Package ‘cmprskQR’

September 15, 2019

Version 0.9.2
Date 2019-09-13
Title Analysis of Competing Risks Using Quantile Regressions
Author Stephan Dlugosz [aut, cre],
Limin Peng [aut],
Ruoshia Li [aut],
Shuolin Shi [ctb]
Maintainer Stephan Dlugosz <stephan.dlugosz@googlemail.com>
Depends R (>= 2.13.0)
Imports graphics, quantreg, stats, survival
Description Estimation, testing and regression modeling of
subdistribution functions in competing risks using quantile regressions,
License GPL (>= 2)
NeedsCompilation yes
Repository CRAN
ByteCompile TRUE
URL https://bitbucket.org/sdlugosz/cmprskqr
BugReports https://bitbucket.org/sdlugosz/cmprskqr/issues
Date/Publication 2019-09-15 14:50:03 UTC

R topics documented:
crrQR ......................................................... 2
plot.crrQR .................................................. 4
plot.predict.crrQR ........................................ 4
predict.crrQR ............................................. 5
summary.crrQR ........................................... 6

Index 8
Description

quantile regression modeling of subdistribution functions in competing risks

Usage

crrQR(ftime, fstatus, X, failcode=1, cencode=0, tau.range=c(0.01,0.99), tau.step=0.01, subset, na.action=na.omit, rq.method="br", variance=TRUE)

## S3 method for class 'crrQR'
print(x, ...)

Arguments

ftime vector of failure/censoring times
fstatus vector with a unique code for each failure type and a separate code for censored observations
X matrix (nobs x ncovs) of covariates
failcode code of fstatus that denotes the failure type of interest
cencode code of fstatus that denotes censored observations
tau.range vector of length 2 denoting the range of quantiles
tau.step grid size on tau.range (spacing between two grid points)
subset a logical vector specifying a subset of cases to include in the analysis
na.action a function specifying the action to take for any cases missing any of ftime, fstatus, cov1, cov2, cengroup, or subset.
rq.method method of computation for quantile regressions. (cf. documentation of method rq.fit in package quantreg for details.)
variance if FALSE, then suppresses computation of asymptotic variances
x crrQR object (output from crrQR()) for method print
... included for compatibility with the generic functions. Not currently used.

Details

Fits the competing risks quantile regression model described in Peng and Fine (2009).

While the use of model formulas is not supported, the model.matrix function can be used to generate suitable matrices of covariates from factors, eg model.matrix(~factor1+factor2)[,-1] will generate the variables for the factor coding of the factors factor1 and factor2. The final [,1] removes the constant term from the output of model.matrix.
If `variance=FALSE`, then some of the functionality in `summary.crrQR` and `print.crrQR` will be lost. This option can be useful in situations where `crrQR` is called repeatedly for point estimates, but standard errors are not required, such as in bootstrapping the cumulative incidence function for confidence intervals.

The print method prints the estimated coefficients, the estimated standard errors, and the two-sided p-values for the test of the individual coefficients equal to 0.

A first implementation of the estimation procedure was prepared by Limin Peng and Ruosha Li.

**Value**

Returns a list of class `crrQR`, with components

- `$beta.seq` the estimated regression coefficients
- `$tau.seq` the sequence of quantiles computed
- `$var.seq` estimated variance covariance matrix of coefficients
- `$inf.func` list of estimated influence functions
- `$call` the call to `crr`
- `$n` the number of observations used in fitting the model
- `$n.missing` the number of observations removed from the input data due to missing values
- `$cvt.length` number of covariates (columns of X)

**References**


**See Also**

`predict.crrQR` `plot.predict.crrQR` `summary.crrQR` `rq.fit`

**Examples**

```r
# simulated data to test
set.seed(10)
ftime <- rexp(200)
fstatus <- sample(0:2,200,replace=TRUE)
X <- matrix(runif(600),nrow=200)
dimnames(X)[[2]] <- c('x1','x2','x3')
# compute model
print(z <- crrQR(ftime,fstatus,X))
summary(z)
# predict and plot cumulative incedences
reference <- as.matrix(rbind(c(.1,.5,.8),c(.1,.5,.2)))
dimnames(reference)[[2]] <- c('x1','x2','x3')
z.p <- predict(z,reference)
print(z.p)
plot(z.p,lty=1,color=2:3)
crrQR(ftime,fstatus,X,failcode=2)
```
Description

plot method for \text{crrQR}

Usage

\texttt{## S3 method for class '{crrQR}'
plot(x, subset=NULL, main=NULL, ...)
}

Arguments

- \texttt{x} \hspace{1cm} \text{output from crrQR}
- \texttt{subset} \hspace{1cm} \text{plot a subset of coefficients}
- \texttt{main} \hspace{1cm} \text{main title of the plot}
- \texttt{...} \hspace{1cm} \text{other arguments to plot}

Side Effects

plots the variable profiles for each curve

See Also

crrQR

Description

plot method for \text{predict.crrQR}

Usage

\texttt{## S3 method for class 'predict.crrQR'
plot(x, lty=1:(ncol(x)-1), color=1,
ylim=c(0, max(x[, ncol(x)])), xlin=0, xmax=max(x[, -ncol(x)]), ...)
}
Arguments

- **x**: output from `predict.crrQR`
- **lty**: vector of line types. If length is < \# curves, then lty[1] is used for all.
- **color**: vector of line colors. If length is < \# curves, then color[1] is used for all.
- **ylim**: range of y-axis (vector of length two)
- **xmin**: lower limit of x-axis (often 0, the default)
- **xmax**: upper limit of x-axis
- **...**: other arguments to plot

Side Effects

plots the subdistribution functions estimated by `predict.crrQR`, by default using a different line type for each curve

See Also

`crrQR`, `predict.crrQR`

---

**predict.crrQR**

*Estimate subdistribution functions from crrQR output*

---

Description

`predict` method for `crrQR`

Usage

```r
## S3 method for class 'crrQR'
predict(object, x, rearrangement, ...)
```

Arguments

- **object**: output from `crrQR`
- **x**: vector of covariate values for which the conditional distribution function is to be estimated. The columns of x must be named the same as in the original call to `crrQR`. Each must be given if present in the original call to `crrQR`.
- **rearrangement**: set rearrangement=TRUE to perform a rearrangement of the predicted probabilities as suggested in Chernozhukov V, Fernández-Val I and Galichon A (2010).
- **...**: additional parameters (currently ignored).

Details

Computes the conditional estimate given values of covariates from \( sup(\tau : \tau \leq \zeta_{x,j}^{-1}[\ln(t)]) \), for \( \zeta_{x,j}(\tau) = x'\beta_j(\tau) \) (see Długosz S, Lo S and Wilke RA (2014) for details)
**Value**

Returns a matrix with the unique type 1 failure times in the first column, and the other columns giving the estimated subdistribution function corresponding to the covariate combinations in the rows of x, at each failure time (the value that the estimate jumps to at that failure time).

**References**


**See Also**

crrQR plot.predict.crrQR

---

**summary.crrQR** Summary method for crrQR

**Description**

generate and print summaries of crrQR output

**Usage**

```r
## S3 method for class 'crrQR'
summary(object, conf.int = 0.95, digits = max(options()$digits - 5, 2), ...)

## S3 method for class 'summary.crrQR'
print(x, digits=max(options()$digits - 4, 3), ...)
```

**Arguments**

- `object` an object of class crrQR (output from the crrQR function)
- `conf.int` the level for a two-sided confidence interval on the coefficients. Default is 0.95.
- `digits` in summary.crrQR, digits determines the number of significant digits retained in the p-values. In print.summary.crrQR, digits sets the values of the digits option for printing the output.
- `...` included for compatibility with the generic functions. Not currently used.
- `x` an object of class summary.crrQR (output from the summary method for crrQR)

**Details**

The summary method calculates the average effects, the variances and p-values of the test on the effect being 0. Furthermore it performs a test for constant coefficients. The print method prints a fairly standard format tabular summary of the results.
Value

`summary.crrQR` returns a list of class `summary.crrQR`, which contains components:

- `call`: the call to `crr`
- `n`: the number of observations used in fitting the model
- `n.missing`: the number of observations removed by `crr` from the input data due to missing values
- `ave.eff`: vector of average effects of covariates
- `var.ave.eff`: vector of corresponding variances
- `p.signf.test`: p-values for testing average effect=0
- `cnst.test`: scores of test on constant effect
- `var.cnst.test`: variances of the score
- `p.cnst.test`: p-values for the test

See Also

- `crrQR`

Examples

```r
## see examples in the crrQR help file
```
Index

*Topic survival
  crrQR, 2
  plot.crrQR, 4
  plot.predict.crrQR, 4
  predict.crrQR, 5
  summary.crrQR, 6

crrQR, 2, 4–7

plot.crrQR, 4
plot.predict.crrQR, 3, 4, 6
predict.crrQR, 3, 5, 6
print.crrQR (crrQR), 2
print.summary.crrQR (summary.crrQR), 6

rq.fit, 2, 3

summary.crrQR, 3, 6