Package 'tranSurv'

January 12, 2021

Title Transformation Model Based Estimation of Survival and Regression
Under Dependent Truncation and Independent Censoring

Version 1.2.2

Description A latent, quasi-independent truncation time is assumed to be linked with the observed dependent truncation time, the event time, and an unknown transformation parameter via a structural transformation model. The transformation parameter is chosen to minimize the conditional Kendall's tau (Martin and Betensky, 2005) <doi:10.1198/016214504000001538> or the regression coefficient estimates (Jones and Crowley, 1992) <doi:10.2307/2336782>. The marginal distribution for the truncation time and the event time are completely left unspecified. The methodology is applied to survival curve estimation and regression analysis.

License GPL (>= 3)

URL https://github.com/stc04003/tranSurv

BugReports https://github.com/stc04003/tranSurv/issues

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports rootSolve, truncSP, survival, SQUAREM, methods

Suggests MASS, boot

NeedsCompilation yes

Author Sy Han (Steven) Chiou [aut, cre],
Jing Qian [aut]

Maintainer Sy Han (Steven) Chiou <schiou@utdallas.edu>

Repository CRAN

Date/Publication 2021-01-12 09:10:02 UTC

R topics documented:

tranSurv-package	*tranSurv:Transformation Model Based Survival Curve Estimation with Dependent Left Truncation*

Description

A package that estimates survival curve under a dependent truncation and independent right censoring via a structural transformation method. The package also includes hypothesis test of quasi-independence based on the conditional Kendall's tau of Martin and Betensky (2005) and two versions of the inverse probability weighted Kendall's tau of Austin and Betensky (2014).

Author(s)

Maintainer: Sy Han (Steven) Chiou <schiou@utdallas.edu>

Authors:

- Jing Qian <qian@schoolph.umass.edu>

References

Martin E. and Betensky R. A. (2005), Testing quasi-independence of failure and truncation times via conditional Kendall's tau, *Journal of the American Statistical Association*, **100** (470): 484-492.

Austin, M. D. and Betensky R. A. (2014), Eliminating bias due to censoring in Kendall's tau estimators for quasi-independence of truncation and failure, *Computational Statistics & Data Analysis*, **73**: 16-26.

Chiou, S., Austin, M., Qian, J. and Betensky R. A. (2016), Transformation model estimation of survival under dependent truncation and independent censoring, *Statistical Methods in Medical Research*, **28** (12): 3785-3798.

See Also

Useful links:

- https://github.com/stc04003/tranSurv
- Report bugs at https://github.com/stc04003/tranSurv/issues

---

cKendall	*Conditional Kendall's tau*

Description

Computes the conditional Kendall's tau and inference

Usage

```
cKendall(
  trun,
  obs,
  delta = NULL,
  method = "MB",
  weights = NULL,
  a = 0,
  trans = "linear",
  ...
)
```

Arguments

trun	left truncation time satisfying `trun <= obs`.
obs	observed failure time, must be the same length as `trun`, might be right-censored.
delta	an optional 0-1 vector of censoring indicator (0 = censored, 1 = event) for `obs`. If this vector is not specified, `cKendall` assumes no censoring and all observed failure time denote events.
method	a character string specifying the different version of conditional Kendall's tau to be computed. The following are permitted: MB conditional Kendall's tau proposed in Martin and Betensky (2005) as $\hat{\tau}_c$, IPW1 inverse probability weighted estimator proposed in Austin and Betensky (2014) as $\hat{\tau}_{c2}$, IPW2 restricted inverse probability weighted estimator proposed in Austin and Betensky (2014) as $\hat{\tau}_{c3}$.
weights	an optional vector of sampling weights used when `method = IPW1` or `method = IPW2`. Inverse probability censored weighting (IPCW) is the default.
a	a numeric transformation parameter. The default value is 0, which applies no transformation. This parameter must be greater than `-1`. See `?tranSurvfit` for the transformation model structure.
trans	a character string specifying the transformation structure. The following are permitted: **linear** linear transformation structure, **log** log-linear transformation structure, **exp** exponential transformation structure.
...	for future methods.

Details

This function performs statistical test for quasi-independence between truncation time and failure time. The hypothesis test is based on the conditional Kendall's tau of Martin and Betensky (2005) and the two versions of the inverse probability weighted Kendall's tau of Austin and Betensky (2014).

The output contains the following components:

PE consistent point estimate of the conditional Kendall's tau.

SE asymptotic standard error of the conditional Kendall's tau estimator.

STAT the value of the normal test statistic.

p.value the (Wald) p-value of the test.

trans the transformation model (if applied).

a the estimated transformation parameter.

References

Martin E. and Betensky R. A. (2005), Testing quasi-independence of failure and truncation times via conditional Kendall's tau, *Journal of the American Statistical Association*, **100** (470): 484-492.

Austin, M. D. and Betensky R. A. (2014), Eliminating bias due to censoring in Kendall's tau estimators for quasi-independence of truncation and failure, *Computational Statistics & Data Analysis*, **73**: 16-26.

See Also

trSurvfit

Examples

```
data(channing, package = "boot")
chan <- subset(channing, sex == "Male" & entry < exit)
attach(chan)
cKendall(entry, exit, cens)
cKendall(entry, exit, cens, method = "IPW1")
cKendall(entry, exit, cens, method = "IPW2")
detach(chan)
```

---

gof *Goodness of fit based on left-truncated regression model*

Description

Provide goodness-of-fit diagnostics for the transformation model.

Usage

```
gof(x, B = 200, P = 1)
```

Arguments

x	an object of class `trSurvfit` returned by the `trSurvfit()` or the `trReg()` function or a survival object returned by the `Surv()`.
B	an integer value specifies the bootstrap size for the left-truncated regression model. A value greater than 2 is required.
P	an integer value specifies number of breakpoints to test the linearity of the transformation model. When `P > 0`, P breakpoints divides the event times into P equal spaced segments. Piecewise linear function constructed from those segments of event times are used in the left-truncated regression model, and the overall significance testing if the coefficient estimates are equal is reported. Default value for `P` is 1. See **Details** for a description of the goodness of fit procedure.

Details

The googness of fit assessment of the transformation model focus on the structure of the transformation model, which has the form:

$$h(U) = (1 + a)^{-1} \times (h(T) + ah(X)),$$

where T is the truncation time, X is the observed failure time, U is the transformed truncation time that is quasi-independent from X and $h(\cdot)$ is a monotonic transformation function. With the condition, $T < X$, assumed to be satisfied, the structure of the transformation model implies

$$X - T = -(1 + a)E(U) + (1 + a)X - (1 + a) \times [U - E(U)] := \beta_0 + \beta_1 X + \epsilon.$$

The regression estimates can be obtained by the left-truncated regression model (Karlsson and Lindmark, 2014). To evaluate the goodness of fit of the transformation model, the `gof()` function directly test the inearity in X by considering larger model that are nonlinear in X. In particular, we expand the covariates X to `P` piecewise linearity terms and test for equality of the associated coefficients.

Value

A list containing the following elements

coefficients the regression coefficients of the left-truncated regression model.

pval the p-value for the equality of the piecewise linearity terms in the expanded model. See **Details**.

References

Karlsson, M., Lindmark, A. (2014) truncSP: An R Package for Estimation of Semi-Parametric Truncated Linear Regression Models, *Journal of Statistical Software*, **57** (14), pp 1–19.

Examples

```
data(channing, package = "boot")
chan <- subset(channing, entry < exit)
fit <- trReg(Surv(entry, exit, cens) ~ sex, data = chan)
gof(fit, B = 10)
```

kendall	*Kendall's tau*

Description

Computes the unconditional Kendall's tau

Usage

```
kendall(x, y = NULL)
```

Arguments

x   a numeric vector.

y   a numeric vector with compatible length to x.

Details

This function computes the unconditional Kendall's tau (the Kendall rank correlation coefficient) for two variables. The returned value is equivalent to that from cor with method = "kendall", but kendall is implemented in C.

Value

The output is a numeric value for the unconditional Kendall's tau.

References

Kendall, M. G. (1938), A new measure of rank correlation, *Biometrika*, 81–93.

See Also

[cor](cor)

Examples

```
library(MASS)
set.seed(1)
dat <- mvrnorm(5000, c(0, 0), matrix(c(1, .5, .5, 1), 2))
## True kendall's tau is 2 * asin(.5) / pi
system.time(print(kendall(dat)))
system.time(print(cor(dat, method = "kendall")))
```

---

plot.trSurvfit	*Plot the survival estimation based on the structural transformation model*

Description

Plot the survival estimation for an `trSurvfit/trReg` object.

Usage

```
## S3 method for class 'trSurvfit'
plot(x, ...)
```

Arguments

x	an object of class `trSurvfit` returned by the `trSurvfit()` or the `trReg()` function.
...	graphical parameters to be passed to methods.

Value

A `ggplot` object.

Examples

```
data(channing, package = "boot")
chan <- subset(channing, entry < exit)

plot(trReg(Surv(entry, exit, cens) ~ 1, data = chan))
```

---

pmcc	*Product-Moment Correlation Coefficient*

Description

pmcc computes the conditional product-moment correlation coefficient proposed by Chen et al. (1996). The conditional product-moment correlation coefficient uses only the uncensored events.

Usage

```
pmcc(trun, obs, a = 0, trans = "linear")
```

Arguments

trun	left truncation time satisfying `trun <= obs`.
obs	observed failure time, must be the same length as `trun`, might be right-censored.
a	a numeric transformation parameter. The default value is 0, which applies no transformation. This parameter must be greater than `-1`. See `?tranSurvfit` for the transformation model structure.
trans	a character string specifying the transformation structure. The following are permitted:

> **linear** linear transformation structure,
>
> **log** log-linear transformation structure,
>
> **exp** exponential transformation structure.

See Also

trSurvfit

Examples

```
data(channing, package = "boot")
chan <- subset(channing, sex == "Male" & entry < exit & cens == 1)
with(chan, pmcc(entry, exit)) ## cannot handle right censored data
```

Surv	*This is the* Surv *function imported from* survival

Description

This function is imported from the `survival` package, used to create survival objects to be fitted with trReg() and trSurvfit. See Surv for the documentation on Surv.

---

trReg	*Fitting regression model via structural transformation model*

Description

`trReg` fits transformation model under dependent truncation and independent censoring via a structural transformation model.

Usage

```
trReg(
  formula,
  data,
  subset,
  tFun = "linear",
  method = c("kendall", "adjust"),
  B = 0,
  control = list()
)
```

Arguments

formula	a formula expression, of the form response ~ predictors. The response is assumed to be a survival::Surv object with both left truncation and right censoring. When there is no covariates, e.g., when the right hand side of the formula is ~ 1, the trReg() function returns a trSurvfit object. See ?survival::Surv for more details.
data	an optional data frame in which to interpret the variables occurring in the formula.
subset	an optional vector specifying a subset of observations to be used in the fitting process.
tFun	a character string specifying the transformation function or a user specified function indicating the relationship between X, T, and a. When tFun is a character, the following are permitted: **linear** linear transformation structure, **log** log-linear transformation structure, **exp** exponential transformation structure.
method	a character string specifying how the transformation parameter is estimated. The available options are "kendall" and "adjust". See **Details**.
B	a numerical value specifies the bootstrap size for estimating the standard error. When B = 0 (default), the bootstrap standard errors will not be computed.
control	a list of control parameters. The following arguments are allowed: lower The lower bound to search for the transformation parameter; default at -1. upper The upper bound to search for the transformation parameter; default at 20. tol The tolerance used in the search for the transformation parameter; default at 0.01. G The number of grids used in the search for the transformation parameter; default is 50. A smaller G could results in faster search, but might be inaccurate. Q The number of cutpoints for the truncation time used when method = "adjust". The default is 0.

P The number of breakpoints to divide the event times into equally spaced segmenets. When P > 1, the latent truncation time, $T'(a)$ will be computed in each subset. The transformation model is then applied to the aggregated data.

a The transformation parameter. When this is specified, the transformation model is applied based on the specified a. When this is not specified, an optimized a will be determined by optimization one of the quasi-independence measure. See **Details**.

parallel an logical value indicating whether parallel computation will be applied when B > 0.

parCl an integer value specifying the number of CPU cores to be used when parallel = TRUE. The default value is half the CPU cores on the current host.

Details

The main assumption on the structural transformation model is that it assumes there is a latent, quasi-independent truncation time that is associated with the observed dependent truncation time, the event time, and an unknown dependence parameter through a specified funciton. The structure of the transformation model is of the form:

$$h(U) = (1 + a)^{-1} \times (h(T) + ah(X)),$$

where T is the truncation time, X is the observed failure time, U is the transformed truncation time that is quasi-independent from X and $h(\cdot)$ is a monotonic transformation function. The condition, $T < X$, is assumed to be satisfied. The quasi-independent truncation time, U, is obtained by inverting the test for quasi-independence by one of the following methods:

method = "kendall" by minimizing the absolute value of the restricted inverse probability weighted Kendall's tau or maximize the corresponding p-value. This is the same procedure used in the trSUrvfit() function.

method = "adjust" includes a function of latent truncation time, U, as a covariate. A piece-wise function is constructed based on $(Q+1)$ indicator functions on whether U falls in the Qth and the $(Q + 1)$th percentile, where Q is the number of cutpoints used. See control for details. The transformation parameter, a, is then chosen to minimize the significance of the coefficient parameter.

See Also

trSurvfit

Examples

```
data(channing, package = "boot")
chan <- subset(channing, entry < exit)
trReg(Surv(entry, exit, cens) ~ sex, data = chan)
```

---

trSurv.control   *Auxiliary for Controlling trSurvfit Fitting*

Description

Auxiliary function as user interface for `trSurvfit` fitting.

Usage

```
trSurv.control(
  interval = c(-1, 20),
  lower = min(interval),
  upper = max(interval)
)
```

Arguments

interval	a vector containing the end-points of the interval to be searched the transformation parameter.
lower	the lower end-point of the interval to be searched.
upper	the upper end-point of the interval to be searched.

See Also

[trSurvfit](#)

---

trSurvfit   *Estimating survival curves via structural transformation model*

Description

`trSurvfit` estimates survival curves under dependent truncation and independent censoring via a structural transformation model.

Usage

```
trSurvfit(
  trun,
  obs,
  delta = NULL,
  tFun = "linear",
  plots = FALSE,
  control = trSurv.control(),
  ...
)
```

Arguments

trun	left truncation time satisfying `trun <= obs`.
obs	observed failure time, must be the same length as `trun`, might be right-censored.
delta	an optional 0-1 vector of censoring indicator (0 = censored, 1 = event) for `obs`. If this vector is not specified, `cKendall` assumes no censoring and all observed failure time denote events.
tFun	a character string specifying the transformation function or a user specified function indicating the relationship between X, T, and a. When `tFun` is a character, the following are permitted:

> **linear** linear transformation structure;
>
> **log** log-linear transformation structure;
>
> **exp** exponential transformation structure.

plots	an optional logical value; if TRUE, a series of diagnostic plots as well as the survival curve for the observed failure time will be plotted.
control	controls the lower and upper bounds when `trans` is an user specified function.
...	for future methods.

Details

A structural transformation model assumes there is a latent, quasi-independent truncation time that is associated with the observed dependent truncation time, the event time, and an unknown dependence parameter through a specified funciton. The dependence parameter is chosen to either minimize the absolute value of the restricted inverse probability weighted Kendall's tau or maximize the corresponding p-value. The marginal distribution for the truncation time and the event time are completely left unspecified.

The structure of the transformation model is of the form:

$$h(U) = (1 + a)^{-1} \times (h(T) + ah(X)),$$

where T is the truncation time, X is the observed failure time, U is the transformed truncation time that is quasi-independent from X and $h(\cdot)$ is a monotonic transformation function. The condition, $T < X$, is assumed to be satisfied. The quasi-independent truncation time, U, is obtained by inverting the test for quasi-independence by either minimizing the absolute value of the restricted inverse probability weighted Kendall's tau or maximize the corresponding p-value.

At the current version, three transformation structures can be specified. `trans = "linear"` corresponds to

$$h(X) = 1;$$

`trans = "log"` corresponds to

$$h(X) = log(X);$$

`trans = "exp"` corresponds to

$$h(X) = exp(X).$$

Value

The output contains the following components:

surv is a data.frame contains the survival probabilities estimates.

byTau a list contains the estimator of transformation parameter:

    par is the best set of transformation parameter found;

    obj is the value of the inverse probability weighted Kendall's tau corresponding to 'par'.

byP a list contains the estimator of transformation parameter:

    par is the best set of transformation parameter found;

    obj is the value of the inverse probability weighted Kendall's tau corresponding to 'par'.

qind a data frame consists of two quasi-independent variables:

    trun is the transformed truncation time;

    obs is the corresponding uncensored failure time.

References

Martin E. and Betensky R. A. (2005), Testing quasi-independence of failure and truncation times via conditional Kendall's tau, *Journal of the American Statistical Association*, **100** (470): 484-492.

Austin, M. D. and Betensky R. A. (2014), Eliminating bias due to censoring in Kendall's tau estimators for quasi-independence of truncation and failure, *Computational Statistics & Data Analysis*, **73**: 16-26.

Chiou, S., Austin, M., Qian, J. and Betensky R. A. (2018), Transformation model estimation of survival under dependent truncation and independent censoring, *Statistical Methods in Medical Research*, **28** (12): 3785-3798.

Examples

```
data(channing, package = "boot")
chan <- subset(channing, sex == "Male" & entry < exit)

## No display
(fit <- with(chan, trSurvfit(entry, exit, cens)))

## With diagnostic plots and the survival estimate
with(chan, trSurvfit(entry, exit, cens, plots = TRUE))

## Plots survival estimate

plot(fit)
```

wKendall	*Weighted conditional Kendall's tau*

Description

This is function computes the perturbed version of the conditional Kendall's tau.

Usage

```
wKendall(trun, obs, delta = NULL, weights = NULL)
```

Arguments

trun	left truncation time satisfying `trun <= obs`.
obs	observed failure time, must be the same length as `trun`, might be right-censored.
delta	an optional 0-1 vector of censoring indicator (0 = censored, 1 = event) for `obs`. If this vector is not specified, `cKendall` assumes no censoring and all observed failure time denote events.
weights	an optional perturbation weights.

Examples

```
data(channing, package = "boot")
chan <- subset(channing, sex == "Male" & entry < exit)
## When weights is not specified, this function reduces to condKendall()
with(chan, wKendall(entry, exit, cens))
mean(replicate(1000, with(chan, wKendall(entry, exit, cens, rexp(nrow(chan))))))
```

Index