.int = FALSE,
weight.perturb = NULL)
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.
deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
t	time of interest for treatment effect.

`std`	TRUE or FALSE; indicates whether standard error estimates should be provided, default is FALSE. Estimates are calculated using perturbation-resampling. Two versions are provided: one that takes the standard deviation of the perturbed estimates (denoted as "sd") and one that takes the median absolute deviation (denoted as "mad").
`conf.int`	TRUE or FALSE; indicates whether 95% confidence intervals should be provided. Confidence intervals are calculated using the percentiles of perturbed estimates, default is FALSE. If this is TRUE, standard error estimates are automatically provided.
`weight.perturb`	weights used for perturbation resampling.

Details

Let $G \in \{1, 0\}$ be the randomized treatment indicator and T denote the time of the primary outcome of interest. We use potential outcomes notation such that $T^{(G)}$ denotes the time of the primary outcome under treatment G, for $G \in \{1, 0\}$. We define the treatment effect as the difference in restricted mean survival time up to a fixed time t under treatment 1 versus under treatment 0,

$$\Delta(t) = E\{T^{(1)} \wedge t\} - E\{T^{(0)} \wedge t\}$$

where $\wedge$ indicates the minimum. Due to censoring, data consist of $n = n_1 + n_0$ independent observations $\{X_{gi}, \delta_{gi}, i = 1, ..., n_g, g = 1, 0\}$, where $X_{gi} = T_{gi} \wedge C_{gi}$, $\delta_{gi} = I(T_{gi} < C_{gi})$, C_{gi} denotes the censoring time, T_{gi} denotes the time of the primary outcome, and $\{(T_{gi}, C_{gi}), i = 1, ..., n_g\}$ are identically distributed within treatment group. The quantity $\Delta(t)$ is estimated using inverse probability of censoring weights:

$$\hat{\Delta}(t) = n_1^{-1} \sum_{i=1}^{n_1} \hat{M}_{1i}(t) - n_0^{-1} \sum_{i=1}^{n_0} \hat{M}_{0i}(t)$$

where $\hat{M}_{gi}(t) = I(X_{gi} > t)t/\hat{W}_g^C(t) + I(X_{gi} < t)X_{gi}\delta_{gi}/\hat{W}_g^C(X_{gi})$ and $\hat{W}_g^C(t)$ is the Kaplan-Meier estimator of $P(C_{gi} \geq t)$.

Value

A list is returned:

`delta`	the estimate, $\hat{\Delta}(t)$, described above.
`rmst.1`	the estimated restricted mean survival time in group 1, described above.
`rmst.0`	the estimated restricted mean survival time in group 0, described above.
`delta.sd`	the standard error estimate of $\hat{\Delta}(t)$; if std = TRUE or conf.int = TRUE.
`delta.mad`	the standard error estimate of $\hat{\Delta}(t)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`conf.int.delta`	a vector of size 2; the 95% confidence interval for $\hat{\Delta}(t)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.

Author(s)

Layla Parast

References

Parast L, Tian L, and Cai T (2020). Assessing the Value of a Censored Surrogate Outcome. Lifetime Data Analysis, 26(2):245-265.

Tian, L, Zhao, L, & Wei, LJ (2013). Predicting the restricted mean event time with the subject's baseline covariates in survival analysis. Biostatistics, 15(2), 222-233.

Royston, P, & Parmar, MK (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), 2409-2421.

Examples

```
data(ExampleData)
names(ExampleData)

delta.estimate(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, t = 5)


delta.estimate(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, t = 5, std = TRUE, conf.int = TRUE)
```

---

delta.estimate.RMST *Helper function*

Description

Helper function; used by delta.estimate function

Usage

```
delta.estimate.RMST(xone, xzero, deltaone, deltazero, t, weight = NULL, delta.only = F)
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.
deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
t	time of interest for treatment effect.
weight	weight; optional
delta.only	TRUE or FALSE; if TRUE then only delta is returned, if FALSE then delta and restricted mean survival time for each treatment group is also returned

Value

A list is returned:

delta	the difference in the restricted mean survival between the two treatment groups.
s1	only if delta.only = F; the restricted mean survival in group 1.
s0	only if delta.only = F; the restricted mean survival in group 0.

Author(s)

Layla Parast

`delta.q.event.RMST` *Calculates the residual treatment effect (the difference in restricted mean survival time at time t) after accounting for the treatment effect on the surrogate outcome information up to the landmark time*

Description

Calculates the residual treatment effect (the difference in restricted mean survival time at time t) after accounting for the treatment effect on the surrogate outcome information up to the landmark time; uses nonparametric estimation.

Usage

```
delta.q.event.RMST(xone, xzero, deltaone, deltazero, sone, szero, t, weight = NULL,
 landmark = landmark, deltaslist = TRUE, transform = FALSE, extrapolate=TRUE,
 number, warn.extrapolate=TRUE)
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.
deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
sone	numeric vector, observed event times for the surrogate outcome in the treatment group.
szero	numeric vector, observed event times for the surrogate outcome in the control group.
t	time of interest for treatment effect.
weight	optional weight.
landmark	landmark time of interest, t_0.
deltaslist	TRUE or FALSE; if TRUE, each component of the residual treatment effect is returned along with the residual treatment effect itself, if FALSE, only the residual treatment effect is returned.

transform	TRUE or FALSE; indicates whether a transformation should be used, default is FALSE.
extrapolate	TRUE or FALSE; indicates whether local constant extrapolation should be used, default is TRUE.
number	number of points for RMST calculation, default is 40.
warn.extrapolate	TRUE or FALSE; indicates whether user prefers a warning message when extrapolation is used, default is TRUE.

Details

See documentation for R.q.event for details.

Value

A list is returned:

delta.q	the estimated residual treatment effect
first.term	the first term of the residual treatment effect, if deltaslist = TRUE
second.term	the second term of the residual treatment effect, if deltaslist = TRUE
third.term	the third term of the residual treatment effect, if deltaslist = TRUE

Author(s)

Layla Parast

References

Parast L, Tian L, and Cai T (2020). Assessing the Value of a Censored Surrogate Outcome. Lifetime Data Analysis, 26(2):245-265.

Examples

```
data(ExampleData)
names(ExampleData)

delta.q.event.RMST(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5, landmark=2,
number = 40)
delta.q.event.RMST(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 4, landmark=2,
number = 40)
```

`delta.q.event.semi.RMST` *Calculates the residual treatment effect (the difference in restricted mean survival time at time t) after accounting for the treatment effect on the surrogate outcome information up to the landmark time*

Description

Calculates the residual treatment effect (the difference in restricted mean survival time at time t) after accounting for the treatment effect on the surrogate outcome information up to the landmark time; uses semi-parametric estimation.

Usage

```
delta.q.event.semi.RMST(xone, xzero, deltaone, deltazero, sone, szero, t,
weight = NULL, landmark = landmark, deltaslist = TRUE, number)
```

Arguments

`xone`	numeric vector, observed event times for the primary outcome in the treatment group.
`xzero`	numeric vector, observed event times for the primary outcome in the control group.
`deltaone`	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
`deltazero`	numeric vector, event/censoring indicators for the primary outcome in the control group.
`sone`	numeric vector, observed event times for the surrogate outcome in the treatment group.
`szero`	numeric vector, observed event times for the surrogate outcome in the control group.
`t`	time of interest for treatment effect.
`weight`	optional weight.
`landmark`	landmark time of interest, t_0.
`deltaslist`	TRUE or FALSE; if TRUE, each component of the residual treatment effect is returned along with the residual treatment effect itself, if FALSE, only the residual treatment effect is returned.
`number`	number of points for RMST calculation, default is 40.

Details

See documentation for R.q.event for details.

Value

A list is returned:

delta.q	the estimated residual treatment effect
first.term	the first term of the residual treatment effect, if deltaslist = TRUE
second.term	the second term of the residual treatment effect, if deltaslist = TRUE
third.term	the third term of the residual treatment effect, if deltaslist = TRUE

Author(s)

Layla Parast

References

Parast L, Tian L, and Cai T (2020). Assessing the Value of a Censored Surrogate Outcome. Lifetime Data Analysis, 26(2):245-265.

Examples

```
data(ExampleData)
names(ExampleData)

delta.q.event.semi.RMST(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone =
ExampleData$delta1, deltazero = ExampleData$delta0, sone = ExampleData$s1,
szero = ExampleData$s0, t = 5, landmark=2, number = 40)
delta.q.event.semi.RMST(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone =
ExampleData$delta1, deltazero = ExampleData$delta0, sone = ExampleData$s1,
szero = ExampleData$s0, t = 3, landmark=2, number = 40)
```

---

delta.t.RMST *Calculates the residual treatment effect (the difference in restricted mean survival time at time t) after accounting for the treatment effect on the primary outcome up to the landmark time*

Description

Calculates the residual treatment effect (the difference in restricted mean survival time at time t) after accounting for the treatment effect on the primary outcome up to the landmark time

Usage

```
delta.t.RMST(xone, xzero, deltaone, deltazero, t, weight = NULL, landmark = landmark)
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.

deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
t	time of interest for treatment effect.
weight	optional weight.
landmark	landmark time of interest, t_0.

Details

See documentation for R.t.estimate for details.

Value

delta.t	the estimated residual treatment effect after accounting for the treatment effect on the primary outcome up to the landmark time

Author(s)

Layla Parast

References

Parast L, Tian L, and Cai T (2020). Assessing the Value of a Censored Surrogate Outcome. Lifetime Data Analysis, 26(2):245-265.

Examples

```
data(ExampleData)
names(ExampleData)
delta.t.RMST(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, t = 5, landmark=2)
```

ExampleData *Hypothetical data*

Description

Hypothetical data to be used in examples; t=5 and the landmark time = 2.

Usage

```
data(ExampleData)
```

Format

A list with 6 elements representing 1000 observations from a control group and 1000 observations from a treatment group:

`s1` Time of the occurrence of the surrogate outcome for treated observations.

`x1` The observed event or censoring time for treated observations; X = min(T, C) where T is the time of the primary outcome and C is the censoring time.

`delta1` The indicator identifying whether the treated observation was observed to have the event or was censored; D =1*(T<C) where T is the time of the primary outcome and C is the censoring time.

`s0` Time of the occurrence of the surrogate outcome for control observations.

`x0` The observed event or censoring time for control observations; X = min(T, C) where T is the time of the primary outcome and C is the censoring time.

`delta0` The indicator identifying whether the control observation was observed to have the event or was censored; D =1*(T<C) where T is the time of the primary outcome and C is the censoring time.

Details

Note that the time of the surrogate outcome is used in all functions only if the surrogate outcome occurs before the minimum of the event time and censoring time.

Examples

```
data(ExampleData)
names(ExampleData)
```

---

helper.si *Helper function*

Description

Helper function; should not be called directly by user.

Usage

```
helper.si(yy,FUN,Yi,Vi=NULL)
```

Arguments

yy	yy
FUN	FUN
Yi	Yi
Vi	Vi

Value

out	matrix

Author(s)

Layla Parast

IV.event *Calculates the incremental value of the surrogate outcome information*

Description

Calculates the incremental value of the surrogate outcome information

Usage

```
IV.event(xone, xzero, deltaone, deltazero, sone, szero, t, landmark, number = 40,
transform = FALSE, extrapolate = TRUE, std = FALSE, conf.int = FALSE,
weight.perturb = NULL, type = "np")
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.
deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
sone	numeric vector, observed event times for the surrogate outcome in the treatment group.
szero	numeric vector, observed event times for the surrogate outcome in the control group.
t	time of interest for treatment effect.
landmark	landmark time of interest, t_0.
number	number of points for RMST calculation, default is 40.
transform	TRUE or FALSE; indicates whether a transformation should be used, default is FALSE.
extrapolate	TRUE or FALSE; indicates whether local constant extrapolation should be used, default is FALSE.
std	TRUE or FALSE; indicates whether standard error estimates should be provided, default is FALSE. Estimates are calculated using perturbation-resampling. Two versions are provided: one that takes the standard deviation of the perturbed estimates (denoted as "sd") and one that takes the median absolute deviation (denoted as "mad").
conf.int	TRUE or FALSE; indicates whether 95% confidence intervals should be provided. Confidence intervals are calculated using the percentiles of perturbed estimates, default is FALSE. If this is TRUE, standard error estimates are automatically provided.
weight.perturb	weights used for perturbation resampling.
type	Type of estimate that should be provided; options are "np" for the nonparametric estimate or "semi" for the semiparametric estimate, default is "np".

Details

The incremental value of the surrogate outcome information only is quantified as $IV_S(t, t_0) = R_Q(t, t_0) - R_T(t, t_0)$ where the definition and estimation procedures for $R_Q(t, t_0)$ and $R_T(t, t_0)$ are described in the documentation for R.q.event and R.t.estimate, respectively. The estimate of the incremental value is $\hat{IV}_S(t, t_0) = \hat{R}_Q(t, t_0) - \hat{R}_T(t, t_0)$.

Value

A list is returned:

`delta`	the estimate, $\hat{\Delta}(t)$, described in delta.estimate documentation.
`delta.q`	the estimate, $\hat{\Delta}_Q(t, t_0)$, described in R.q.event documention.
`R.q`	the estimate, $\hat{R}_Q(t, t_0)$, described in R.q.event documention.
`delta.t`	the estimate, $\hat{\Delta}_T(t, t_0)$, described in R.t.estimate documention.
`R.t`	the estimate, $\hat{R}_T(t, t_0)$, described in R.t.estimate documention.
`IV`	the estimated incremental value of the surrogate outcome information, described above.
`delta.sd`	the standard error estimate of $\hat{\Delta}(t)$; if std = TRUE or conf.int = TRUE.
`delta.mad`	the standard error estimate of $\hat{\Delta}(t)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`delta.q.sd`	the standard error estimate of $\hat{\Delta}_Q(t, t_0)$; if std = TRUE or conf.int = TRUE.
`delta.q.mad`	the standard error estimate of $\hat{\Delta}_Q(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`R.q.sd`	the standard error estimate of $\hat{R}_Q(t, t_0)$; if std = TRUE or conf.int = TRUE.
`R.q.mad`	the standard error estimate of $\hat{R}_Q(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`delta.t.sd`	the standard error estimate of $\hat{\Delta}_T(t, t_0)$; if std = TRUE or conf.int = TRUE.
`delta.t.mad`	the standard error estimate of $\hat{\Delta}_T(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`R.t.sd`	the standard error estimate of $\hat{R}_T(t, t_0)$; if std = TRUE or conf.int = TRUE.
`R.t.mad`	the standard error estimate of $\hat{R}_T(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`IV.sd`	the standard error estimate of the incremental value; if std = TRUE or conf.int = TRUE.
`IV.mad`	the standard error estimate of the incremental value using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`conf.int.delta`	a vector of size 2; the 95% confidence interval for $\hat{\Delta}(t)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.
`conf.int.delta.q`	a vector of size 2; the 95% confidence interval for $\hat{\Delta}_Q(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.
`conf.int.R.q`	a vector of size 2; the 95% confidence interval for $\hat{R}_Q(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.
`conf.int.delta.t`	a vector of size 2; the 95% confidence interval for $\hat{\Delta}_T(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.

conf.int.R.t	a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.
conf.int.IV	a vector of size 2; the 95% confidence interval for the incremental value based on sample quantiles of the perturbed values; if conf.int = TRUE.

Author(s)

Layla Parast

References

Parast L, Tian L, and Cai T (2020). Assessing the Value of a Censored Surrogate Outcome. Lifetime Data Analysis, 26(2):245-265.

Examples

```
data(ExampleData)
names(ExampleData)


IV.event(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5,
 landmark=2, type = "np")

IV.event(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5,
 landmark=2, type = "np", std = TRUE, conf.int = TRUE)
```

Kern.FUN *Calculates kernel matrix*

Description

Helper function; this calculates the kernel matrix

Usage

```
Kern.FUN(zz, zi, bw, kern0)
```

Arguments

zz	zz
zi	zi
bw	bandwidth
kern0	kernel distribution

Value

the kernel matrix

Author(s)

Layla Parast

new.q *Helper function*

Description

Helper function; should not be called directly by user. This function differs from the quantile function in that it returns an NA when there are NAs in the supplied vector, rather than causing an error.

Usage

```
new.q(x, p)
```

Arguments

x	vector of numeric values
p	sample quantile that is desired

Value

returns either NA or the desired sample quantile

Author(s)

Layla Parast

pred.smooth.surv *Calculates the conditional probability of survival for control group values*

Description

Helper function; calculates the estimated probability of survival for control control group surrogate values using treatment group surrogate and outcome information.

Usage

```
pred.smooth.surv(xone.f, deltaone.f, sone.f, szero.one, myt, bw = NULL,
weight, transform, extrapolate = T)
```

Arguments

xone.f	observed event times in the treatment group.
deltaone.f	censoring indicators in the treatment group.
sone.f	surrogate marker values in the treatment group.
szero.one	surrogate marker values in the control group.
myt	time of interest.
bw	bandwidth.

weight	weight used for perturbation resampling.
transform	TRUE or FALSE; indicates whether a transformation should be used, default is FALSE.
extrapolate	TRUE or FALSE; indicates whether local constant extrapolation should be used, default is TRUE.

Value

A list is returned:

Phat.ss	conditional probability of survival past t for control group
warn.flag	warning flag equal to 1 if extrapolation was used; 0 otherwise

Author(s)

Layla Parast

R.opt.event *Calculates the proportion of the treatment effect (the difference in survival at time t) explained by surrogate outcome information observed up to the landmark time, using the optimal transformation of the surrogate*

Description

Calculates the proportion of the treatment effect (the difference in survival at time t) explained by surrogate outcome information observed up to the landmark time, using the optimal transformation of the surrogate; also provides standard error estimate and confidence interval. Details are provided in: Wang X, Cai T, Tian L, Parast L (2025). Model-free Approach to Evaluate a Censored Intermediate Outcome as a Surrogate for Overall Survival. arXiv preprint arXiv:2412.14129.

Usage

```
R.opt.event(xone, xzero, deltaone, deltazero, sone, szero, t, landmark, std = FALSE,
conf.int = FALSE, gopt = FALSE, ind = FALSE)
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.
deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
sone	numeric vector, observed event times for the surrogate outcome in the treatment group.
szero	numeric vector, observed event times for the surrogate outcome in the control group.

t	time of interest for treatment effect.
landmark	landmark time of interest, t_0.
std	TRUE or FALSE; indicates whether standard error estimates should be provided, default is FALSE. Estimates are calculated using perturbation-resampling.
conf.int	TRUE or FALSE; indicates whether a 95% confidence interval for the PTE should be provided. Confidence intervals are calculated using the percentiles of perturbed estimates, default is FALSE.
gopt	TRUE or FALSE; indicates whether the estimates of the optimal g1(s) and g2 should be provided, default is FALSE.
ind	TRUE or FALSE; indicates whether the estimate of PTE for primary outcome information only up to the landmark time should be provided, default is FALSE.

Value

A list is returned:

R.opt	the estimate of the PTE of the surrogate outcome
R.opt.ind	the estimate of the PTE for primary outcome information only up to the landmark time; if ind=TRUE
g1.opt	the estimate of the optimal transformation g1(s); if gopt=TRUE
g2.opt	the estimate of the optimal transformation g2; if gopt=TRUE
R.opt.std	the standard error estimate of the PTE of the surrogate outcome; if std=TRUE
conf.int.R	the confidence interval for the PTE of the surrogate outcome; if cont.int=TRUE
g1.opt.sd	the standard error estimate of the optimal transformation g1(s); if gopt=TRUE and std=TRUE
g2.opt.sd	the standard error estimate of the optimal transformation g2; if gopt=TRUE and std=TRUE

Author(s)

Xuan Wang

References

Wang X, Cai T, Tian L, Parast L (2025). Model-free Approach to Evaluate a Censored Intermediate Outcome as a Surrogate for Overall Survival. arXiv preprint arXiv:2412.14129.

Examples

```
data(ExampleData)
names(ExampleData)

R.opt.event(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5,
landmark=2)

#with all options as TRUE
R.opt.event(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5,
landmark=2, std = TRUE, conf.int = TRUE, gopt = TRUE, ind = TRUE)
```

R.q.event *Calculates the proportion of the treatment effect (the difference in restriced mean survival time at time t) explained by surrogate outcome information observed up to the landmark time*

Description

Calculates the proportion of the treatment effect (the difference in restriced mean survival time at time t) explained by surrogate outcome information observed up to the landmark time; also provides standard error estimate and confidence interval.

Usage

```
R.q.event(xone, xzero, deltaone, deltazero, sone, szero, t, landmark, number = 40,
transform = FALSE, extrapolate = TRUE, std = FALSE, conf.int = FALSE,
weight.perturb = NULL, type = "np")
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.
deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
sone	numeric vector, observed event times for the surrogate outcome in the treatment group.
szero	numeric vector, observed event times for the surrogate outcome in the control group.
t	time of interest for treatment effect.
landmark	landmark time of interest, t_0.
number	number of points for RMST calculation, default is 40.
transform	TRUE or FALSE; indicates whether a transformation should be used, default is FALSE.
extrapolate	TRUE or FALSE; indicates whether local constant extrapolation should be used, default is FALSE.
std	TRUE or FALSE; indicates whether standard error estimates should be provided, default is FALSE. Estimates are calculated using perturbation-resampling. Two versions are provided: one that takes the standard deviation of the perturbed estimates (denoted as "sd") and one that takes the median absolute deviation (denoted as "mad").
conf.int	TRUE or FALSE; indicates whether 95% confidence intervals should be provided. Confidence intervals are calculated using the percentiles of perturbed estimates, default is FALSE. If this is TRUE, standard error estimates are automatically provided.

`weight.perturb`	weights used for perturbation resampling.
`type`	Type of estimate that should be provided; options are "np" for the nonparametric estimate or "semi" for the semiparametric estimate, default is "np".

Details

Let $G \in \{1, 0\}$ be the randomized treatment indicator, T denote the time of the primary outcome of interest, and S denote the time of the surrogate outcome. We use potential outcomes notation such that $T^{(G)}$ and $S^{(G)}$ denote the respective times of the primary and surrogate outcomes under treatment G, for $G \in \{1, 0\}$. In the absence of censoring, we only observe $(T, S) = (T^{(1)}, S^{(1)})$ or $(T^{(0)}, S^{(0)})$ for each individual depending on whether $G = 1$ or 0. Due to censoring, data consist of $n = n_1 + n_0$ independent observations $\{X_{gi}, \delta_{gi}, I(S_{gi} < t_0)I(X_{gi} > t_0), S_{gi} \wedge t_0 I(X_{gi} > t_0), i = 1, ..., n_g, g = 1, 0\}$, where $X_{gi} = T_{gi} \wedge C_{gi}$, $\delta_{gi} = I(T_{gi} < C_{gi})$, C_{gi} denotes the censoring time, T_{gi} denotes the time of the primary outcome, S_{gi} denotes the time of the surrogate outcome, $\{(T_{gi}, C_{gi}, S_{gi}), i = 1, ..., n_g\}$ are identically distributed within treatment group, and t_0 is the landmark time of interest.

We define the treatment effect as the difference in restricted mean survival time up to a fixed time t under treatment 1 versus under treatment 0,

$$\Delta(t) = E\{T^{(1)} \wedge t\} - E\{T^{(0)} \wedge t\}$$

where $\wedge$ indicates the minimum. To define the proportion of treatment effect explained by the surrogate outcome information, let

$$Q_{t_0}^{(g)} = (Q_{t_0 1}, Q_{t_0 2})' = \{S^{(g)} \wedge t_0 I(T^{(g)} > t_0), T^{(g)} I(T^{(g)} \leq t_0)\}', g = 1, 0$$

and define the residual treatment effect after accounting for the treatment effect on the surrogate outcome information as:

$$\Delta_Q(t, t_0) = P_{t_0,2}^{(0)} \int_0^{t_0} \phi_1(t|t_0, s) dF_0(s) + P_{t_0,3}^{(0)} \psi_1(t|t_0) - P(T^{(0)} > t_0)\nu_0(t|t_0)$$

where $P_{t_0,2}^{(0)} = P(T^{(0)} > t_0, S^{(0)} < t_0)$ and $P_{t_0,3}^{(0)} = P(T^{(0)} > t_0, S^{(0)} > t_0)$, $\psi_1(t \mid t_0) = E(T^{(1)} \wedge t \mid T^{(1)} > t_0, S^{(1)} > t_0)$, $\phi_1(t \mid t_0, s) = E(T^{(1)} \wedge t \mid T^{(1)} > t_0, S^{(1)} = s)$, $\nu_0(t|t_0) = E(T^{(0)} \wedge t | T^{(0)} > t_0)$, and $F_0(\cdot \mid t_0)$ is the cumulative distribution function of $S^{(0)}$ conditional on $T^{(0)} > t_0$ and $S^{(0)} < t_0$. Then, the proportion of treatment effect on the primary outcome that is explained by surrogate information up to t_0, Q_{t_0}, can be expressed as a contrast between $\Delta(t)$ and $\Delta_Q(t, t_0)$:

$$R_Q(t, t_0) = \{\Delta(t) - \Delta_Q(t, t_0)\}/\Delta(t) = 1 - \Delta_Q(t, t_0)/\Delta(t).$$

The quantity $\Delta(t)$ is estimated using inverse probability of censoring weights:

$$\hat{\Delta}(t) = n_1^{-1} \sum_{i=1}^{n_1} \hat{M}_{1i}(t) - n_0^{-1} \sum_{i=1}^{n_0} \hat{M}_{0i}(t)$$

where $\hat{M}_{gi}(t) = I(X_{gi} > t)t/\hat{W}_g^C(t) + I(X_{gi} < t)X_{gi}\delta_{gi}/\hat{W}_g^C(X_{gi})$ and $\hat{W}_g^C(t)$ is the Kaplan-Meier estimator of $P(C_{gi} \geq t)$. The residual treatment effect $\Delta_Q(t, t_0)$ can be estimated non-parametrically or semi-parametrically. For nonparametric estimation, $\psi_1(t|t_0)$ is estimated by $\hat{\psi}_1(t|t_0) = \sum_{i=1}^{n_1} \frac{\hat{W}_1^C(t_0) I(S_{1i} > t_0, X_{1i} > t_0)}{\sum_{i=1}^{n_1} I(S_{1i} > t_0, X_{1i} > t_0)} \hat{M}_{1i}(t)$, and $\phi_1(t \mid t_0, s) = E(T^{(1)} \wedge t \mid X^{(1)} > t_0, S^{(1)} = s)$ is estimated using a nonparametric kernel Nelson-Aalen estimator for $\Lambda_1(t \mid t_0, s)$, the cumulative hazard function of $T^{(1)}$ conditional on $S^{(1)} = s$ and $T^{(1)} > t_0$, as

$$\hat{\phi}_1(t \mid t_0, s) = t_0 + \int_{t_0}^{t} \exp\{-\hat{\Lambda}_1(t \mid t_0, s)\} dt,$$

where

$$\hat{\Lambda}_1(t \mid t_0, s) = \int_{t_0}^{t} \frac{\sum_{i=1}^{n_1} I(X_{1i} > t_0, S_{1i} < t_0) K_h\{\gamma(S_{1i}) - \gamma(s)\} dN_{1i}(z)}{\sum_{i=1}^{n_1} I(X_{1i} > t_0, S_{1i} < t_0) K_h\{\gamma(S_{1i}) - \gamma(s)\} Y_{1i}(z)},$$

is a consistent estimate of $\Lambda_1(t \mid t_0, s)$, $Y_{1i}(t) = I(X_{1i} \geq t)$, $N_{1i}(t) = I(X_{1i} \leq t)\delta_i$, $K(\cdot)$ is a smooth symmetric density function, $K_h(x) = K(x/h)/h$, $\gamma(\cdot)$ is a given monotone transformation function, and $h = O(n_1^{-\eta})$ is a specified bandwidth with $\eta \in (1/2, 1/4)$. Finally, we let

$$\hat{\nu}_0(t|t_0) = \sum_{i=1}^{n_0} \frac{\hat{W}_0^C(t_0) I(X_{0i} > t_0)}{\sum_{i=1}^{n_0} I(X_{0i} > t_0)} \hat{M}_{0i}(t).$$

We then estimate $\Delta_Q(t, t_0)$ as $\hat{\Delta}_Q(t, t_0)$ defined as

$$n_0^{-1} \sum_{i=1}^{n_0} \left\{ \frac{I_{t_0,2}(X_{0i}, S_{0i}) \hat{\phi}_1(t \mid t_0, S_{0i}) + I_{t_0,3}(X_{0i}, S_{0i}) \hat{\psi}_1(t \mid t_0) - I_{t_0}(X_{0i}) \hat{\nu}(t|t_0)}{\hat{W}_0^C(t_0)} \right\}$$

where $I_{t_0,2}(x, s) = I(x > t_0, s < t_0)$ and $I_{t_0,3}(x, s) = I(x > t_0, s > t_0)$ and $I_{t_0}(x) = I(x > t_0)$ and thus, $\hat{R}_Q(t, t_0) = 1 - \hat{\Delta}_Q(t, t_0)/\hat{\Delta}(t)$.

For the semi-parametric estimate, $\hat{\phi}_1(t|t_0, s)$ is replaced with an estimate obtained using a landmark Cox proportional hazards model

$$P(T^{(1)} > t \mid T^{(1)} > t_0, S^{(1)} < t_0, S^{(1)}) = \exp\{-\Lambda_0(t|t_0) \exp(\beta_0 S^{(1)})\}$$

where $\Lambda_0(t|t_0)$ is the unspecified baseline cumulative hazard among $\Omega_{t_0} = \{T^{(1)} > t_0, S^{(1)} < t_0\}$ and β_0 is unknown. That is, let $\tilde{\phi}_1(t|t_0, s) = t_0 + \int_{t_0}^{t} \exp\{-\hat{\Lambda}_0(t|t_0) \exp(\hat{\beta} s)\} dt$, where $\hat{\beta}$ is estimated by fitting a Cox model to the subpopulation Ω_{t_0} with a single predictor S and $\hat{\Lambda}_0(\cdot|t_0)$ is the corresponding Breslow estimator. Then the semiparametric estimator for $\Delta_Q(t, t_0)$ is $\tilde{\Delta}_Q(t, t_0)$ defined as

$$n_0^{-1} \sum_{i=1}^{n_0} \left\{ \frac{I_{t_0,2}(X_{0i}, S_{0i}) \tilde{\phi}_1(t \mid t_0, S_{0i}) + I_{t_0,3}(X_{0i}, S_{0i}) \hat{\psi}_1(t \mid t_0) - I_{t_0}(X_{0i}) \hat{\nu}(t|t_0)}{\hat{W}_0^C(t_0)} \right\}$$

and $\tilde{R}_Q(t, t_0) = 1 - \tilde{\Delta}_Q(t, t_0)/\hat{\Delta}(t)$.

Value

A list is returned:

`delta`	the estimate, $\hat{\Delta}(t)$, described in delta.estimate documentation.
`delta.q`	the estimate, $\hat{\Delta}_Q(t, t_0)$, described above.
`R.q`	the estimate, $\hat{R}_Q(t, t_0)$, described above.
`delta.sd`	the standard error estimate of $\hat{\Delta}(t)$; if std = TRUE or conf.int = TRUE.
`delta.mad`	the standard error estimate of $\hat{\Delta}(t)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`delta.q.sd`	the standard error estimate of $\hat{\Delta}_Q(t, t_0)$; if std = TRUE or conf.int = TRUE.
`delta.q.mad`	the standard error estimate of $\hat{\Delta}_Q(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`R.q.sd`	the standard error estimate of $\hat{R}_Q(t, t_0)$; if std = TRUE or conf.int = TRUE.

`R.q.mad` the standard error estimate of $\hat{R}_Q(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.

`conf.int.delta` a vector of size 2; the 95% confidence interval for $\hat{\Delta}(t)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.

`conf.int.delta.q` a vector of size 2; the 95% confidence interval for $\hat{\Delta}_Q(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.

`conf.int.R.q` a vector of size 2; the 95% confidence interval for $\hat{R}_Q(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.

Author(s)

Layla Parast

References

Parast L, Tian L, and Cai T (2020). Assessing the Value of a Censored Surrogate Outcome. Lifetime Data Analysis, 26(2):245-265.

Parast, L and Cai, T (2013). Landmark risk prediction of residual life for breast cancer survival. Statistics in Medicine, 32(20), 3459-3471.

Examples

```
data(ExampleData)
names(ExampleData)


R.q.event(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5,
landmark=2, type = "np")
R.q.event(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5,
landmark=2, type = "semi")
R.q.event(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, sone = ExampleData$s1, szero = ExampleData$s0, t = 5,
landmark=2, type = "np", std = TRUE, conf.int = TRUE)
```

R.t.estimate *Calculates the proportion of the treatment effect (the difference in restricted mean survival time at time t) explained by primary outcome information observed up to the landmark time*

Description

Calculates the proportion of the treatment effect (the difference in restricted mean survival time at time t) explained by primary outcome information observed up to the landmark time; also provides standard error estimate and confidence interval.

Usage

```
R.t.estimate(xone, xzero, deltaone, deltazero, t, landmark, std = FALSE, conf.int
 = FALSE, weight.perturb = NULL)
```

Arguments

xone	numeric vector, observed event times for the primary outcome in the treatment group.
xzero	numeric vector, observed event times for the primary outcome in the control group.
deltaone	numeric vector, event/censoring indicators for the primary outcome in the treatment group.
deltazero	numeric vector, event/censoring indicators for the primary outcome in the control group.
t	time of interest for treatment effect.
landmark	landmark time of interest, t_0.
std	TRUE or FALSE; indicates whether standard error estimates should be provided, default is FALSE. Estimates are calculated using perturbation-resampling. Two versions are provided: one that takes the standard deviation of the perturbed estimates (denoted as "sd") and one that takes the median absolute deviation (denoted as "mad").
conf.int	TRUE or FALSE; indicates whether 95% confidence intervals should be provided. Confidence intervals are calculated using the percentiles of perturbed estimates, default is FALSE. If this is TRUE, standard error estimates are automatically provided.
weight.perturb	weights used for perturbation resampling.

Details

Let $G \in \{1, 0\}$ be the randomized treatment indicator, T denote the time of the primary outcome of interest, and S denote the time of the surrogate outcome. We use potential outcomes notation such that $T^{(G)}$ and $S^{(G)}$ denote the respective times of the primary and surrogate outcomes under treatment G, for $G \in \{1, 0\}$. In the absence of censoring, we only observe $(T, S) = (T^{(1)}, S^{(1)})$ or $(T^{(0)}, S^{(0)})$ for each individual depending on whether $G = 1$ or 0. Due to censoring, data consist of $n = n_1 + n_0$ independent observations $\{X_{gi}, \delta_{gi}, I(S_{gi} < t_0)I(X_{gi} > t_0), S_{gi} \wedge t_0 I(X_{gi} > t_0), i = 1, ..., n_g, g = 1, 0\}$, where $X_{gi} = T_{gi} \wedge C_{gi}$, $\delta_{gi} = I(T_{gi} < C_{gi})$, C_{gi} denotes the censoring time, T_{gi} denotes the time of the primary outcome, S_{gi} denotes the time of the surrogate outcome, $\{(T_{gi}, C_{gi}, S_{gi}), i = 1, ..., n_g\}$ are identically distributed within treatment group, and t_0 is the landmark time of interest.

The proportion of treatment effect explained by primary outcome information observed up to the landmark time, t_0, is defined as $R_T(t, t_0) = 1 - \Delta_T(t, t_0)/\Delta(t)$ where

$$\Delta_T(t, t_0) = P(T^{(0)} > t_0)E\{T^{(1)} \wedge t - T^{(0)} \wedge t \mid T > t_0\}$$

and $\Delta(t)$ is the treatment effect on the primary outcome, defined in the documentation for delta.estimate. The quantity $\Delta_T(t, t_0)$ is estimated using

$$\hat{\Delta}_T(t, t_0) = n_0^{-1} \sum_{i=1}^{n_0} I(X_{0i} > t_0)/\hat{W}_0^C(t_0)\{\hat{\nu}_1(t|t_0) - \hat{\nu}_0(t|t_0)\}$$

where $\hat{W}_0^C(t)$ is the Kaplan-Meier estimator of $P(C_{gi} \geq t)$, $\hat{\nu}_0(t|t_0)$ is defined in the documentation for R.q.event and $\hat{\nu}_1(t|t_0)$ is obtained by replacing 0 with 1.

Value

A list is returned:

`delta`	the estimate, $\hat{\Delta}(t)$, described in delta.estimate documentation.
`delta.t`	the estimate, $\hat{\Delta}_T(t, t_0)$, described above.
`R.t`	the estimate, $\hat{R}_T(t, t_0)$, described above.
`delta.sd`	the standard error estimate of $\hat{\Delta}(t)$; if std = TRUE or conf.int = TRUE.
`delta.mad`	the standard error estimate of $\hat{\Delta}(t)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`delta.t.sd`	the standard error estimate of $\hat{\Delta}_T(t, t_0)$; if std = TRUE or conf.int = TRUE.
`delta.t.mad`	the standard error estimate of $\hat{\Delta}_T(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`R.t.sd`	the standard error estimate of $\hat{R}_T(t, t_0)$; if std = TRUE or conf.int = TRUE.
`R.t.mad`	the standard error estimate of $\hat{R}_T(t, t_0)$ using the median absolute deviation; if std = TRUE or conf.int = TRUE.
`conf.int.delta`	a vector of size 2; the 95% confidence interval for $\hat{\Delta}(t)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.
`conf.int.delta.t`	a vector of size 2; the 95% confidence interval for $\hat{\Delta}_T(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.
`conf.int.R.t`	a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)$ based on sample quantiles of the perturbed values; if conf.int = TRUE.

Author(s)

Layla Parast

References

Parast L, Tian L, and Cai T (2020). Assessing the Value of a Censored Surrogate Outcome. Lifetime Data Analysis, 26(2):245-265.

Examples

```
data(ExampleData)
names(ExampleData)


R.t.estimate(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone =ExampleData$delta1,
deltazero = ExampleData$delta0, t = 5, landmark=2)

R.t.estimate(xone = ExampleData$x1, xzero = ExampleData$x0, deltaone = ExampleData$delta1,
deltazero = ExampleData$delta0, t = 5, landmark=2, std = TRUE, conf.int = TRUE)
```

resam *Resampling for standard error estimation*

Description

Resamples data to obtain standard error estimates

Usage

```
resam(vv, t, t.0, tt, data, data1, data2, indexindex)
```

Arguments

vv	random exponential weight
t	number, the time of interest
t.0	landmark time of interest, t_0
tt	set of time points between t.0 and t
data	full dataset
data1	a random half from data
data2	the remaining random half from data
indexindex	the indices of data contained within data1

Value

A vector with the first value being the resampled estimate of the PTE, the second value being the resampled estimate of the optimal g2, and the remaining values being the resampled estimate of the optimal g1(s)

Author(s)

Xuan Wang

VTM *Repeats a row*

Description

Helper function; this function creates a matrix that repeats vc, dm times where each row is equal to the vc vector.

Usage

```
VTM(vc, dm)
```

Arguments

vc	the vector to repeat.
dm	number of rows.

Value

a matrix that repeats vc, dm times where each row is equal to the vc vector

---

WEIGHT *Weight function*

Description

Creates a matrix of weights and estimates to be used in estimation of the PTE within R.opt.event

Usage

```
WEIGHT(xob, deltaob, aob, n, t.0,t)
```

Arguments

xob	numeric vector, the observed event time: X = min(T, C) where T is the time of the primary outcome, C is the censoring time
deltaob	numeric vector of 0/1, the censoring indicator: D = I(T<C) where T is the time of the primary outcome, C is the censoring time
aob	numeric vector of 0/1, indicating treatment group
n	sample size
t.0	landmark time of interest, t_0
t	time of interest for treatment effect.

Value

Returns a matrix of weights and estimates to be used in estimation of the PTE within R.opt.event

Author(s)

Xuan Wang

WEIGHT.p *Weight function for resampling*

Description

Creates a matrix of weights and estimates to be used in estimation of the standard error of the PTE within resam

Usage

```
WEIGHT.p(xob, deltaob, aob, n, v, t.0, t)
```

Arguments

xob	numeric vector, the observed event time: X = min(T, C) where T is the time of the primary outcome, C is the censoring time
deltaob	numeric vector of 0/1, the censoring indicator: D = I(T<C) where T is the time of the primary outcome, C is the censoring time
aob	numeric vector of 0/1, indicating treatment group
n	sample size
v	random exponential weight
t.0	landmark time of interest, t_0
t	time of interest for treatment effect.

Value

Returns a matrix of weights and estimates to be used in estimation of the PTE within resam

Author(s)

Xuan Wang

Index

∗ **RMST**
delta.estimate, 3
delta.q.event.RMST, 6
delta.q.event.semi.RMST, 8
delta.t.RMST, 9
R.q.event, 18
R.t.estimate, 21
∗ **arith**
cumsum2, 3
helper.si, 11
new.q, 15
∗ **array**
VTM, 24
∗ **datasets**
ExampleData, 10
∗ **internal**
censor.weight, 2
cumsum2, 3
delta.estimate.RMST, 5
helper.si, 11
Kern.FUN, 14
new.q, 15
pred.smooth.surv, 15
resam, 24
VTM, 24
WEIGHT, 25
WEIGHT.p, 25
∗ **nonparametric**
censor.weight, 2
delta.estimate, 3
delta.q.event.RMST, 6
delta.t.RMST, 9
IV.event, 12
Kern.FUN, 14
pred.smooth.surv, 15
R.q.event, 18
R.t.estimate, 21
∗ **robust**
Kern.FUN, 14
pred.smooth.surv, 15
∗ **smooth**
Kern.FUN, 14
pred.smooth.surv, 15
∗ **survival**
censor.weight, 2
delta.estimate, 3
delta.q.event.RMST, 6
delta.q.event.semi.RMST, 8
delta.t.RMST, 9
IV.event, 12
pred.smooth.surv, 15
R.q.event, 18
R.t.estimate, 21

censor.weight, 2
cumsum2, 3

delta.estimate, 3
delta.estimate.RMST, 5
delta.q.event.RMST, 6
delta.q.event.semi.RMST, 8
delta.t.RMST, 9

ExampleData, 10

helper.si, 11

IV.event, 12

Kern.FUN, 14

new.q, 15

pred.smooth.surv, 15

R.opt.event, 16
R.q.event, 18
R.t.estimate, 21
resam, 24

VTM, 24

WEIGHT, 25
WEIGHT.p, 25