

Package ‘coxphw’

June 5, 2009

Type Package

Title Weighted Cox regression

Version 1.3

Date 2009-06-04

Author R by Meinhard Ploner, Fortran by Georg Heinze

Maintainer Georg Heinze <georg.heinze@meduniwien.ac.at>

Depends R (>= 2.4.0), survival

Description Weighted estimation for Cox regression

License GPL

Repository CRAN

Date/Publication 2009-06-05 13:40:34

R topics documented:

coxphw	1
plotw	7
Index	8

 coxphw

Weighted estimation in Cox regression

Description

This package implements weighted estimation in Cox regression, as proposed by Schemper (1992) and Sasieni (1993). Weighted estimation provides unbiased average hazard ratio estimates in case of non-proportional hazards (Schemper, Wakounig and Heinze (2009)).

Usage

```
coxphw(formula,
  data=parent.frame(),
  breslow=NA,
  prentice=NA,
  taroneware=NA,
  id=NULL,
  robust=FALSE,
  jack=FALSE,
  normalize=TRUE,
  scale.weights=1,
  offset=NULL,
  alpha=0.05,
  maxit=50,
  maxhs=5,
  epsilon=1e-6,
  maxstep=2.5,
  censcorr=FALSE,
  x=TRUE,
  trunc.weights=1,
  ...)
```

Arguments

formula	a formula object, with the response on the left of the operator, and the model 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' argument.
breslow	a righthand formula with the terms which should be estimated using Breslow (N at risk) weights
prentice	a righthand formula with the terms which should be estimated using Prentice (survival function) weights
taroneware	a righthand formula with the terms which should be estimated using Tarone-Ware (square of N at risk) weights
normalize	if T, weights are normalized such that their sum is equal to the number of events. May speed up or enable convergence if for some variables no weighting is used.

alpha	the significance level ($1-\alpha$ = the confidence level), 0.05 as default.
maxit	maximum number of iterations (default value is 50)
maxhs	maximum number of step-halvings per iterations (default value is 5). The increments of the parameter vector in one Newton-Rhaphson iteration step are halved, unless the new likelihood is greater than the old one, maximally doing maxhs halvings.
epsilon	specifies the maximum allowed change in penalized log likelihood to declare convergence. Default value is 0.0001.
maxstep	specifies the maximum change of (standardized) parameter values allowed in one iteration. Default value is 2.5.
id	a vector of patient identification numbers, must be integers starting from 1. These IDs are used for computing the robust covariance matrix. If id=NA (the default) the program assumes that each line of the data set refers to a distinct individual.
robust	set to TRUE if the robust variance estimate should be computed. This robust estimate is computed from $V=D'D$ where D is the matrix of dfbeta residuals which are computed by $D'D = A^{-1}U'UA^{-1}$ with A denoting the weighted sum of contributions to the second derivative of the log likelihood and U denoting the nxp matrix of score residuals.
jack	set to TRUE if the variance should be based on a complete jackknife. Each individual (as identified by distinct values of the variable specified in id) is left out in turn. The resulting matrix of dfbeta residuals D is then used to compute the variance matrix: $V=D'D$.
offset	specifies a variable which is included in the model but its parameter estimate is fixed at 1.
scale.weights	specifies a scaling factor for the weights
censcorr	If set to TRUE, the weights are multiplied by the inverse marginal Kaplan-Meier estimator with reverse status indicator (default=F).
x	If set to TRUE, adds the model matrix to the output object.
trunc.weights	Specifies quantile for weight truncation (default=1=no truncation). Weight truncation (restricting the weight values to some maximum value) may improve efficiency of estimates. Use values of 0.99 for slight truncation, 0.95 for stronger truncation.
...	additional parameters

Details

If Cox's proportional hazards regression model is used in the presence of non-proportional hazards, i.e., with underlying time-dependent hazard ratios of prognostic factors, the average relative risk for such a factor is under- or overestimated and testing power for the corresponding regression parameter is reduced. In such a situation weighted estimation of this parameter provides a parsimonious alternative to more elaborate modelling of time-dependent effects. Weighted estimation in Cox regression (WCR) extends the tests by Breslow and Prentice to a multi-covariate situation as does the

Cox model to Mantel's logrank test. WCR can also be seen as a robust alternative to the standard Cox estimator, reducing the influence of outlying survival times on parameter estimates. Schemper (1992) first demonstrated the suitability of WCR for estimating average hazard ratios when hazards are non-proportional and Sasieni (1993) extensively investigated the favorable properties of WCR.

Weighted estimation assigns weights to risk sets, according to the number of subjects at risk (Breslow weights), according to their square roots (Tarone-Ware weights) or according to the survival function estimates (Prentice weights). These weights are applied to the summands of the score function. The final estimate is the vector of parameter values which equates the score function to 0. Since there is one score function corresponding to each parameter of the model, weights may be applied to some but not necessarily to all parameters of a model.

Breslow's tie-handling method is used by the program, other methods to handle ties are currently not available.

By default, the program estimates the covariance matrix using Lin (1991) and Sasieni (1993) sandwich estimate $A^{-1}BA^{-1}$ with $-A$ and $-B$ denoting the sum of contributions to the second derivative of the log likelihood, weighted by $w(t_j)$ and $w(t_j)^2$, respectively. This estimate is independent from the scaling of the weights and reduces to the inverse of the information matrix in case of no weighting.

In case of censoring Schemper, Wakounig and Heinze (2009) proposed to multiply the weights by a correction which particularly upweights late risk sets by the inverse "follow-up" Kaplan-Meier estimate. Using this correction and Prentice weighting one obtains an average hazard ratio which is a consistent estimator of the odds of concordance: $Pr(T_1 > T_0)/Pr(T_1 < T_0)$, where T_1 and T_0 are the survival times of two randomly drawn subjects that differ in the independent variable by one unit. The weighting by the number of subjects at risk (Breslow weights) might be preferred if robust estimation is desired. In case of mixed weighting (weights only used for model terms that exhibit non-proportional hazards), the jackknife estimate of the variance should be used. Therefore, we recommend the following settings:

Estimation of a representative average hazard ratio if there are censored survival times:

```
prentice=~<all model terms>, censcorr=T, robust=T.
```

Robust estimation:

```
breslow=~<all model terms>, censcorr=F, robust=T.
```

Mixed weighting (proportional hazards assumed for some but not all model terms):

```
prentice=~<non-ph terms>, censcorr=T, jack=T.
```

Value

<code>coefficients</code>	the parameter estimates
<code>alpha</code>	the significance level = 1 - confidence level
<code>var</code>	the estimated covariance matrix
<code>df</code>	the degrees of freedom
<code>method.ties</code>	the ties handling method
<code>iter</code>	the number of iterations needed to converge
<code>n</code>	the number of observations
<code>y</code>	the response

formula	the model formula
means	the means of the covariates
linear.predictors	the linear predictors
method	the estimation method (usually weighted estimation)
method.ci	the confidence interval estimation method (Profile Likelihood or Wald)
ci.lower	the lower confidence limits
ci.upper	the upper confidence limits
prob	the p-values
call	the function call
dfbeta.resid	the dfbeta residuals D , only computed if robust=T, or if jack=T. For jack=T, the dfbeta residuals are the change in parameter estimates if each individual in turn is left out. For robust=T, the dfbeta residuals are computed via UA^{-1} with U and A denoting the matrix of score residuals and minus the weighted sum of contributions to the second derivative of the likelihood. If jack=T or robust=T, the covariance matrix of the parameter estimates is based on $D'D$.
cov.ls	the covariance matrix computed by the Lin-Sasieni method (the default method)
cov.lw	the covariance matrix computed by the Lin-Wei method (robust covariance, only computed if robust==T)
cov.j	the covariance matrix computed by the jackknife method (only computed if jack==T)
cov.method	the method used to compute the (displayed) covariance matrix and the standard errors. This method is either "jack" if jack==T, or "Lin-Wei" if robust==T and jack==F, or "Lin-Sasieni" if robust==F and jack==F.
w.matrix	A matrix with 4 columns and rows according to the number of uncensored failure times. The first column contains the failure times, the remaining columns (labeled w.raw, w.obskm, and w) contain the raw weights, the weights according to the inverse of the Kaplan-Meier estimates with reverse status indicator and the normalized product of both.

Note

A SAS macro WCM with similar functionality is offered for download at:

<http://www.meduniwien.ac.at/msi/biometrie>

Author(s)

Georg Heinze and Meinhard Ploner

References

- Lin D and Wei L (1989). The robust inference for the Cox proportional hazards model. *Journal of the American Statistical Association* 84, 1074-1078.
- Lin D (1991). Goodness-of-fit analysis for the Cox regression model based on a class of parameter estimators. *Journal of the American Statistical Association* 86, 725-728.


```

    ))

# weighted estimation, prentice (survival curve, Kaplan-Meier) weights

fit2<-coxphw(data=gastric, Surv(time,status)~treat, prentice= ~treat, robust=TRUE, censcorr=
# equivalent: fit2<-coxphw(data=gastric, Surv(time,status)~treat, km= ~treat)

summary(fit2)
fit2$cov.lw
fit2$cov.ls
fit2$cov.j
plotw(fit2)

# weighted estimation, breslow (N at risk) weights

fit3<-coxphw(data=gastric, Surv(time,status)~treat, breslow= ~treat)
# equivalent: fit2<-coxphw(data=gastric, Surv(time,status)~treat, km= ~treat)

summary(fit3)
fit3$cov.lw
fit3$cov.ls
fit3$cov.j

```

plotw

Plot weights of weighted estimation in Cox regression

Description

This function plots the weights used in a weighted Cox regression analysis against time.

Usage

```
plotw(object, rank=F, log=F, ...)
```

Arguments

object	a coxphw object
rank	if set to TRUE, plots the weights against ranked time (default=F)
log	if set to TRUE, shows logarithm of weights (default=F)
...	additional parameters

Details

The function plots the raw, inverse-censoring-Kaplan-Meier, and normalized total weights against (ranked) time.

Author(s)

Georg Heinze

See Also

coxphw

Index

*Topic **survival**

coxphw, 1

plotw, 7

coxphw, 1

plotw, 7