Package 'sglg'

April 25, 2020

Type Package

Title Fitting Semi-Parametric Generalized log-Gamma Regression Models

Version 0.1.6

Author
Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com> and G. Paula and L. Vanegas

Maintainer Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>

Description Set of tools to fit a linear multiple or semi-parametric regression
models and non-informative right-censoring may be considered. Under this setup, the localization parameter of the response variable distribution is modeled by using linear multiple regression or semi-parametric functions, whose non-parametric components may be approximated
by natural cubic spline or P-
splines. The supported distribution for the model error is a generalized log-
gamma distribution which includes
the generalized extreme value distribution as important special case.

Depends R (>= 3.1.0)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Suggests testthat

Imports ssym, Formula, survival, methods, graphics, stats,
AdequacyModel, ggplot2, moments, gridExtra, pracma

NeedsCompilation no

Repository CRAN

Date/Publication 2020-04-25 14:40:02 UTC

R topics documented:

`deBoor2` *Build the basis matrix and the penalty matrix of cubic B-spline basis.*

Description

`deBoor` builds the basis matrix and penalty matrix to approximate a smooth function using cubic B-spline cubic.

Usage

```
deBoor2(t, knots)
```

Arguments

`t`	a vector of values.
`knots`	a set of internal knot.

Value

nknot number of knots.

knots set of knots.

N basis matrix.

K penalty matrix.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
t <- runif(120)
knot <- 6
deBoor2(t,knot)
```

---

deviance_residuals *Deviance Residuals for a Generalized Log-gamma Regression Model*

Description

`deviance_residuals` is used to generate deviance residuals for a generalized log-gamma regression model. Also, under the presence of a right-censored sample. The function plot() produces a graph of the survival function of the error distribution.

Usage

```
deviance_residuals(object, ...)
```

Arguments

`object`	an object of the class sglg. This object is returned from the call to glg(), sglg(), survglg() or ssurvglg().
`...`	other arguments.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
# Example 1
n <- 300
error <- rglg(n,0,1,1)
y <- 0.5 + error
fit <- glg(y~1,data=as.data.frame(y))
deviance_residuals(fit)
# Example 2
n <- 300
error <- rglg(n,0,1,1)
x <- runif(n,-3,3)
y <- 0.5 +  2*x + error
fit <- glg(y~x,data=as.data.frame(y,x))
deviance_residuals(fit)
```

`dglg` *Density distribution function for a generalized log-gamma variable*

Description

`dglg` is used to calculate the density distribution function of a generalized log-gamma variable at x.

Usage

```
dglg(x, location, scale, shape)
```

Arguments

`x`	numeric, a real number.
`location`	numeric, represent the location parameter of a generalized log-gamma distribution. Default value is 0.
`scale`	numeric, represent the scale parameter of a generalized log-gamma distribution. Default value is 1.
`shape`	numeric, represent the shape parameter of a generalized log-gamma distribution. Default value is 1.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
x <- runif(60,-3,3)
dglg(sort(x),location=0,scale=1,shape=1)
plot(sort(x),dglg(sort(x),location=0.5,scale=1,shape=1),type="l",xlab="x",ylab="Density")
```

`entropy` *Tool to calculate the entropy for a generalized log-gamma distribution.*

Description

`entropy` is used to obtain the entropy for a generalized log-gamma distribution.

Usage

```
entropy(mu, sigma, lambda)
```

Arguments

`mu`	numeric, represent the location parameter of a generalized log-gamma distribution. Default value is 0.
`sigma`	numeric, represent the scale parameter of a generalized log-gamma distribution. Default value is 1.
`lambda`	numeric, represent the shape parameter of a generalized log-gamma distribution. Default value is 1.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
entropy(0,1,-1)    # Extreme value type I distribution, maximum case.
entropy(0,1,1)     # Extreme value type I distribution, minimum case.
entropy(0,1,0.077) # Standard normal distribution.
```

envelope.sglg *envelope.sglg*

Description

Build a Normal probability plot with simulated envelope for a deviance-type residuals in semiparametric or multiple linear generalized log-gamma regression models.

Usage

```
envelope.sglg(fit, Rep)
```

Arguments

fit	an object of the class sglg. This object is returned from the call to glg(), sglg().
Rep	a positive integer. This is the number of replications on which to build the simulated envelope. Default is Rep=50.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Ortega, E., Paula, G. A. and Bolfarine, H. (2008) Deviance residuals in generalized log-gamma regression models with censored observations. Journal of Statistical Computation and Simulation, 78, 747-764.

Examples

```
rows <- 120
columns <- 2
t_beta  <- c(0.5, 2)
t_sigma <- 0.5
t_lambda <- 1
set.seed(8142031)
x1 <- rbinom(rows, 1, 0.5)
x2 <- runif(columns, 0, 1)
X <- cbind(x1,x2)
error <- rglg(rows, 0, 1, t_lambda)
y1 <- X %*%t_beta + t_sigma * error
data.example <- data.frame(y1,X)
fit <- glg(y1 ~ x1 + x2 - 1,data=data.example)
envelope.sglg(fit,Rep=50)
```

glg *Fitting multiple linear Generalized Log-gamma Regression Models*

Description

glg is used to fit a multiple linear regression model suitable for analysis of data sets in which the response variable is continuous, strictly positive, and asymmetric. In this setup, the location parameter of the response variable is explicitly modeled by a linear function of the parameters.

Usage

```
glg(formula, data, shape, Tolerance, Maxiter)
```

Arguments

formula	a symbolic description of the systematic component of the model to be fitted. See details for further information.
data	an optional data frame, list containing the variables in the model.
shape	an optional value for the shape parameter of the error distribution of a generalized log-gamma distribution. Default value is 1.
Tolerance	an optional positive value, which represents the convergence criterion. Default value is 1e-04.
Maxiter	an optional positive integer giving the maximal number of iterations for the estimating process. Default value is 1e03.

Value

mu a vector of parameter estimates asociated with the location parameter.

sigma estimate of the scale parameter associated with the model.

lambda estimate of the shape parameter associated with the model.

interval estimate of a 95% confidence interval for each estimate parameters associated with the model.

Deviance the deviance associated with the model.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
rows <- 200
columns <- 2
x1 <- rbinom(rows, 1, 0.5)
x2 <- runif(columns, 0, 1)
X <- cbind(x1,x2)
t_beta  <- c(0.5, 2)
t_sigma <- 1

######################
#                    #
# Extreme value case #
#                    #
######################

t_lambda <- 1
set.seed(8142031)
error <- rglg(rows, 0, 1, t_lambda)
y1 <- X %*%t_beta + t_sigma * error
data.example <- data.frame(y1,X)
fit1 <- glg(y1 ~ x1 + x2 - 1,data=data.example)
summary(fit1)
deviance_residuals(fit1)

###############
#             #
# Normal case #
#             #
###############

t_lambda <- 0.001
set.seed(8142031)
error <- rglg(rows, 0, 1, t_lambda)
y1 <- X %*%t_beta + t_sigma * error
data.example <- data.frame(y1, X)
fit0 <- glg(y1 ~ x1 + x2 - 1,data=data.example)
logLik(fit0)
fit0$AIC
fit0$mu

##########################################
#                                        #
#  A comparison with a normal linear model #
#                                        #
##########################################

fit2 <- lm(y1 ~ x1 + x2 - 1,data=data.example)
logLik(fit2)
AIC(fit2)
coefficients(fit2)
```

gnfit *gnfit*

Description

This function provides some useful statistics to assess the quality of fit of generalized log-gamma probabilistic model, including the statistics Cramer-von Mises and Anderson-Darling. It can also calculate other goodness of fit such as Hannan-Quin Information Criterion and Kolmogorov-Smirnov test.

Usage

```
gnfit(starts, data)
```

Arguments

starts	numeric vector. Initial parameters to maximize the likelihood function
data	numeric vector. A sample of a generalized log-gamma distribution.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
## Not run:
set.seed(1)
# The size of the sample must be median or large to obtain a good estimates
n <- 100
sample <- rglg(n,location=0,scale=0.5,shape=0.75)
# This step takes a few minutes.
result <- gnfit(starts=c(0.1,0.75,1),data=sample)
result

## End(Not run)
```

Gu *Tool to build the basis matrix and the penalty matrix of natural cubic splines.*

Description

Gu builds the basis matrix and penalty matrix to approximate a smooth function using natural cubic splines based on the Gu basis form.

Usage

```
Gu(t, knot)
```

Arguments

t	the covariate.
knot	a integer value that represent the number of knots of the natural cubic spline.

Value

nknot number of knots.

knots set of knots.

N basis matrix.

K penalty matrix.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Wood, S. (2006) Generalized additive models: An R introduction. Chapman and Hall.

Carlos Alberto Cardozo Delgado. Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
t <- runif(120)
knot <- 6
Gu(t,knot)
```

`influence.sglg` *influence*

Description

influence.sglg extracts from a object of class sglg the local influence measures and displays their graphs versus the index of the observations.

Usage

```
## S3 method for class 'sglg'
influence(model, ...)
```

Arguments

`model`	an object of the class sglg. This object is returned from the call to glg(), sglg(), survglg() or ssurvglg().
`...`	other arguments.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
rows <- 100
columns <- 2
t_beta  <- c(0.5, 2)
t_sigma <- 1
t_lambda <- 1
set.seed(8142031)
x1 <- rbinom(rows, 1, 0.5)
x2 <- runif(columns, 0, 1)
X <- cbind(x1,x2)
error <- rglg(rows, 0, 1, t_lambda)
y1 <- X %*%t_beta + t_sigma * error
data.example <- data.frame(y1,X)
fit1 <- glg(y1 ~ x1 + x2 - 1,data=data.example)
influence(fit1)
```

logLik.sglg *Extract Log-Likehood*

Description

logLik.sglg extracts log-likehood from a model from an object of class 'sglg'.

Usage

```
## S3 method for class 'sglg'
logLik(object, ...)
```

Arguments

object	an object of the class sglg. This object is returned from the call to glg(), sglg(), survglg() or ssurvglg() function.
...	other arguments.

lss *Measures of location, scale and shape measures for a generalized log-gamma distribution*

Description

lss is used to obtain the mean, variance, skewness and kurtosis for a generalized log-gamma distribution.

Usage

```
lss(mu, sigma, lambda)
```

Arguments

mu	numeric, represent the location parameter of a generalized log-gamma distribution. Default value is 0.
sigma	numeric, represent the scale parameter of a generalized log-gamma distribution. Default value is 1.
lambda	numeric, represent the shape parameter of a generalized log-gamma distribution. Default value is 1.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

National Institute of Standards and Technology, NIST. Engineering Statistics Handbook. http://www.itl.nist.gov/div898/handl

Examples

```
lss(0,1,-1)    # Extreme value type I distribution, maximum case.
lss(0,1,1)     # Extreme value type I distribution, minimum case.
lss(0,1,0.005) # Standard normal distribution.
```

`order_glg` *Random Sampling of Order Statistics from a Generalized Log-gamma Distribution*

Description

`order_glg` is used to obtain a random sample of order statistics from a Generalized Log-gamma Distribution.

Usage

```
order_glg(size, mu, sigma, lambda, k, n)
```

Arguments

`size`	numeric, represents the size of the sample.
`mu`	numeric, represents the location parameter. Default value is 0.
`sigma`	numeric, represents the scale parameter. Default value is 1.
`lambda`	numeric, represents the shape parameter. Default value is 1.
`k`	numeric, represents the Kth smallest value from a sample.
`n`	numeric, represents the size of the sample to compute the order statistic from.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>.

References

Gentle, J, Computational Statistics, First Edition. Springer - Verlag, 2009.

Naradajah, S. and Rocha, R. (2016) Newdistns: An R Package for New Families of Distributions, Journal of Statiscal Software.

Examples

```
# A random sample of size 10 of order statistics from a Extreme Value Distribution.
order_glg(10,0,1,1,1,50)
## Not run:  # A small comparison between two random sampling methods of order statistics
# Method 1
m <- 10
output <- rep(0,m)
order_sample <- function(m,n,k){
for(i in 1:m){
sample <- rglg(n)
order_sample <- sort(sample)
output[i] <- order_sample[k]
}
return(output)
}
N <- 10000
n <- 200
k <- 100
system.time(order_sample(N,n,k))
sample_1 <- order_sample(N,n,k)
hist(sample_1)
summary(sample_1)
# Method 2
system.time(order_glg(N,0,1,1,k,n))
sample_2 <- order_glg(N,0,1,1,k,n)$sample
hist(sample_2)
summary(sample_2)

## End(Not run)
```

`pglg` *Cumulative distribution function for a generalized log-gamma variable*

Description

`pglg` is used to calculate the cumulative distribution function of a generalized log-gamma variable at x.

Usage

```
pglg(x, location, scale, shape)
```

Arguments

`x`	numeric, a real number.
`location`	numeric, represent the location parameter of a generalized log-gamma distribution. Default value is 0.

scale	numeric, represent the scale parameter of a generalized log-gamma distribution. Default value is 1.
shape	numeric, represent the shape parameter of a generalized log-gamma distribution. Default value is 1.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
x <- runif(3,-1,1)
pglg(sort(x),location=0.5,scale=1,shape=1)
```

---

plotnpc *Plotting a natural cubic splines or P-splines.*

Description

plotnpc displays a graph of a fitted nonparametric effect, either natural cubic spline or P-spline, from an object of class sglg.

Usage

```
plotnpc(fit, conf_lev)
```

Arguments

fit	an object of the class sglg. This object is returned from the call to glg(), sglg(), survglg() or ssurvglg().
conf_lev	is the confidence level of the asymptotic confidence band. Default value is 0.05.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Eilers P.H.C. and Marx B.D. (1996). Flexible smoothing with B-splines and penalties. Statistical Science. 11, 89-121.

Wood, S. (2006). Additive generalized models: An R introduction. Chapman and Hall.

Examples

```
library(sglg)
library(ssym)
library(ggplot2)
set.seed(1)
n <- 200
error <- rglg(n,0,1,1)
t <- as.matrix((2*1:n - 1)/(2*n))
colnames(t) <- "t"
f_t <- cos(4*pi*t)
y <- 0.5 + f_t + error
colnames(y) <- "y"
data <- as.data.frame(cbind(y,1,t))
fit1 <- sglg(y ~ 1,npc=t,data=data,basis = "deBoor",alpha0=seq(0.001,0.005,by=0.001),nknts=5)
quantile_residuals(fit1)
# The adjusted (black) and true (red) non-linear component
plotnpc(fit1) + geom_line(aes(t,f_t),colour="red")
fit2 <- sglg(y ~ 1,npc=t,data=data,basis = "Gu",alpha0=seq(0.001,0.005,by=0.001),nknts=9)
quantile_residuals(fit2)
# The adjusted (black) and true (red) non-linear component
plotnpc(fit2,conf_lev=0.01) + geom_line(aes(t,f_t),colour="red",size=1.2)
```

---

plotsurv.sglg *Plot simultaneously the Kaplan-Meier and parametric estimators of the survival function.*

Description

`plotsurv.sglg` is used to plot simultaneously the Kaplan-Meier and parametric estimators of the survival function.

Usage

```
plotsurv.sglg(fit)
```

Arguments

`fit`	an object of the class sglg. This object is returned from the call to survglg() or ssurvglg().

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos A. Cardozo, G. Paula and L. Vanegas. Semi-parametric accelerated failure time models with generalized log-gamma erros. In preparation.

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
rows  <- 240
columns <- 2
t_beta  <- c(0.5, 2)
t_sigma <- 1
t_lambda <- 1
set.seed(8142031)
library(ssym)
x1 <- rbinom(rows, 1, 0.5)
x2 <- runif(columns, 0, 1)
X <- cbind(x1,x2)
s         <- t_sigma^2
a         <- 1/s
t_ini1    <- exp(X %*% t_beta) * rgamma(rows, scale = s, shape = a)
cens.time <- rweibull(rows, 0.6, 14)
delta1     <- ifelse(t_ini1 > cens.time, 1, 0)
obst1 <- t_ini1
for (i in 1:rows) {
if (delta1[i] == 1) {
   obst1[i] <- cens.time[i]
  }
}
data.example <- data.frame(obst1,delta1,X)
fit3 <- survglg(Surv(log(obst1),delta1) ~ x1 + x2 - 1, data=data.example,shape=0.9)
plotsurv.sglg(fit3)
```

---

qglg *Quantile function for a generalized log-gamma variable*

Description

qglg is used to calculate the quantile function of a generalized log-gamma variable at x.

Usage

```
qglg(x, location, scale, shape)
```

Arguments

x	numeric, a real number.
location	numeric, represent the location parameter of a generalized log-gamma distribution. Default value is 0.
scale	numeric, represent the scale parameter of a generalized log-gamma distribution. Default value is 1.
shape	numeric, represent the shape parameter of a generalized log-gamma distribution. Default value is 1.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
x <- runif(3,0,1)
qglg(sort(x),location=0,scale=1,shape=1)
```

---

quantile_residuals *Quantile Residuals for a Generalized Log-gamma Regression Model*

Description

quantile_residuals is used to generate quantile residuals for a generalized log-gamma regression model.

Usage

```
quantile_residuals(fit)
```

Arguments

fit	is an object sglg. This object is returned from the call to glg(), sglg(), survglg() or ssurvglg().

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
# Example 1
n <- 300
error <- rglg(n,0,1,1)
y <- 0.5 + error
fit <- glg(y~1,data=as.data.frame(y))
quantile_residuals(fit)
# Example 2
n <- 300
```

```
error <- rglg(n,0,1,1)
x <- runif(n,-3,3)
y <- 0.5 +  2*x + error
fit <- glg(y~x,data=as.data.frame(y,x))
quantile_residuals(fit)
```

`residuals.sglg` *Extract Model Residuals*

Description

residuals.sglg extracts the deviance-type residuals for a model from an object of class 'sglg'.

Usage

```
## S3 method for class 'sglg'
residuals(object, ...)
```

Arguments

`object`	an object of the class sglg. This object is returned from the call to glg(), sglg(), survglg() or ssurvglg() function.
`...`	other arguments.

`rglg` *Random number generation for a generalized log-gamma distribution*

Description

`rglg` is used to generate random numbers for a generalized log-gamma distribution.

Usage

```
rglg(n, location, scale, shape)
```

Arguments

`n`	numeric, size of the random sample.
`location`	numeric, represent the location parameter of a generalized log-gamma distribution. Default value is 0.
`scale`	numeric, represent the scale parameter of a generalized log-gamma distribution. Default value is 1.
`shape`	numeric, represent the shape parameter of a generalized log-gamma distribution. Default value is 1.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
rglg(10,location=-1,scale=0.5,shape=1)
```

---

sglg *Fitting semi-parametric generalized log-gamma regression models*

Description

sglg is used to fit a semi-parametric regression model suitable for analysis of data sets in which the response variable is continuous, strictly positive, and asymmetric. In this setup, the location parameter of the response variable is explicitly modeled by semi-parametric functions, whose non-parametric components may be approximated by natural cubic splines or cubic P-splines.

Usage

```
sglg(
  formula,
  npc,
  basis,
  data,
  shape,
  method,
  alpha0,
  nknts,
  Tolerance,
  Maxiter
)
```

Arguments

formula	a symbolic description of the systematic component of the model to be fitted. See details for further information.
npc	a data frame with potential nonparametric variables of the systematic part of the model to be fitted.
basis	a name of the cubic spline basis to be used in the model. Supported basis include deBoor and Gu basis which are a B-spline basis and a natural cubic spline basis, respectively.

data	an optional data frame, list containing the variables in the model.
shape	an optional value for the shape parameter of the error distribution of a generalized log-gamma distribution. Default value is 1.
method	There are two possibles algorithms to estimate the parameters. The default algorithm is 'FS' Fisher-Scoring, the other option is 'GSFS' an adequate combination between the block matrix version of non-linear Gauss-Seidel algorithm and Fisher-Scoring algorithm.
alpha0	is a vector of initial values for the smoothing parameter alpha.
nknts	is a vector of the number of knots in each non-linear component of the model.
Tolerance	an optional positive value, which represents the convergence criterion. Default value is 1e-04.
Maxiter	an optional positive integer giving the maximal number of iterations for the estimating process. Default value is 1e03.

Value

mu a vector of parameter estimates asociated with the location parameter.

sigma estimate of the scale parameter associated with the model.

lambda estimate of the shape parameter associated with the model.

interval estimate of a 95% confidence interval for each estimate parameters associated with the model.

Deviance the deviance associated with the model.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos A. Cardozo, G. Paula and L. Vanegas. Semi-parametric generalized log-gamma regression models. In preparation.

Examples

```
library(ssym)
library(sglg)
set.seed(1)
n <- 200
error <- rglg(n,0,1,1)
x1 <- runif(n,-3,3)
x2 <- rbinom(n,1,0.5)
t <- as.matrix((2*1:n - 1)/(2*n))
colnames(t) <- "t"
f_t <- cos(4*pi*t)
y <- 2*x1 + 0.5*x2 + f_t + error
colnames(y) <- "y"
data <- as.data.frame(cbind(y,x1,x2,t))
```

```
fit1 <- sglg(y ~ x1 + x2 - 1,npc=t,data=data,basis = "deBoor")
fit2 <- sglg(y ~ x1 + x2 - 1,npc=t,data=data,basis = "Gu",alpha0=c(0.05,0.1),nknts=7)
```

---

shape *shape*

Description

Tool that supports the estimation of the shape parameter in semi-parametric or multiple linear accelerated failure time model with generalized log-gamma erros under the presence of censored data. The estimation is based on the profiled likelihood function for the shape parameter of the model.

Usage

```
shape(formula, npc, data, interval, semi, step)
```

Arguments

formula	a symbolic description of the systematic component of the model to be fitted.
npc	a data frame with potential nonparametric variables of the systematic part of the model to be fitted.
data	a data frame which contains the variables in the model.
interval	an optional numerical vector of length 2. In this interval is the maximum likelihood estimate of the shape parameter of the model. By default is [0.1,1.5].
semi	a logical value. TRUE means that the model has a non-parametric component. By default is FALSE.
step	an optional positive value. This parameter represents the length of the step of the partition of the interval parameter. By default is 0.1.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos Alberto Cardozo Delgado, Semi-parametric generalized log-gamma regression models. Ph. D. thesis. Sao Paulo University.

Examples

```
rows  <- 200
columns <- 2
t_beta  <- c(0.5, 2)
t_sigma <- 1
t_lambda <- 1
set.seed(8142031)
library(ssym)
```

```
x1 <- rbinom(rows, 1, 0.5)
x2 <- runif(columns, 0, 1)
X <- cbind(x1,x2)
s         <- t_sigma^2
a         <- 1/s
t_ini1    <- exp(X %*% t_beta) * rgamma(rows, scale = s, shape = a)
cens.time <- rweibull(rows, 0.3, 14)
delta     <- ifelse(t_ini1 > cens.time, 1, 0)
obst1 = t_ini1
for (i in 1:rows) {
if (delta[i] == 1) {
   obst1[i] = cens.time[i]
  }
}
example <- data.frame(obst1,delta,X)
lambda <- shape(Surv(log(obst1),delta) ~ x1 + x2 - 1, data=example)
lambda
# To change interval or step or both options
lambda <- shape(Surv(log(obst1),delta) ~ x1 + x2 - 1, data=example, interval=c(0.95,1.3), step=0.05)
lambda
```

`ssurvglg` *Fitting semi-parametric generalized log-gamma regression models under the presence of right censored data.*

Description

`ssurvglg` is used to fit a semi-parametric regression model in which the response variable is continuous, strictly positive, asymmetric and there are right censored observations. In this setup, the location parameter of the logarithm of the variable is explicitly modeled by semi-parametric functions, whose nonparametric components may be approximated by natural cubic splines or cubic P-splines.

Usage

```
ssurvglg(formula, npc, basis, data, shape, Maxiter, Tolerance)
```

Arguments

`formula`	a symbolic description of the systematic component of the model to be fitted. See details for further information.
`npc`	a data frame with potential nonparametric variables of the systematic part of the model to be fitted.
`basis`	a name of the cubic spline basis to be used in the model. Supported basis include deBoor and Gu basis which are a B-spline basis and a natural cubic spline basis, respectively.
`data`	an optional data frame, list containing the variables in the model.
`shape`	an optional value for the shape parameter of the model.

Maxiter	an optional positive integer giving the maximal number of iterations for the estimating process. Default value is 1e03.
Tolerance	an optional positive value, which represents the convergence criterion. Default value is 1e-04.

Value

mu a vector of parameter estimates asociated with the location parameter.

sigma estimate of the scale parameter associated with the model.

lambda estimate of the shape parameter associated with the model.

interval estimate of a 95% confidence interval for each estimate parameters associated with the model.

Deviance the deviance associated with the model.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos A. Cardozo, G. Paula and L. Vanegas. Semi-parametric accelerated failure time models with generalized log-gamma erros: Censored case. In preparation.

Examples

```
rows    <- 150
columns <- 2
t_beta  <- c(0.5, 2)
t_sigma <- 0.75
t_lambda <- 1
set.seed(8142030)
library(ssym)
x1 <- rbinom(rows, 1, 0.5)
x2 <- runif(rows, 0, 1)
X <- cbind(x1,x2)
t_knot1 <- 6
ts1 <- seq(0, 1, length = t_knot1)
t_g1 <- 0.4 * sin(pi * ts1)
BasisN <- function(n, knot) {
N <- matrix(0, n, knot)
m <- n/knot
block <- rep(1,m)
for (i in 1:knot) {
   l <- (i - 1) * m + 1
 r <- i * m
  N[l:r, i] <- block }
return(N)
 }
s_N1 <- BasisN(rows, length(ts1))
x3 <- s_N1 %*% ts1
```

```
colnames(x3) <- 'x3'
sys       <- X %*% t_beta + s_N1%*%t_g1
t_ini1    <- exp(sys) * rweibull(rows,1/t_sigma,1)
cens.time <- rweibull(rows, 1.5, 14)
delta     <- ifelse(t_ini1 > cens.time, 1, 0)
obst1 = t_ini1
for(i in 1:rows) {
   if (delta[i] == 1) {
       obst1[i] = cens.time[i]
      }
}
data.example <- data.frame(obst1, delta, X, x3)
fit4 <- ssurvglg(Surv(log(obst1),delta)~ x1 + x2 - 1, npc=x3, data=data.example, shape=0.8)
```

---

summary.sglg *summary.sglg*

Description

summary.sglg extracts displays the summary of the fitted model including parameter estimates, associated (approximated) standard errors and goodness-of-fit statistics from a model from an object of class 'sglg'.

Usage

```
## S3 method for class 'sglg'
summary(object, ...)
```

Arguments

object	an object of the class sglg. This object is returned from the call to glg(), sglg(), survglg() or ssurvglg() function.
...	other arguments.

---

survglg *Fitting linear generalized log-gamma regression models under the presence of right censored data.*

Description

survglg is used to fit a multiple linear regression model in which the response variable is continuous, strictly positive, asymmetric and there are right censored observations. In this setup, the location parameter of the logarithm of the response variable is modeled by a linear model of the parametes.

Usage

```
survglg(formula, data, shape, Maxiter, Tolerance)
```

Arguments

formula	a symbolic description of the systematic component of the model to be fitted. See details for further information.
data	an optional data frame, list containing the variables in the model.
shape	an optional value for the shape parameter of the model.
Maxiter	an optional positive integer giving the maximal number of iterations for the estimating process. Default value is 1e03.
Tolerance	an optional positive value, which represents the convergence criterion. Default value is 1e-04.

Value

mu a vector of parameter estimates asociated with the location parameter.

sigma estimate of the scale parameter associated with the model.

lambda estimate of the shape parameter associated with the model.

interval estimate of a 95% confidence interval for each estimate parameters associated with the model.

Deviance the deviance associated with the model.

Author(s)

Carlos Alberto Cardozo Delgado <cardozorpackages@gmail.com>, G. Paula and L. Vanegas.

References

Carlos A. Cardozo, G. Paula and L. Vanegas. Semi-parametric accelerated failure time models with generalized log-gamma erros. In preparation.

Examples

```
rows  <- 240
columns <- 2
t_beta  <- c(0.5, 2)
t_sigma <- 1
t_lambda <- 1
set.seed(8142031)
library(ssym)
x1 <- rbinom(rows, 1, 0.5)
x2 <- runif(columns, 0, 1)
X <- cbind(x1,x2)
s         <- t_sigma^2
a         <- 1/s
t_ini1    <- exp(X %*% t_beta) * rgamma(rows, scale = s, shape = a)
cens.time <- rweibull(rows, 0.3, 14)
```

```
delta1     <- ifelse(t_ini1 > cens.time, 1, 0)
obst1 <- t_ini1
for (i in 1:rows) {
if (delta1[i] == 1) {
   obst1[i] <- cens.time[i]
  }
}
data.example <- data.frame(obst1,delta1,X)
fit3 <- survglg(Surv(log(obst1),delta1) ~ x1 + x2 - 1, data=data.example,shape=0.9)
summary(fit3)
```

Index