Package 'coxrobust'

February 9, 2012

Version 1.0

Depends R (>= 2.0.0), survival

Date 2006-03-05

Title Robust Estimation in Cox Model

LazyLoad Yes

Description Fit robustly proportional hazards regression model

Author Tadeusz Bednarski <t.bednarski@prawo.uni.wroc.pl>, Filip Borowicz <f.borowicz@prawo.uni.wroc.pl>

Maintainer Filip Borowicz <f.borowicz@prawo.uni.wroc.pl>

License GPL (>= 2)

Repository CRAN

Date/Publication 2006-03-15 09:12:51

R topics documented:

coxr *Fit Robustly Proportional Hazards Regression Model*

Description

Fits efficiently and robustly Cox proportional hazards regression model in its basic form, where explanatory variables are time independent with one event per subject. Method is based on a smooth modification of the partial likelihood.

Usage

```
coxr(formula, data, subset, na.action, trunc = 0.95,
     f.weight = c("linear", "quadratic", "exponential"),
     singular.ok = TRUE, model = FALSE)
```

Arguments

formula	a formula object, with the response on the left of a ~ operator, and the terms on the right. The response must be a survival object as returned by the Surv function.
data	a data frame in which to interpret the variables named in the formula, or in the subset.
subset	expression saying that only a subset of the rows of the data should be used in the fit.
na.action	a missing-data filter function, applied to the model.frame, after any subset argument has been used.
trunc	roughtly, quantile of the sample $T_i exp(\beta' Z_i)$, it determines the trimming level for the robust estimator
f.weight	type of weighting function, default is "quadratic"
singular.ok	logical value indicating how to handle collinearity in the model matrix. If TRUE, the program will automatically skip over columns of the X matrix that are linear combinations of earlier columns. In this case the coefficients for such columns will be NA, and the variance matrix will contain zeros. For ancillary calculations, such as the linear predictor, the missing coefficients are treated as zeros.
model	a logical value indicating whether model frame should be included as a component of the returned value.

Details

The method consists in maximization of an objective function which is a smooth modification of the partial likelihood. Observations with excessive values of $\Lambda(T)exp(\beta' Z)$, where Λ is the cumulated hazard, β vector of parameters, Z explanatory variables and T possibly censored survival time, are down-weighted. Both Λ and β are iteratively robustly estimated.

Numerical results are supported by a graphical tool `plot`, which in a series of 5 graphs let us compare how well data are explained by the estimated proportional hazards model with non-robust (black color) and robust method (green color). The first graph shows standardized difference of two estimated survival functions; one via the Cox model and the other via Kaplan Meier estimator. The following four graphs show the same differences for four strata, defined by the quartiles of the estimated linear predictor. Comparison of estimation results along with analysis of the graphs leads frequently to a very detailed information about the model fit (see examples).

Value

An object of class `coxr`. See `coxr.object` for details.

References

Bednarski, T. (1993). Robust estimation in Cox's regression model. Scandinavian Journal of Statistics. Vol. 20, 213–225.

Bednarski, T. (1989). On sensitivity of Cox's estimator. Statistics and Decisions. 7, 215–228.

Grzegorek, K.(1993). On robust estimation of baseline hazard under the Cox model and via Frechet differentiability. Preprint of the Institute of Mathematics of the Polish Academy of Sciences.518.

Minder, C.E. & Bednarski, T. (1996). A robust method for proportional hazards regression. Statistics in Medicine Vol. 15, 1033–1047.

See Also

`coxph`, `coxr.object`, `gen_data`, `plot.coxr`, `Surv`

Examples

```
# Create a simple test data set using the attached function gen_data
a <- gen_data(200, c(1, 0.1, 2), cont = 0.05, p.censor = 0.30)
result <- coxr(Surv(time, status) ~ X1 + X2 + X3, data = a , trunc = 0.9)
result
plot(result)

#use the lung cancer data at Mayo Clinic to
#compare results of non-robust and robust estimation
result <- coxr(Surv(time, status) ~ age + sex + ph.karno + meal.cal + wt.loss, data = lung)
result
plot(result)

#use the Veteran's Administration Lung Cancer Data
#to compare results of non-robust and robust estimation
result <- coxr(Surv(time,status) ~ age + trt + celltype + karno + diagtime + prior, data = veteran)
result
plot(result)
```

coxr.object *Fit Robustly Proportional Hazards Regression Object*

Description

This class of objects is returned by coxr function to represent efficiently and robustly fitted proportional hazards regression model. Objects of this class have methods for the functions print, plot and predict.

Value

The following components must be included in a legitimate coxr object.

coefficients	robust estimate of the regression parameter.
ple.coefficients	non-robust (efficient) estimate of the regression parameter.
var	an approximate variance matrix of the coefficients (estimated robustly). Rows and columns corresponding to any missing coefficients are set to zero.
ple.var	an approximate variance matrix of the coefficients (estimated non-robustly). Rows and columns corresponding to any missing coefficients are set to zero.
lambda	cumulated hazard (estimated robustly).
lambda.ple	cumulated hazard (estimated non-robustly).
wald.test	the value of Wald test.
ewald.test	the value of extended Wald test.
skip	skipped columns.
na.action	the na.action attribute, if any, that was returned by the na.action routine.

The object also contain the following, for documentation see the lm object: terms, call, x, y and optionally model.

See Also

coxr

gen_data *Generate Data from the Proportional Hazards Regression Model*

Description

Generates data set from the proportional hazards regression model without or with contamination.

Usage

```
gen_data(n, beta, cont = 0, p.censor = 0)
```

Arguments

n	number of observations.
beta	vector of regression coefficients.
cont	fraction of contaminated observations.
p.censor	probability of censoring.

Details

Covariates are generated independently, each from the standard normal distribution. Baseline hazard is equal to 1. After generation of survival times, covariates are replaced by independent $2N(0,1)+1$ variables in fraction cont of observations.

Value

Data frame containing the following variables:

time	vector of survival times.
status	vector of censoring status.
X1, X2, ...	explanatory variables (their number is determined by the dimension of vector of regression coefficients).

See Also

coxph, coxr

Examples

```
gen_data(50, c(2,-2), cont = 0.05)
```

plot.coxr *Plot Diagnostics for a coxr Object*

Description

Graphical tool which in a series of 5 graphs let us compare how well data are explained by the estimated proportional hazards model with non-robust (black color) and robust method (green color). The first graph gives standardized difference of two estimated survival functions; one via the Cox model and the other via Kaplan Meier estimator. The following four graphs show the same differences for four strata, defined by the quartiles of the estimated linear predictor. Comparison of estimation results along with analysis of the graphs leads frequently to a very detailed information about the model fit (see examples).

Usage

```
## S3 method for class 'coxr'
plot(x, caption = c("Full data set", "First quartile",
    "Second quartile", "Third quartile", "Fourth quartile"), main = NULL,
    xlab = "log time", ylab = "standardized survival differences", ...,
    color = TRUE)
```

Arguments

x	coxr object, typically result of coxr.
caption	captions to appear above the plots.
xlab	title for the x axis.
ylab	title for the y axis.
main	overall title for the plot.
...	other parameters to be passed through to plotting functions.
color	if FALSE grayscale mode is used.

See Also

coxr

Examples

```
#use the lung cancer data at Mayo Clinic
#to compare results of non-robust and robust estimation
result <- coxr(Surv(time, status) ~ age + sex + ph.karno + meal.cal + wt.loss, data = lung)
plot(result, main = "Mayo Clinic Lung Cancer Data")
```

Index