Package ‘CatPredi’

May 4, 2016

Type Package
Title Optimal categorisation of continuous variables in prediction models
Version 1.1
Date 2016-04-29
Author Irantzu Barrio
Maintainer Irantzu barrio <irantzu.barrio@ehu.eus>
Depends survival, rms, CPE, rgenoud, mgcv
Description The CatPredi package allows the user to categorise a continuous predictor variable in a logistic or a Cox proportional hazards regression setting.
License GPL

R topics documented:

CatPredi-package ................................................................. 1
catpredi ................................................................. 2
catpredi.survival ............................................................. 4
comp.cutpoints ............................................................... 6
comp.cutpoints.survival ..................................................... 7
controlcatpredi ................................................................. 8
controlcatpredi.survival ..................................................... 9
plot.catpredi ................................................................. 10
plot.catpredi.survival ..................................................... 11
summary.catpredi ........................................................... 12
summary.catpredi.survival ............................................... 13

Index 15

Description

This package allows for the optimal categorisation of continuous predictor variables in a logistic regression model or a Cox proportional hazards regression model. The categorisation can be done either in a univariate or a multivariate setting.
Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

Maintainer: Irantzu barrio <irantzu.barrio@ehu.eus>

References


catpredi  

*Function to obtain optimal cut points to categorise a continuous predictor variable in a logistic regression model*

Description

Returns an object with the optimal cut points to categorise a continuous predictor variable in a logistic regression model

Usage

```r
catpredi(formula, cat.var, cat.points = 1L, data, 
method = c("addfor", "genetic"), range = NULL, 
correct.AUC = TRUE, control = controlcatpredi(), ...)
```

Arguments

- `formula`: An object of class `family` giving the model to be fitted in addition to the continuous covariate is aimed to categorise. This argument allows the user to specify whether the continuous predictor should be categorised in a univariable context, or in presence of other covariates or cofounders, i.e in a multiple logistic regression model. For instance, `Y ~ 1` indicates that the categorisation should be done in a univariable setting, with `Y` being the response variable. If the predictor variable should be categorised in a multivariable setting, this argument allows to specify whether the covariates should be modelled using linear or non linear effects. In the latest, the effects are estimated using the `mgcv` package.

- `cat.var`: Name of the continuous variable to categorise.

- `cat.points`: Number of cut points to look for.

- `data`: Data frame containing all needed variables.

- `method`: The algorithm selected to search for the optimal cut points. "addfor" if the AddFor algorithm is choosen and "genetic" otherwise.

- `range`: The range of the continuous variable in which to look for the cut points. By default NULL, i.e, all the range.

- `correct.AUC`: A logical value. If TRUE the bias corrected AUC is estimated.

- `control`: Output of the controlcatpredi() function.

- `...`: Further arguments for passing on to the function `genoud` of the package `rgenoud`. 
Value

Returns an object of class "catpredi" with the following components:

- call: the matched call.
- method: the algorithm selected in the call.
- formula: an object of class family giving the model to be fitted in addition to the continuous covariate is aimed to categorise.
- cat.var: name of the continuous variable to categorise.
- data: the data frame with the variables used in the call.
- correct.AUC: the control parameters used in the call.
- results: a list with the estimated cut points, AUC and bias corrected AUC.
- control: the control parameters used in the call.

For each of the methods used in the call, a list with the following components is obtained:

- "cutpoints": Estimated optimal cut points.
- "AUC": Estimated AUC.
- "AUC.cor": Estimated bias corrected AUC.

Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

References


See Also

See Also as controlcatpredi, comp.cutpoints, plot.catpredi.

Examples

```r
set.seed(127)
#Simulate data
n = 200
#Predictor variable
xh <- rnorm(n, mean = 0, sd = 1)
xd <- rnorm(n, mean = 1.5, sd = 1)
x <- c(xh, xd)
# Response
y <- c(rep(0,n), rep(1,n))
# Covariate
zh <- rnorm(n, mean=1.5, sd=1)
zd <- rnorm(n, mean=1, sd=1)
z <- c(zh, zd)
# Data frame
df <- data.frame(y = y, x = x, z = z)
```
# Select optimal cut points using the AddFor algorithm
res.addfor <- catpredi(formula = y ~ z, cat.var = "x", cat.points = 3, 
data = df, method = "addfor", range=NULL, correct.AUC=FALSE)

---

**catpredi.survival**  
*Function to obtain optimal cut points to categorise a continuous predictor variable in a Cox proportional hazards regression model*

---

**Description**

Returns an object with the optimal cut points to categorise a continuous predictor variable in a Cox proportional hazards regression model

**Usage**

```r
catpredi.survival(formula, cat.var, cat.points = 1, data, 
method = c("addfor", "genetic"), conc.index = c("cin dex", "cpe"), 
range = NULL, correct.index = TRUE, control = controlcatpredi.survival(), ...)
```

**Arguments**

- `formula`  
  An object of class `family` giving the model to be fitted in addition to the continuous covariate is aimed to categorise. This argument allows the user to specify whether the continuous predictor should be categorised in a univariable context, or in presence of other covariates or cofounders, i.e in a multiple Cox proportional hazards regression model. For instance, `Y ~ 1` indicates that the categorisation should be done in a univariable setting, with `Y` being the response variable.

- `cat.var`  
  Name of the continuous variable to categorise.

- `cat.points`  
  Number of cut points to look for.

- `data`  
  Data frame containing all needed variables.

- `method`  
  The algorithm selected to search for the optimal cut points. "addfor" if the AddFor algorithm is choosen and "genetic" otherwise.

- `conc.index`  
  The concordance probability estimator selected for maximisation purposes. "cindex" if the c-index concordance probability is choosen and "cpe" otherwise. The c-index and CPE are estimated using the `rms` and `cpe` packages, respectively.

- `range`  
  The range of the continuous variable in which to look for the cut points. By default NULL, i.e, all the range.

- `correct.index`  
  A logical value. If TRUE the bias corrected concordance probability is estimated.

- `control`  
  Output of the controlcatpredi.survival() function.

- `...`  
  Further arguments for passing on to the function `genoud` of the package `rgenoud`.
Value

Returns an object of class "catpredi.survival" with the following components:

call         the matched call.
method       the algorithm selected in the call.
formula      an object of class family giving the model to be fitted in addition to the continuous covariate is aimed to categorise.
cat.var      name of the continuous variable to categorise.
data         the data frame with the variables used in the call.
correct.index The logical value used in the call.
results      a list with the estimated cut points, concordance probability and bias corrected concordance probability.
control      the control parameters used in the call.

When the c-index concordance probability is choosen, a list with the following components is obtained for each of the methods used in the call:

"cutpoints"    Estimated optimal cut points.
"Cindex"      Estimated c-index.
"Cindex.cor"  Estimated bias corrected c-index.

When the CPE concordance probability is choosen, a list with the following components is obtained for each of the methods used in the call:

"cutpoints"    Estimated optimal cut points.
"CPE"         Estimated c-index.
"CPE.cor"     Estimated bias corrected c-index.

Author(s)

Irantzuz Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

References


See Also

See Also controlcatpredi.survival, comp.cutpoints.survival, plot.catpredi.survival, catpredi.
**comp.cutpoints**

### Description

Compares two objects of type catpredi.

### Usage

```r
comp.cutpoints(obj1, obj2, V = 100)
```

### Arguments

- `obj1`: catpredi type object for k number of cut points
- `obj2`: catpredi type object for k+1 number of cut points
- `V`: Number of bootstrap resamples. By default V=100

### Value

This function returns an object of class "comp.cutpoints" with the following components:

- `auc.cor.diff`: the difference of the bias corrected AUCs for the two categorical variables.
- `icb.auc.diff`: bootstrap based confidence interval for the bias corrected AUC difference.

### Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

### References

comp.cutpoints.survival

Selection of optimal number of cut points

Description

Compares two objects of class "catpredi.survival"

Usage

comp.cutpoints.survival(obj1, obj2, V = 100)

Arguments

obj1 catpredi.survival type object for k number of cut points
obj2 catpredi type object for k+1 number of cut points
V Number of bootstrap resamples. By default V=100

Value

This function returns an object of class "comp.cutpoints.survival" with the following components:

CI.cor.diff the difference of the bias corrected concordance probability for the two categorical variables.

icb.CI.diff bootstrap based confidence interval for the bias corrected concordance probability difference.

See Also

See Also as catpredi.

Examples

set.seed(127)
# Simulate data
n = 100
# Predictor variable
xh <- rnorm(n, mean = 0, sd = 1)
xd <- rnorm(n, mean = 1.5, sd = 1)
x <- c(xh, xd)
# Response
y <- c(rep(0, n), rep(1, n))
# Data frame
df <- data.frame(y = y, x = x)

# Select 2 optimal cut points using the AddFor algorithm. Correct the AUC
res.addfor.k2 <- catpredi(formula = y ~ 1, cat.var = "x", cat.points = 2,
data = df, method = "addfor", range=NULL, correct.AUC=TRUE)
# Select 3 optimal cut points using the AddFor algorithm. Correct the AUC
res.addfor.k3 <- catpredi(formula = y ~ 1, cat.var = "x", cat.points = 3,
data = df, method = "addfor", range=NULL, correct.AUC=TRUE)
# Select optimal number of cut points
comp <- comp.cutpoints(res.addfor.k2, res.addfor.k3, V = 100)
Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

References


See Also

See Also as `catpredi.survival`.

Examples

```r
set.seed(123)
# Simulate data
n = 500
tauc = 1
x <- rnorm(n=n, mean=0, sd=2)
survT <- exp(2*x + rweibull(n = n, shape=1, scale = 1)) + rnorm(n, mean=0, sd=0.25)
# Censoring time
CensTime <- runif(n=n, min=0, max=tauc)
# Status
SurvS <- as.numeric(survT <= CensTime)
# Data frame
dat <- data.frame(x = x, survT = pmin(survT, CensTime), SurvS = SurvS)

data = dat, method = "addfor", conc.index = "cindex", range = NULL, correct.index = TRUE)
# Select 3 optimal cut points using the AddFor algorithm. Correct the c-index
res.k3 <- catpredi.survival (formula= Surv(survT,SurvS)~1, cat.var="X", cat.points = 3, data = dat, method = "addfor", conc.index = "cindex", range = NULL, correct.index = TRUE)
# Select optimal number of cut points
comp <- comp.cutpoints.survival(res.k2, res.k3, V = 100)
```

controlcatpredi

**Control function**

**Description**

Function used to set several parameters to control the selection of the optimal cut points in a logistic regression model

**Usage**

```r
controlcatpredi(min.p.cat = 1, addfor.g = 100, B = 50, b.method = c("ncoutcome", "outcome"), print.gen = 0)
```
controlcatpredi.survival

Arguments

- `min.p.cat` Set the minimum number of individuals in each category.
- `addfor.g` Grid size for the AddFor algorithm.
- `B` Number of bootstrap replicates for the AUC bias correction procedure.
- `b.method` Allows to specify whether the bootstrap resampling should be done considering or not the outcome variable. The option "ncoutcome" indicates that the data is resampled without taking into account the response variable, while "coutcome" indicates that the data is resampled in regard to the response variable.
- `print.gen` corresponds to the argument print.level of the `genoud` function of the package `rgenoud`.

Value

A list with components for each of the possible arguments.

Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

References


See Also

See Also as `catpredi`.

---

controlcatpredi.survival

*Control function*

Description

Function used to set several parameters to control the selection of the optimal cut points in a Cox proportional hazards regression model.

Usage

```r
controlcatpredi.survival(min.p.cat = 5, addfor.g = 100, B = 50, b.method = c("ncoutcome", "coutcome"), print.gen = 0)
```

Arguments

- `min.p.cat` Set the minimum number of individuals in each category.
- `addfor.g` Grid size for the AddFor algorithm.
- `B` Number of bootstrap replicates for the AUC bias correction procedure.
b.method

Allows to specify whether the bootstrap resampling should be done considering or not the outcome variable. The option "ncoutcome" indicates that the data is resampled without taking into account the response variable, while "coutcome" indicates that the data is resampled in regard to the response variable.

print.gen

Corresponds to the argument print.level of the genoud function of the package rgenoud.

Value

A list with components for each of the possible arguments.

Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

References


See Also

See Also as catpredi.survival.

---

plot.catpredi

*Plot the optimal cut points.*

Description

Plots the relationship between the predictor variable is aimed to categorise and the response variable based on a GAM model. Additionally, the optimal cut points obtained with the catpredi() function are drawn on the graph.

Usage

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

Arguments

- `x`: An object of type catpredi.
- `...`: Additional arguments to be passed on to other functions. Not yet implemented.

Value

This function returns the plot of the relationship between the predictor variable and the outcome.

Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui
plot.catpredi.survival

References


See Also

See Also as catpredi.

Examples

```r
set.seed(127)
# Simulate data
n = 100
# Predictor variable
xh <- rnorm(n, mean = 0, sd = 1)
xd <- rnorm(n, mean = 1.5, sd = 1)
x <- c(xh, xd)
# Response
y <- c(rep(0, n), rep(1, n))
# Data frame
df <- data.frame(y = y, x = x)

# Select optimal cut points using the AddFor algorithm
res.addfor <- catpredi(formula = y ~ 1, cat.var = "x", cat.points = 3,
data = df, method = "addfor", range = NULL, correctAUC = FALSE)
# Plot
plot(res.addfor)
```

---

plot.catpredi.survival

*Plot the optimal cut points.*

Description

Plots the functional form of the predictor variable we want to categorise. Additionally, the optimal cut points obtained with the catpredi.survival() function are drawn on the graph.

Usage

```r
## S3 method for class 'catpredi.survival'
plot(x, ...)
```

Arguments

- `x` An object of type catpredi.survival.
- `...` Additional arguments to be passed on to other functions. Not yet implemented.

Value

This function returns the plot of the relationship between the predictor variable and the outcome.
Author(s)
Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

References

See Also
See Also as catpredi.survival.

Examples
```r
set.seed(123)
# Simulate data
n = 500
tauc = 1
X <- rnorm(n=n, mean=0, sd=2)
SurvT <- exp((2*X + rweibull(n = n, shape=1, scale = 1)) + rnorm(n, mean=0, sd=0.25)
# Censoring time
CensTime <- runif(n=n, min=0, max=tauc)
# Status
SurvS <- as.numeric(Surv(SurvT <= CensTime)
# Data frame
dat <- data.frame(X = X, SurvT = pmin(SurvT, CensTime), SurvS = SurvS)

# Select optimal cut points using the AddFor algorithm
res <- catpredi.survival (formula= Surv(SurvT,SurvS)^1, cat.var="X", cat.points = 2, data = dat, method = "addfor"
# Plot
plot(res)
```

summary.catpredi  Summary method for catpredi objects

Description
Produces a summary of a catpredi object. The following are printed: the call to the catpredi() function; the estimated optimal cut points obtained with the method selected and the estimated AUC and bias corrected AUC (if the argument correct.AUC is TRUE) for the categorised variable.

Usage
```r
## S3 method for class 'catpredi'
summary(object, digits = 4, ...)
```

Arguments
- `object`  an object of class catpredi as produced by catpredi()
- `digits` .
- `...`  further arguments passed to or from other methods.
Value

Returns an object of class "summary.catpredi" with the same components as the catpredi function (see catpredi). plus:

fit.gam  fitted model according to the model specified in the call, based on the function gam of the package mgcv.

Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

References


See Also

See Also as catpredi.

Examples

set.seed(127)
#Simulate data
n <- 200
#Predictor variable
xh <- rnorm(n, mean = 0, sd = 1)
xd <- rnorm(n, mean = 1.5, sd = 1)
x <- c(xh, xd)
#Response
y <- c(rep(0, n), rep(1, n))
#Covariate
zh <- rnorm(n, mean = 1.5, sd = 1)
zd <- rnorm(n, mean = 1, sd = 1)
z <- c(zh, zd)
# Data frame
df <- data.frame(y = y, x = x, z = z)

# Select optimal cut points using the AddFor algorithm
res.addfor <- catpredi(formula = y ~ z, cat.var = "x", cat.points = 3, data = df, method = "addfor", range=NULL, correct.AUC=FALSE)
# Summary
summary(res.addfor)
Usage

```r
## S3 method for class 'catpredi.survival'
summary(object, digits = 4, ...)
```

Arguments

- `object`: an object of class `catpredi.survival` as produced by `catpredi.survival`
- `digits`: numeric
- `...`: further arguments passed to or from other methods.

Value

Returns an object of class "summary.catpredi.survival" with the same components as the `catpredi.survival` function (see `catpredi.survival`).

Author(s)

Irantzu Barrio

References


See Also

See Also as `catpredi.survival`.

Examples

```r
set.seed(123)
#Simulate data
n = 500
tauc = 1
X <- rnorm(n=n, mean=0, sd=2)
SurvT <- exp(2*X + rweibull(n = n, shape=1, scale = 1)) + rnorm(n, mean=0, sd=0.25)
# Censoring time
CensTime <- runif(n=n, min=0, max=tauc)
# Status
SurvS <- as.numeric(SurvT <= CensTime)
# Data frame
dat <- data.frame(X = X, SurvT = pmin(SurvT, CensTime), SurvS = SurvS)

# Select optimal cut points using the AddFor algorithm
res <- catpredi.survival (formula= Surv(SurvT,SurvS)-1, cat.var="X", cat.points = 2, data = dat, method = "addfor")
# Summary
summary(res)
```
Index

«Topic package
  CatPredi-package, 1

CatPredi (CatPredi-package), 1
catpredi, 2, 5, 7, 9, 11, 13
CatPredi-package, 1
catpredi.survival, 4, 8, 10, 12, 14
comp.cutpoints, 3, 6
comp.cutpoints.survival, 5, 7
controlcatpredi, 3, 8
controlcatpredi.survival, 5, 9

family, 2–5

gam, 13
genoud, 2, 4, 9, 10

mgcv, 2

plot.catpredi, 3, 10
plot.catpredi.survival, 5, 11

summary.catpredi, 12
summary.catpredi.survival, 13