

Package ‘concreg’

July 2, 2014

Type Package

Title Concordance regression

Version 0.5

Date 2013-04-05

Author Georg Heinze, Meinhard Ploner, Daniela Dunkler

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

Depends R (>= 2.15.1), survival

Description Concordance regression

License GPL

NeedsCompilation yes

Repository CRAN

Date/Publication 2013-04-05 16:26:43

R topics documented:

cindex	2
coef.concreg	3
concreg	5
confint.concreg	9
plotw	11
vcov.concreg	12
Index	14

cindex	<i>Method to extract the estimated generalized c-index from a concreg object</i>
--------	--

Description

This method returns the estimated generalized c-indices by transforming the regression coefficients from a concreg object.

Usage

```
cindex(object, confint=FALSE, level=0.95)
```

Arguments

object	a concreg object
confint	if TRUE, request confidence interval(s) for c-indices.
level	if confint==TRUE, specifies confidence level.

Details

In case of a regression analysis with one explanatory variable, and with npar=TRUE, this returns the classical nonparametric c-index: $P(Y_i < Y_j | X_i > X_j)$, where i and j are all possible pairs of individuals.

In case of a multivariable regression, it will return the partial c-index for each variable X, conditional on the covariates Z: $P(Y_i < Y_j | X_i > X_j \& Z_i == Z_j)$.

In case that npar=FALSE, it will return the generalized (partial) concordance index as defined by Dunkler et al, Bioinformatics 2010: $P(Y_i < Y_j | X_i - X_j == 1)$.

Value

A pxp covariance matrix, where p is the number of regression coefficients.

Author(s)

Georg Heinze

References

Dunkler D, Schemper M and Heinze G (2010). Gene selection in microarray survival studies under possibly non-proportional hazards. *Bioinformatics* 26, 784-790.

See Also

coef.concreg confint.concreg

Examples

```

gastric <-
  structure(list(patnr = as.integer(c(46, 1, 2, 3, 4, 5, 47, 6,
    7, 8, 9, 48, 10, 11, 49, 12, 13, 14, 50, 15, 16, 17, 18, 19,
    20, 51, 21, 22, 52, 23, 53, 54, 55, 24, 25, 56, 57, 58, 59, 60,
    61, 62, 63, 64, 26, 65, 27, 66, 28, 29, 67, 68, 69, 70, 30, 71,
    31, 72, 32, 73, 33, 34, 74, 75, 76, 77, 78, 35, 79, 36, 80, 81,
    82, 37, 38, 39, 83, 84, 40, 85, 41, 86, 87, 88, 42, 43, 44, 89,
    90, 45)),
    treat = as.integer(c(0, 1, 1, 1, 1, 1, 0, 1, 1, 1,
    1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0,
    0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0,
    0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0,
    1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1)),
    time = as.integer(c(1,
    17, 42, 44, 48, 60, 63, 72, 74, 95, 103, 105, 108, 122, 125,
    144, 167, 170, 182, 183, 185, 193, 195, 197, 208, 216, 234, 235,
    250, 254, 262, 301, 301, 307, 315, 342, 354, 356, 358, 380, 383,
    383, 388, 394, 401, 408, 445, 460, 464, 484, 489, 499, 523, 524,
    528, 535, 542, 562, 567, 569, 577, 580, 675, 676, 748, 778, 786,
    795, 797, 855, 955, 968, 977, 1174, 1214, 1232, 1245, 1271, 1366,
    1420, 1455, 1460, 1516, 1551, 1585, 1622, 1626, 1690, 1694, 1736
    )),
    status = as.integer(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0))),
  .Names = c("patnr",
    "treat", "time", "status"), class = "data.frame",
  row.names = c("1",
    "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13",
    "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",
    "25", "26", "27", "28", "29", "30", "31", "32", "33", "34", "35",
    "36", "37", "38", "39", "40", "41", "42", "43", "44", "45", "46",
    "47", "48", "49", "50", "51", "52", "53", "54", "55", "56", "57",
    "58", "59", "60", "61", "62", "63", "64", "65", "66", "67", "68",
    "69", "70", "71", "72", "73", "74", "75", "76", "77", "78", "79",
    "80", "81", "82", "83", "84", "85", "86", "87", "88", "89", "90"
  ))

fit<-concreg(data=gastric, Surv(time,status)~treat, npar=TRUE)
cindex(fit, confint=TRUE) ### nonparametric c-index with 95% confidence interval

```



```

      1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
      1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0,
      0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0)))
.Names = c("patnr",
  "treat", "time", "status"), class = "data.frame",
row.names = c("1",
  "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13",
  "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",
  "25", "26", "27", "28", "29", "30", "31", "32", "33", "34", "35",
  "36", "37", "38", "39", "40", "41", "42", "43", "44", "45", "46",
  "47", "48", "49", "50", "51", "52", "53", "54", "55", "56", "57",
  "58", "59", "60", "61", "62", "63", "64", "65", "66", "67", "68",
  "69", "70", "71", "72", "73", "74", "75", "76", "77", "78", "79",
  "80", "81", "82", "83", "84", "85", "86", "87", "88", "89", "90"
))

```

```

fit<-concreg(data=gastric, Surv(time,status)~treat)
coef(fit)

```

concreg

Concordance regression for survival data

Description

This package implements concordance regression for survival and other outcome data types, where each summand of the log likelihood consists of a pair of observations. The parameter estimates are estimated log odds of concordance and straightforwardly translate into partial concordance indices.

Usage

```

concreg(formula=attr(data, "formula"),
  data=sys.parent(),
  id=NULL,
  normalize=TRUE,
  scale.weights=1,
  offset=NULL,
  alpha=0.05,
  maxit=50,
  maxhs=5,
  epsilon=1e-6,
  maxstep=2.5,
  x=TRUE,
  y=TRUE,
  print=TRUE,
  c.risk=NULL,
  strata.var=NULL,
  trunc.weights=1,

```

```
npar=FALSE,
...)
```

Arguments

formula	a formula object, with the response on the left of the operator, and the model terms on the right. The response can be a survival object as returned by the 'Surv' function, or a single variable.
data	a data.frame in which to interpret the variables named in the 'formula' argument.
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.
offset	specifies a variable which is included in the model but its parameter estimate is fixed at 1.
scale.weights	specifies a scaling factor (a multiplicative constant) for the weights
x	includes covariates in output object
y	includes outcome in output object
print	prints fitting information on the screen
c.risk	competing risk indicator: 0 for end-of-follow-up, 1 for event of interest, 2 for competing event. status variable in formula must be 0 for censored (by end-of-follow-up or competing event), 1 for event
strata.var	variable for defining strata in stratified analysis
trunc.weights	quantile at which weights are truncated. set to 1 for no weight truncation.
npar	estimation of nonparametric log odds of concordance?
...	further arguments

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 or type-1 error inflated. In such a situation concordance regression provides an alternative, as it summarizes a time-dependent effect into a scalar estimate that can be interpreted as log odds of concordance.

Concordance regression is conditional logistic regression on all pairs of observations. In each pair, the subject that dies earlier is assumed to be a case, and the other subject the control. Pairs with equal survival time or covariate vector are uninformative. Pairs where the shorter time is censored are also not used. To correct for the loss of information due to censoring, a weighting scheme is used that upweights eligible pairs by inverse probability of censoring, and at the same time restores the number of pairs at each failure time that would be expected if there was no censoring.

Inference is based on a robust covariance matrix similar to that of Lin and Wei (1989) proposed for the Cox model. Competing risks can be accommodated by an additional weighting of subjects who experience a competing risk. These subjects remain in the risk sets, but their weights in the analysis resemble their probability to be still under follow-up (following Fine and Gray, 1999).

Value

coefficients	the parameter estimates
alpha	the significance level = 1 - confidence level
var	the estimated robust covariance matrix
cov.mb	the model-based covariance matrix
iter	the number of iterations needed to converge
n	the number of observations
y	the response
x	the covariates
formula	the model formula
means	the means of the covariates
linear.predictors	the linear predictors
Wald	the global Wald statistic
df	the degrees of freedom
ci.lower	the lower confidence limits
ci.upper	the upper confidence limits
prob	the p-values
call	the function call
W	A matrix with 3 columns and rows according to the number of uncensored failure times. The first column contains the stratum numbers, the second column the failure times, the third column the weight for each pair at each failure time.
G	A vector containing the probability of censoring for each observation.

Author(s)

Georg Heinze, Daniela Dunkler and Meinhard Ploner

References

Dunkler D, Schemper M and Heinze G (2010). Gene selection in microarray survival studies under possibly non-proportional hazards. *Bioinformatics* 26, 784-790.

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

Lin D and Wei L (1989). The robust inference for the Cox proportional hazards model. *Journal of the American Statistical Association* 84, 1074-1078.

See Also

coxph

Examples

```
# gastric cancer data set
gastric <-
  structure(list(patnr = as.integer(c(46, 1, 2, 3, 4, 5, 47, 6,
    7, 8, 9, 48, 10, 11, 49, 12, 13, 14, 50, 15, 16, 17, 18, 19,
    20, 51, 21, 22, 52, 23, 53, 54, 55, 24, 25, 56, 57, 58, 59, 60,
    61, 62, 63, 64, 26, 65, 27, 66, 28, 29, 67, 68, 69, 70, 30, 71,
    31, 72, 32, 73, 33, 34, 74, 75, 76, 77, 78, 35, 79, 36, 80, 81,
    82, 37, 38, 39, 83, 84, 40, 85, 41, 86, 87, 88, 42, 43, 44, 89,
    90, 45)),
    treat = as.integer(c(0, 1, 1, 1, 1, 1, 0, 1, 1, 1,
    1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0,
    0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0,
    0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0,
    1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1)),
    time = as.integer(c(1,
    17, 42, 44, 48, 60, 63, 72, 74, 95, 103, 105, 108, 122, 125,
    144, 167, 170, 182, 183, 185, 193, 195, 197, 208, 216, 234, 235,
    250, 254, 262, 301, 301, 307, 315, 342, 354, 356, 358, 380, 383,
    383, 388, 394, 401, 408, 445, 460, 464, 484, 489, 499, 523, 524,
    528, 535, 542, 562, 567, 569, 577, 580, 675, 676, 748, 778, 786,
    795, 797, 855, 955, 968, 977, 1174, 1214, 1232, 1245, 1271, 1366,
    1420, 1455, 1460, 1516, 1551, 1585, 1622, 1626, 1690, 1694, 1736
    )),
    status = as.integer(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0))),
  .Names = c("patnr",
    "treat", "time", "status"), class = "data.frame",
  row.names = c("1",
    "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13",
    "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",
```



```

"25", "26", "27", "28", "29", "30", "31", "32", "33", "34", "35",
"36", "37", "38", "39", "40", "41", "42", "43", "44", "45", "46",
"47", "48", "49", "50", "51", "52", "53", "54", "55", "56", "57",
"58", "59", "60", "61", "62", "63", "64", "65", "66", "67", "68",
"69", "70", "71", "72", "73", "74", "75", "76", "77", "78", "79",
"80", "81", "82", "83", "84", "85", "86", "87", "88", "89", "90"
))

fit<-concreg(data=gastric, Surv(time,status)~treat)
summary(fit)

plotw(fit)

fit2<-concreg(data=gastric, Surv(time,status)~treat, trunc.weights=0.95)
summary(fit2)

# nonparametric c-index
y<-rnorm(200)
x<-rnorm(200)
dat<-data.frame(x=x, y=y)
fit3<-concreg(data=dat, y~x, npar=TRUE)
cindex(fit3)
confint(fit3, what="cindex")

# Mann-Whitney-U (AUROC) statistic
y<-rbinom(200, 1, 1-pnorm(x))
dat<-data.frame(x=x, y=y)
fit4<-concreg(data=dat, y~x, npar=TRUE)
cindex(fit4)
confint(fit4, what="cindex")
# symmetry of the univariate nonparametric model:
fit5<-concreg(data=dat, x~y, npar=TRUE)
cindex(fit5)
confint(fit5, what="cindex")

```

confint.concreg *Method to extract confidence intervals from a concreg object.*

Description

This method extracts confidence intervals from a concreg object.

Usage

```

## S3 method for class 'concreg'
confint(object, parm, level = 0.95, what="coefficients", ...)

```

Arguments

object	a concreg object
parm	a specification of which parameters are to be given confidence intervals, a vector of numbers. If missing, all parameters are considered.)
level	the confidence level (default=0.95)
what	the type of parameter for which confidence intervals are requested: one of c("coefficients", "OC", "cindex")
...	additional argument(s) for methods

Details

This method returns confidence intervals based on a normal approximation using the sandwich covariance matrix, for the regression coefficients (what="coefficients"), the odds of concordance (what="OC"), which are defined as $\exp(\text{coefficients})$, or the c-index (what="cindex"), which are $OC/(1+OC)$.

Value

A $p \times 2$ matrix of confidence limits, where $p = \text{length}(\text{parms})$.

Author(s)

Georg Heinze

See Also

[coef.concreg](#)

Examples

```
gastric <-
  structure(list(patnr = as.integer(c(46, 1, 2, 3, 4, 5, 47, 6,
    7, 8, 9, 48, 10, 11, 49, 12, 13, 14, 50, 15, 16, 17, 18, 19,
    20, 51, 21, 22, 52, 23, 53, 54, 55, 24, 25, 56, 57, 58, 59, 60,
    61, 62, 63, 64, 26, 65, 27, 66, 28, 29, 67, 68, 69, 70, 30, 71,
    31, 72, 32, 73, 33, 34, 74, 75, 76, 77, 78, 35, 79, 36, 80, 81,
    82, 37, 38, 39, 83, 84, 40, 85, 41, 86, 87, 88, 42, 43, 44, 89,
    90, 45)),
    treat = as.integer(c(0, 1, 1, 1, 1, 1, 0, 1, 1, 1,
    1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0,
    0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0,
    0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0,
    1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1)),
    time = as.integer(c(1,
    17, 42, 44, 48, 60, 63, 72, 74, 95, 103, 105, 108, 122, 125,
    144, 167, 170, 182, 183, 185, 193, 195, 197, 208, 216, 234, 235,
    250, 254, 262, 301, 301, 307, 315, 342, 354, 356, 358, 380, 383,
    383, 388, 394, 401, 408, 445, 460, 464, 484, 489, 499, 523, 524,
    528, 535, 542, 562, 567, 569, 577, 580, 675, 676, 748, 778, 786,
```

```

      795, 797, 855, 955, 968, 977, 1174, 1214, 1232, 1245, 1271, 1366,
      1420, 1455, 1460, 1516, 1551, 1585, 1622, 1626, 1690, 1694, 1736
    )),
  status = as.integer(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0)),
  .Names = c("patnr",
    "treat", "time", "status"), class = "data.frame",
  row.names = c("1",
    "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13",
    "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",
    "25", "26", "27", "28", "29", "30", "31", "32", "33", "34", "35",
    "36", "37", "38", "39", "40", "41", "42", "43", "44", "45", "46",
    "47", "48", "49", "50", "51", "52", "53", "54", "55", "56", "57",
    "58", "59", "60", "61", "62", "63", "64", "65", "66", "67", "68",
    "69", "70", "71", "72", "73", "74", "75", "76", "77", "78", "79",
    "80", "81", "82", "83", "84", "85", "86", "87", "88", "89", "90"
  ))

fit<-concreg(data=gastric, Surv(time,status)~treat)
confint(fit)

```

plotw

Plot pair weights of concordance regression versus time

Description

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

Usage

```
plotw(x, rank=FALSE, log=FALSE, xlim=c(0,max(time)),
  ...)
```

Arguments

x	a concreg 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)
xlim	limits for time axis, defaults to range of time variable
...	further arguments for plotting

Details

The function plots the (optionally log-transformed) pair weights against (ranked) time, separately for each stratum.

Value

no return value.

Author(s)

Georg Heinze

See Also

concreg

vcov.concreg

Method to extract the sandwich covariance matrix for the regression coefficients from a concreg object

Description

This method returns the sandwich covariance matrix for the estimated regression coefficients from a concreg object.

Usage

```
## S3 method for class 'concreg'  
vcov(object, ...)
```

Arguments

object a concreg object
... additional argument(s) for methods.

Value

A $p \times p$ covariance matrix for the p regression coefficients.

Author(s)

Georg Heinze

See Also

confint.concreg coef.concreg

Examples

```

gastric <-
  structure(list(patnr = as.integer(c(46, 1, 2, 3, 4, 5, 47, 6,
    7, 8, 9, 48, 10, 11, 49, 12, 13, 14, 50, 15, 16, 17, 18, 19,
    20, 51, 21, 22, 52, 23, 53, 54, 55, 24, 25, 56, 57, 58, 59, 60,
    61, 62, 63, 64, 26, 65, 27, 66, 28, 29, 67, 68, 69, 70, 30, 71,
    31, 72, 32, 73, 33, 34, 74, 75, 76, 77, 78, 35, 79, 36, 80, 81,
    82, 37, 38, 39, 83, 84, 40, 85, 41, 86, 87, 88, 42, 43, 44, 89,
    90, 45)),
    treat = as.integer(c(0, 1, 1, 1, 1, 1, 0, 1, 1, 1,
    1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0,
    0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0,
    0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0,
    1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1)),
    time = as.integer(c(1,
    17, 42, 44, 48, 60, 63, 72, 74, 95, 103, 105, 108, 122, 125,
    144, 167, 170, 182, 183, 185, 193, 195, 197, 208, 216, 234, 235,
    250, 254, 262, 301, 301, 307, 315, 342, 354, 356, 358, 380, 383,
    383, 388, 394, 401, 408, 445, 460, 464, 484, 489, 499, 523, 524,
    528, 535, 542, 562, 567, 569, 577, 580, 675, 676, 748, 778, 786,
    795, 797, 855, 955, 968, 977, 1174, 1214, 1232, 1245, 1271, 1366,
    1420, 1455, 1460, 1516, 1551, 1585, 1622, 1626, 1690, 1694, 1736
    )),
    status = as.integer(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0))),
  .Names = c("patnr",
    "treat", "time", "status"), class = "data.frame",
  row.names = c("1",
    "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13",
    "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",
    "25", "26", "27", "28", "29", "30", "31", "32", "33", "34", "35",
    "36", "37", "38", "39", "40", "41", "42", "43", "44", "45", "46",
    "47", "48", "49", "50", "51", "52", "53", "54", "55", "56", "57",
    "58", "59", "60", "61", "62", "63", "64", "65", "66", "67", "68",
    "69", "70", "71", "72", "73", "74", "75", "76", "77", "78", "79",
    "80", "81", "82", "83", "84", "85", "86", "87", "88", "89", "90"
  ))

fit<-concreg(data=gastric, Surv(time,status)~treat)
coef(fit)
vcov(fit)

```

Index

*Topic **models**

- cindex, [2](#)
- coef.concreg, [3](#)
- concreg, [5](#)
- confint.concreg, [9](#)
- vcov.concreg, [12](#)

*Topic **regression**

- cindex, [2](#)
- coef.concreg, [3](#)
- concreg, [5](#)
- confint.concreg, [9](#)
- vcov.concreg, [12](#)

*Topic **survival**

- cindex, [2](#)
- concreg, [5](#)
- confint.concreg, [9](#)
- vcov.concreg, [12](#)

- cindex, [2](#)
- coef.concreg, [3](#), [10](#)
- concreg, [5](#)
- confint.concreg, [9](#)

- plotw, [11](#)

- vcov.concreg, [12](#)