Package ‘Cprob’

September 26, 2017

Title The Conditional Probability Function of a Competing Event
Version 1.4
Author Arthur Allignol
Description Permits to estimate the conditional probability function of a competing event, and to fit, using the temporal process regression or the pseudo-value approach, a proportional-odds model to the conditional probability function (or other models by specifying another link function). See <doi:10.1111/j.1467-9876.2010.00729.x>.
Maintainer Arthur Allignol <arthur.allignol@gmail.com>
Depends prodlim
Imports tpr, geepack, lgtdl, graphics, stats, lattice
License GPL (>= 2)
NeedsCompilation no
Repository CRAN
Date/Publication 2017-09-25 22:01:04 UTC

R topics documented:

<table>
<thead>
<tr>
<th>R topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cprob-package</td>
<td>2</td>
</tr>
<tr>
<td>cpf</td>
<td>3</td>
</tr>
<tr>
<td>cpfpo</td>
<td>5</td>
</tr>
<tr>
<td>lines.cpf</td>
<td>7</td>
</tr>
<tr>
<td>mgs</td>
<td>8</td>
</tr>
<tr>
<td>plot.cpf</td>
<td>9</td>
</tr>
<tr>
<td>predict.cpf</td>
<td>10</td>
</tr>
<tr>
<td>print.cpf</td>
<td>11</td>
</tr>
<tr>
<td>print.cpfpo</td>
<td>12</td>
</tr>
<tr>
<td>pseudocpf</td>
<td>12</td>
</tr>
<tr>
<td>summary.cpf</td>
<td>14</td>
</tr>
<tr>
<td>summary.pseudocpf</td>
<td>15</td>
</tr>
<tr>
<td>xyplot.cpfpo</td>
<td>16</td>
</tr>
</tbody>
</table>

Index 17
Conditional probability function of a competing event

Description

Estimates the conditional probability function of a competing event, and fits, using the temporal process regression or the pseudo-value approach, a proportional-odds model to the conditional probability function.

Details

```
Package: Cprob
Version: 1.0
Depends: prodlim, tpr, lattice, geepack
License: GPL (>=2)
```

Index:

- `cpf` Conditional Probability Function of a Competing Event
- `cpfpo` Proportional-odds Model for the Conditional Probability Function
- `lines.cpf` Lines method for 'cpf' objects
- `mgus` Monoclonal Gammapathy of Undetermined Significance
- `plot.cpf` Plot method for cpf objects
- `predict.cpf` Conditional Probability Estimates at chosen timepoints
- `print.cpf` Print a cpf object
- `print.cpfpo` Print Method for cpfpo objects
- `pseudocpf` Pseudo values for the conditional probability function
- `summary.cpf` Summary method for cpf
- `summary.pseudocpf` Summary method for pseudocpf objects
- `xyplot.cpfpo` 'xyplot' method for object of class 'cpfpo'

The `cpf` function computes the conditional probability function of a competing event and can test equality of (only) two conditional probability curves.

A proportional-odds model for the conditional probability function can be fitted using either `cpfpo` or `pseudocpf`. The former function uses the temporal process regression methodology while the latter uses the pseudo value technique.

Author(s)

Arthur Allignol
Maintainer: Arthur Allignol <arthur.allignol@gmail.com>

References


See Also
tpr

cpf

*Conditional Probability Function of a Competing Event*

Description

This function computes estimates of the conditional probability function of a competing event and its variance. It also tests equality of conditional probability functions in two samples.

Usage

cpf(formula, data, subset, na.action, conf.int = 0.95, failcode)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>formula</td>
<td>A formula object that has a <code>Hist</code> object on the left of a <code>~</code> operator, and if desired, terms separated by <code>+</code> on the right. Note that any subsetting, i.e., <code>data$var</code> or <code>data[, &quot;var&quot;]</code>, is invalid for this function.</td>
</tr>
<tr>
<td>data</td>
<td>A data frame in which the variables in the formula can be interpreted.</td>
</tr>
<tr>
<td>subset</td>
<td>Expression identifying a subset of the data to be used for conditional probability estimation.</td>
</tr>
<tr>
<td>na.action</td>
<td>A missing-data filter function, applied to the model frame, after any <code>subset</code> argument has been used. Default option is <code>options()$na.action</code>.</td>
</tr>
<tr>
<td>conf.int</td>
<td>Level for pointwise two-sided confidence intervals. Default is 0.95.</td>
</tr>
<tr>
<td>failcode</td>
<td>Failure code of the event of interest. Default is the smallest event type provided in the data.</td>
</tr>
</tbody>
</table>
Details

The conditional probability function is defined as the probability of having failed due to one competing event (the event of interest), given that no other event has previously occurred (Pepe, 1993). The cpf function aims at estimating this quantity along with its variance at each event times. It also computes a test of equality of conditional probability curves in two samples (and only in two samples).

Of note, if there is more than 2 competing events, the failure types that are not of interest are aggregated into one competing event.

Value

cpf returns an object of class cpf with components

- **cp**: Estimates of the conditional probability function given at all event times
- **var**: Variance estimates
- **time**: Event times
- **lower**: Lower confidence limit for the conditional probability curve
- **upper**: Upper confidence limit for the conditional probability curve
- **n.risk**: Number of individuals at risk just before $t$
- **n.event**: A matrix giving the number of events of interest at time $t$ in the first column, and the number of competing events at time $t$ in the second column
- **n.lost**: Number of censored observations at time $t$
- **size.strata**: Displays the size of each strata
- **X**: Gives covariate’s name and labels
- **strata**: Gives the covariate labels that will be used by default for plotting the conditional probability curves, for example.
- **call**: Call that produced the object
- **z**: Test statistic
- **p**: p value of the test
- **failcode**: Same as in function call

Author(s)

Arthur Allignol, <arthur.allignol@gmail.com>

References


See Also

Hist.print.cpf, summary.cpf

Examples

data(mgus)

CP <- cpf(Hist(time, ev), data = mgus)
CP

## With age dichotomised according to its median
mgus$AGE <- ifelse(mgus$age < 64, 0, 1)
CP <- cpf(Hist(time, ev) ~ AGE, data = mgus)
CP
summary(CP)

## Conditional probability of the competing event
CP.death <- cpf(Hist(time, ev), data = mgus, failcode = 2)
CP.death

---

**cpfpo**

Proportional-odds Model for the Conditional Probability Function

Description

This function uses the Temporal Process Regression framework to fit a proportional-odds model to
the conditional probability function.

Usage

cpfpo(formula, data, subset, na.action, failcode, tis, w, ...)

Arguments

- **formula**: A formula object whose response, on the left of a ~ operator, is a Hist object, and the terms on the right of ~
- **data**: A data.frame in which to interpret the variable names in the formula and subset
- **subset**: Expression specifying that only a subset of the data set should be used
- **na.action**: A missing data filter function applied to the model.frame, after any subset argument has been used. Default is options()$na.action
- **failcode**: Integer specifying the code for the event of interest
- **tis**: Vector of timepoints on which the model is fitted
- **w**: Vector of weights. Should be of the same length as tis. Default is rep(1, length(tis))
- **...**: Further arguments for tpr
Details

The conditional probability function of a competing event is the probability of having failed due to one risk (the event of interest) given that no other failure has previously occurred.

The cpfpo function fits a proportional-odds model for the conditional probability function within the Temporal Process Regression framework, which is a marginal mean model, where the mean of a response \( Y(t) \) at time \( t \) is specified conditionally on a vector of covariates \( Z \) and a time-dependent stratification factor \( S(t) \)

\[
E\{Y(t)|Z, S(t) = 1\} = g^{-1}\{\beta(t)'Z\}
\]

This approach enables the application of standard binary regression models in continuous time.

The regression model is fitted using the \texttt{tpr} package. See \texttt{tpr} for further details.

Value

cpfpo returns an object of class cpfpo and tpr. See tpr for further details.

Note

As the returned value is also a tpr object, all the methods defined in the \texttt{tpr} package are available.

Author(s)

Arthur Allignol, <arthur.allignol@gmail.com>

References


See Also

tpr, print.cpfpo, xyplot.cpfpo

Examples

data(mgus)
mgus$A <- ifelse(mgus$age < 64, 0, 1)

## fit the model for 2 covariates
fit.cpfpo <- cpfpo(Hist(time, ev)=factor(A) + creat,
                   data = mgus, tis=seq(10, 30, 0.3),
                   w=rep(1,67))

## and plot the odds-ratios
if(require("lattice")) {

lines.cpf

Lines method for `cpf` objects

Description

Lines method for cpf objects

Usage

```r
## S3 method for class 'cpf'
lines(x, conf.int = FALSE, mark.time = FALSE, mark = 3,
col = 1, lty, ci.lty = 3, ...)
```

Arguments

- `x` An object of class cpf
- `conf.int` Logical. Whether to add a pointwise confidence interval
- `mark.time` Controls the labelling of the curves. If set to TRUE, then the curves are marked at each occurrence of a competing event
- `mark` Mark parameter which will be used to label the curves. Same as pch
- `col` A vector of colours
- `lty` A vector specifying the line types for the curves
- `ci.lty` A vector specifying the line type for the confidence intervals
- `...` Other arguments

Value

No value returned

Author(s)

Arthur Allignol, <arthur.allignol@gmail.com>

See Also

`plot.cpf`
Description

All 241 patients diagnosed with monoclonal gammopathy of undetermined significance at the Mayo Clinic before Jan. 1, 1971, and followed forward until 1992. The interest lies in the possible transformation of mgus towards a cancer of the plasma cells, death from other causes acting as a competing risk.

Usage

data(mgus)

Format

A data frame with 241 observations and the following 10 variables.

- id  Patient identification number
- time  Event time
- ev  Event type. 0=censored, 1=cancer of the plasma cells, 2=other
- age  Age at diagnostic of mgus
- sex  Sex. 1=male, 2=female
- y.diag  Calendar year of diagnosis
- albu  Albumine level at mgus diagnosis
- creat  Serum creatinine level at mgus diagnosis
- hb  Hemoglobine level at mgus diagnosis
- size  Size of the monoclonal protein peak at mgus diagnosis

Source


Examples

data(mgus)
plot.cpf  

Plot method for cpf objects

Description

A plot of conditional probability curves is produced, one for each strata. Pointwise confidence intervals and legend can also be displayed.

Usage

```r
## S3 method for class 'cpf'
plot(x, conf.int = FALSE, mark.time = FALSE, mark = 3,
     col = 1, lty, ci.lty = 3, xlim, ylim = c(0, 1),
     xlab = "", ylab = "", bty = "n", legend = TRUE,
     curvlab = NULL, legend.pos = NULL,
     legend.bty = "n", ...)
```

Arguments

- `x` An object of class `cpf`
- `conf.int` A logical indicating whether to plot a pointwise confidence interval. Default is `FALSE`
- `mark.time` Controls the labelling of the curves. If set to `TRUE`, then the curves are marked at each occurrence of a competing event
- `mark` Mark parameter which will be used to label the curve. The same as `pch`
- `col` Vector of colours for the curves
- `lty` Vector of integers specifying the line types
- `ci.lty` A vector of integer controlling the line types for the pointwise confidence intervals
- `xlim` x-axis limits for the plot area. Default is `c(0, 1)`
- `ylim` y-axis limits for the plot area
- `xlab` x-axis label
- `ylab` y-axis label
- `bty` see `par`
- `legend` Whether or nor draw a legend. Default is `TRUE`
- `curvlab` Text for legend
- `legend.pos` Position for the legend. Default is the upper left corner
- `legend.bty` Box type. See `legend`
- `...` Further arguments for plot

Value

No value returned
Author(s)

Arthur Allignol, <arthur.allignol@gmail.com>

See Also

par.cpf

Examples

data(mgus)
mgus$A <- ifelse(mgus$age < 64, 0, 1)

fit <- cpf(Hist(time, ev)-A, mgus)

plot(fit, curvlab = c("Age < 64", "Age >= 64"),
     main = "Conditional Probability of Cancer", xlab = "Years")

predict.cpf  Conditional Probability Estimates at chosen timepoints

Description

This function displays estimates of the conditional probability function at timepoints chosen by the user.

Usage

## S3 method for class 'cpf'
predict(object, timepoints, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>An object of class cpf</td>
</tr>
<tr>
<td>timepoints</td>
<td>Vector of timepoints</td>
</tr>
<tr>
<td>...</td>
<td>Not used</td>
</tr>
</tbody>
</table>

Value

A data.frame with the following columns

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>time</td>
<td>The given timepoints</td>
</tr>
<tr>
<td>cp</td>
<td>Conditional probability estimates</td>
</tr>
<tr>
<td>var</td>
<td>Variance estimates</td>
</tr>
<tr>
<td>lower</td>
<td>Lower confidence limit</td>
</tr>
<tr>
<td>upper</td>
<td>Upper confidence limit</td>
</tr>
<tr>
<td>n.risk</td>
<td>Number of individuals at risk just before the timepoints</td>
</tr>
<tr>
<td>group</td>
<td>Group, if any</td>
</tr>
</tbody>
</table>
Print a cpf object

Description
Print method for a cpf object

Usage
```r
## S3 method for class 'cpf'
print(x, ...)  
```

Arguments
- `x` : A cpf object
- `...` : Further arguments to the print method. Not used here.

Value
No value returned.

Author(s)
Arthur Allignol, <arthur.allignol@gmail.com>

See Also
cpf
**pseudocpf**

---

### print.cpfpo

**Print Method for cpfpo objects**

**Description**

A print method for an object of class `cpfpo`. It displays the results of test for non-significant effects, along with the results of the test for time-independent effects.

**Usage**

```r
## S3 method for class 'cpfpo'
print(x, ...)  
```

**Arguments**

- `x`:
  - An object of class `cpfpo`
- `...`:
  - Not used

**Value**

An invisible list with 2 components:

- `sig`:
  - Results of the test on non-significant effects
- `tdep`:
  - Results of the test on the time-dependence of the effects

See [tpr.test](#).

**Author(s)**

Arthur Allignol, <arthur.allignol@gmail.com>

**See Also**

- [tpr.test](#), [cpfpo](#)

### pseudocpf

**Pseudo values for the conditional probability function**

**Description**

The function computes pseudo values and then fit a proportional-odds model to the conditional probability function using GEE.

**Usage**

```r
pseudocpf(formula, data, id, subset, na.action, timepoints, failcode = 1, ...)
```
Arguments

- **formula**: A formula object, whose terms are on the right of a `~` operator and the response, a `Hist` object, on the left.
- **data**: A data frame in which to interpret the formula.
- **id**: Individual patient id.
- **subset**: Expression specifying that only a subset of the data set should be used.
- **na.action**: A missing data filter function applied to the model.frame, after any subset argument has been used. Default is `options()$na.action`.
- **timepoints**: Time points at which to compute the pseudo values.
- **failcode**: Integer that specifies which event is of interest.
- **...**: Other arguments for the `geese` function.

Details

The regression model is fitted using a method based on the pseudo-values from a jackknife statistic constructed from the conditional probability curve. Then a GEE model is used on the pseudovalues to obtain the odds-ratios.

Value

Returns an object of class `pseudocpf` containing the following components:

- **fit**: A `geese` object.
- **pseudo**: The pseudo values computed at the specified time points.
- **timepoints**: Same as in the function call.
- **call**: The matched call.

Note

Besides the estimated regression coefficients, the function returns the computed pseudo-values, so that one can fit a different model, e.g., with a different link function.

Author(s)

Arthur Allignol, <arthur.allignol@gmail.com>

References


See Also

`geese`, `summary.pseudocpf`
Examples

data(mgus)

cutoffs <- quantile(mgus$time, probs = seq(0, 1, 0.05))[-1]

### with fancy variance estimation
fit1 <- pseudocpf(Hist(time, ev) ~ age + creat, mgus, id = id,
    timepoints = cutoffs, corstr = "independence",
    scale.value = TRUE)
summary(fit1)

### with jackknife variance estimation
fit2 <- pseudocpf(Hist(time, ev) ~ age + creat, mgus, id = id,
    timepoints = cutoffs, corstr = "independence",
    scale.value = TRUE, jack = TRUE)
summary(fit2)

summary.cpf

Summary method for cpf

Description

Provides a summary of a cpf object.

Usage

## S3 method for class 'cpf'
summary(object, ...)
## S3 method for class 'summary.cpf'
print(x, ...)

Arguments

object An object of class cpf
x An object of class summary.cpf
... Not used

Value

A list with the following components

est A list of data.frame according to the covariate number of levels. Each data.frame contains conditional probability estimates, variance estimates, number of individual at risk at each event times, number of events for the event of interest, number of competing events, and the event times.
call Call that produced the cpf object
X Covariate's name and levels
z, p Test statistic and p value
Summary method for pseudocpf objects

**Description**

Summary method for pseudocpf objects

**Usage**

```r
## S3 method for class 'pseudocpf'
summary(object, conf.int = 0.95, scale = 1, ...)
```

**Arguments**

- `object` : An object of class `pseudocpf`
- `conf.int` : Level for the two-sided confidence intervals
- `scale` : Vector of scale factors for the coefficients, defaults to 1. The confidence limits are for the risk change associated with one scale unit
- `...` : Other arguments

**Value**

Returns an object of class `summary.pseudocpf` that includes the following components:

- `call` : The matched call
- `coefficients` : A matrix with 5 columns including the regression coefficients, odds-ratios, standard-errors, wald statistics and corresponding two-sided p-values
- `conf.int` : A matrix with 4 columns that consists of the odds-ratios, exp(-coef) and the lower and upper bounds of the confidence interval

**Author(s)**

Arthur Allignol, <arthur.allignol@gmail.com>

**See Also**

`pseudocpf`
Description

This function plots the results of the proportional-odds model fitted to the conditional probability function using `cpfpo`.

Usage

```r
## S3 method for class 'cpfpo'
xyplot(x, data = NULL, conf.int = TRUE, level = 0.95,
       odds = TRUE, intercept = TRUE, ylab, xlab, lty = c(1, 3, 3),
       col = c(1, 1, 1), ...)```

Arguments

- `x`: An object of class `cpfpo`
- `data`: `Useless. Can be kept to NULL`
- `conf.int`: Logical. Whether to plot pointwise confidence intervals. Default is `TRUE`
- `level`: Level of the pointwise confidence interval. Default is `0.95`
- `odds`: If set to `TRUE`, the odds-ratios are displayed. Otherwise their logs are plotted. Default is `TRUE`
- `intercept`: Logical. Controls whether the intercept should also be displayed. Default is `TRUE`
- `ylab`: Label for the y-axis
- `xlab`: Label for the x-axis
- `lty`: Vector of line types. Default is 1 for the odds-ratio and 3 for the confidence interval.
- `col`: A vector of colours. Default is black
- `...`: Further arguments

Value

The function returns a `trellis` object. See `xyplot` for further details.

Author(s)

Arthur Allignol, <arthur.allignol@gmail.com>

See Also

`cpfpo, xyplot`
Index

*Topic **datasets**
  mgus, 8

*Topic **hplot**
  lines.cpf, 7
  plot.cpf, 9
  xyplot.cpfpo, 16

*Topic **package**
  Cprob-package, 2

*Topic **print**
  print.cpf, 11
  print.cpfpo, 12
  summary.cpf, 14
  summary.pseudocpf, 15

*Topic **regression**
  cpfpo, 5
  pseudocpf, 12

*Topic **survival**
  cpf, 3
  cpfpo, 5
  Cprob-package, 2
  lines.cpf, 7
  mgus, 8
  predict.cpf, 10
  pseudocpf, 12
  summary.cpf, 14
  summary.pseudocpf, 15
  xyplot.cpfpo, 16

[.cpf (cpf), 3

cpf, 3, 10, 11, 15
cpfpo, 5, 12, 16
Cprob (Cprob-package), 2
Cprob-package, 2

glese, 13

Hist, 5

legend, 9

lines.cpf, 7

mgus, 8

par, 9, 10
plot.cpf, 7, 9
predict.cpf, 10
print.cpf, 5, 11
print.cpfpo, 6, 12
print.summary.cpf (summary.cpf), 14
print.summary.pseudocpf
  (summary.pseudocpf), 15
pseudocpf, 12, 15

summary.cpf, 5, 14
summary.pseudocpf, 13, 15

tpr, 3, 5, 6

tpr.test, 12

xyplot, 16

xyplot.cpfpo, 6, 16