

# Package ‘coxrt’

January 7, 2019

**Type** Package

**Title** Cox Proportional Hazards Regression for Right Truncated Data

**Version** 1.0.1

**Author** Bella Vakulenko-Lagun, Micha Mandel, Rebecca A. Betensky

**Maintainer** Bella Vakulenko-Lagun <[bella.vakulenko-lagun@mail.huji.ac.il](mailto:bella.vakulenko-lagun@mail.huji.ac.il)>

**Description** Fits Cox regression based on retrospectively ascertained times-to-event. The method uses Inverse-Probability-Weighting estimating equations.

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 3.0.0)

**Imports** survival, BB, inline, gss, ggplot2

**LinkingTo** Rcpp, RcppArmadillo

**RoxygenNote** 6.1.0

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**URL** <https://github.com/Bella2001/coxrt>

**BugReports** <https://github.com/Bella2001/coxrt/issues>

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2019-01-05 20:43:46

## R topics documented:

coxph.RT . . . . .	2
coxph.RT.a0 . . . . .	4
coxrt . . . . .	6

<b>Index</b>	<b>7</b>
--------------	----------

---

 coxph.RT

*Fits Cox Regression Model Using Right Truncated Data*


---

### Description

Estimates covariate effects in a Cox proportional hazard regression from right-truncated survival data assuming positivity, that is  $P(\text{lifetime} > \max(\text{right}) \mid Z=0) = 0$ .

### Usage

```
coxph.RT(formula, right, data, bs = FALSE, nbs.rep = 500,
          conf.int = 0.95)
```

### Arguments

formula	a formula object, with the response on the left of a ~ operator, and covariates on the right. The response is a target lifetime variable.
right	a right truncation variable.
data	a data frame that includes the variables used in both sides of formula and in right. The observations with missing values in one of the variables are dropped.
bs	logical value: if TRUE, the bootstrap estimator of standard error, confidence interval, and confidence upper and lower limits for one-sided confidence intervals based on the bootstrap distribution are calculated. The default value is FALSE.
nbs.rep	number of bootstrap replications. The default number is 200.
conf.int	The confidence level for confidence intervals and hypotheses tests. The default level is 0.95.

### Details

When positivity does not hold, the estimator of regression coefficients will be biased. But if all the covariates are independent in the population, the Wald test performed by this function is still valid and can be used for testing partial hypotheses about regression coefficients even in the absence of positivity. If the covariates are not independent and positivity does not hold, the partial tests cannot guarantee the correct level of type I error.

### Value

A list with components:

coef	an estimate of regression coefficients
var	covariance matrix of estimates of regression coefficients based on the analytic formula
n	the number of observations used to fit the model
summary	a data frame with a summary of fit:

- coef a vector of coefficients

- `exp.coef` exponent of regression coefficients (=hazard ratio)
  - SE asymptotic standard error estimate based on the analytic formula derived in Vakulenko-Lagun et al. (2018)
  - `CI.L` lower confidence limit for two-sided hypothesis  $H_0: \beta_i = 0$
  - `CI.U` upper confidence limit for two-sided hypothesis  $H_0: \beta_i = 0$
  - `pvalue` p-value from a Wald test for a two-sided hypothesis  $H_0: \beta_i = 0$
  - `pvalue.H1.b.gr0` p-value from the Wald test for a one-sided partial hypothesis  $H_0: \beta_i \leq 0$  based on the analytical asymptotic standard error estimate
  - `pvalue.H1.b.le0` p-value from the Wald test a for one-sided partial hypothesis  $H_0: \beta_i \geq 0$  based on the analytical asymptotic standard error estimate
- `bs` if the input argument `bs` was TRUE, then an output list also includes an element `bs` with statistics from the bootstrap distribution of estimated coefficients:
- `num.bs.rep` number of bootstrap replications used to obtain the sample distribution
  - `var` estimated variance
  - `summary` a data frame with a summary of bootstrap distribution that includes: SE, a bootstrap estimated standard error; `CI.L`, a quantile estimated lower confidence limit for two-sided hypothesis  $H_0: \beta_i = 0$ ; `CI.U`, a quantile estimated upper confidence limit for two-sided hypothesis  $H_0: \beta_i = 0$ ; `CI.L.H1.b.gr0`, a quantile estimated the limit for one-sided hypothesis  $H_0: \beta_i \leq 0$ ; `CI.U.H1.b.le0`, a quantile estimated the limit for one-sided hypothesis  $H_0: \beta_i \geq 0$ .

### See Also

[coxph.RT.a0](#), [coxrt](#), [coxph](#)

### Examples

```
# loading AIDS data set
library(gss)
data(aids)
s <- data.frame(age=aids$age, ageg=as.numeric(aids$age<=4), T=aids$incu, R=aids$infe)
s$T[s$T==0] <- 0.5 # as in Kalbfleisch and Lawless (1989)

# analysis assuming positivity
# we request bootstrap SE estimate as well:
sol <- coxph.RT(T~ageg, right=R, data=s, bs=FALSE)
sol
sol$summary # print the summary of fit based on the analytic Asymptotic Standard Error estimate
```

---

 coxph.RT.a0

*Fits Cox Regression with Adjustment for the Lack of Positivity*


---

### Description

Estimates covariate effects in a Cox proportional hazard regression from right truncated survival data for a given value of  $a_0 = P(\text{lifetime} > \max(\text{right}) \mid Z=0)$ . This probability reflects the chance of falling to the right of the upper bound of the support of the right truncation variable in the reference stratum where all the covariates are zero. Right truncation might result in a completely unobserved right tail of the distribution of the target lifetime. That means that it can happen there will be no "representatives" in a sample from the right tail. Ignoring this and using `coxph.RT` in general will result in biased estimation of regression coefficients, whereas `coxph.RT.a0` allows to account for this violation.

### Usage

```
coxph.RT.a0(formula, right, data, a0 = 0, bs = FALSE, nbs.rep = 200,
  conf.int = 0.95)
```

### Arguments

<code>formula</code>	a formula object, with the response on the left of a <code>~</code> operator, and covariates on the right. The response is a target lifetime variable.
<code>right</code>	a right truncation variable.
<code>data</code>	a data frame that includes the variables used in <code>formula</code> and in <code>right</code> .
<code>a0</code>	probability of falling into the unobservable region in the stratum of $Z=0$ , i.e. $P(\text{lifetime} > \max(\text{right}) \mid Z=0)$ . By default $a_0=0$ , which is equivalent to assuming positivity.
<code>bs</code>	logical value: if TRUE, the bootstrap estimator of standard error, confidence interval, and confidence upper and lower limits for one-sided confidence intervals based on the bootstrap distribution are calculated. The default value is FALSE.
<code>nbs.rep</code>	number of bootstrap replications. The default number is 200.
<code>conf.int</code>	The confidence level for confidence intervals and hypotheses tests. The default level is 0.95.

### Value

a list with components:

<code>convergence</code>	convergence code as returned by <code>BBsolve</code> . convergence > 0 means that the algorithm diverged and a solution was not found. <code>BBsolve</code> is used with a default parameters setting.
<code>coef</code>	a vector of estimated regression coefficients.
<code>var</code>	covariance matrix of regression coefficients, if the input argument <code>bs</code> was TRUE; NULL, otherwise.

n the number of observations used to fit the model.  
 a0 plugged-in value of  $a_0$ .  
 bs if the input argument bs was TRUE, then an output list also includes an element bs with statistics from the bootstrap distribution of estimated coefficients:

- num.bs.rep number of successful bootstrap replications used to obtain the sample distribution
- var estimated variance of regression coefficients
- summary a data frame with a summary of bootstrap distribution that includes: coef, a vector of estimated regression coefficients; exp.coef, an exponent of regression coefficients (=hazard ratio); SE, a bootstrap estimated standard error; CI.L, a quantile estimated lower confidence limit for two-sided hypothesis  $H_0: \beta_i = 0$ ; CI.U, a quantile estimated upper confidence limit for two-sided hypothesis  $H_0: \beta_i = 0$ ; CI.L.H1.b.gr0, a quantile estimated the limit for one-sided hypothesis  $H_0: \beta_i \leq 0$ ; CI.U.H1.b.le0, a quantile estimated the limit for one-sided hypothesis  $H_0: \beta_i \geq 0$ .

### See Also

[coxph.RT, BBSolve](#)

### Examples

```
# loading AIDS data set
library(gss)
data(aids)
s <- data.frame(age=aids$age, ageg=as.numeric(aids$age<=4), T=aids$incu, R=aids$infe)
s$T[s$T==0] <- 0.5 # as in Kalbfleisch and Lawless (1989)

# analysis using adjusted estimating equations for a0=0.2
sol.02 <- try(coxph.RT.a0(T~ageg, right=R, data=s, a0=0.2, bs=FALSE))
sol.02
# for a0=0
sol <- try(coxph.RT(T~ageg, right=R, data=s, bs=FALSE) )
sol$summary # print the summary of fit based on the asymptotic SE estimate

# sensitivity analysis for different values of a0
a_ <- seq(0.05, 0.55, by=0.05)
est <- NULL

for(q in 1:length(a_))
{
  sol.a <- try(coxph.RT.a0(T~ageg, right=R, data=s, a0=a_[q], bs=FALSE))
  if (sol.a$convergence!=0)
  {
    cat("a0=", a_[q], ". Error occurred in BBSolve.\n")
  } else
  {
    cat("a=", a_[q], " ", " IPW.adj.est=", sol.a$coef, "\n")
    est <- c(est, sol.a$coef)
  }
}
```

```

}
require(ggplot2)
res.d <- data.frame(a0=c(0, a_), beta=c(sol$coef, est))

p <- ggplot(res.d, aes(x=a0, y=beta)) +
  geom_line() + geom_point() +
  geom_hline(yintercept=0)
p + xlab(expression( paste(a[0], "=P(T>", r['*']," | z=0)" , sep=""))) +
  ylab(expression( paste(hat(beta), "(", a[0], ")" , sep=""))) ) +
  scale_x_continuous(breaks=res.d$a0, labels=res.d$a0) +
  theme(axis.text.x = element_text(face="bold", angle=45),
        axis.text.y = element_text(face="bold"))

```

---

coxrt

*A Package to Fit the Cox Regression from Right Truncated Data*


---

## Description

The method assumes that truncation is independent of covariates, and of lifetime, and that there is no censoring. The method uses Inverse-Probability-Weighting estimating equations with stabilized weights, IPW-S and IPW-SA, as described in Vakulenko-Lagun et al. (2018). Currently the code allows only time-independent covariates.

## Details

The **coxrt** package provides two functions: `coxph.RT` (IPW-S) that assumes positivity and `coxph.RT.a0` (IPW-SA) that allows for adjustment of estimation using plugged-in  $a_0$ . The illustrative examples in these functions include analysis of AIDS latency data with age as a covariate, where the AIDS cases were retrospectively ascertained at June 30, 1986, and only those who developed AIDS by that time were included in the analysis (Kalbfleisch and Lawless, 1989).

## References

- Vakulenko-Lagun, B., Mandel, M., Betensky, R.A. Inverse-Probability-Weighting methods for Cox regression with right-truncated data. 2018, submitted to *Biometrics*
- Kalbfleisch, J.D. and Lawless, J.F. Inference based on retrospective ascertainment: an analysis of the data on transfusion-related AIDS. *Journal of the American Statistical Association*, 84 (406):360-372, 1989.

# Index

BBsolve, [4](#), [5](#)

coxph, [3](#)

coxph.RT, [2](#), [4–6](#)

coxph.RT.a0, [3](#), [4](#), [6](#)

coxrt, [3](#), [6](#)

coxrt-package (coxrt), [6](#)