Package 'CoxAIPW'

May 7, 2026

Type Package

Title Doubly Robust Inference for Cox Marginal Structural Model with Informative Censoring

Version 0.0.3

Description Doubly robust estimation and inference of log hazard ratio under the Cox marginal structural model with informative censoring. An augmented inverse probability weighted estimator that involves 3 working models, one for conditional failure time T, one for conditional censoring time C and one for propensity score. Both models for T and C can depend on both a binary treatment A and additional baseline covariates Z, while the propensity score model only depends on Z. With the help of cross-fitting techniques, achieves the rate-doubly robust property that allows the use of most machine learning or non-parametric methods for all 3 working models, which are not permitted in classic inverse probability weighting or doubly robust estimators. When the proportional hazard assumption is violated, CoxAIPW estimates a causal estimated that is a weighted average of the time-varying log hazard ratio. Reference: Luo, J. (2023). Statistical Robustness - Distributed Linear Regression, Informative Censoring, Causal Inference, and Non-Proportional Hazards [Unpublished doctoral dissertation]. University of California San Diego.; Luo & Xu (2022) <doi:10.48550/arXiv.2206.02296>; Rava (2021) <https://escholarship.org/uc/item/8h1846gs>.

License GPL-3

URL https://github.com/charlesluo1002/CoxAIPW

BugReports https://github.com/charlesluo1002/CoxAIPW/issues

Imports survival, randomForestSRC, polspline, tidyr, ranger, pracma, gbm

Encoding UTF-8

Language en-US

RoxygenNote 7.2.3

NeedsCompilation no

Author Jiyu Luo [cre, aut], Dennis Rava [aut], Ronghui Xu [aut]

Maintainer Jiyu Luo <charlesluo1002@gmail.com>

Repository CRAN

Date/Publication 2023-09-20 07:30:06 UTC

Contents

CoxAIPW 2

Index **5**

`CoxAIPW` *CoxAIPW*

Description

Doubly robust estimation of two-group log hazard ratio under the Cox marginal structural model, works for observational data with informative censoring.

Usage

```
CoxAIPW(
  data,
  Tmod = "Cox",
  Cmod = "Cox",
  PSmod = "logit",
  tau = NULL,
  k = 5,
  beta0 = 0,
  min.S = 0.05,
  min.PS = 0.1,
  weights = NULL,
  augmentation = "AIPTCW",
  crossFit = TRUE
)
```

Arguments

`data`	A matrix or dataframe object, where the first column is observed event time, second column is event indicator (1 = event, 0 = censored), third column is a binary group indicator A (numeric 0 or 1), the other columns are baseline covariates Z.
`Tmod`	working conditional model for failure time T given A,Z, currently supports values ('Cox', 'Spline', 'RSF'), corresponding to Cox PH, hazard regression, and Random survival forest.
`Cmod`	working conditional model for censoring time C given A,Z, currently supports values ('Cox', 'Spline', 'RSF'), corresponding to Cox PH, hazard regression, and Random survival forest.

PSmod	working model for propensity score P(A=1|Z), currently supports values ('logit','RF','twang'), corresponding to logistic regression, random forest, and the package 'twang' (gradient boosted logistic regression).
tau	the cutoff time of the study, which should be no less than all observed event times. Default to NULL, which sets tau to the largest observed event time.
k	the number of folds used in cross-fitting, default to 5.
beta0	initial value for beta in the optimization algorithm, default to 0.
min.S	the minimum threshold for estimated survival functions P(C>t|A,Z) and P(T>t|A,Z), any estimated value below min.S is set as min.S to avoid exploding gradient. Default to 0.05
min.PS	the minimum threshold for estimated propensity score, any estimated value below min.PS or above 1-min.PS is set to min.PS or 1-min.PS to avoid exploding gradient. Default to 0.1
weights	a vector of length equal to the samples size that assigns weight to each observation. Also used in Bayesian Bootstrap. Default to NULL (equal weights).
augmentation	a string indicating the type of augmention to consider. 'AIPTCW' performs augmented inverse probability of treatment and censoring weighting, which works for observational data with informative censoring. 'AIPTW' performs augmented inverse probability of treatment weighting, which only works under observational data with random censoring. 'AIPCW' performs augmented inverse probability of censoring weighting, which only works under randomized study with informative censoring. Default to 'AIPTCW'.
crossFit	a logical variable indicating whether to apply cross-fitting. If set to FALSE, all three nuisance function estimators 'Tmod', 'Cmod', and 'PSmod' need to be root-n. Default to TRUE.

Value

a list object containing 'beta', 'model_se', 'Lambda0', 'survival0', 'survival1' and 'beta_t_plot'. 'beta' is the estimated log hazard ratio, 'model_se' is the model-based asymptotic standard error estimator used for inference, 'Lambda0' is the estimated cumulative baseline hazard function, evaluated on each follow up time, and 'survival0' and 'survival1' are the estimated survival curves, evaluated on each follow up time, for treatment group 0 and 1 respectively. 'beta_t_plot' is a list made up of an x-values vector and a y-values vector, which leads to a plot of time-varying log hazard ratio beta(t) against time t after applying loess or spline smoothing.

Examples

```
# run Cox AIPW on the cancer data set from survival package
data(cancer, package = 'survival')
data = data.frame(
  time = cancer$time,
  status = cancer$status - 1,
  A = cancer$sex - 1,
  age = cancer$age)
aipw = CoxAIPW(data)
```

```
# extract beta and model SE.
logHR = aipw$beta
modelSE = aipw$model_se

# extract cumulative baseline hazard function.
cumBaseHaz = aipw$Lambda0

# extract survival curve for group 0 and survival curve for group 1.
surv0 = aipw$survival0
surv1 = aipw$survival1

# extract x and y values for a plot of time-varying log hazard ratio beta(t).
x = aipw$beta_t_plot$x
y = aipw$beta_t_plot$y
# library(splines)
# spline_fit = lm(y ~ ns(x, df=2, intercept =F))
# X = seq(0, max(x), length = 40)
# plot(X, predict(spline_fit, newdata = data.frame(x = X)), type = 'l')
```

Index

CoxAIPW, 2