Package 'CoxICPen'

Title Variable Selection for Cox's Model with Interval-Censored Data

Version 1.1.0

Maintainer Qiwei Wu <qw235@mail.missouri.edu>

Imports foreach

Description Perform variable selection for Cox regression model with interval-censored data. Can deal with both low-dimensional and high-dimensional data. Case-cohort design can be incorporated. Two sets of covariates scenario can also be considered. The references are listed in the URL below.

License Apache License (>= 2)

Encoding UTF-8

URL https://doi.org/10.1080/01621459.2018.1537922,

https://doi.org/10.1002/sim.8594,

https://doi.org/10.1002/bimj.201900180

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

Author Qiwei Wu [aut, cre],
Hui Zhao [aut],
Jianguo Sun [aut]

Repository CRAN

Date/Publication 2021-02-23 14:10:07 UTC

Contents

CoxICPen	*Variable Selection for Cox's Model with Interval-Censored Data*

Description

Perform variable selection for Cox regression model with interval-censored data by using the methods proposed in Zhao et al. (2020a), Wu et al. (2020) and Zhao et al. (2020b). Can deal with both low-dimensional and high-dimensional data.

Usage

```
CoxICPen(LR = LR,
         x = x,
         lamb = log(nrow(x))/2-2,
         beta.initial = rep(0,ncol(x)),
         pen = "BAR",
         nfold = 5,
         BernD = 3,
         subj.wt = rep(1,nrow(x)))
```

Arguments

LR	An n by 2 matrix that contains interval-censored failure times (L, R]. Please set time point R to "Inf" if a subject is right-censored.
x	An n by p covariate matrix.
lamb	The value of the tuning parameter of the penalty term. Can either be a single value or a vector. Cross-validation will be employed to select the optimal lambda if a vector is provided. Default is log(n)/2-2.
beta.initial	The initial values for the regression coefficients in the Cox's model. Default is 0.
pen	The penalty function. Choices include "RIDGE", "BAR", "LASSO", "ALASSO", "SCAD", "MCP", "SICA", "SELO". Default is "BAR".
nfold	Number of folds for cross-validation. Will be ignored if a single lambda value is provided. Default is 5.
BernD	The degree of Bernstein polynomials. Default is 3.
subj.wt	Weight for each subject in the likelihood function. Can be used to incorporate case-cohort design. Default is 1 for each subject.

Value

beta: Penalized estimates of the regression coefficients in the Cox's model.

phi: Estimates of the coefficients in Bernstein Polynomials.

logL: Log likelihood function based on current parameter estimates and lambda value.

Lamb0: Estimate of the cumulative baseline hazard function at each observation time point.

cv.out: Cross-validation outcome for each lambda. Will be NULL if cross-validation is not performed.

References

Zhao, H., Wu, Q., Li, G., Sun, J. (2020a). Simultaneous Estimation and Variable Selection for Interval-Censored Data with Broken Adaptive Ridge Regression. Journal of the American Statistical Association. 115(529):204-216.

Wu, Q., Zhao, H., Zhu, L., Sun, J. (2020). Variable Selection for High-dimensional Partly Linear Additive Cox Model with Application to Alzheimer's disease. Statistics in Medicines.39(23):3120-3134.

Zhao, H., Wu, Q., Gilbert, P. B., Chen, Y. Q., Sun, J. (2020b). A Regularized Estimation Approach for Case-cohort Periodic Follow-up Studies with An Application to HIV Vaccine Trials. Biometrical Journal. 62(5):1176-1191.

Examples

```
# Generate an example data

require(foreach)

n <- 300  # Sample size
p <- 20   # Number of covariates

bet0 <- c(1, -1, 1, -1, rep(0,p-4))  # True values of regression coefficients

set.seed(1)
x.example <- matrix(rnorm(n*p,0,1),n,p)  # Generate covariates matrix

T.example <- c()
for (i in 1:n){
  T.example[i] <- rexp(1,exp(x.example%*%bet0)[i])  # Generate true failure times
}

timep <- seq(0,3,,10)
LR.example <- c()
for (i in 1:n){
  obsT <- timep*rbinom(10,1,0.5)
  if (max(obsT) < T.example[i]) {LR.example <- rbind(LR.example,c(max(obsT), Inf))} else {
   LR.example <- rbind(LR.example,c(max(obsT[obsT<T.example[i]]), min(obsT[obsT>=T.example[i]])))
   }
}  # Generate interval-censored failure times


# Fit Cox's model with penalized estimation

model1 <- CoxICPen(LR = LR.example, x = x.example, lamb = 100, pen = "RIDGE")
beta.initial <- model1$beta

model2 <- CoxICPen(LR = LR.example, x = x.example, beta.initial = beta.initial, pen = "BAR")
model2$beta

#model3 <- CoxICPen(LR = LR.example, x = x.example, lamb = seq(0.1,1,0.1),
#                   beta.initial = beta.initial, pen = "SELO")
#model3$beta
```

CoxICPen.XZ   *CoxICPen with two sets of covariates*

Description

Perform variable selection for Cox regression model with two sets of covariates by using the method in Wu et al. (2020). Variable selection is performed on the possibly high-dimensional covariates x with linear effects. Covariates z with possibly nonlinear effects are always kept in the model.

Usage

```
CoxICPen.XZ(LR = LR,
            x = x,
            z = z,
            lamb = log(nrow(x))/2-2,
            beta.initial = rep(0,ncol(x)),
            pen = "BAR",
            nfold = 5,
            BernD = 3,
            subj.wt = rep(1,nrow(x)))
```

Arguments

LR	An n by 2 matrix that contains interval-censored failure times (L, R]. Please set time point R to "Inf" if a subject is right-censored.
x	An n by p covariate matrix. Variable selection will be performed on x. Linear covariates effects are assumed. Both p>n and p<n are allowed.
z	An n by q covariate matrix. Variable selection will NOT be performed on z. Non-linear covariates effects are assumed. Only q<n is allowed.
lamb	The value of the tuning parameter of the penalty term. Can either be a single value or a vector. Cross-validation will be employed to select the optimal lambda if a vector is provided. Default is log(n)/2-2.
beta.initial	The initial values for the regression coefficients in the Cox's model. Default is 0.
pen	The penalty function. Choices include "RIDGE", "BAR", "LASSO", "ALASSO", "SCAD", "MCP", "SICA", "SELO". Default is "BAR".
nfold	Number of folds for cross-validation. Will be ignored if a single lambda value is provided. Default is 5.
BernD	The degree of Bernstein polynomials for both cumulative baseline hazard and covariate effects of z. Default is 3.
subj.wt	Weight for each subject in the likelihood function. Can be used to incorporate case-cohort design. Default is 1 for each subject.

Value

beta: Penalized estimates of the regression coefficients in the Cox's model.

phi: Estimates of the coefficients in Bernstein Polynomials.

logL: Log likelihood function based on current parameter estimates and lambda value.

Lamb0: Estimate of the cumulative baseline hazard function at each observation time point.

cv.out: Cross-validation outcome for each lambda. Will be NULL if cross-validation is not performed.

f.est.all: A matrix that contains the values of covariates z and the corresponding estimated effects.

References

Wu, Q., Zhao, H., Zhu, L., Sun, J. (2020). Variable Selection for High-dimensional Partly Linear Additive Cox Model with Application to Alzheimer's disease. Statistics in Medicines.39(23):3120-3134.

Examples

```
# Generate an example data

require(foreach)

n <- 300  # Sample size
p <- 20   # Number of covariates

bet0 <- c(1, -1, 1, -1, rep(0,p-4))  # True values of regression coefficients
f1 <- function(z) sin(2*pi*z)  # True effects of z1
f2 <- function(z) cos(2*pi*z)  # True effects of z2
set.seed(1)
x.example <- matrix(rnorm(n*p,0,1),n,p)  # Generate x covariates matrix
z.example <- cbind(runif(n,0,1),runif(n,0,1))  # Generate z covariates matrix

T.example <- c()
for (i in 1:n){
  T.example[i] <- rexp(1,exp(x.example%*%bet0+
    f1(z.example[,1])+f2(z.example[,2]))[i])  # Generate true failure times
}

timep <- seq(0,3,,10)
LR.example <- c()
for (i in 1:n){
  obsT <- timep*rbinom(10,1,0.5)
  if (max(obsT) < T.example[i]) {LR.example <- rbind(LR.example,c(max(obsT), Inf))} else {
   LR.example <- rbind(LR.example,c(max(obsT[obsT<T.example[i]]), min(obsT[obsT>=T.example[i]])))
   }
}  # Generate interval-censored failure times


# Fit Cox's model with penalized estimation
```

```
model1 <- CoxICPen.XZ(LR = LR.example, x = x.example, z = z.example, lamb = 100, pen = "RIDGE")
beta.initial <- model1$beta

model2 <- CoxICPen.XZ(LR = LR.example, x = x.example, z = z.example,
                      beta.initial = beta.initial, pen = "BAR")
model2$beta

# Plots of covariate effects of z

par(mfrow=c(1,2))

plot(model2$f.est.all$z1, model2$f.est.all$f1, type="l", ylim=c(-1,2),
     xlab="z1", ylab="f1")
lines(model2$f.est.all$z1, f1(model2$f.est.all$z1), col="blue")
legend("topright", col=c("black","blue"), lty=rep(1,2), c("Estimate", "True"))

plot(model2$f.est.all$z2, model2$f.est.all$f2, type="l", ylim=c(-1,2),
     xlab="z2", ylab="f2")
lines(model2$f.est.all$z2, f2(model2$f.est.all$z2), col="blue")
legend("topright", col=c("black","blue"), lty=rep(1,2), c("Estimate", "True"))
```

Index