Package ‘Rsurrogate’

July 15, 2019

Type Package

Title Robust Estimation of the Proportion of Treatment Effect Explained by Surrogate Marker Information

Version 2.1

Date 2019-07-10

Author Layla Parast

Maintainer Layla Parast <parast@rand.org>

Description Provides functions to estimate the proportion of treatment effect on the primary outcome that is explained by the treatment effect on the surrogate marker.

License GPL

Imports stats, survival, Matrix

NeedsCompilation no

R topics documented:

Aug.R.s.surv.estimate *Calculates the augmented estimator of the proportion of treatment effect explained by the surrogate marker information measured at a specified time and primary outcome information up to that specified time*

Description

This function calculates the augmented version of the proportion of treatment effect on the primary outcome explained by the surrogate marker information measured at t_0 and primary outcome information up to t_0. Variance estimates and 95 % confidence intervals for the augmented estimates are provided automatically; three versions of the confidence interval are provided: a normal approximation based interval, a quantile based interval and Fieller's confidence interval, all using perturbation-resampling. The user can also request an estimate of the incremental value of surrogate marker information.

Usage

```
Aug.R.s.surv.estimate(xone, xzero, deltaone, deltazero, sone, szero, t,
weight.perturb = NULL, landmark, extrapolate = FALSE, transform = FALSE,
basis.delta.one, basis.delta.zero, basis.delta.s.one = NULL,
basis.delta.s.zero = NULL, incremental.value = FALSE)
```

Arguments

xone	numeric vector, the observed event times in the treatment group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
xzero	numeric vector, the observed event times in the control group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
deltaone	numeric vector, the event indicators for the treatment group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
deltazero	numeric vector, the event indicators for the control group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
sone	numeric vector; surrogate marker measurement at t_0 for treated observations, assumed to be continuous. If $X_{1i} < t_0$, then the surrogate marker measurement should be NA.
szero	numeric vector; surrogate marker measurement at t_0 for control observations, assumed to be continuous. If $X_{1i} < t_0$, then the surrogate marker measurement should be NA.
t	the time of interest.
weight.perturb	weights used for perturbation resampling.
landmark	the landmark time t_0 or time of surrogate marker measurement.
extrapolate	TRUE or FALSE; indicates whether the user wants to use extrapolation.
transform	TRUE or FALSE; indicates whether the user wants to use a transformation for the surrogate marker.
basis.delta.one	either a vector of length n_1 or a matrix with n_1 rows; this is the basis transformation used for augmentation of $\hat{\Delta}(t)$ for treated observations only, all values must be numeric

`basis.delta.zero`
either a vector of length n_0 or a matrix with n_0 rows; this is the basis transformation used for augmentation of $\hat{\Delta}(t)$ for control observations only, all values must be numeric

`basis.delta.s.one`
either a vector of length n_1 or a matrix with n_1 rows; this is the basis transformation used for augmentation of $\hat{\Delta}_S(t, t_0)$ for treated observations only, all values must be numeric; default is to assume this is the same as basis.delta.one i.e. that the same basis transformation is used for both quantities

`basis.delta.s.zero`
either a vector of length n_0 or a matrix with n_0 rows; this is the basis transformation used for augmentation of $\hat{\Delta}_S(t, t_0)$ for control observations only, all values must be numeric; default is to assume this is the same as basis.delta.zero i.e. that the same basis transformation is used for both quantities

`incremental.value`
TRUE or FALSE; indicates whether the user would like to see the incremental value of the surrogate marker information, default is FALSE.

Details

Please see R.s.surv.estimate documention for details about the estimates before augmentation is performed. Recent work has shown that augmentation can lead to improvements in efficiency by taking advantage of the association between baseline information, denoted here as Z, and the primary outcome. This function calculates the augmented estimates of the quantities of interest. For example, the augmented version of $\hat{\Delta}(t)$ is defined as:

$$\hat{\Delta}(t)^{AUG} = \hat{\Delta}(t) + \gamma\{n_1^{-1}\sum_{i=1}^{n_1} h(Z_{1i}) - n_0^{-1}\sum_{i=1}^{n_0} h(Z_{0i})\}$$

where $Z_{gi}, i = 1, 2, \cdots, n_g$ are i.i.d. random vectors of baseline covariates from treatment group g and $h(\cdot)$ is a basis transformation given a priori. Due to treatment randomization, $\{n_1^{-1}\sum_{i=1}^{n_1} h(Z_{1i}) - n_0^{-1}\sum_{i=1}^{n_0} h(Z_{0i})\}$ converges to zero in probability as the sample size goes to infinity and thus the augmented estimator converges to the same limit as the original counterparts. The quantity γ is selected such that the variance of $\hat{\Delta}(t)^{AUG}$ is minimized. That is, $\gamma = (\Xi_{12})(\Xi_{22})^{-1}$ where

$$\Xi_{12} = \text{cov}\{\hat{\Delta}(t), n_1^{-1}\sum_{i=1}^{n_1} h(Z_{1i}) - n_0^{-1}\sum_{i=1}^{n_0} h(Z_{0i})\},$$

$$\Xi_{22} = \text{var}\{n_1^{-1}\sum_{i=1}^{n_1} h(Z_{1i}) - n_0^{-1}\sum_{i=1}^{n_0} h(Z_{0i})\}$$

and thus we can obtain $\hat{\Delta}(t)^{AUG}$ by replacing γ with a consistent estimator, $\hat{\gamma}$ obtained using perturbation-resampling. A similar approach is used to obtain $\hat{\Delta}_S(t)^{AUG}$ and thus construct

$$\hat{R}_S(t, t_0)^{AUG} = 1 - \frac{\hat{\Delta}_S(t, t_0)^{AUG}}{\hat{\Delta}(t)^{AUG}}.$$

Value

A list is returned:

`aug.delta` the estimate, $\hat{\Delta}(t)^{AUG}$.

`aug.delta.s` the estimate, $\hat{\Delta}_S(t, t_0)^{AUG}$.

`aug.R.s` the estimate, $\hat{R}_S(t, t_0)^{AUG}$.

`aug.delta.var` the variance estimate of $\hat{\Delta}(t)^{AUG}$.

`aug.delta.s.var` the variance estimate of $\hat{\Delta}_S(t, t_0)^{AUG}$.

`aug.R.s.var` the variance estimate of $\hat{R}_S(t, t_0)^{AUG}$.

`conf.int.normal.aug.delta` a vector of size 2; the 95% confidence interval for $\hat{\Delta}(t)^{AUG}$ based on a normal approximation.

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

`conf.int.normal.aug.delta.s` a vector of size 2; the 95% confidence interval for $\hat{\Delta}_S(t, t_0)^{AUG}$ based on a normal approximation.

`conf.int.quantile.aug.delta.s` a vector of size 2; the 95% confidence interval for $\hat{\Delta}_S(t, t_0)^{AUG}$ based on sample quantiles of the perturbed values.

`conf.int.normal.R.s` a vector of size 2; the 95% confidence interval for $\hat{R}_S(t, t_0)^{AUG}$ based on a normal approximation.

`conf.int.quantile.aug.R.s` a vector of size 2; the 95% confidence interval for $\hat{R}_S(t, t_0)^{AUG}$ based on sample quantiles of the perturbed values..

`conf.int.fieller.aug.R.s` a vector of size 2; the 95% confidence interval for $\hat{R}_S(t, t_0)^{AUG}$ based on Fieller's approach.

`aug.delta.t` the estimate, $\hat{\Delta}_T(t, t_0)^{AUG}$; if incremental.vaue = TRUE.

`aug.R.t` the estimate, $\hat{R}_T(t, t_0)^{AUG}$; if incremental.vaue = TRUE.

`aug.incremental.value` the estimate, $I\hat{V}_S(t, t_0)^{AUG}$; if incremental.vaue = TRUE.

`aug.delta.t.var` the variance estimate of $\hat{\Delta}_T(t, t_0)^{AUG}$; if incremental.vaue = TRUE.

`aug.R.t.var` the variance estimate of $\hat{R}_T(t, t_0)^{AUG}$; if incremental.vaue = TRUE.

`aug.incremental.value.var` the variance estimate of $I\hat{V}_S(t, t_0)^{AUG}$; if incremental.vaue = TRUE.

`aug.conf.int.normal.delta.t` a vector of size 2; the 95% confidence interval for $\hat{\Delta}_T(t, t_0)^{AUG}$ based on a normal approximation; if incremental.vaue = TRUE.

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

`aug.conf.int.normal.R.t` a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)^{AUG}$ based on a normal approximation; if incremental.vaue = TRUE.

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

`aug.conf.int.fieller.R.t`
: a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)^{AUG}$ based on Fieller's approach, described above; if incremental.vaue = TRUE.

`aug.conf.int.normal.iv`
: a vector of size 2; the 95% confidence interval for $\hat{IV}_S(t, t_0)^{AUG}$ based on a normal approximation; if incremental.vaue = TRUE.

`aug.conf.int.quantile.iv`
: a vector of size 2; the 95% confidence interval for $\hat{IV}_S(t, t_0)^{AUG}$ based on sample quantiles of the perturbed values; if incremental.vaue = TRUE.

Note

If the treatment effect is not significant, the user will receive the following message: "Warning: it looks like the treatment effect is not significant; may be difficult to interpret the residual treatment effect in this setting". If the treatment effect is negative, the user will receive the following message: "Warning: it looks like you need to switch the treatment groups" as this package assumes throughout that larger values of the event time are better. If the observed support of the surrogate marker for the control group is outside the observed support of the surrogate marker for the treatment group, the user will receive the following message: "Warning: observed supports do not appear equal, may need to consider a transformation or extrapolation".

Author(s)

Layla Parast

References

Tian L, Cai T, Zhao L,Wei L. On the covariate-adjusted estimation for an overall treatment difference with data from a randomized comparative clinical trial. Biostatistics 2012; 13(2): 256-273.

Garcia TP, Ma Y, Yin G. Efficiency improvement in a class of survival models through model-free covariate incorporation. Lifetime Data Analysis 2011; 17(4): 552-565.

Zhang M, Tsiatis AA, Davidian M. Improving efficiency of inferences in randomized clinical trials using auxiliary covariates. Biometrics 2008; 64(3): 707-715.

Parast L, Cai T and Tian L. Evaluating Surrogate Marker Information using Censored Data. Under Review.

Examples

```
#computationally intensive
#Aug.R.s.surv.estimate(xone = d_example_surv$x1, xzero = d_example_surv$x0,
#deltaone = d_example_surv$delta1, deltazero = d_example_surv$delta0,
#sone = d_example_surv$s1, szero = d_example_surv$s0, t=3, landmark = 1,
#basis.delta.one = d_example_surv$z1 , basis.delta.zero = d_example_surv$z0)
```

`delta.estimate` *Calculates treatment effect*

Description

This function calculates the treatment effect estimate, the difference in the average outcome in the treatment group minus the control group. This function is intended to be used for a fully observed continuous outcome. The user can also request a variance estimate, estimated using perturbating-resampling, and a 95% confidence interval. If a confidence interval is requested two versions are provided: a normal approximation based interval and a quantile based interval, both use perturbation-resampling.

Usage

```
delta.estimate(yone,yzero, var = FALSE, conf.int = FALSE, weight = NULL,
weight.perturb = NULL)
```

Arguments

`yone`	numeric vector; primary outcome for treated observations.
`yzero`	numeric vector; primary outcome for control observations.
`var`	TRUE or FALSE; indicates whether a variance estimate for delta is requested, default is FALSE.
`conf.int`	TRUE or FALSE; indicates whether a 95% confidence interval for delta is requested, default is FALSE.
`weight`	a n1+n0 by x matrix of weights where n1 = length of yone and n0 = length of yzero, default is null; generally not supplied by use but only used by other functions.
`weight.perturb`	a n1+n0 by x matrix of weights where n1 = length of yone and n0 = length of yzero, default is null; generally used for confidence interval construction and may be supplied by user.

Details

Let $Y^{(1)}$ and $Y^{(0)}$ denote the primary outcome under the treatment and primary outcome under the control,respectively. The treatment effect, Δ, is the expected difference in $Y^{(1)}$ compared to $Y^{(0)}$, $\Delta = E(Y^{(1)} - Y^{(0)})$. We estimate Δ as

$$\hat{\Delta} = n_1^{-1}\sum_{i=1}^{n_1} Y_{1i} - n_0^{-1}\sum_{i=1}^{n_0} Y_{0i}$$

where Y_{1i} is the observed primary outcome for person i in the treated group, Y_{0i} is the observed primary outcome for person i in the control group, and n_1 and n_0 are the number of individuals in the treatment and control group, respectively. Randomized treatment assignment is assumed throughout this package.

Variance estimation and confidence interval construction are performed using perturbation-resampling. Specifically, let $\left\{V^{(b)} = (V_{11}^{(b)}, ...V_{1n_1}^{(b)}, V_{01}^{(b)}, ...V_{0n_0}^{(b)})^T, b = 1, ..., D\right\}$ be $n \times D$ independent copies of a positive random variables V from a known distribution with unit mean and unit variance. Let

$$\hat{\Delta}^{(b)} = \frac{\sum_{i=1}^{n_1} V_{1i}^{(b)} Y_{1i}}{\sum_{i=1}^{n_1} V_{1i}^{(b)}} - \frac{\sum_{i=1}^{n_0} V_{0i}^{(b)} Y_{0i}}{\sum_{i=1}^{n_0} V_{0i}^{(b)}}.$$

The variance of $\hat{\Delta}$ is obtained as the empirical variance of $\{\hat{\Delta}^{(b)}, b = 1, ..., D\}$. In this package, we use weights generated from an Exponential(1) distribution and use $D = 500$. We construct two versions of the 95% confidence interval for $\hat{\Delta}$: one based on a normal approximation confidence interval using the estimated variance and another taking the 2.5th and 97.5th empirical percentiles of $\hat{\Delta}^{(b)}$.

Value

A list is returned:

`delta` the estimate, $\hat{\Delta}$, described above.

`var` the variance estimate of $\hat{\Delta}$; if var = TRUE or conf.int = TRUE.

`conf.int.normal`
a vector of size 2; the 95% confidence interval for $\hat{\Delta}$ based on a normal approximation; if conf.int = TRUE.

`conf.int.quantile`
a vector of size 2; the 95% confidence interval for $\hat{\Delta}$ based on sample quantiles of the perturbed values, described above; if conf.int = TRUE.

Author(s)

Layla Parast

Examples

```
data(d_example)
names(d_example)
delta.estimate(yone=d_example$y1, yzero=d_example$y0)
```

`delta.s.estimate` *Calculates model-based or robust residual treatment effect*

Description

This function calculates the model-based or robust estimate of the residual treatment effect i.e. the hypothetical treatment effect if the distribution of the surrogate in the treatment group looks like the distribution of the surrogate in the control group. Ideally, this function is only used as a helper function and is not directly called.

Usage

```
delta.s.estimate(sone, szero, yone, yzero, weight.perturb = NULL, number="single",
type="robust", warn.te = FALSE, warn.support = FALSE, extrapolate = FALSE,
transform = FALSE)
```

Arguments

`sone` numeric vector or matrix; surrogate marker for treated observations, assumed to be continuous. If there are multiple surrogates then this should be a matrix with n_1 (number of treated observations) rows and n.s (number of surrogate markers) columns.

szero	numeric vector or matrix; surrogate marker for control observations, assumed to be continuous. If there are multiple surrogates then this should be a matrix with n_0 (number of control observations) rows and n.s (number of surrogate markers) columns.
yone	numeric vector; primary outcome for treated observations.
yzero	numeric vector; primary outcome for control observations.
weight.perturb	a $n_1 + n_0$ by x matrix of weights where n_1 = length of yone and n_0 = length of yzero; generally used for variance estimation and confidence interval construction, default is null.
number	specifies the number of surrogate markers; choices are "multiple" or "single", default is "single".
type	specifies the type of estimation; choices are "robust" or "model", default is "robust".
warn.te	value passed from R.s.estimate function to control warnings; user does not need to specify.
warn.support	value passed from R.s.estimate function to control warnings; user does not need to specify.
extrapolate	TRUE or FALSE; indicates whether the user wants to use extrapolation.
transform	TRUE or FALSE; indicates whether the user wants to use a transformation for the surrogate marker.

Details

Details are included in the documentation for R.s.estimate.

Value

$\hat{\Delta}_S$, the model-based or robust residual treatment effect estimate.

Note

If the treatment effect is not significant, the user will receive the following message: "Warning: it looks like the treatment effect is not significant; may be difficult to interpret the residual treatment effect in this setting". If the treatment effect is negative, the user will receive the following message: "Warning: it looks like you need to switch the treatment groups" as this package assumes throughout that higher values are better. In the single marker case with the robust estimation approach, if the observed support of the surrogate marker for the control group is outside the observed support of the surrogate marker for the treatment group, the user will receive the following message: "Warning: observed supports do not appear equal, may need to consider a transformation or extrapolation".

Author(s)

Layla Parast

References

Parast, L., McDermott, M., Tian, L. (2015). Robust estimation of the proportion of treatment effect explained by surrogate marker information. Statistics in Medicine, 35(10):1637-1653.

Wang, Y., & Taylor, J. M. (2002). A measure of the proportion of treatment effect explained by a surrogate marker. Biometrics, 58(4), 803-812.

Examples

```
data(d_example)
names(d_example)
delta.s.estimate(yone=d_example$y1, yzero=d_example$y0, sone=d_example$s1.a, szero=
d_example$s0.a, number = "single", type = "robust")
delta.s.estimate(yone=d_example$y1, yzero=d_example$y0, sone=d_example$s1.a, szero=
d_example$s0.a, number = "single", type = "model")
delta.s.estimate(yone=d_example$y1, yzero=d_example$y0, sone=cbind(d_example$s1.a,
d_example$s1.b, d_example$s1.c), szero=cbind(d_example$s0.a, d_example$s0.b, d_example$s0.c),
number = "multiple", type = "robust")
delta.s.estimate(yone=d_example$y1, yzero=d_example$y0, sone=cbind(d_example$s1.a,
d_example$s1.b, d_example$s1.c), szero=cbind(d_example$s0.a, d_example$s0.b, d_example$s0.c),
number = "multiple", type = "model")
```

`delta.s.surv.estimate` *Calculates robust residual treatment effect accounting for surrogate marker information measured at a specified time and primary outcome information up to that specified time*

Description

This function calculates the robust estimate of the residual treatment effect accounting for surrogate marker information measured at t_0 and primary outcome information up to t_0 i.e. the hypothetical treatment effect if both the surrogate marker distribution at t_0 and survival up to t_0 in the treatment group look like the surrogate marker distribution and survival up to t_0 in the control group. Ideally this function is only used as a helper function and is not directly called.

Usage

```
delta.s.surv.estimate(xone, xzero, deltaone, deltazero, sone, szero, t,
weight.perturb = NULL, landmark, extrapolate = FALSE, transform = FALSE)
```

Arguments

`xone`	numeric vector, the observed event times in the treatment group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`xzero`	numeric vector, the observed event times in the control group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`deltaone`	numeric vector, the event indicators for the treatment group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`deltazero`	numeric vector, the event indicators for the control group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`sone`	numeric vector; surrogate marker measurement at t_0 for treated observations, assumed to be continuous. If $X_{1i} < t_0$, then the surrogate marker measurement should be NA.
`szero`	numeric vector; surrogate marker measurement at t_0 for control observations, assumed to be continuous. If $X_{1i} < t_0$, then the surrogate marker measurement should be NA.
`t`	the time of interest.

`weight.perturb`	weights used for perturbation resampling.
`landmark`	the landmark time t_0 or time of surrogate marker measurement.
`extrapolate`	TRUE or FALSE; indicates whether the user wants to use extrapolation.
`transform`	TRUE or FALSE; indicates whether the user wants to use a transformation for the surrogate marker.

Details

Details are included in the documentation for R.s.surv.estimate.

Value

$\hat{\Delta}_S(t, t_0)$, the robust residual treatment effect estimate accounting for surrogate marker information measured at t_0 and primary outcome information up to t_0.

Note

If the treatment effect is not significant, the user will receive the following message: "Warning: it looks like the treatment effect is not significant; may be difficult to interpret the residual treatment effect in this setting". If the treatment effect is negative, the user will receive the following message: "Warning: it looks like you need to switch the treatment groups" as this package assumes throughout that larger values of the event time are better. If the observed support of the surrogate marker for the control group is outside the observed support of the surrogate marker for the treatment group, the user will receive the following message: "Warning: observed supports do not appear equal, may need to consider a transformation or extrapolation".

Author(s)

Layla Parast

References

Parast L, Cai T and Tian L. Evaluating Surrogate Marker Information using Censored Data. Under Review.

Examples

```
data(d_example_surv)
names(d_example_surv)
```

delta.surv.estimate *Calculates treatment effect in a survival setting*

Description

This function calculates the treatment effect in the survival setting i.e. the difference in survival at time t between the treatment group and the control group. The user can also request a variance estimate, estimated using perturbating-resampling, and a 95% confidence interval. If a confidence interval is requested two versions are provided: a normal approximation based interval and a quantile based interval, both use perturbation-resampling.

Usage

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

Arguments

`xone`	numeric vector, the observed event times in the treatment group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`xzero`	numeric vector, the observed event times in the control group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`deltaone`	numeric vector, the event indicators for the treatment group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`deltazero`	numeric vector, the event indicators for the control group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`t`	the time of interest.
`var`	TRUE or FALSE; indicates whether a variance estimate for delta is requested, default is FALSE.
`conf.int`	TRUE or FALSE; indicates whether a 95% confidence interval for delta is requested, default is FALSE.
`weight`	a $n_1 + n_0$ by x matrix of weights where n_1 = sample size in treatment group and n_0 = sample size in the control group, default is null; generally not supplied by use but only used by other functions.
`weight.perturb`	a $n_1 + n_0$ by x matrix of weights where n_1 =sample size in treatment group and n_0 = sample size in the control group, default is null; generally used for confidence interval construction and may be supplied by user.

Details

Let G be the binary treatment indicator with $G = 1$ for treatment and $G = 0$ for control and we assume throughout that subjects are randomly assigned to a treatment group at baseline. Let T denote the time of the primary outcome of interest, death for example. We use potential outcomes notation such that $T^{(g)}$ denotes the time of the primary outcome under treatment $G = g$. We define the treatment effect, $\Delta(t)$, as the difference in survival rates by time t under treatment versus control,

$$\Delta(t) = E\{I(T^{(1)} > t)\} - E\{I(T^{(0)} > t)\} = P(T^{(1)} > t) - P(T^{(0)} > t)$$

where $t > t_0$

Due to censoring, our data consist of n_1 observations $\{(X_{1i}, \delta_{1i}), i = 1, ..., n_1\}$ from the treatment group $G = 1$ and n_0 observations $\{(X_{0i}, \delta_{0i}), i = 1, ..., n_0\}$ from the control group $G = 0$ where $X_{gi} = \min(T_{gi}, C_{gi})$, $\delta_{gi} = I(T_{gi} < C_{gi})$, and C_{gi} denotes the censoring time for $g = 1, 0$, for individual i. Throughout, we estimate the treatment effect $\Delta(t)$ as

$$\hat{\Delta}(t) = n_1^{-1} \sum_{i=1}^{n_1} \frac{I(X_{1i} > t)}{\hat{W}_1^C(t)} - n_0^{-1} \sum_{i=1}^{n_0} \frac{I(X_{0i} > t)}{\hat{W}_0^C(t)}$$

where $\hat{W}_g^C(\cdot)$ is the Kaplan-Meier estimator of survival for censoring for $g = 1, 0$.

Variance estimation and confidence interval construction are performed using perturbation-resampling. Specifically, let $\left\{V^{(b)} = (V_{11}^{(b)}, ...V_{1n_1}^{(b)}, V_{01}^{(b)}, ...V_{0n_0}^{(b)})^T, b = 1, ..., D\right\}$ be $n \times D$ independent copies of a positive random variables V from a known distribution with unit mean and unit variance. Let

$$\hat{\Delta}^{(b)}(t) = \frac{\sum_{i=1}^{n_1} V_{1i}^{(b)} I(X_{1i} > t)}{\sum_{i=1}^{n_1} V_{1i}^{(b)} \hat{W}_1^{C(b)}(t)} - \frac{\sum_{i=1}^{n_0} V_{0i}^{(b)} I(X_{0i} > t)}{\sum_{i=1}^{n_0} V_{0i}^{(b)} \hat{W}_0^{C(b)}(t)}.$$

In this package, we use weights generated from an Exponential(1) distribution and use $D = 500$. The variance of $\hat{\Delta}(t)$ is obtained as the empirical variance of $\{\hat{\Delta}(t)^{(b)}, b = 1, ..., D\}$. We construct two versions of the 95%confidence interval for $\hat{\Delta}(t)$: one based on a normal approximation confidence interval using the estimated variance and another taking the 2.5th and 97.5th empirical percentiles of $\hat{\Delta}(t)^{(b)}$.

Value

A list is returned:

`delta` the estimate, $\hat{\Delta}(t)$, described above.

`var` the variance estimate of $\hat{\Delta}(t)$; if var = TRUE or conf.int = TRUE.

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

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

Author(s)

Layla Parast

Examples

```
data(d_example_surv)
names(d_example_surv)
delta.surv.estimate(xone = d_example_surv$x1, xzero = d_example_surv$x0,
deltaone = d_example_surv$delta1, deltazero = d_example_surv$delta0, t = 3)
```

`delta.t.surv.estimate` *Calculates robust residual treatment effect accounting only for primary outcome information up to a specified time*

Description

This function calculates the robust estimate of the residual treatment effect accounting only for primary outcome information up to t_0 i.e. the hypothetical treatment effect if survival up to t_0 in the treatment group looks like survival up to t_0 in the control group. Ideally this function is only used as a helper function and is not directly called.

Usage

```
delta.t.surv.estimate(xone, xzero, deltaone, deltazero, t, weight.perturb = NULL,
landmark)
```

Arguments

`xone`	numeric vector, the observed event times in the treatment group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`xzero`	numeric vector, the observed event times in the control group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`deltaone`	numeric vector, the event indicators for the treatment group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`deltazero`	numeric vector, the event indicators for the control group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`t`	the time of interest.
`weight.perturb`	weights used for perturbation resampling.
`landmark`	the landmark time t_0 or time of surrogate marker measurement.

Details

Details are included in the documentation for R.t.surv.estimate.

Value

$\hat{\Delta}_T(t, t_0)$, the robust residual treatment effect estimate accounting only for survival up to t_0.

Note

If the treatment effect is not significant, the user will receive the following message: "Warning: it looks like the treatment effect is not significant; may be difficult to interpret the residual treatment effect in this setting". If the treatment effect is negative, the user will receive the following message: "Warning: it looks like you need to switch the treatment groups" as this package assumes throughout that larger values of the event time are better.

Author(s)

Layla Parast

References

Parast L, Cai T and Tian L. Evaluating Surrogate Marker Information using Censored Data. Under Review.

Examples

```
data(d_example_surv)
names(d_example_surv)
```

`d_example` *Hypothetical data*

Description

Hypothetical data to be used in examples.

Usage

```
data(d_example)
```

Format

A list with 8 elements representing 500 observations from a control group and 500 observations from a treatment group:

`s1.a` First surrogate marker measurement for treated observations.

`s1.b` Second surrogate marker measurement for treated observations.

`s1.c` Third surrogate marker measurement for treated observations.

`y1` Primary outcome for treated observations.

`s0.a` First surrogate marker measurement for control observations.

`s0.b` Second surrogate marker measurement for control observations.

`s0.c` Third surrogate marker measurement for control observations.

`y0` Primary outcome for control observations.

Examples

```
data(d_example)
names(d_example)
```

d_example_surv — *Hypothetical survival data*

Description

Hypothetical survival data to be used in examples.

Usage

```
data(d_example_surv)
```

Format

A list with 8 elements representing 500 observations from a control group and 500 observations from a treatment group:

- `s1` Surrogate marker measurement for treated observations; this marker is measured at time = 0.5. For observations that experience the primary outcome or are censored before 0.5, this value is NA.
- `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` Surrogate marker measurement for control observations; this marker is measured at time = 0.5. For observations that experience the primary outcome or are censored before 0.5, this value is NA.
- `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.
- `z1` A baseline covariate value for treated observations.
- `z0` A baseline covariate value for control observations.

Examples

```
data(d_example_surv)
names(d_example_surv)
```

`fieller.ci` *Constructs Fieller's confidence interval.*

Description

Constructs Fieller's confidence interval.

Usage

```
fieller.ci(perturb.delta.s, perturb.delta, delta.s, delta)
```

Arguments

`perturb.delta.s`
: numeric vector; the perturbed values for $\hat{\Delta}_S$, the residual treatment effect estimate, used in variance estimation and confidence interval construction.

`perturb.delta`
: numeric vector; the perturbed values for $\hat{\Delta}$, the treatment effect estimate, used in variance estimation and confidence interval construction.

`delta.s`
: the residual treatment effect, Δ_S, estimate, $\hat{\Delta}_S$.

`delta`
: the treatment effect, Δ, estimate, $\hat{\Delta}$.

Details

See documention for R.s.estimate for more detail.

Value

Returns a vector of length 2, lower bound of the 95% confidence interval and upper bound of the 95% confidence interval.

Author(s)

Layla Parast

References

Fieller, Edgar C. (1954). Some problems in interval estimation. Journal of the Royal Statistical Society. Series B (Methodological), 175-185.

Fieller, E. C. (1940). The biological standardization of insulin. Supplement to the Journal of the Royal Statistical Society, 1-64.

Parast, L., McDermott, M., Tian, L. (2016). Robust estimation of the proportion of treatment effect explained by surrogate marker information. Statistics in Medicine, 35(10):1637-1653.

`me.variance.estimate` *Estimates measurement error variance given replicate data.*

Description

Estimates measurement error variance given replicate data using a simple components of variance analysis.

Usage

```
me.variance.estimate(replicates)
```

Arguments

`replicates`	matrix of data where each row indicates a subject and each column is a replicated measurement; columns can have NAs when subjects have different numbers of measurements.

Details

Estimates measurement error variance given replicate data using a simple components of variance analysis.

Value

estimate of measurement error variance

Author(s)

Layla Parast

References

Carroll, R. J., Ruppert, D., Crainiceanu, C. M., and Stefanski, L. A. (2006). Measurement error in nonlinear models: a modern perspective. Chapman and Hall/CRC.

Parast, L., Garcia, TP, Prentice, RL, Carroll, RJ (2019+). Robust Methods to Correct for Measurement Error when Evaluating a Surrogate Marker. Under Review.

`R.s.estimate` *Calculates the proportion of treatment effect explained*

Description

This function calculates the proportion of treatment effect on the primary outcome explained by the treatment effect on the surrogate marker(s). This function is intended to be used for a fully observed continuous outcome. The user can also request a variance estimate and a 95% confidence interval, both estimated using perturbating-resampling. If a confidence interval is requested three versions are provided: a normal approximation based interval, a quantile based interval, and Fieller's confidence interval.

Usage

```
R.s.estimate(sone, szero, yone, yzero, var = FALSE, conf.int = FALSE,
weight.perturb = NULL, number = "single", type = "robust",extrapolate = FALSE,
transform = FALSE)
```

Arguments

`sone`	numeric vector or matrix; surrogate marker for treated observations, assumed to be continuous. If there are multiple surrogates then this should be a matrix with n_1 (number of treated observations) rows and n.s (number of surrogate markers) columns.
`szero`	numeric vector; surrogate marker for control observations, assumed to be continuous.If there are multiple surrogates then this should be a matrix with n_0 (number of control observations) rows and n.s (number of surrogate markers) columns.
`yone`	numeric vector; primary outcome for treated observations, assumed to be continuous.
`yzero`	numeric vector; primary outcome for control observations, assumed to be continuous.
`var`	TRUE or FALSE; indicates whether a variance estimate is requested, default is FALSE.
`conf.int`	TRUE or FALSE; indicates whether a 95% confidence interval is requested, default is FALSE
`weight.perturb`	a $n_1 + n_0$ by x matrix of weights where n_1 = length of yone and n_0 = length of yzero; used for perturbation-resampling, default is null.
`number`	specifies the number of surrogate markers; choices are "multiple" or "single", default is "single"
`type`	specifies the type of estimation; choices are "robust" or "model" or "freedman", default is "robust"
`extrapolate`	TRUE or FALSE; indicates whether the user wants to use extrapolation.
`transform`	TRUE or FALSE; indicates whether the user wants to use a transformation for the surrogate marker.

Details

Let $Y^{(1)}$ and $Y^{(0)}$ denote the primary outcome under the treatment and primary outcome under the control,respectively. Let $S^{(1)}$ and $S^{(0)}$ denote the surrogate marker under the treatment and the surrogate marker under the control,respectively. The residual treatment effect is defined as

$$\Delta_S = \int_{-\infty}^{\infty} E(Y^{(1)}|S^{(1)} = s)dF_0(s) - \int_{-\infty}^{\infty} E(Y^{(0)}|S^{(0)} = s)dF_0(s),$$

where $\Delta_S(s) = E(Y^{(1)}|S^{(1)} = s) - E(Y^{(0)}|S^{(0)} = s)$ and $F_0(\cdot)$ is the marginal cumulative distribution function of $S^{(0)}$, the surrogate marker measure under the control. The proportion of treatment effect explained by the surrogate marker, which we denote by R_S, can be expressed using a contrast between Δ_S and Δ:

$$R_S = \{\Delta - \Delta_S\}/\Delta = 1 - \Delta_S/\Delta.$$

The definition and estimation of Δ is described in the delta.estimate documentation.

A flexible model-based approach to estimate Δ_S in the single marker setting is to specify:

$$E(S^{(0)}) = \alpha_0 \quad \text{and} \quad E(S^{(1)}) - E(S^{(0)}) = \alpha_1,$$

$$E(Y^{(0)}|S^{(0)}) = \beta_0 + \beta_1 S^{(0)} \quad \text{and} \quad E(Y^{(1)}|S^{(1)}) = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) S^{(1)}.$$

It can be shown that when these models hold, $\Delta_S = \beta_2 + \beta_3 \alpha_0$. Thus, reasonable estimates for Δ_S and R_S using this approach would be $\hat{\Delta}_S = \hat{\beta}_2 + \hat{\beta}_3 \hat{\alpha}_0$ and $\hat{R}_S = 1 - \hat{\Delta}_S / \hat{\Delta}$.

For robust estimation of Δ_S in the single marker setting, we estimate $\mu_1(s) = E(Y^{(1)}|S^{(1)} = s)$ nonparametrically using kernel smoothing:

$$\hat{\mu}_1(s) = \frac{\sum_{i=1}^{n_1} K_h\left(S_{1i} - s\right) Y_{1i}}{\sum_{i=1}^{n_1} K_h\left(S_{1i} - s\right)}$$

where S_{1i} is the observed $S^{(1)}$ for person i, Y_{1i} is the observed $Y^{(1)}$ for person i, $K(\cdot)$ is a smooth symmetric density function with finite support, $K_h(\cdot) = K(\cdot/h)/h$ and h is a specified bandwidth. As in most nonparametric functional estimation procedures, the choice of the smoothing parameter h is critical. To eliminate the impact of the bias of the conditional mean function on the resulting estimator, we require the standard undersmoothing assumption of $h = O(n_1^{-\delta})$ with $\delta \in (1/4, 1/3)$. To obtain an appropriate h we first use bw.nrd to obtain h_{opt}; and then we let $h = h_{opt} n_1^{-c_0}$ with $c_0 = 0.25$. We then estimate Δ_S as

$$\hat{\Delta}_S = \sum_{i=1}^{n_0} \frac{\hat{\mu}_1(S_{0i}) - Y_{0i}}{n_0}$$

where S_{0i} is the observed $S^{(0)}$ for person i and Y_{0i} is the observed $Y^{(0)}$ for person i. Lastly, we estimate R_S as $\hat{R}_S = 1 - \hat{\Delta}_S/\hat{\Delta}$.

This function also allows for estimation of R_S using Freedman's approach. Let Y denote the primary outcome, S denote the surrogate marker, and G denote the treatment group (0 for control, 1 for treatment). Freedman's approach to calculating the proportion of treatment effect explained by the surrogate marker is to fit the following two regression models:

$$E(Y|G) = \gamma_0 + \gamma_1 I(G = 1) \quad \text{and} \quad E(Y|G, S) = \gamma_{0S} + \gamma_{1S} I(G = 1) + \gamma_{2S} S$$

and estimating the proportion of treatment effect explained, denoted by R_S, as $1 - \hat{\gamma}_{1S}/\hat{\gamma}_1$.

This function also estimates R_S in a multiple marking setting. A flexible model-based approach to estimate Δ_S in the multiple marker setting is to specify models for $E(Y|G,S)$ and $E(S_j|G)$ for each S_j in $S = \{S_1, ...S_p\}$ (where p is the number of surrogate markers). Without loss of generality, consider the case where there are three surrogate markers, $S = \{S_1, S_2, S_3\}$ and one specifies the following linear models:

$$E(Y^{(0)}|S^{(0)}) = \beta_0 + \beta_1 S_1^{(0)} + \beta_2 S_2^{(0)} + \beta_3 S_3^{(0)}$$

$$E(Y^{(1)}|S^{(1)}) = (\beta_0 + \beta_4) + (\beta_1 + \beta_5) S_1^{(1)} + (\beta_2 + \beta_6) S_2^{(1)} + (\beta_3 + \beta_7) S_3^{(1)}$$

$$E(S_j^{(0)}) = \alpha_j, \quad j = 1, 2, 3.$$

It can be shown that when these models hold

$$\Delta_S = \beta_4 + \beta_5 \alpha_1 + \beta_6 \alpha_2 + \beta_7 \alpha_3.$$

Thus, reasonable estimates for Δ_S and R_S here would be easily obtained by replacing the unknown regression coefficients in the models above by their consistent estimators.

For robust estimation of S Δ_S in the multiple marker setting, we use a two-stage procedure combining the model-based approach and the nonparametric estimation procedure from the single marker setting. Specifically, we use a working semiparametric model:

$$E(Y^{(1)}|S^{(1)} = S) = \beta_0 + \beta_1 S_1^{(1)} + \beta_2 S_2^{(1)} + \beta_3 S_3^{(1)}$$

and define $Q^{(1)} = \hat{\beta}_0 + \hat{\beta}_1 S_1^{(1)} + \hat{\beta}_2 S_2^{(1)} + \hat{\beta}_3 S_3^{(1)}$ and $Q^{(0)} = \hat{\beta}_0 + \hat{\beta}_1 S_1^{(0)} + \hat{\beta}_2 S_2^{(0)} + \hat{\beta}_3 S_3^{(0)}$ to reduce the dimension of S in the first stage and in the second stage, we apply the robust approach used in the single marker setting to estimate its surrogacy.

To use Freedman's approach in the presence of multiple markers, the markers are simply additively entered into the second regression model.

Variance estimation and confidence interval construction are performed using perturbation-resampling. Specifically, let $\left\{V^{(b)} = (V_{11}^{(b)}, ...V_{1n_1}^{(b)}, V_{01}^{(b)}, ...V_{0n_0}^{(b)})^T, b = 1, ..., D\right\}$ be $n \times D$ independent copies of a positive random variables V from a known distribution with unit mean and unit variance. Let

$$\hat{\Delta}^{(b)} = \frac{\sum_{i=1}^{n_1} V_{1i}^{(b)} Y_{1i}}{\sum_{i=1}^{n_1} V_{1i}^{(b)}} - \frac{\sum_{i=1}^{n_0} V_{0i}^{(b)} Y_{0i}}{\sum_{i=1}^{n_0} V_{0i}^{(b)}}.$$

The variance of $\hat{\Delta}$ is obtained as the empirical variance of $\{\hat{\Delta}^{(b)}, b = 1, ..., D\}$. In this package, we use weights generated from an Exponential(1) distribution and use $D = 500$. Variance estimates for $\hat{\Delta}_S$ and $\hat{R}_S$ are calculated similarly. We construct two versions of the 95% confidence interval for each estimate: one based on a normal approximation confidence interval using the estimated variance and another taking the 2.5th and 97.5th empirical percentile of the perturbed quantities. In addition, we use Fieller's method to obtain a third confidence interval for R_S as

$$\left\{1 - r : \frac{(\hat{\Delta}_S - r\hat{\Delta})^2}{\hat{\sigma}_{11} - 2r\hat{\sigma}_{12} + r^2\hat{\sigma}_{22}} \leq c_\alpha\right\},$$

where $\hat{\Sigma} = (\hat{\sigma}_{ij})_{1 \leq i,j \leq 2}$ and c_α is the $(1-\alpha)$th percentile of

$$\left\{\frac{\{\hat{\Delta}_S^{(b)} - (1 - \hat{R}_S)\hat{\Delta}^{(b)}\}^2}{\hat{\sigma}_{11} - 2(1 - \hat{R}_S)\hat{\sigma}_{12} + (1 - \hat{R}_S)^2\hat{\sigma}_{22}}, b = 1, \cdots, C\right\}$$

where $\alpha = 0.05$.

Note that if the observed supports for S are not the same, then $\hat{\mu}_1(s)$ for $S_{0i} = s$ outside the support of S_{1i} may return NA (depending on the bandwidth). If extrapolation = TRUE, then the $\hat{\mu}_1(s)$ values for these surrogate values are set to the closest non-NA value. If transform = TRUE, then S_{1i} and S_{0i} are transformed such that the new transformed values, S_{1i}^{tr} and S_{0i}^{tr} are defined as: $S_{gi}^{tr} = F([S_{gi} - \mu]/\sigma)$ for $g = 0, 1$ where $F(\cdot)$ is the cumulative distribution function for a standard normal random variable, and μ and σ are the sample mean and standard deviation, respectively, of $(S_{1i}, S_{0i})^T$.

Value

A list is returned:

`R.s` the estimate, $\hat{R}_S$, described above.

`R.s.var` the variance estimate of $\hat{R}_S$; if var = TRUE or conf.int = TRUE.

`conf.int.normal.R.s`
a vector of size 2; the 95% confidence interval for $\hat{R}_S$ based on a normal approximation; if conf.int = TRUE.

`conf.int.quantile.R.s`
: a vector of size 2; the 95% confidence interval for $\hat{R}_S$ based on sample quantiles of the perturbed values, described above; if conf.int = TRUE.

`conf.int.fieller.R.s`
: a vector of size 2; the 95% confidence interval for $\hat{R}_S$ based on Fieller's approach, described above; if conf.int = TRUE.

For all options other then "freedman", the following are also returned:

`delta`
: the estimate, $\hat{\Delta}$, described in delta.estimate documentation.

`delta.s`
: the estimate, $\hat{\Delta}_S$, described above.

`delta.var`
: the variance estimate of $\hat{\Delta}$; if var = TRUE or conf.int = TRUE.

`delta.s.var`
: the variance estimate of $\hat{\Delta}_S$; if var = TRUE or conf.int = TRUE.

`conf.int.normal.delta`
: a vector of size 2; the 95% confidence interval for $\hat{\Delta}$ based on a normal approximation; if conf.int = TRUE.

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

`conf.int.normal.delta.s`
: a vector of size 2; the 95% confidence interval for $\hat{\Delta}_S$ based on a normal approximation; if conf.int = TRUE.

`conf.int.quantile.delta.s`
: a vector of size 2; the 95% confidence interval for $\hat{\Delta}_S$ based on sample quantiles of the perturbed values, described above; if conf.int = TRUE.

Note

If the treatment effect is not significant, the user will receive the following message: "Warning: it looks like the treatment effect is not significant; may be difficult to interpret the proportion of treatment effect explained in this setting". If the treatment effect is negative, the user will receive the following message: "Warning: it looks like you need to switch the treatment groups" as this package assumes throughout that higher values are better. In the single marker case with the robust estimation approach, if the observed support of the surrogate marker for the control group is outside the observed support of the surrogate marker for the treatment group, the user will receive the following message: "Warning: observed supports do not appear equal, may need to consider a transformation or extrapolation"

Author(s)

Layla Parast

References

Freedman, L. S., Graubard, B. I., & Schatzkin, A. (1992). Statistical validation of intermediate endpoints for chronic diseases. Statistics in medicine, 11(2), 167-178.

Parast, L., McDermott, M., Tian, L. (2016). Robust estimation of the proportion of treatment effect explained by surrogate marker information. Statistics in Medicine, 35(10):1637-1653.

Wang, Y., & Taylor, J. M. (2002). A measure of the proportion of treatment effect explained by a surrogate marker. Biometrics, 58(4), 803-812.

Fieller, Edgar C. (1954). Some problems in interval estimation. Journal of the Royal Statistical Society. Series B (Methodological), 175-185.

Fieller, E. C. (1940). The biological standardization of insulin. Supplement to the Journal of the Royal Statistical Society, 1-64.

Examples

```
data(d_example)
names(d_example)
R.s.estimate(yone=d_example$y1, yzero=d_example$y0, sone=d_example$s1.a, szero=d_example$s0.a,
number = "single", type = "robust")
R.s.estimate(yone=d_example$y1, yzero=d_example$y0, sone=cbind(d_example$s1.a,d_example$s1.b,
d_example$s1.c), szero=cbind(d_example$s0.a, d_example$s0.b, d_example$s0.c),
number = "multiple", type = "model")
```

R.s.estimate.me *Calculates the proportion of treatment effect explained correcting for measurement error in the surrogate marker*

Description

This function calculates the proportion of treatment effect on the primary outcome explained by the treatment effect on a surrogate marker, correcting for measurement error in the surrogate marker. This function is intended to be used for a fully observed continuous outcome. The user must specify what type of estimation they would like (parametric or nonparametric estimation of the proportion explained, denoted by R) and what estimator they would like (see below for details).

Usage

```
R.s.estimate.me(sone, szero, yone, yzero, parametric = FALSE, estimator = "n",
me.variance, extrapolate = TRUE, transform = FALSE, naive = FALSE, Ronly = TRUE)
```

Arguments

`sone`	numeric vector or matrix; surrogate marker for treated observations, assumed to be continuous. If there are multiple surrogates then this should be a matrix with n_1 (number of treated observations) rows and n.s (number of surrogate markers) columns.
`szero`	numeric vector; surrogate marker for control observations, assumed to be continuous.If there are multiple surrogates then this should be a matrix with n_0 (number of control observations) rows and n.s (number of surrogate markers) columns.
`yone`	numeric vector; primary outcome for treated observations, assumed to be continuous.
`yzero`	numeric vector; primary outcome for control observations, assumed to be continuous.
`parametric`	TRUE or FALSE; indicates whether the user wants the parametric approach to be used (TRUE) or nonparametric (FALSE).
`estimator`	options are "d","q","n" for parametric and "q","n" for nonparametric; "d" stands for the disattenuated estimator, "q" stands for the SIMEX estimator with quadratic extrapolation, "n" stands for the SIMEX estimator with a nonlinear extrapolation.

`me.variance` the variance of the measurement error; must be provided.

`extrapolate` TRUE or FALSE; indicates whether the user wants to use extrapolation.

`transform` TRUE or FALSE; indicates whether the user wants to use a transformation for the surrogate marker.

`naive` TRUE or FALSE; indicates whether the user wants the naive estimate (not correcting for measurement error) to also be calculated

`Ronly` TRUE or FALSE; indicates whether the user wants only R (and corresponding variance and confidence intervals) to be returned.

Details

Details can be found in Parast, L., Garcia, TP, Prentice, RL, Carroll, RJ (2019+). Robust Methods to Correct for Measurement Error when Evaluating a Surrogate Marker. Under Review.

Please email parast@rand.org if you would like a copy of this article.

Value

A list is returned:

`R.naive` the naive estimate of the proportion of treatment effect explained by the surrogate marker; only if naive = TRUE

`R.naive.var` the estimated variance of the naive estimate of the proportion of treatment effect explained by the surrogate marker; only if naive = TRUE

`R.naive.CI.normal`
the 95% confidence interval using the normal approximation for the naive estimate of the proportion of treatment effect explained by the surrogate marker; only if naive = TRUE

`R.naive.CI.fieller`
the 95% confidence interval using Fieller's approach for the naive estimate of the proportion of treatment effect explained by the surrogate marker; only if naive = TRUE

`B1star.naive` the naive estimate of the adjusted regression coefficient for treatment; only if naive = TRUE and Ronly = FALSE and parametric = TRUE

`B1star.naive.var`
the estimated variance of the naive estimate of the adjusted regression coefficient for treatment; only if naive = TRUE and Ronly = FALSE and parametric = TRUE

`B1star.naive.CI.normal`
the 95% confidence interval using the normal approximation for the naive estimate of the adjusted regression coefficient for treatment; only if naive = TRUE and Ronly = FALSE and parametric = TRUE

`deltas.naive` the naive estimate of the residual treatment effect; only if naive = TRUE and Ronly = FALSE and parametric = FALSE

`deltas.naive.var`
the estimated variance of the naive estimate of the residual treatment effect; only if naive = TRUE and Ronly = FALSE and parametric = FALSE

`deltas.naive.CI.normal`
the 95% confidence interval using the normal approximation for the naive estimate of the residual treatment effect; only if naive = TRUE and Ronly = FALSE and parametric = FALSE

`R.corrected.dis`
: the corrected disattenuated estimate of the proportion of treatment effect explained by the surrogate marker; only if parametric = TRUE and estimator ="d"

`R.corrected.var.dis`
: the estimated variance of the corrected disattenuated estimate of the proportion of treatment effect explained by the surrogate marker; only if naive = TRUE

`R.corrected.CI.normal.dis`
: the 95% confidence interval using the normal approximation for the corrected disattenuated estimate of the proportion of treatment effect explained by the surrogate marker; only if parametric = TRUE and estimator ="d"

`R.corrected.CI.fieller.dis`
: the 95% confidence interval using Fieller's approach for the corrected disattenuated estimate of the proportion of treatment effect explained by the surrogate marker; only if parametric = TRUE and estimator ="d"

`B1star.corrected.dis`
: the corrected disattenuated estimate of the adjusted regression coefficient for treatment; only if parametric = TRUE and estimator = "d" and Ronly = FALSE

`B1star.corrected.var.dis`
: the estimated variance of the corrected disattenuated estimate of the adjusted regression coefficient for treatment; only if parametric = TRUE and estimator = "d" and Ronly = FALSE

`B1star.corrected.CI.normal.dis`
: the 95% confidence interval using the normal approximation for the corrected disattenuated estimate of the adjusted regression coefficient for treatment; only if parametric = TRUE and estimator = "d" and Ronly = FALSE

`R.corrected.q`
: the corrected SIMEX (quadratic) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`R.corrected.var.q`
: the estimated variance of the corrected SIMEX (quadratic) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`R.corrected.CI.normal.q`
: the 95% confidence interval using the normal approximation for the corrected SIMEX (quadratic) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`R.corrected.CI.fieller.q`
: the 95% confidence interval using Fieller's approach for the corrected SIMEX (quadratic) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`B1star.corrected.q`
: the corrected SIMEX (quadratic) estimate of the adjusted regression coefficient for treatment; only if estimator = "q" and Ronly = FALSE and parametric = TRUE

`B1star.corrected.var.q`
: the estimated variance of the corrected SIMEX (quadratic) estimate of the adjusted regression coefficient for treatment; only if estimator = "q" and Ronly = FALSE and parametric = TRUE

`B1star.corrected.CI.normal.q`
: the 95% confidence interval using the normal approximation for the corrected SIMEX (quadratic) estimate of the adjusted regression coefficient for treatment; only if estimator = "q" and Ronly = FALSE and parametric = TRUE

`deltas.corrected.q`
: the corrected SIMEX (quadratic) estimate of the residual treatment effect; only if estimator = "q" and Ronly = FALSE and parametric = FALSE

`deltas.corrected.var.q`
: the estimated variance of the corrected SIMEX (quadratic) estimate of the residual treatment effect; only if estimator = "q" and Ronly = FALSE and parametric = FALSE

`deltas.corrected.CI.normal.q`
: the 95% confidence interval using the normal approximation for the corrected SIMEX (quadratic) estimate of the residual treatment effect; only if estimator = "q" and Ronly = FALSE and parametric = FALSE

`R.corrected.nl`
: the corrected SIMEX (nonlinear) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`R.corrected.var.nl`
: the estimated variance of the corrected SIMEX (nonlinear) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`R.corrected.CI.normal.nl`
: the 95% confidence interval using the normal approximation for the corrected SIMEX (nonlinear) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`R.corrected.CI.fieller.nl`
: the 95% confidence interval using Fieller's approach for the corrected SIMEX (nonlinear) estimate of the proportion of treatment effect explained by the surrogate marker; only if estimator = "q"

`B1star.corrected.nl`
: the corrected SIMEX (nonlinear) estimate of the adjusted regression coefficient for treatment; only if estimator = "q" and Ronly = FALSE and parametric = TRUE

`B1star.corrected.var.nl`
: the estimated variance of the corrected SIMEX (nonlinear) estimate of the adjusted regression coefficient for treatment; only if estimator = "q" and Ronly = FALSE and parametric = TRUE

`B1star.corrected.CI.normal.nl`
: the 95% confidence interval using the normal approximation for the corrected SIMEX (nonlinear) estimate of the adjusted regression coefficient for treatment; only if estimator = "q" and Ronly = FALSE and parametric = TRUE

`deltas.corrected.nl`
: the corrected SIMEX (nonlinear) estimate of the residual treatment effect; only if estimator = "q" and Ronly = FALSE and parametric = FALSE

`deltas.corrected.var.nl`
: the estimated variance of the corrected SIMEX (nonlinear) estimate of the residual treatment effect; only if estimator = "q" and Ronly = FALSE and parametric = FALSE

`deltas.corrected.CI.normal.nl`
: the 95% confidence interval using the normal approximation for the corrected SIMEX (nonlinear) estimate of the residual treatment effect; only if estimator = "q" and Ronly = FALSE and parametric = FALSE

Author(s)

Layla Parast

References

Parast, L., Garcia, TP, Prentice, RL, Carroll, RJ (2019+). Robust Methods to Correct for Measurement Error when Evaluating a Surrogate Marker. Under Review.

Examples

```
data(d_example)
names(d_example)
R.s.estimate.me(yone=d_example$y1, yzero=d_example$y0, sone=d_example$s1.a, szero=d_example$s0.a,
parametric = TRUE, estimator = "d", me.variance = 0.5, naive= TRUE, Ronly = FALSE)
R.s.estimate.me(yone=d_example$y1, yzero=d_example$y0, sone=d_example$s1.a, szero=d_example$s0.a,
parametric = TRUE, estimator = "q", me.variance = 0.5, naive= FALSE, Ronly = TRUE)
R.s.estimate.me(yone=d_example$y1, yzero=d_example$y0, sone=d_example$s1.a, szero=d_example$s0.a,
parametric = FALSE, estimator = "q", me.variance = 0.5, naive= FALSE, Ronly = TRUE)
```

R.s.surv.estimate — *Calculates the proportion of treatment effect explained by the surrogate marker information measured at a specified time and primary outcome information up to that specified time*

Description

This function calculates the proportion of treatment effect on the primary outcome explained by the surrogate marker information measured at t_0 and primary outcome information up to t_0. The user can also request a variance estimate, estimated using perturbating-resampling, and a 95% confidence interval. If a confidence interval is requested three versions are provided: a normal approximation based interval, a quantile based interval and Fieller's confidence interval, all using perturbation-resampling. The user can also request an estimate of the incremental value of surrogate marker information.

Usage

```
R.s.surv.estimate(xone, xzero, deltaone, deltazero, sone, szero, t,
weight.perturb = NULL, landmark, extrapolate = FALSE, transform = FALSE,
conf.int = FALSE, var = FALSE, incremental.value = FALSE)
```

Arguments

`xone`	numeric vector, the observed event times in the treatment group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`xzero`	numeric vector, the observed event times in the control group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`deltaone`	numeric vector, the event indicators for the treatment group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`deltazero`	numeric vector, the event indicators for the control group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`sone`	numeric vector; surrogate marker measurement at t_0 for treated observations, assumed to be continuous. If $X_{1i} < t_0$, then the surrogate marker measurement should be NA.

`szero`	numeric vector; surrogate marker measurement at t_0 for control observations, assumed to be continuous. If $X_{1i} < t_0$, then the surrogate marker measurement should be NA.
`t`	the time of interest.
`weight.perturb`	weights used for perturbation resampling.
`landmark`	the landmark time t_0 or time of surrogate marker measurement.
`extrapolate`	TRUE or FALSE; indicates whether the user wants to use extrapolation.
`transform`	TRUE or FALSE; indicates whether the user wants to use a transformation for the surrogate marker.
`conf.int`	TRUE or FALSE; indicates whether a 95% confidence interval for delta is requested, default is FALSE.
`var`	TRUE or FALSE; indicates whether a variance estimate for delta is requested, default is FALSE.
`incremental.value`	TRUE or FALSE; indicates whether the user would like to see the incremental value of the surrogate marker information, default is FALSE.

Details

Let G be the binary treatment indicator with $G = 1$ for treatment and $G = 0$ for control and we assume throughout that subjects are randomly assigned to a treatment group at baseline. Let $T^{(1)}$ and $T^{(0)}$ denote the time of the primary outcome of interest, death for example, under the treatment and under the control, respectively. Let $S^{(1)}$ and $S^{(0)}$ denote the surrogate marker measured at time t_0 under the treatment and the control, respectively.

The residual treatment effect is defined as

$$\Delta_S(t,t_0) = P(T^{(0)} > t_0)\left\{\int \psi_1(t|s,t_0)dF_0(s|t_0) - P(T^{(0)} > t|T^{(0)} > t_0)\right\}$$

where $F_0(\cdot|t_0)$ is the cumulative distribution function of $S^{(0)}$ conditional on $T^{(0)} > t_0$ and $\psi_1(t|s,t_0) = P(T^{(1)} > t|S^{(1)} = s, T^{(1)} > t_0)$. The proportion of treatment effect explained by the surrogate marker information measured at t_0 and primary outcome information up to t_0, which we denote by $R_S(t,t_0)$, can be expressed using a contrast between $\Delta_S(t,t_0)$ and $\Delta(t)$:

$$R_S(t,t_0) = \{\Delta(t) - \Delta_S(t,t_0)\}/\Delta(t) = 1 - \Delta_S(t,t_0)/\Delta(t).$$

The definition and estimation of $\Delta(t)$ is described in the delta.surv.estimate documentation.

Due to censoring, our data consist of n_1 observations $\{(X_{1i}, \delta_{1i}, S_{1i}), i = 1, ..., n_1\}$ from the treatment group $G = 1$ and n_0 observations $\{(X_{0i}, \delta_{0i}, S_{0i}), i = 1, ..., n_0\}$ from the control group $G = 0$ where $X_{gi} = \min(T_{gi}, C_{gi})$, $\delta_{gi} = I(T_{gi} < C_{gi})$, C_{gi} denotes the censoring time, and S_{gi} denotes the surrogate marker information measured at time t_0, for $g = 1, 0$, for individual i. Note that if $X_{gi} < t_0$, then S_{gi} should be NA (not available).

To estimate $\Delta_S(t,t_0)$, we use a nonparametric kernel Nelson-Aalen estimator to estimate $\psi_1(t|s,t_0)$ as $\hat{\psi}_1(t|s,t_0) = \exp\{-\hat{\Lambda}_1(t|s,t_0)\}$, where

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

is a consistent estimate of $\Lambda_1(t|s,t_0) = -\log[\psi_1(t|s,t_0)], Y_{1i}(t) = I(X_{1i} \geq t), N_{1i}(t) = I(X_{1i} \leq t)\delta_i, K(\cdot)$ is a smooth symmetric density function, $K_h(x) = K(x/h)/h, \gamma(\cdot)$ is a given monotone transformation function, and h is a specified bandwidth. To obtain an appropriate h we first use bw.nrd to obtain h_{opt}; and then we let $h = h_{opt} n_1^{-c_0}$ with $c_0 = 0.11$.

Since $F_0(s|t_0) = P(S_{0i} \le s | X_{0i} > t_0)$, we empirically estimate $F_0(s|t_0)$ using all subjects with $X_{0i} > t_0$ as

$$\hat{F}_0(s|t_0) = \frac{\sum_{i=1}^{n_0} I(S_{0i} \le s, X_{0i} > t_0)}{\sum_{i=1}^{n_0} I(X_{0i} > t_0)}.$$

Subsequently, we construct an estimator for $\Delta_S(t, t_0)$ as

$$\hat{\Delta}_S(t, t_0) = n_0^{-1} \sum_{i=1}^{n_0} \left[\hat{\psi}_1(t|S_{0i}, t_0) \frac{I(X_{0i} > t_0)}{\hat{W}_0^C(t_0)} - \frac{I(X_{0i} > t)}{\hat{W}_0^C(t)} \right]$$

where $\hat{W}_g^C(\cdot)$ is the Kaplan-Meier estimator of survival for censoring for $g = 1, 0$. Finally, we estimate $R_S(t, t_0)$ as $\hat{R}_S(t, t_0) = 1 - \hat{\Delta}_S(t, t_0)/\hat{\Delta}(t)$.

Variance estimation and confidence interval construction are performed using perturbation-resampling. Specifically, let $\left\{ V^{(b)} = (V_{11}^{(b)}, ...V_{1n_1}^{(b)}, V_{01}^{(b)}, ...V_{0n_0}^{(b)})^T, b = 1, ..., D \right\}$ be $n \times D$ independent copies of a positive random variables V from a known distribution with unit mean and unit variance. Let

$$\hat{\Delta}^{(b)}(t) = \frac{\sum_{i=1}^{n_1} V_{1i}^{(b)} I(X_{1i} > t)}{\sum_{i=1}^{n_1} V_{1i}^{(b)} \hat{W}_1^{C(b)}(t)} - \frac{\sum_{i=1}^{n_0} V_{0i}^{(b)} I(X_{0i} > t)}{\sum_{i=1}^{n_0} V_{0i}^{(b)} \hat{W}_0^{C(b)}(t)}.$$

In this package, we use weights generated from an Exponential(1) distribution and use $D = 500$. The variance of $\hat{\Delta}(t)$ is obtained as the empirical variance of $\{\hat{\Delta}(t)^{(b)}, b = 1, ..., D\}$. Variance estimates for $\hat{\Delta}_S(t, t_0)$ and $\hat{R}_S(t, t_0)$ are calculated similarly. We construct two versions of the 95% confidence interval for each estimate: one based on a normal approximation confidence interval using the estimated variance and another taking the 2.5th and 97.5th empirical percentile of the perturbed quantities. In addition, we use Fieller's method to obtain a third confidence interval for $R_S(t, t_0)$ as

$$\left\{ 1 - r : \frac{(\hat{\Delta}_S(t, t_0) - r\hat{\Delta}(t))^2}{\hat{\sigma}_{11} - 2r\hat{\sigma}_{12} + r^2\hat{\sigma}_{22}} \le c_\alpha \right\},$$

where $\hat{\Sigma} = (\hat{\sigma}_{ij})_{1 \le i,j \le 2}$ and c_α is the $(1 - \alpha)$th percentile of

$$\left\{ \frac{\{\hat{\Delta}_S^{(b)}(t) - (1 - \hat{R}_S(t, t_0))\hat{\Delta}(t)^{(b)}\}^2}{\hat{\sigma}_{11} - 2(1 - \hat{R}_S(t, t_0))\hat{\sigma}_{12} + (1 - \hat{R}_S(t, t_0))^2\hat{\sigma}_{22}}, b = 1, \cdots, C \right\}$$

where $\alpha = 0.05$.

Since the definition of $R_S(t, t_0)$ considers the surrogate information as a combination of both S information and T information up to t_0, a logical inquiry would be how to assess the incremental value of the S information in terms of the proportion of treatment effect explained, when added to T information up to t_0. The proportion of treatment effect explained by T information up to t_0 only is denoted as $R_T(t, t_0)$ and is described in the documentation for R.t.surv.estimate. The incremental value of S information is defined as:

$$IV_S(t, t_0) = R_S(t, t_0) - R_T(t, t_0) = \frac{\Delta_T(t, t_0) - \Delta_S(t, t_0)}{\Delta(t)}.$$

For estimation of $R_T(t_0)$, see documentation for R.t.surv.estimate. The quantity $IV_S(t, t_0)$ is then estimated by $\hat{IV}_S(t, t_0) = \hat{R}_S(t, t_0) - \hat{R}_T(t, t_0)$. Perturbation-resampling is used for variance estimation and confidence interval construction for this quantity, similar to the other quantities in this package.

Note that if the observed supports for S are not the same, then $\hat{\Lambda}_1(t|s,t_0)$ for $S_{0i} = s$ outside the support of S_{1i} may return NA (depending on the bandwidth). If extrapolation = TRUE, then the $\hat{\Lambda}_1(t|s,t_0)$ values for these surrogate values are set to the closest non-NA value. If transform = TRUE, then S_{1i} and S_{0i} are transformed such that the new transformed values, S_{1i}^{tr} and S_{0i}^{tr} are defined as: $S_{gi}^{tr} = F([S_{gi} - \mu]/\sigma)$ for $g = 0, 1$ where $F(\cdot)$ is the cumulative distribution function for a standard normal random variable, and μ and σ are the sample mean and standard deviation, respectively, of $\{(S_{1i}, S_{0i})^T, i \quad s.t. X_{gi} > t_0\}$.

Value

A list is returned:

`delta` the estimate, $\hat{\Delta}(t)$, described in delta.estimate documentation.

`delta.s` the estimate, $\hat{\Delta}_S(t,t_0)$, described above.

`R.s` the estimate, $\hat{R}_S(t,t_0)$, described above.

`delta.var` the variance estimate of $\hat{\Delta}(t)$; if var = TRUE or conf.int = TRUE.

`delta.s.var` the variance estimate of $\hat{\Delta}_S(t,t_0)$; if var = TRUE or conf.int = TRUE.

`R.s.var` the variance estimate of $\hat{R}_S(t,t_0)$; if var = TRUE or conf.int = TRUE.

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

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

`conf.int.normal.delta.s`
a vector of size 2; the 95% confidence interval for $\hat{\Delta}_S(t,t_0)$ based on a normal approximation; if conf.int = TRUE.

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

`conf.int.normal.R.s`
a vector of size 2; the 95% confidence interval for $\hat{R}_S(t,t_0)$ based on a normal approximation; if conf.int = TRUE.

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

`conf.int.fieller.R.s`
a vector of size 2; the 95% confidence interval for $\hat{R}_S(t,t_0)$ based on Fieller's approach, described above; if conf.int = TRUE.

`delta.t` the estimate, $\hat{\Delta}_T(t,t_0)$, described above; if incremental.vaue = TRUE.

`R.t` the estimate, $\hat{R}_T(t,t_0)$, described above; if incremental.vaue = TRUE.

`incremental.value`
the estimate, $\hat{IV}_S(t,t_0)$, described above; if incremental.vaue = TRUE.

`delta.t.var` the variance estimate of $\hat{\Delta}_T(t,t_0)$; if var = TRUE or conf.int = TRUE and incremental.vaue = TRUE.

`R.t.var` the variance estimate of $\hat{R}_T(t,t_0)$; if var = TRUE or conf.int = TRUE and incremental.vaue = TRUE.

incremental.value.var
: the variance estimate of $\hat{IV}_S(t, t_0)$; if var = TRUE or conf.int = TRUE and incremental.vaue = TRUE.

conf.int.normal.delta.t
: a vector of size 2; the 95% confidence interval for $\hat{\Delta}_T(t, t_0)$ based on a normal approximation; if conf.int = TRUE and incremental.vaue = TRUE.

conf.int.quantile.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, described above; if conf.int = TRUE and incremental.vaue = TRUE.

conf.int.normal.R.t
: a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)$ based on a normal approximation; if conf.int = TRUE and incremental.vaue = TRUE.

conf.int.quantile.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, described above; if conf.int = TRUE and incremental.vaue = TRUE.

conf.int.fieller.R.t
: a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)$ based on Fieller's approach, described above; if conf.int = TRUE and incremental.vaue = TRUE.

conf.int.normal.iv
: a vector of size 2; the 95% confidence interval for $\hat{IV}_S(t, t_0)$ based on a normal approximation; if conf.int = TRUE and incremental.vaue = TRUE.

conf.int.quantile.iv
: a vector of size 2; the 95% confidence interval for $\hat{IV}_S(t, t_0)$ based on sample quantiles of the perturbed values, described above; if conf.int = TRUE and incremental.vaue = TRUE.

Note

If the treatment effect is not significant, the user will receive the following message: "Warning: it looks like the treatment effect is not significant; may be difficult to interpret the residual treatment effect in this setting". If the treatment effect is negative, the user will receive the following message: "Warning: it looks like you need to switch the treatment groups" as this package assumes throughout that larger values of the event time are better. If the observed support of the surrogate marker for the control group is outside the observed support of the surrogate marker for the treatment group, the user will receive the following message: "Warning: observed supports do not appear equal, may need to consider a transformation or extrapolation".

Author(s)

Layla Parast

References

Parast L, Cai T and Tian L. Evaluating Surrogate Marker Information using Censored Data. Under Review.

Examples

```
data(d_example_surv)
names(d_example_surv)
```

`R.t.surv.estimate` *Calculates the proportion of treatment effect explained by the primary outcome information up to a specified time*

Description

This function calculates the proportion of treatment effect on the primary outcome explained by the treatment effect on the primary outcome up to t_0. The user can also request a variance estimate, estimated using perturbating-resampling, and a 95% confidence interval. If a confidence interval is requested three versions are provided: a normal approximation based interval, a quantile based interval and Fieller's confidence interval, all using perturbation-resampling.

Usage

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

Arguments

`xone`	numeric vector, the observed event times in the treatment group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`xzero`	numeric vector, the observed event times in the control group, X = min(T,C) where T is the time of the primary outcome and C is the censoring time.
`deltaone`	numeric vector, the event indicators for the treatment group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`deltazero`	numeric vector, the event indicators for the control group, D = I(T<C) where T is the time of the primary outcome and C is the censoring time.
`t`	the time of interest.
`weight.perturb`	weights used for perturbation resampling.
`landmark`	the landmark time t_0 or time of surrogate marker measurement.
`var`	TRUE or FALSE; indicates whether a variance estimate for delta is requested, default is FALSE.
`conf.int`	TRUE or FALSE; indicates whether a 95% confidence interval for delta is requested, default is FALSE.

Details

Let G be the binary treatment indicator with $G = 1$ for treatment and $G = 0$ for control and we assume throughout that subjects are randomly assigned to a treatment group at baseline. Let T denote the time of the primary outcome of interest, death for example. We use potential outcomes notation such that $T^{(g)}$ denotes the time of the primary outcome under treatment $G = g$. The proportion of treatment effect explained by T observed up to t_0 only is $R_T(t, t_0) = 1 - \Delta_T(t, t_0)/\Delta(t)$ where

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

To estimate $R_T(t, t_0)$, we use the estimator $\hat{R}_T(t, t_0) = 1 - \hat{\Delta}_T(t, t_0)/\hat{\Delta}(t)$ where $\hat{\Delta}_T(t, t_0) = \hat{\phi}_0(t_0)\hat{\phi}_1(t)/\hat{\phi}_1(t_0) - \hat{\phi}_0(t)$ and $\hat{\phi}_g(u) = n_g^{-1}\sum_{i=1}^{n_g} \frac{I(X_{gi}>u)}{\hat{W}_g^C(u)}$ for $g = 1, 0$ where $\widehat{W}_g^C(\cdot)$ is the Kaplan-Meier estimator of survival for censoring for $g = 1, 0$.

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.var` the variance estimate of $\hat{\Delta}(t)$; if var = TRUE or conf.int = TRUE.

`delta.t.var` the variance estimate of $\hat{\Delta}_T(t, t_0)$; if var = TRUE or conf.int = TRUE.

`R.t.var` the variance estimate of $\hat{R}_T(t, t_0)$; if var = TRUE or conf.int = TRUE.

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

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

`conf.int.normal.delta.t`
a vector of size 2; the 95% confidence interval for $\hat{\Delta}_T(t, t_0)$ based on a normal approximation; if conf.int = TRUE.

`conf.int.quantile.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, described above; if conf.int = TRUE.

`conf.int.normal.R.t`
a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)$ based on a normal approximation; if conf.int = TRUE.

`conf.int.quantile.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, described above; if conf.int = TRUE.

`conf.int.fieller.R.t`
a vector of size 2; the 95% confidence interval for $\hat{R}_T(t, t_0)$ based on Fieller's approach, described above; if conf.int = TRUE.

Note

If the treatment effect is not significant, the user will receive the following message: "Warning: it looks like the treatment effect is not significant; may be difficult to interpret the residual treatment effect in this setting". If the treatment effect is negative, the user will receive the following message: "Warning: it looks like you need to switch the treatment groups" as this package assumes throughout that larger values of the event time are better.

Author(s)

Layla Parast

References

Parast L, Cai T and Tian L. Evaluating Surrogate Marker Information using Censored Data. Under Review.

Examples

```
data(d_example_surv)
names(d_example_surv)
```

Index