

# Package ‘CatPredi’

October 9, 2017

**Type** Package

**Title** Optimal Categorisation of Continuous Variables in Prediction Models

**Version** 1.1

**Date** 2017-10-04

**Imports** survival, rms, CPE, rgenoud, mgcv, stats, graphics

**Description** Allows the user to categorise a continuous predictor variable in a logistic or a Cox proportional hazards regression setting, by maximising the discriminative ability of the model. I Barrio, I Arostegui, MX Rodriguez-Alvarez, JM Quintana (2015) <doi:10.1177/0962280215601873>. I Barrio, MX Rodriguez-Alvarez, L Meira-Machado, C Esteban, I Arostegui (2017) <<https://www.idescat.cat/sort/sort411/41.1.3.barrio-et-al.pdf>>.

**License** GPL

**NeedsCompilation** no

**Author** Irantzu Barrio [aut, cre],  
Maria Xose Rodriguez-Alvarez [aut],  
Inmaculada Arostegui [ctb]

**Maintainer** Irantzu Barrio <irantzu.barrio@ehu.eus>

**Repository** CRAN

**Date/Publication** 2017-10-09 14:16:41 UTC

## R topics documented:

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

summary.catpredi . . . . .	14
summary.catpredi.survival . . . . .	15

<b>Index</b>	<b>17</b>
--------------	-----------

---

CatPredi-package	<i>Categorisation of Continuous Predictor Variables in Regression Models.</i>
------------------	---

---

### Description

Allows the user to categorise a continuous predictor variable in a logistic or a Cox proportional hazards regression setting, by maximising the discriminative ability of the 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

I Barrio, I Arostegui, M.X Rodriguez-Alvarez and J.M Quintana (2015). A new approach to categorising continuous variables in prediction models: proposal and validation. *Statistical Methods in Medical Research*, in press.

I Barrio, M.X Rodriguez-Alvarez, L Meira-Machado, C Esteban and I Arostegui (2017). Comparison of two discrimination indexes in the categorisation of continuous predictors in time-to-event studies. *SORT*, 41:73-92

---

catpredi	<i>Function to obtain optimal cut points to categorise a continuous predictor variable in a logistic regression model</i>
----------	---

---

### Description

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

### Usage

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

**Arguments**

formula	An object of class <code>formula</code> 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, $Y \sim 1$ indicates that the categorisation should be done in a univariable setting, with Y being the response variable. If the predictor variable is aimed to 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 <code>mgcv</code> 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 <code>genoud</code> of the package <code>rgenoud</code> .

**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 <code>formula</code> 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 logical value 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

I Barrio, I Arostegui, M.X Rodriguez-Alvarez and J.M Quintana (2015). A new approach to categorising continuous variables in prediction models: proposal and validation. *Statistical Methods in Medical Research* (in press).

S.N Wood (2006). *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC.

## See Also

See Also as [controlcatpredi](#), [comp.cutpoints](#), [plot.catpredi](#), [summary.catpredi](#).

## Examples

```
library(CatPredi)

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	<i>Function to obtain optimal cut points to categorise a continuous predictor variable in a Cox proportional hazards regression model</i>
-------------------	---

---

## Description

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

## Usage

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

**Arguments**

formula	An object of class <code>formula</code> giving the model to be fitted in addition to the continuous covariate is aimed to categorise. The response must be a survival object as returned by the <code>Surv</code> function. 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, <code>Surv(SurvT,SurvS)~1</code> indicates that the categorisation should be done in a univariable setting.
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 chosen and "genetic" otherwise.
conc.index	The concordance probability estimator selected for maximisation purposes. "cindex" if the c-index concordance probability is chosen 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 <code>controlcatpredi.survival()</code> function.
...	Further arguments for passing on to the function <code>genoud</code> of the package <code>rgenoud</code> .

**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 <code>formula</code> 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 chosen, 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 chosen, a list with the following components is obtained for each of the methods used in the call:

"cutpoints"	Estimated optimal cut points.
"CPE"	Estimated CPE.
"CPE.cor"	Estimated bias corrected CPE.

### Author(s)

Irantzu Barrio and Maria Xose Rodriguez-Alvarez

### References

I Barrio, M.X Rodriguez-Alvarez, L Meira-Machado, C Esteban and I Arostegui (2017). Comparison of two discrimination indexes in the categorisation of continuous predictors in time-to-event studies. *SORT*, 41:73-92

M Gonen and G Heller (2005). Concordance probability and discriminatory power in proportional hazards regression. *Biometrika*, 92:965-970.

F Harrell (2001). Regression modeling strategies: with applications to linear models, logistic and ordinal regression, and survival analysis. Springer.

### See Also

See Also [controlcatpredi.survival](#), [comp.cutpoints.survival](#), [plot.catpredi.survival](#), [catpredi](#).

### Examples

```
library(CatPredi)
library(survival)
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", conc.index = "cindex", range = NULL,
  correct.index = FALSE)
```

---

comp.cutpoints	<i>Selection of optimal number of cut points</i>
----------------	--

---

**Description**

Compares two objects of class "catpredi".

**Usage**

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

**Arguments**

obj1	an object inheriting from class "catpredi" for k number of cut points
obj2	an object inheriting from class "catpredi" 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**

I Barrio, I Arostegui, M.X Rodriguez-Alvarez and J.M Quintana (2015). A new approach to categorising continuous variables in prediction models: proposal and validation. *Statistical Methods in Medical Research* (in press).

**See Also**

See Also as [catpredi](#).

**Examples**

```
library(CatPredi)
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,
control=controlcatpredi(addfor.g=100))
# 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,
control=controlcatpredi(addfor.g=100))
# Select optimal number of cut points
comp <- comp.cutpoints(res.addfor.k2, res.addfor.k3, V = 100)

```

---

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	an object inheriting from class "catpredi.survival" for k number of cut points
obj2	an object inheriting from class "catpredi.survival" 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.

## Author(s)

Irantzu Barrio and Maria Xose Rodriguez-Alvarez



## References

I Barrio, M.X Rodriguez-Alvarez, L Meira-Machado, C Esteban and I Arostegui (2017). Comparison of two discrimination indexes in the categorisation of continuous predictors in time-to-event studies. *SORT*, 41:73-92

## See Also

See Also as [catpredi.survival](#).

## Examples

```
library(CatPredi)
library(survival)
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 2 optimal cut points using the AddFor algorithm. Correct the c-index
res.k2 <- catpredi.survival (formula= Surv(SurvT, SurvS)~1, cat.var="X", cat.points = 2,
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**

```
controlcatpredi(min.p.cat = 1, 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 <a href="#">genoud</a> 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**

Mebane Jr, W. R., & Sekhon, J. S. (2011). Genetic optimization using derivatives: the rgenoud package for R. *Journal of Statistical Software* 42**11**, 1-26.

**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**

```
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 <a href="#">genoud</a> function of the package rgenoud.

**Value**

A list with components for each of the possible arguments.

**Author(s)**

Irantzu Barrio and Maria Xose Rodriguez-Alvarez

**References**

Mebane Jr, W. R., & Sekhon, J. S. (2011). Genetic optimization using derivatives: the rgenoud package for R. *Journal of Statistical Software* 42**11**, 1-26.

**See Also**

See Also as [catpredi.survival](#).

---

plot.catpredi	<i>Plot the optimal cut points.</i>
---------------	-------------------------------------

---

**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**

```
## 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

**References**

I Barrio, I Arostegui, M.X Rodriguez-Alvarez and J.M Quintana (2015). A new approach to categorising continuous variables in prediction models: proposal and validation. *Statistical Methods in Medical Research* (in press).

**See Also**

See Also as [catpredi](#).

**Examples**

```
library(CatPredi)
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, correct.AUC = 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**

```
## 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 and Maria Xose Rodriguez-Alvarez

**References**

I Barrio, M.X Rodriguez-Alvarez, L Meira-Machado, C Esteban and I Arostegui (2017). Comparison of two discrimination indexes in the categorisation of continuous predictors in time-to-event studies. *SORT*, 41:73-92

**See Also**

See Also as [catpredi.survival](#).

**Examples**

```
library(CatPredi)
library(survival)
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", conc.index = "cindex", range = NULL,
correct.index = FALSE)
# 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

```
## 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 <a href="#">gam</a> of the package mgcv.
---------	---

### Author(s)

Irantzu Barrio, Maria Xose Rodriguez-Alvarez and Inmaculada Arostegui

### References

I Barrio, I Arostegui, M.X Rodriguez-Alvarez and J.M Quintana (2015). A new approach to categorising continuous variables in prediction models: proposal and validation. *Statistical Methods in Medical Research* (in press).

### See Also

See Also as [catpredi](#).

**Examples**

```

library(CatPredi)
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 = 2,
data = df, method = "addfor", range=NULL, correct.AUC=FALSE)
# Summary
summary(res.addfor)

```

---

```
summary.catpredi.survival
```

*Summary method for objects of class "catpredi.survival"*

---

**Description**

Produces a summary of a "catpredi.survival" object. The following are printed: the call to the catpredi.survival() function; the estimated optimal cut points obtained with the method and concordance probability estimator selected and the estimated and bias corrected concordance probability for the categorised variable (whenever the argument correct.index is set to TRUE) .

**Usage**

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

**Arguments**

object	an object of class "catpredi.survival" as produced by catpredi.survival()
digits	.
...	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 and Maria Xose Rodriguez-Alvarez

**References**

I Barrio, M.X Rodriguez-Alvarez, L Meira-Machado, C Esteban and I Arostegui (2017). Comparison of two discrimination indexes in the categorisation of continuous predictors in time-to-event studies. *SORT*, 41:73-92

**See Also**

See Also as [catpredi.survival](#).

**Examples**

```
library(CatPredi)
library(survival)
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", conc.index = "cindex", range = NULL,
  correct.index = FALSE)
# Summary
summary(res)
```



# Index

## \*Topic **package**

CatPredi-package, 2

CatPredi (CatPredi-package), 2

catpredi, 2, 6, 7, 10, 12, 14

CatPredi-package, 2

catpredi.survival, 4, 9, 11, 13, 16

comp.cutpoints, 4, 7

comp.cutpoints.survival, 6, 8

controlcatpredi, 4, 9

controlcatpredi.survival, 6, 10

formula, 3, 5

gam, 14

genoud, 3, 5, 10, 11

mgcv, 3

plot.catpredi, 4, 11

plot.catpredi.survival, 6, 12

summary.catpredi, 4, 14

summary.catpredi.survival, 15

Surv, 5