Package 'goftte'

November 30, 2017

Type Package

Title Goodness-of-Fit for Time-to-Event Data

Version 1.0.5

Date 2017-11-30

Author Patrick Sfumato and Jean-Marie Boher.

Maintainer Patrick Sfumato <sfumatop@gmail.com>

Description Extension of 'gof' package to survival models.

Depends survival, R (>= 3.0.3)

Suggests cmprsk, rms, gof

License GPL

LazyLoad yes

NeedsCompilation yes

Repository CRAN

Date/Publication 2017-11-30 09:26:03 UTC

R topics documented:

goftte-package *Omnibus tests to check the proportional regression models assumptions*

Description

Let's denote $Z_i = (Z_{i,k})_{1 \leq k \leq p}$ a matrix of p individual regression covariates, and $\widehat{\beta} = (\widehat{\beta}_k)_{1 \leq k \leq p}$ the vector of estimated regression coefficients assuming a Cox (1972) proportional hazards (PH) regression model or a Fine & Gray (1999) proportional subdistribution hazards (PSH) regression model. The package implements objective diagnotic tools to assess the key modeling assumptions using cumulative sums of model residuals of the form :

$$U(\widehat{\beta}, t, z) = \sum_{i=1}^{n} \int_0^t f(Z_i) 1\{Z_i \leq z\} d\widehat{M}_i(u).$$

Here $f(.)$ denotes a smooth function, z the covariates values and $\widehat{M}_i(t)$ the model residuals, so-called martingales residuals in Cox regression settings.

To test for PH or PSH assumptions, Lin & al. (1993) and Li & al. (2015) proposed test statistics obtained by letting $f(Z_i) = Z_i$ and $z \longrightarrow \infty$. In that case, the test statistics reduce to the supremum value of the score process $U(\beta, t)$ evaluated at $\beta = \widehat{\beta}$ (Kolmogorov-Smirnov type test statistic). With respect to covariate k, the statistic is written as follows :

$$KS = \sup_t \mid\mid U_k(\widehat{\beta}, t) \mid\mid$$

For checking the functional form of a covariate included in the model, same type of statistics were adopted by considering the process $U(\widehat{\beta}, z)$ obtained by letting $f(Z_i) = 1$ and $t \longrightarrow \infty$.

Null limiting distribution of cumulative sum process are obtained by drawing R independent realizations of an asymtpotically equivalent gaussian process (Monte-Carlo simulations). The significance level of KS test statistics were derived from the approximated null distribution.

Our package can be viewed as an extension of the gof package (Holst, 2014) allowing to check the functional form assumptions of covariates in the Cox model and to check Fine and Gray model assumptions. Besides this, it proposes goodness-of-fit tests for PH and PSH based on Cramer-Von Mises (CvM) and Anderson-Darling (AD) type test statistics as recommanded by Kvaloy (2004) :

$$CvM = \int_0^\infty U_k(\widehat{\beta}, t)^2 dq(t)$$

$$AD = \int_0^\infty \frac{U_k(\widehat{\beta}, t)^2}{q(t)(1 - q(t))} dq(t)$$

Where $q(t) = \frac{I_{k,k}(\widehat{\beta}, t)}{I_{k,k}(\widehat{\beta}, \infty)}$, and $I(\widehat{\beta}, t)$ denotes the consistent sample estimate of the score process covariance matrix.

Author(s)

Patrick Sfumato and Jean-Marie Boher

References

Cox DR (1972). *Regression models and life-tables.* Journal of the Royal Statistical Society. Series B (Methodological), 34(2), 187-220.

Fine JP and Gray RJ (1999). *A Proportional Hzards Model for the Subdistribution of Competing Risk.* American Statistical Association, 94(446), 496-509.

Lin DY, Wei JL and Ying Z (1993). *Checking the Cox model with cumulative sums of martingale-based residuals.* Biometrika, 80(3), 557-572.

Li J, Scheike TH and Zhang MJ (2015). *Checking Fine & Gray subditribution hazards model with cumulative sums of residuals.* Lifetime Data Analysis, 21(2), 197-217.

Holst KK (2014). *gof: Model-diagnostics based on cumulative residuals.* R package version 0.9.1, URL https://CRAN.R-project.org/package=gof.

Kvaloy JT and Neef LR (2004). *Tests for the Proportional Intensity Assumption Based on the Score Process.* Lifetime Data Analysis, 10(2), 139-157.

`fcov.coxph` *Checking the functional form assumption for Cox model*

Description

Diagnostics for the linear functional form of specific covariate(s) of the Cox model. The method is based on Lin's approximation of the randomly pertubed sum process (1993). Another approximation method recommended by Liu (2008) is also proposed. P-values are derived for supremum KS test statistics.

Usage

```
## S3 method for class 'coxph'
fcov(model, variable=NULL, type.test=c("Lin"), R=1000, plots=min(R,50),
seed=NULL,...)
```

Arguments

`model`	Model object (`coxph`).
`variable`	Vector corresponding to the labels of each covariate. This is also an output argument.
`type.test`	Type of approximation. Values are "Lin" or "Liu". Default is "Lin".
`R`	Generation number used for Monte-Carlo simulations. This is also an output argument.
`plots`	Realizations number of Monte-Carlo simulations to save for use in the plot-routine.
`seed`	Random seed.
`...`	additional arguments.

Details

A fcov.cph function accepting models of cph type (rms package) is also included in the package, and uses the same arguments as coxph. Contrary to fcov.coxph, the use of 'x=TRUE' option for covariates design is now required in cph model. Strata and cluster options are not supported.

Value

Returns an object of class 'scproc'. The main items of this object are :

obs	l x p matrix of unique covariates values for functional form. l is the maximum number of unique observations between the p covariates.
W	The process $U(\widehat{\beta}, z)$.
What	The simulated limiting processes for the R-plots first Monte-Carlo realizations.
sdw	Standard error over covariates values of What.
cvalues	R x p matrix whose components are the supremum of the standardized What process for each Monte-Carlo realization . The quantiles of this output argument are used to calculate the prediction bands in the plot-routine.
KS	Vector of the p rejection probabilities using KS type statistic.

Author(s)

Patrick Sfumato and Jean-Marie Boher.

References

Lin DY, Wei JL and Ying Z (1993).*Checking the Cox model with cumulative sums of martingale-based residuals.* Biometrika, 80(3), 557-572.

Liu M, Lu W and Shao (2008). *A Monte Carlo approach for change-point detection in the Cox proportional hazards model.* Statistics in Medecine, 27(19), 3894-3909.

Examples

```
require(survival)

#Simulating survival data
simcox <- function(n=100, seed=1) {
  if (!is.null(seed))
  set.seed(seed)
  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);

#Fitting Cox model
fit.coxph <- coxph(Surv(time,status)~ X1 + X2, data=d)
```

```
#Checking the covariates functional form assumption
fcov(fit.coxph)
```

`fcov.crr` *Checking the functional form assumption for Fine & Gray model*

Description

Diagnostics for the linear functional form of specific covariate(s) of the Fine & Gray model. Similarly to Li (2015), we extend the Lin's approximation method to take into account competing events. We also provide an adaptation of Liu's approximation method (2008) for Fine & Gray models. P-values are derived for supremum KS test statistics.

Usage

```
## S3 method for class 'crr'
fcov(model, fstatus, ftime, cov1, cencode=0, failcode=1,  type.test=c("Lin"), R=1000,
plots=min(R,50),  seed=NULL, variable=NULL,...)
```

Arguments

`model`	Model object (`crr`).
`fstatus`	Vector corresponding to the failures of the n patients.
`ftime`	Vector corresponding to the failure times.
`cov1`	Matrix whose columns consist in the components of the p covariates.
`cencode`	Censoring code.
`failcode`	Interest event code. All the failures differing from cencode and faildcode are considered as competing events.
`type.test`	Type of approximation. Values are "Lin" or "Liu". Default is "Lin".
`R`	Generation number used for Monte-Carlo simulations. This is also an output argument.
`plots`	Realizations number of Monte-Carlo simulations to save for use in the plot-routine.
`seed`	Random seed.
`variable`	Vector corresponding to the labels of each covariate. This is also an output argument.
`...`	additional arguments.

Value

Returns an object of class 'scproc'. The main items of this object are :

obs	l x p matrix of unique covariates values. l is the maximum number of unique observation between the p covariates.
W	The process $U(\widehat{\beta}, z)$ adapted to Fine & Gray model.
What	The simulated limiting processes for the R-plots first Monte-Carlo realizations.
sdw	Standard error over covariates values of What.
cvalues	R x p matrix whose components are the supremum of the standardized What process for each Monte-Carlo realization. The quantiles of this output argument are used to calculate the prediction bands in the plot-routine.
KS	Vector of the p rejection probabilities using KS type statistic.

Author(s)

Patrick Sfumato and Jean-Marie Boher.

References

Li J, Scheike TH and Zhang MJ (2015). *Checking Fine & Gray subditribution hazards model with cumulative sums of residuals.* Lifetime Data Analysis, 21(2), 197-217.

Lin DY, Wei JL and Ying Z (1993).*Checking the Cox model with cumulative sums of martingale-based residuals.* Biometrika, 80(3), 557-572.

Liu M, Lu W and Shao (2008). *A Monte Carlo approach for change-point detection in the Cox proportional hazards model.* Statistics in Medecine, 27(19), 3894-3909.

Examples

```
require(cmprsk)

#Simulating survival data with competing events
set.seed(10)
ftime <- rexp(200)
fstatus <- sample(0:2,200,replace=TRUE)
cov <- matrix(runif(200),nrow=200)

# Fine & Gray regression
fit.crr <- crr(ftime,fstatus,cov)

#Checking the covariates functional form assumption
fcov(model=fit.crr, ftime=ftime,fstatus=fstatus,cov1=cov)
```

`plot.scproc` *Graphical model diagnostic*

Description

Produce a graphical model diagnostic, with the possibility of adding the prediction bands calculated by `predband.scproc`.

Usage

```
## S3 method for class 'scproc'
plot(x, idx=1:length(x$variable), col=c("grey"), ci=FALSE,
                      col.ci="darkblue", col.alpha=0.3, lty.ci=0, level=0.95,
                      legend=c("type1","type2","none"), xlab=NULL, ylab=NULL,
                    ylim=NULL, xlim=NULL, title=NULL, cex.lab=1,cex.main=1,...)
```

Arguments

`x`	An object of class `scproc`.
`idx`	The index number of covariates whose the plot is required.
`col`	Boolean argument controlling for the plot of prediction bands. Values are TRUE or FALSE.
`ci`	Color of the processes realizations.
`col.ci`	Color of the prediction bands.
`col.alpha`	Color intensity of the prediction bands.
`lty.ci`	Line thickness of the prediction bands border.
`level`	The required level for prediction bands.
`legend`	Argument controling the type of legend on the plot. "type1" for p-values, "type2" for usual type of legends and "none" for no legend.
`xlab`	Label of the x-axis.
`ylab`	Label of the y-axis.
`xlim`	Vector of two components which indicates the extremum values of x-axis.
`ylim`	Vector of two components which indicates the extremum values of y-axis.
`title`	Vector of length idx whose components are titles of plots.
`cex.lab`	Magnification of x and y labels relative to cex.
`cex.main`	Magnification of titles relative to cex.
`...`	additional arguments.

Details

This function is derived from plot.cumres of gof package.

Author(s)

Patrick Sfumato and Jean-Marie Boher.

References

Holst KK (2014). *gof: Model-diagnostics based on cumulative residuals.* R package version 0.9.1, URL https://CRAN.R-project.org/package=gof.

Examples

```
require(cmprsk)

#Simulating survival data with competing events
set.seed(10)
ftime <- rexp(200)
fstatus <- sample(0:2,200,replace=TRUE)
cov <- matrix(runif(200),nrow=200)

# Fine & Gray regression
fit.crr <- crr(ftime,fstatus,cov)

#Checking the covariates functional form assumption
k<-fcov(model=fit.crr, ftime=ftime,fstatus=fstatus,cov1=cov)

#Plotting a graphical model diagnostic
plot(k)
```

---

predband.scproc *Prediction bands for 'scproc' object*

Description

Calculates the prediction bands under the null hypothesis.

Usage

```
## S3 method for class 'scproc'
predband(x, idx=1:length(x$variable), level=0.95, cval=NULL, ...)
```

Arguments

x	An object of class scproc.
idx	The index number of covariates whose the prediction band is calculated.
level	The required level for prediction bands.
cval	Parameter used to calculate the symmetric prediction bands defined by the standard error multiplied by cval.
...	additional arguments.

Details

This function is derived from confint.cumres of gof package.

Value

t	Ordered values of the variable used : unique times for proportionality and unique observations of covariates for functional form.
yu	Upper simultaneous confidence limit.

Author(s)

Patrick Sfumato and Jean-Marie Boher.

References

Holst KK (2014). *gof: Model-diagnostics based on cumulative residuals.* R package version 0.9.1, URL https://CRAN.R-project.org/package=gof.

Examples

```
require(cmprsk)

#Simulating survival data with competing events
set.seed(10)
ftime <- rexp(200)
fstatus <- sample(0:2,200,replace=TRUE)
cov <- matrix(runif(200),nrow=200)

# Fine & Gray regression
fit.crr <- crr(ftime,fstatus,cov)

#Checking the covariates functional form assumption
k<-fcov(model=fit.crr, ftime=ftime,fstatus=fstatus,cov1=cov)

#Prediction bands
predband(k)
```

---

print.scproc *Printing method for 'scproc' objects*

Description

Produces a printed summary of assumption diagnostics.

Usage

```
## S3 method for class 'scproc'
print(x, idx=NULL, ...)
```

Arguments

x	An object of class scproc.
idx	The index number of covariates whose print is required.
...	additional arguments.

Details

This function is derived from print.cumres of gof package.

Author(s)

Patrick Sfumato and Jean-Marie Boher.

References

Holst KK (2014). *gof: Model-diagnostics based on cumulative residuals.* R package version 0.9.1, URL https://CRAN.R-project.org/package=gof.

Examples

```
require(cmprsk)

#Simulating survival data with competing events
set.seed(10)
ftime <- rexp(200)
fstatus <- sample(0:2,200,replace=TRUE)
cov <- matrix(runif(200),nrow=200)

# Fine & Gray regression
fit.crr <- crr(ftime,fstatus,cov)

#Checking the covariates functional form assumption
k<-fcov(model=fit.crr, ftime=ftime,fstatus=fstatus,cov1=cov)

#Summary of covariates functional form assumption diagnostics
print(k)
```

---

prop.coxph *Checking proportional hazards assumption for Cox model*

Description

Diagnostics for proportional hazards of specific covariate(s) of the Cox model. The limiting null distribution of the score process is approximated using either Lin's method (1993) or Liu's (2008). P-values are derived for KS, CvM and AD statistics.

Usage

```
## S3 method for class 'coxph'
prop(model, variable=NULL, type.test=c("Lin"), R=1000, plots=min(R,50), seed=NULL,...)
```

Arguments

model	Model object (coxph).
variable	Vector corresponding to the labels of each covariate. This is also an output argument.
type.test	Type of approximation. Values are "Lin" or "Liu". Default is "Lin".
R	Generation number used for Monte-Carlo simulations. This is also an output argument.
plots	Realizations number of Monte-Carlo simulations to save for use in the plot-routine.
seed	Random seed.
...	additional arguments.

Details

- The same type of function was proposed in gof package (cumres.coxph) at the difference that the autors only used Lin's method to approximate the limiting null distribution of the score process.

- A prop.cph function accepting models of cph type (rms package) is also included in the package, and uses the same arguments as prop.coxph. Contrary to prop.coxph, the use of 'x=TRUE' option for covariates design is now required in cph model. Strata and cluster options are not supported.

Value

Returns an object of class 'scproc'. The main items of this object are :

obs	m x p matrix of unique times. m is the length of unique times.
W	The process $U(\widehat{\beta}, t)$.
What	The simulated limiting processes for the R-plots first Monte-Carlo realizations.
sdw	Standard error over time of What.
cvalues	R x p matrix whose components are the supremum of the standardized What process for each Monte-Carlo realization . The quantiles of this output argument are used to calculate the prediction bands in the plot-routine.
KS	Vector of the p rejection probabilities using KS type statistic.
CvM	Vector of the p rejection probabilities using CvM type statistic.
AD	Vector of the p rejection probabilities using AD type statistic.

Author(s)

Patrick Sfumato and Jean-Marie Boher.

References

Holst KK (2014). *gof: Model-diagnostics based on cumulative residuals.* R package version 0.9.1, URL https://CRAN.R-project.org/package=gof.

Lin DY, Wei JL and Ying Z (1993).*Checking the Cox model with cumulative sums of martingale-based residuals.* Biometrika, 80(3), 557-572.

Liu M, Lu W and Shao (2008). *A Monte Carlo approach for change-point detection in the Cox proportional hazards model.* Statistics in Medecine, 27(19), 3894-3909.

Examples

```
require(survival)

#Simulating survival data
simcox <- function(n=100, seed=1) {
  if (!is.null(seed))
  set.seed(seed)
  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);

#Fitting Cox's model
fit.coxph <- coxph(Surv(time,status)~ X1 + X2, data=d)

#Checking the proportional hazards assumption
prop(fit.coxph)
```

`prop.crr` *Checking proportional subdistribution hazard assumption for Fine & Gray model*

Description

Diagnostics for the proportional subdistribution hazards of specific covariate(s) of the Fine & Gray model. Similarly to Li (2015), the limiting null distribution of the score process is approximated by extending Lin's method (1993) to take into account competing events. An adaptation of Liu's approximation method (2008) for Fine & Gray model is also provided. P-values are derived for KS, CvM and AD statistics.

Usage

```
  ## S3 method for class 'crr'
prop(model, fstatus, ftime, cov1, cencode=0, failcode=1, variable=NULL,
  type.test=c("Liu"), R=1000, plots=min(R,50), seed=NULL,...)
```

Arguments

model	Model object (crr).
fstatus	Vector corresponding to the failures of the n patients.
ftime	Vector corresponding to the failure times.
cov1	matrix whose columns consist in the components of the p covariates.
cencode	Censoring code.
failcode	Interest event code. All the failures differing from cencode and failcode are considered as competing events.
type.test	Type of approximation. Values are "Lin" or "Liu". Default is "Liu".
R	Generation number used for Monte-Carlo simulations. This is also an output argument.
plots	Realizations number of Monte-Carlo simulations to save for use in the plot-routine.
seed	Random seed.
variable	Vector corresponding to the labels of each covariate. This is also an output argument.
...	additional arguments.

Value

Returns an object of class 'scproc'. The main items of this object are :

obs	m x p matrix of unique times. m is the length of unique times.
W	The process $U(\widehat{\beta}, t)$ adapted to Fine & Gray model.
What	The simulated limiting processes for the R-plots first Monte-Carlo realizations.
sdw	Standard error over time of What.
cvalues	R x p matrix whose components are the supremum of the standardized What process for each Monte-Carlo realization . The quantiles of this output argument are used to calculate the prediction bands in the plot-routine.
KS	Vector of the p rejection probabilities using KS type statistic.
CvM	Vector of the p rejection probabilities using CvM type statistic.
AD	Vector of the p rejection probabilities using AD type statistic.

Author(s)

Patrick Sfumato and Jean-Marie Boher.

References

Li J, Scheike TH and Zhang MJ (2015). *Checking Fine & Gray subditribution hazards model with cumulative sums of residuals.* Lifetime Data Analysis, 21(2), 197-217.

Lin DY, Wei JL and Ying Z (1993).*Checking the Cox model with cumulative sums of martingale-based residuals.* Biometrika, 80(3), 557-572.

Liu M, Lu W and Shao (2008). *A Monte Carlo approach for change-point detection in the Cox proportional hazards model.* Statistics in Medecine, 27(19), 3894-3909.

Examples

```
require(cmprsk)

#Simulating survival data with competing events
set.seed(10)
ftime <- rexp(200)
fstatus <- sample(0:2,200,replace=TRUE)
cov <- matrix(runif(200),nrow=200)

# Fine & Gray regression
fit.crr <- crr(ftime,fstatus,cov)

#Checking the proportional subdistribution hazards assumption
prop(model=fit.crr, ftime=ftime,fstatus=fstatus,cov1=cov)
```

Index