

Package ‘survC1’

February 20, 2015

Type Package

Title C-statistics for risk prediction models with censored survival data

Version 1.0-2

Date 2013-02-14

Author Hajime Uno

Maintainer Hajime Uno <huno@jimmy.harvard.edu>

Depends survival

Description Performs inference for C of risk prediction models with censored survival data, using the method proposed by Uno et al. (2011). Inference for the difference in C between two competing prediction models is also implemented.

License GPL-2

LazyLoad yes

Repository CRAN

Date/Publication 2013-02-14 12:31:24

NeedsCompilation yes

R topics documented:

survC1-package	2
CompCase	3
conc	4
cvC	4
Est.Cval	5
Est.PH	6
Inf.Cval	7
Inf.Cval.Delta	8
kmcens	9
unoCW	9
unoU2P	10
VEC2MAT	10

Index**11**

survC1-package	<i>C-statistics for risk prediction models with censored survival data</i>
----------------	--

Description

Performs inference of overall adequacy of risk prediction models with censored survival data.

Details

Package:	survC1
Type:	Package
Version:	1.0-2
Date:	2012-02-14
License:	GPL-2
LazyLoad:	yes

Performs inference for C of risk prediction models with censored survival data, using the method proposed by Uno et al. (2011). Inference for the difference in C between two competing prediction models is also implemented.

Author(s)

Hajime Uno

Maintainer: Hajime Uno <huno@jimmy.harvard.edu>

References

Hajime Uno, Tianxi Cai, Michael J. Pencina, Ralph B. D'Agostino, and LJ Wei. On the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Statistics in Medicine* 2011, 30:1105-16.

Examples

```
#####
# read sample data (PBC in survival package)
#####
D=CompCase(pbc[1:200,c(2:4,10:14)])
D[,2]=as.numeric(D[,2]==2)

#####
# Inference of C
#####
tau=365.25*8
C=Inf.Cval(D, tau, itr=200)
```

```

round(c(C$Dhat, C$se, C$low95, C$upp95), digits=3)

#####
# Inference of Delta C between 2 models
#####
model0<-D[,c(1:2,4:5)] ;
model1<-D
covs1<-as.matrix(model1[,c(-1,-2)])
covs0<-as.matrix(model0[,c(-1,-2)])

Delta=Inf.Cval.Delta(model0[,1:2], covs0, covs1, tau, itr=200)
round(Delta, digits=3)

#####
# Point estimation via cross-validation
#####
model1=D[,c(1,2,4)]

cvC(model1,tau,cvK=2,Rep=10)

```

CompCase

Creates complete case data

Description

Creates a complete-case data set, removing subjects with any missing value in the outcome or predictors

Usage

```
CompCase(mydata)
```

Arguments

mydata A data matrix that consists of numeric data. No character data is allowed.

Examples

```
D=CompCase(pbc[,c(2:4,10:14)])
```

conc *A child function to calculate concordance*

Description

This is a child function (R-wrapper) called in the main functions in this package.

Usage

```
conc(X, D, W, R)
```

Arguments

X	time
D	event indicator (1:event, 0:ensor)
W	weight
R	risk score

cvC *Cross-validation estimate for C*

Description

Provides a cross-validation estimate for C

Usage

```
cvC(mydata, tau, cvK = 10, Rep = 10)
```

Arguments

mydata	Input data. The 1st column should be time-to-event, and the 2nd column is event indicator (1=event, 0=censor). The rest of the columns are covariates/predictors used in the model. No character variable or missing is allowed.
tau	Truncation time. The resulting C tells how well the given prediction model works in predicting events that occur in the time range from 0 to tau. Note that the survival function for the underlying censoring time distribution needs to be positive at tau.
cvK	The number of fold. A K-fold cross-validation is performed, according to the number given for cvK. Generally, a value from 2 to 10 will be specified.
Rep	The number of iteration of the random splits for the K-fold cross-validation. The average of the Rep times of cvK-fold cross-validation estimates is provided as a resulting estimate for C.

Examples

```

D=CompCase(pbc[,c(2:4,10:14)])
D[,2]=as.numeric(D[,2]==2)

tau=365.25*8

model1=D[1:200,c(1,2,4)]

cvC(model1,tau,cvK=2,Rep=10)

```

Est.Cval

Estimation of C

Description

Performs a point estimation of C

Usage

```
Est.Cval(mydata, tau, nofit=FALSE)
```

Arguments

mydata	Input data. The 1st column should be time-to-event, and the 2nd column is event indicator (1=event, 0=censor). The rest of the columns are covariates/predictors used in the model. No character variable or missing is allowed.
tau	Truncation time. The resulting C tells how well the given prediction model works in predicting events that occur in the time range from 0 to tau. Note that the survival function for the underlying censoring time distribution needs to be positive at tau.
nofit	If TRUE, the 3rd column of mydata is used as the risk score directly in calculation of C. The default is FALSE.

Value

Dhat	Estimate of C
beta	Estimates for regression coefficient in the Cox model
beta.var	Variance-Covariance matrix for beta above
rs	Risk score of each individual
ft	coxph object with the fitted model

Examples

```
D=CompCase(pbc[1:200,c(2:4,10:14)])
D[,2]=as.numeric(D[,2]==2)
tau=365.25*8
model=D[,c(1,2,4,5)]

Est.Cval(model, tau)$Dhat
```

 Est.PH

Derivation of a risk score by a Cox proportional hazards model

Description

Provides risk score by fitting data to a Cox's proportional hazards model with a given set of predictors.

Usage

```
Est.PH(mydata)
```

Arguments

mydata	Input data. The 1st column should be time-to-event, and the 2nd column is event indicator (1=event, 0=censor). The rest of the columns are covariates/predictors used in the model. No character variable or missing is allowed.
--------	--

Value

beta	Estimates for regression coefficient in the Cox model
var	Variance-Covariance matrix for the beta above
rs	Risk score of each individual
ft	coxph object with the fitted model

Examples

```
D=CompCase(pbc[1:200,c(2:4,10:14)])
D[,2]=as.numeric(D[,2]==2)

ft=Est.PH(D)
```

Inf.Cval *Inference of C*

Description

Performs the inference for C

Usage

```
Inf.Cval(mydata, tau, itr = 1000, seed = NULL)
```

Arguments

mydata	Input data. The 1st column should be time-to-event, and the 2nd column is event indicator (1=event, 0=censor). The rest of the columns are covariates/predictors used in the model. No character variable or missing is allowed.
tau	Truncation time. The resulting C tells how well the given prediction model works in predicting events that occur in the time range from 0 to tau. Note that the survival function for the underlying censoring time distribution needs to be positive at tau.
itr	Iteration of perturbation-resampling.
seed	A seed for generating random numbers for the resampling. The default is NULL.

Value

Dhat	Estimate for C
se	Standard error estimate
low95	Lower limit of a 0.95 two-sided confidence interval for C
upp95	Upper limit of a 0.95 two-sided confidence interval for C
ft	coxph object with the fitted model

Examples

```
D=CompCase(pbc[1:200,c(2:4,10:14)])
D[,2]=as.numeric(D[,2]==2)

tau=365.25*8

C=Inf.Cval(D, tau, itr=200)
round(c(C$Dhat, C$se, C$low95, C$upp95), digits=3)
```

 Inf.Cval.Delta

Inference of Delta C

Description

Performs the inference for the difference in C between two competing risk prediction models

Usage

```
Inf.Cval.Delta(mydata, covs0, covs1, tau, itr = 1000, seed = NULL)
```

Arguments

mydata	Input data. The 1st column should be time-to-event, and the 2nd column is event indicator (1=event, 0=censor).
covs0	A matrix that consists of a set of predictors for a base model (Model 0)
covs1	A matrix that consists of a set of predictors for a new model (Model 1)
tau	Truncation time. The resulting C tells how well the given prediction model works in predicting events that occur in the time range from 0 to tau. Note that the survival function for the underlying censoring time distribution needs to be positive at tau.
itr	Iteration of perturbation-resampling.
seed	A seed for generating random numbers for the resampling. The default is NULL.

Value

A 4x3 matrix that consists of point estimates, standard error estimates, and lower and upper limits of 0.95 two-sided intervals for C of Model 1, Model 0 and that for the difference of C

Examples

```
D=CompCase(pbc[1:200,c(2:4,10:14)])
D[,2]=as.numeric(D[,2]==2)

model0<-D[,c(1:2,4:5)] ;
model1<-D
covs1<-as.matrix(model1[,c(-1,-2)])
covs0<-as.matrix(model0[,c(-1,-2)])

tau=365.25*8

Delta=Inf.Cval.Delta(model0[,1:2], covs0, covs1, tau, itr=200)
round(Delta, digits=3)
```

kmcens	<i>Kaplan-Meier for Censoring</i>
--------	-----------------------------------

Description

This is a child function called in the main functions in this package. Specifically, this obtains the survival function for censoring time and gives quantities used in the perturbation-resampling method for C.

Usage

```
kmcens(time, status, tau)
```

Arguments

time	time to event variable
status	indicator of event (1:event, 0: censor)
tau	see cvK

unoCW	<i>A child function used in Inf.Cval and Inf.Cval.Delta</i>
-------	---

Description

This is a child function (R-wrapper) used in the main functions of this package.

Usage

```
unoCW(X, D, W, Wstar, RS, RSstar, REX, Chat)
```

Arguments

X	time
D	event indicator (1:event, 0:censor)
W	Weight used in calculating C
Wstar	Weight used in calculating C (perturbed version)
RS	risk score
RSstar	risk score (perturbed version)
REX	Random numbers assigned to each individual for the perturbation-resampling method
Chat	C-statistic estimated with the present dataset

unoU2P

A child function used in Inf.Cval and Inf.Cval.Delta

Description

This is a child function (R-wrapper) used in the main functions of this package.

Usage

```
unoU2P(A, B)
```

Arguments

A	A nxp matrix
B	A nx1 vector

VEC2MAT

Vector to Matrix

Description

Making a (PxM) matrix (x', x', x', \dots, x') from a (Px1) vector, x

Usage

```
VEC2MAT(vc, dm)
```

Arguments

vc	A vector
dm	The number of row of the matrix returned

Index

*Topic **\textasciitildekwd1**

conc, [4](#)
kmcens, [9](#)
unoCW, [9](#)
unoU2P, [10](#)

*Topic **\textasciitildekwd2**

conc, [4](#)
kmcens, [9](#)
unoCW, [9](#)
unoU2P, [10](#)

*Topic **package**

survC1-package, [2](#)

CompCase, [3](#)

conc, [4](#)

cvC, [4](#)

Est.Cval, [5](#)

Est.PH, [6](#)

Inf.Cval, [7](#)

Inf.Cval.Delta, [8](#)

kmcens, [9](#)

survC1 (survC1-package), [2](#)

survC1-package, [2](#)

unoCW, [9](#)

unoU2P, [10](#)

VEC2MAT, [10](#)