Package ‘gamlss.cens’

October 13, 2022

Type Package
Title Fitting an Interval Response Variable Using ‘gamlss.family’ Distributions
Version 5.0-1
Date 2018-06-08
Depends R (>= 2.2.1), gamlss.dist, gamlss, survival, methods
Author Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk>, Bob Rigby, Nicoleta Mortan
Maintainer Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk>
Description This is an add-on package to GAMLSS. The purpose of this package is to allow users to fit interval response variables in GAMLSS models. The main function gen.cens() generates a censored version of an existing GAMLSS family distribution.
License GPL-2 | GPL-3
URL http://www.gamlss.org/
NeedsCompilation no
Repository CRAN
Date/Publication 2018-06-11 07:57:56 UTC

R topics documented:

  gamlss.cens-package .........................................................  2
cens .................................................................  4
cens.d ...............................................................  6
cens.p ...............................................................  8
cens.q ...............................................................  9
gen.cens ............................................................ 10
lip ................................................................. 12

Index 14
gamlss.cens-package  

Fitting an Interval Response Variable Using ‘gamlss.family’ Distributions

Description

This is an add-on package to GAMLSS. The purpose of this package is to allow users to fit interval response variables in GAMLSS models. The main function gen.cens() generates a censored version of an existing GAMLSS family distribution.

Details

The DESCRIPTION file:

Package:  gamlss.cens  
Type:  Package  
Title:  Fitting an Interval Response Variable Using ‘gamlss.family’ Distributions  
Version:  5.0-1  
Date:  2018-06-08  
Depends:  R (>= 2.2.1), gamlss.dist, gamlss, survival, methods  
Author:  Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk>, Bob Rigby, Nicoleta Mortan  
Maintainer:  Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk>  
Description:  This is an add-on package to GAMLSS. The purpose of this package is to allow users to fit interval response variables in GAMLSS models. The main function gen.cens() generates a censored version of an existing GAMLSS family distribution.  
License:  GPL-2 | GPL-3  
URL:  http://www.gamlss.org/

Index of help topics:

cens  Function to Fit Censored Data Using a gamlss.family Distribution  
cens.d  Censored Probability Density Function of a gamlss.family Distribution  
cens.p  Censored Cumulative Probability Density Function of a gamlss.family Distribution  
cens.q  Censored Inverse Cumulative Probability Density Function of a gamlss.family Distribution  
gamlss.cens-package  Fitting an Interval Response Variable Using ‘gamlss.family’ Distributions  
gen.cens  A Function to Generate Appropriate Functions to Be Used to Fit a Censored Response variable in GAMLSS  
lip  Data for lip

Author(s)

Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk>, Bob Rigby, Nicoleta Mortan
Maintainer: Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk>

References


(see also [http://www.gamlss.org/](http://www.gamlss.org/)).

See Also

gamlss, gamlss.family

Examples

```r
library(survival)
library(gamlss)
library(gamlss.dist)

# comparing results with package survival
# fitting the exponential distribution
ms1 <- survreg(Surv(futime, fustat) ~ ecog.ps + rx, ovarian,
               dist='exponential')
mg1 <- gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian,
              family=cens(EXP), c.crit=0.00001)
if(abs(-2*ms1$loglik[2]-deviance(mg1))>0.001) stop(paste("descrepancies in exp"))
if(sum(coef(ms1)-coef(mg1))>0.001) warning(paste("descrepancies in coef in exp"))
summary(ms1)
summary(mg1)

# fitting the Weibull distribution
ms2 <- survreg(Surv(futime, fustat) ~ ecog.ps + rx, ovarian, dist='weibull')
mg2 <- gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian,
              family=cens(WEI, delta=c(0.001,0.001)), c.crit=0.00001)
if(abs(-2*ms2$loglik[2]-deviance(mg2))>0.005) stop(paste("descrepancies in deviance in WEI"))
summary(ms2);summary(mg2)

# compare the scale parameter
1/exp(coef(mg2,"sigma"))

# now fit the Weibull in different parameterizations
mg21 <- gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian,
                family=cens(WEI2), method=mixed(2,30))
mg21 <- gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian,
                family=cens(WEI3))
```
cens

Function to Fit Censored Data Using a gamlss.family Distribution

Description

This function can be used to fit censored or interval response variables. It takes as an argument an existing `gamlss.family` distribution and generates a new `gamlss.family` object which then can be used to fit right, left or interval censored data.

Usage

cens(family = "NO", type = c("right", "left", "interval"), name = "cens",
local = TRUE, delta = NULL, ...)

Arguments

- **family**: a `gamlss.family` object, which is used to define the distribution and the link functions of the various parameters. The distribution families supported by gamlss() can be found in `gamlss.family` and in the package `gamlss.dist`.
- **name**: the characters you want to add to the name of new functions, by default is `cens`
- **type**: what type of censoring is required, right, left or interval.
- **local**: if TRUE the function will try to find the environment of gamlss to generate the d and p functions required for the fitting, if FALSE the functions will be generated in the global environment.
- **delta**: the delta increment used in the numerical derivatives
- ... for extra arguments

Details

This function is created to help users to fit censored data using an existing `gamlss.family` distribution. It does this by taking an existing `gamlss.family` and changing some of the components of the distribution to help the fitting process. It particular it (i) creates a (d) function (for calculating the censored likelihood) and a (p) function (for generating the quantile residuals) within gamlss, (ii) changes the global deviance function `G.dev.incr`, the first derivative functions (see note below) and other quantities from the original distribution.

Value

It returns a `gamlss.family` object which has all the components needed for fitting a distribution in gamlss.
Note

This function is experimental and could be changed in the future. The function cens changes the first derivatives of the original gamlss family d function to numerical derivatives for the new censored d function. The default increment delta, for this numerical derivatives function, is \( \text{eps} \times \text{pmax}(\text{abs}(x), 1) \) where \( \text{eps} \leftarrow \sqrt{\text{.Machine}\$\text{double}.\text{eps}} \). The default delta could be inappropriate for specific applications and can be overwritten by using the argument delta.

Note that in order to get the correct standard errors you have to generate the "d" function by using gen.cens().

Author(s)

Mikis Stasinopoulos \(<d.stasinopoulos@londonmet.ac.uk>\) and Bob Rigby \(<r.rigby@londonmet.ac.uk>\)

References


See Also

cens.d, cens.p, gen.cens

Examples

# comparing output with the survreg() of package survival
library(gamlss.dist)
library(survival)
#---------------------------------------------------------------
# right censoring example
# example from survreg()  
# fitting the exponential distribution
mexp<-survreg(Surv(futime, fustat) ~ ecog.ps + rx, ovarian, dist='exponential')
gexp<-gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian,
      family=expo(EXP), c.crit=0.00001)
if(abs(-2*mexp$loglik[2]-deviance(gexp))>0.001)
  stop(paste("discrepancies in exponential models"))
if(sum(coef(mexp)-coef(gexp))>0.001)
  warning(paste("discrepancies in coef in exponential models"))
summary(mexp)
gen.cens(EXP)
summary(gexp)
# fitting different distributions 
# weibull
mwei <- survreg(Surv(futime, fustat) ~ ecog.ps + rx, ovarian, dist='weibull')
gwei <- gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian,  
family=cens(WEI, delta=c(0.0001,0.0001)), c.crit=0.00001)
if(abs(-2*mwei$loglik[2]-deviance(gwei))>0.005)
  stop(paste("descrepancies in deviance in WEI"))
scoef <- sum(coef(mwei)-coef(gwei))
if(abs(scoef)>0.005)
  warning(cat("descrepancies in coef in WEI of ", scoef, "\n"))
# WEI3 is weibull parametrised with mu as the mean
  gwei3 <- gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian, 
          family=cens(WEI3))
# log normal
mlno <- survreg(Surv(futime, fustat) ~ ecog.ps + rx, ovarian, 
  dist='lognormal')
glno <- gamlss(Surv(futime, fustat) ~ ecog.ps + rx, data=ovarian, 
  family=cens(LOGNO, delta=c(0.001,0.001)), c.cyc=0.00001)
if(abs(-2*mlno$loglik[2]-deviance(glno))>0.005)
  stop(paste("descrepancies in deviance in LOGNO"))
coef(mlno);coef(glno)
#--------------------------------------------------------------------
# now interval response variable
data(lip)
with(lip, y)
mg1 <- survreg(y ~ poly(Tem,2)+poly(pH,2)+poly(aw,2), data=lip, dist="weibull")
gg1 <- gamlss(y ~ poly(Tem,2)+poly(pH,2)+poly(aw,2), data=lip, 
          family=cens(WEI,type="interval"), c.crit=0.00001, n.cyc=200, trace=FALSE)
summary(mg1)
gen.cens(WEI,type="interval")
summary(gg1)
#--------------------------------------------------------------------------------
# fitting discretised continuous distribution to count data
# fitting discretised Gamma
data(species)
# first generate the distributions
gen.cens(GA, type="interval")
gen.cens(IG, type="interval")
mGA <- gamlss(Surv(fish,fish+1,type="interval2")~log(lake)+I(log(lake)^2),  
  sigma.fo=~log(lake), data=species, family=GAic)
# fitting discretised inverse Gaussian
mIG <- gamlss(Surv(fish,fish+1,type="interval2")~log(lake)+I(log(lake)*2),  
  sigma.fo=~log(lake), data=species, family=IGic)
AIC(mGA,mIG)
plot(fish~log(lake), data=species)
with(species, lines(log(lake)[order(lake)], fitted(mIG)[order(lake)]))
#--------------------------------------------------------------------------------
**Description**

Creates a probability density function from a current `gamlss.family` distribution to be used for fitting a censored or interval response variable.

**Usage**

```r
cens.d(family = "NO", type = c("right", "left", "interval"), ...)
```

**Arguments**

- `family`: a `gamlss.family` object, which is used to define the distribution and the link functions of the various parameters. The distribution families supported by `gamlss()` can be found in `gamlss.family` and in the package `gamlss.dist`.
- `type`: whether `right`, `left` or in `interval` censoring is required, (right is the default)
- `...`: for extra arguments

**Details**

This function is used to calculate the likelihood function for censored data. This function is not supposed to be used on its own but it is used in function `gen.cens`.

**Value**

Returns a modified d family function. The argument of the original function d function are the same.

**Note**

For an example see `gen.cens()`

**Author(s)**

Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk> and Bob Rigby

**References**


**See Also**

`cens.p, gen.cens`
cens.p

Censored Cumulative Probability Density Function of a gamlss.family Distribution

Description

Creates a cumulative density function from a current gamlss.family distribution suitable for censored or interval response variable data.

Usage

cens.p(family = "NO", type = c("right", "left", "interval"), ...)

Arguments

family a gamlss.family object, which is used to define the distribution and the link functions of the various parameters. The distribution families supported by gamlss() can be found in gamlss.family.

type whether right, left or in interval censoring is required, (right is the default)

... for extra arguments

Details

This function is used to calculate the quantile residuals for censored data distributions. This function is not supposed to be used on its own but it is used in the function gen.cens.

Value

Returns a modified p family function. The argument of the original function d function are the same.

Note

For an example see gen.cens()

Author(s)

Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk> and Bob Rigby <r.rigby@londonmet.ac.uk>
References


See Also

cens.d, gen.cens

Examples

#see the help for function cens for an example

cens.q

Censored Inverse Cumulative Probability Density Function of a 
gamlss.family Distribution

Description

Creates the inverse cumulative density function from a current gamlss.family distribution suitable for censored or interval response variable data. This is a dummy function identical to the uncensored one but it is needed for consistency in centile estimation from censored data.

Usage

cens.q(family = "NO", ...)

Arguments

family a gamlss.family object, which is used to define the distribution and the link functions of the various parameters. The distribution families supported by gamlss() can be found in gamlss.family.

... for extra arguments

Details

This is dummy function, used only to calculate centiles from censored response variable. This function is not supposed to be used on its own but is used by the function gen.cens.

Value

Returns a modified q family function. The argument of the original function q function are the same.
Note

For an example see gen.cens()

Author(s)

Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk> and Bob Rigby <r.rigby@londonmet.ac.uk>

References


See Also

cens.d, cens.p, gen.cens

Examples

#see the help for function cens for an example

generate cens()

A Function to Generate Appropriate Functions to Be Used to Fit a Censored Response variable in GAMLSS

Description

The gen.cens() function allows the user to generate a d, p, (dummy) q and fitting gamlss functions for censor and interval response variables. The function can take any gamlss.family distribution.

Usage

gen.cens(family = "NO", name = "cens",
         type = c("right", "left", "interval"), ...)

Arguments

family
- a `gamlss.family` object, which is used to define the distribution and the link functions of the various parameters. The distribution families supported by `gamlss()` can be found in `gamlss.family` and in the package `gamlss.dist`.

name
- the characters you want to add to the name of new functions, by default is the first letter of type and ci.e WEIic for WEI (weibull) interval response variable

type
- whether right, left or in interval censoring is required, (right is the default)

Value

Returns the d, p, (dummy) q and the fitting used in the fitting `gamlss` algorithm (The one used in the fitting `gamlss` algorithm) of a `gamlss.family` distribution.

Author(s)

Mikis Stasinopoulos <d.stasinopoulos@londonmet.ac.uk> and Bob Rigby

References


See Also

cens.d, cens.p, cens

Examples

```r
library(gamlss.dist)
data(lip)
gen.cens(WEI,type="interval")
WEIic
gg1<- gamlss(y ~ poly(Tem,2)+poly(pH,2)+poly(aw,2), data=lip,
  family=WEIic, c.crit=0.00001, n.cyc=200, trace=FALSE)
```
Description

The data set used in this package are collected by Dr Peggy Braun (University of Leipzig) and passed on to use by professor Jane Sutherland of London Metropolitan University. It consists of experimental enzymology results from a research project which attempted to develop a generic food spoilage model. The data set contains a column called NAMES, which shows the experiment name, three columns with values of the environmental conditions: temperature (Tem), pH and water activity (aw), and the rest of the columns contains the activity of the cocktails, observed at certain days. The researchers recorded the activity of proteases and lipases in each cocktail and were interested in predicting the time when the activity started given the environmental conditions. The activity is a positive integer and enzymes are considered inactive when activity=0.

Usage

```r
data(lip)
```

Format

A data frame with 120 observations on the following 14 variables.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>a factor with levels the different experiment</td>
</tr>
<tr>
<td>Tem</td>
<td>a numeric vector showing the temperature</td>
</tr>
<tr>
<td>pH</td>
<td>a numeric vector PH</td>
</tr>
<tr>
<td>aw</td>
<td>a numeric vector water activity</td>
</tr>
<tr>
<td>X0.d</td>
<td>a numeric vector if enzyme reacted at day 0</td>
</tr>
<tr>
<td>X1.d</td>
<td>a numeric vector if enzyme reacted at day 1</td>
</tr>
<tr>
<td>X2.d</td>
<td>a numeric vector if enzyme reacted at day 2</td>
</tr>
<tr>
<td>X4.d</td>
<td>a numeric vector if enzyme reacted at days 3 or 4</td>
</tr>
<tr>
<td>X11.d</td>
<td>a numeric vector if enzyme reacted at days 5 to 11</td>
</tr>
<tr>
<td>X18d.</td>
<td>a numeric vector if enzyme reacted at days 12 to q18</td>
</tr>
<tr>
<td>X25.d</td>
<td>a numeric vector if enzyme reacted at days 19 to 25</td>
</tr>
<tr>
<td>X32.d</td>
<td>a numeric vector if enzyme reacted at days 26 to 32</td>
</tr>
<tr>
<td>X39.d</td>
<td>a numeric vector if enzyme reacted at days 33 to 39</td>
</tr>
<tr>
<td>y</td>
<td>a matrix with 3 columns: this is a <code>Surv()</code> object indicating the start the finish and censored indicator as defined in function <code>Surv()</code> of <code>survival</code>.</td>
</tr>
</tbody>
</table>

Source

Prof. Jane Sutherland, London Metropolitan University
Examples

data(lip)
with(lip, y)
Index

* datasets
  lip, 12

* distribution
  cens, 4
  cens.d, 6
  cens.p, 8
  cens.q, 9
  gen.cens, 10

* package
  gamlss.cens-package, 2

* regression
  cens, 4
  cens.d, 6
  cens.p, 8
  cens.q, 9
  gen.cens, 10

  cens, 4, 11
  cens.d, 5, 6, 9–11
  cens.p, 5, 7, 8, 10, 11
  cens.q, 9

  gamlss, 3
  gamlss.cens (gamlss.cens-package), 2
  gamlss.cens-package, 2
  gamlss.family, 3, 4, 7–11
  gen.cens, 5, 7, 9, 10, 10

  lip, 12