Package ‘gof’

February 20, 2015

Type Package
Title Model-diagnostics based on cumulative residuals
Version 0.9.1
Date 2014-03-13
Author Klaus K. Holst
Maintainer Klaus. K. Holst <kkho@biostat.ku.dk>
Description Implementation of model-checking techniques for generalized linear models and linear structural equation models based on cumulative residuals
Depends R (>= 2.15)
Suggests numDeriv, survival, lava (>= 1.2.5)
License GPL
LazyLoad yes
Collate 'confint.R' 'cumres.coxph.R' 'cumres.default.R' 'cumres.glm.R'
                 'cumres.lvmfit.R' 'gof-package.R' 'onload.R' 'plot.R' 'print.R'
                 'vcov.R'
NeedsCompilation yes
Repository CRAN
Date/Publication 2014-03-14 07:50:00

R topics documented:
  gof-package .................................................. 2
  confint.cumres ............................................. 2
  cumres.coxph ................................................ 3
  cumres.glm .................................................. 5
  cumres.lvmfit .............................................. 7
  plot.cumres .................................................. 8
  surgunit .................................................... 10

Index 11
Model-diagnostics based on cumulative residuals

Description

Model-diagnostics based on cumulative residuals

Author(s)

Klaus K. Holst <kkho@biostat.ku.dk>

References


See Also

cox.aalen in the timereg-package for similar GoF-methods for survival-data.

Examples

example(cumres)

 Returns prediction bands for 'cumres' object

Description

Calculates prediction bands for the cumulative residual process under the null.

Usage

## S3 method for class 'cumres'
confint(object,
  parm = 1:length(object$variable), level = 0.95,
  cval = NULL, ...)
Arguments

- **object**: Object produced by the function `cumres`.
- **parm**: Vector of numbers indicating which processes from the `x` to calculate prediction bands for.
- **level**: The required prediction level.
- **cval**: Overrules the level-parameter by calculating symmetric prediction bands defined by the standard error multiplied by `cval`.

Value

A list with the following members:

- `"t"`: Ordered values of variable that is used to cumulate residuals after
- `yu`: Upper simultaneous confidence limit

Author(s)

Klaus K. Holst <kkho@biostat.ku.dk>

See Also

- `cumres`

Examples

```r
n <- 500; x <- abs(rnorm(n, sd=0.2)) + 0.01; y <- sqrt(x) + rnorm(n, sd=0.2)
l <- lm(y ~ x)
g <- cumres(l, R=1000)
confint(g,l)
```

---

### cumres.coxph

*Calculates GoF measures for Cox's proportional hazard model for right censored survival times*

Description

Calculates score processes and KS and Cvm tests for proportional hazards via simulation (Martinsussen and Scheike, 2006).

Usage

```r
## S3 method for class 'coxph'
cumres(model,
variable = c(colnames(model.matrix(model))),
type = c("score", "residual"), R = 1000,
plots = min(R, 50), seed = round(runif(1, 1, 1e+09)),
...)
```
Arguments

- `model`: Model object (lm or glm)
- `variable`: List of variable to order the residuals after
- `R`: Number of samples used in simulation
- `type`: Type of GoF-procedure
- `plots`: Number of realizations to save for use in the plot-routine
- `seed`: Random seed
- `...`: additional arguments

Value

Returns an object of class 'cumres'.

Author(s)

Klaus K. Holst and Thomas Scheike

References


See Also

cumresNglm, coxph, and cox.aalen in the timereg package for similar GoF-methods for survival-data.

Examples

```r
library(survival)

simcox <- function(n=100, seed=1) {
  if (!is.null(seed))
    set.seed(seed)
  require(survival)
  time<-rexp(n); cen<-2*rexp(n);
  status<-(time<cen);
  time[status==0]<-cen[status==0];
  X<-matrix(rnorm(2*n),n,2)
  return(data.frame(time=time, status=status, X))
}

n <- 100; d <- simcox(n); m1 <- coxph(Surv(time,status)~ X1 + X2, data=d)
cumres(m1)

## Not run:
## PBC example
```
cumres.glm

Calculates GoF statistics based on cumulative residual processes

Description

Given the generalized linear models model

\[ g(E(Y_i|X_{i1}, ..., X_{ik})) = \sum_{i=1}^{k} \beta_j X_{ij} \]

the `cumres`-function calculates the observed cumulative sum of residual process, cumulating the residuals, \( e_i \), by the jth covariate:

\[ W_j(t) = n^{-1/2} \sum_{i=1}^{n} 1\{X_{ij}<t\} e_i \]

and Kolmogorov-Smirnov and Cramer-von-Mises test statistics are calculated via simulation from the asymptotic distribution of the cumulative residual process under the null (Lin et al., 2002).

Usage

```r
## S3 method for class 'glm'
 cumres(model,     
 variable = c("predicted", colnames(model.matrix(model))), 
 data = data.frame(model.matrix(model)), R = 1000, 
 b = 0, plots = min(R, 50), breakties = 1e-12, 
 seed = round(runif(1, 1, 1e+09)), ...) 
```

Arguments

- `model`: Model object (lm or glm)
- `variable`: List of variable to order the residuals after
- `data`: data.frame used to fit model (complete cases)
- `R`: Number of samples used in simulation
- `b`: Moving average bandwidth (0 corresponds to infinity = standard cumulated residuals)
- `plots`: Number of realizations to save for use in the plot-routine
- `breakties`: Add unif[0,breakties] to observations
- `seed`: Random seed
- `...`: additional arguments
Value

Returns an object of class ‘cumres’.

Note

Currently linear (normal), logistic and poisson regression models with canonical links are supported.

Author(s)

Klaus K. Holst

References


See Also

cox.aalen in the timereg-package for similar GoF-methods for survival-data.

Examples

```r
sim1 <- function(n=100, f=function(x1,x2) {10+x1+x2*x2}, sd=1, seed=1) {
  if (is.null(seed))
    set.seed(seed)
  x1 <- rnorm(n);
  x2 <- rnorm(n)
  X <- cbind(x1,x2)
  y <- f(x1,x2) + rnorm(n,sd=sd)
  d <- data.frame(y,x1,x2)
  return(d)
}
d <- sim1(n=100);  l <- lm(y ~ x1 + x2,d)
system.time(g <- cumres(l, R=100, plots=50))
g
plot(g)
g1 <- cumres(l, c("y"), R=100, plots=50)
g1
g2 <- cumres(l, c("y"), R=100, plots=50, b=0.5)
g2
```
cumres.lvmfit

Cumulative residual processes for structural equation models

Description
Calculates GoF statistics based on cumulative residual processes for structural equation models fitted with the lava package.

Usage
```
# S3 method for class 'lvmfit'
cumres(model, y, x, full = FALSE,
       data = model.frame(model), p, R = 1000, b = 0,
       plots = min(R, 50), seed = round(runif(1, 1, 1e+09)),
       ...)```

Arguments
- `model`: lvm object
- `y`: A formula specifying the association to be checked. Alternatively the outcome specified as a function or a string with the name of the outcome in the model.
- `x`: Predictor. A function, vector or character
- `full`: If FALSE the prediction, Pr, of the variable that are ordered after is only calculated based on the conditional distribution given covariates. If TRUE the conditional expectation is based on the largest set of covariates and endogenous variables such that the residual and Pr are uncorrelated.
- `data`: data.frame (default is the model.frame of the model)
- `p`: Optional parameter vector
- `R`: Number of processes to simulate
- `b`: Moving average parameter
- `plots`: Number of processes to save for use with the plot method
- `seed`: Random seed
- `...`: Additional arguments parsed on to lower-level functions

Details
With y and x given as functions the user can decide which variables to use in the prediction of the outcome and predictor (use the predict method as below).

Value
Returns a cumres object with associated plot, print, confint methods
Author(s)

Klaus K. Holst

References


Examples

```r
library(lava)
m <- lvm(list(c(y1,y2,y3)~eta, eta~x)); latent(m) <- ~eta
## simulate some data with non-linear covariate effect
functional(m,eta~x) <- function(x) 0.3*x^2
d <- sim(m,100)
e <- estimate(m,d)
## Checking the functional form of eta on x
g <- cumres(e,eta~x,R=1000)
plot(g)

x <- function(p) predict(e,x=-y2+y3,p=p)[,"eta"]
## Checking the functional form of y1 on eta
cumres(e,y1~eta,R=1000)
g <- cumres(e,"y1",x=x,R=1000)
plot(g)
```

plot.cumres

Plot cumulative residuals from a 'cumres' object

Description

plot displays the observed cumulative residual process with realizations under the null. 95% prediction bands

Usage

```r
## S3 method for class 'cumres'
plot(x, idx = 1:length(x$variable),
     col = c("grey"), ci = TRUE, col.ci = "darkblue",
     col.alpha = 0.3, lty.ci = 0, level = 0.95,
     legend = c("type1", "type2", "none"), xlab, ylab,
     vs = TRUE, ylim = NULL, title, ...)
```
Arguments

- **x**: Object produced by the function `cumres`.
- **idx**: Vector of numbers (or variable names) indicating which processes from the `x` to plot.
- **col**: Color of the sample processes. By setting this parameter to "none" or NULL no realizations will be drawn. The number of realizations is determined by the `cumres`-object.
- **ci**: Type of prediction bands to plot. Defaults to none. Set to TRUE to obtain simultaneous prediction bands under the null (pointwise can be obtained by setting to "pointwise").
- **col.ci**: Color of prediction band.
- **col.alpha**: Degree of transparency (0-1) of the prediction bands.
- **lty.ci**: Line type of prediction band.
- **level**: The required prediction level.
- **legend**: Type of legend where "type1" gives p-values of GoF-tests and "type2" gives usual type of legends.
- **xlab**: Optional label of x-axis
- **ylab**: Optional label of y-axis
- **vs**: Label of predictor
- **ylim**: Range of y axis
- **title**: Main title
- **...**: Additional arguments passed to the plot-routine.

Author(s)

Klaus K. Holst <kkho@biostat.ku.dk>

See Also

cumres

Examples

```r
n <- 500; x <- abs(rnorm(n, sd=0.2))+0.01; y <- sqrt(x) + rnorm(n, sd=0.2)
l <- lm(y ~ x)
g <- cumres(l, R=500)
plot(g, idx=1, ci="sim", col=NULL, col.ci="purple", legend="type2")
```
**Surgical Unit Data**

**Description**

Surgical Unit Data used in the paper by Lin et al. (2002). Survival time and covariates for 54 patients undergoing liver surgery.

**Source**


**References**

Index

*Topic **datasets**
  - surgunit, 10
*Topic **hplot**
  - plot.cumres, 8
*Topic **models**
  - confint.cumres, 2
  - cumres.coxph, 3
  - cumres.glm, 5
  - cumres.lvmfit, 7
*Topic **package**
  - gof-package, 2
*Topic **regression**
  - confint.cumres, 2
  - cumres.coxph, 3
  - cumres.glm, 5
  - cumres.lvmfit, 7
  - plot.cumres, 8

confint.cumres, 2
cox.aalen, 2, 4, 6
coxph, 4
cumres, 3, 9
cumres (cumres.glm), 5
cumres.coxph, 3
cumres.glm, 4, 5
cumres.lvmfit, 7
gof (gof-package), 2
gof-package, 2
plot.cumres, 8
surgunit, 10