

Package ‘qris’

July 22, 2025

Type Package

Title Quantile Regression Model for Residual Lifetime Using an Induced Smoothing Approach

Version 1.1.1

Date 2024-02-27

Description A collection of functions is provided by this package to fit quantiles regression models for censored residual lifetimes. It provides various options for regression parameters estimation: the induced smoothing approach (smooth), and L1-minimization (non-smooth). It also implements the estimation methods for standard errors of the regression parameters estimator based on an efficient partial multiplier bootstrap method and robust sandwich estimator. Furthermore, a simultaneous procedure of estimating regression parameters and their standard errors via an iterative updating procedure is implemented (iterative). For more details, see Kim, K. H., Caplan, D. J., & Kang, S. (2022), ``Smoothed quantile regression for censored residual life'', Computational Statistics, 1-22 <doi:10.1007/s00180-022-01262-z>.

Depends R (>= 3.6.0)

License GPL (>= 3)

URL <https://github.com/Kyuhyun07/qris>

Encoding UTF-8

RoxygenNote 7.3.1

NeedsCompilation yes

Imports nleqslv, quantreg, stringr, survival, ggplot2, Rcpp

LinkingTo Rcpp, RcppArmadillo

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Author Kyu Hyun Kim [aut, cre],
Sangwook Kang [aut],
Sy Han Chiou [aut]

Maintainer Kyu Hyun Kim <kyuhyunkim07@gmail.com>

Repository CRAN

Date/Publication 2024-03-05 14:40:03 UTC

Contents

qris-package	2
export_Surv	3
plot.qris	3
predict.qris	5
qris	5
qris.control	7
qris.extend	8
residuals.qris	9
Index	10

qris-package	<i>qris: Quantile Regression Model for Residual Lifetime Using an Induced Smoothing Approach</i>
--------------	--

Description

This package offers a collection of functions to fit quantiles regression models for censored residual lifetimes. It provides various options for regression parameters estimation: the induced smoothing approach (smooth), and L1-minimization (non-smooth). It also implements the estimation methods for standard errors of the regression parameters estimates based on an efficient partial multiplier bootstrap method and robust sandwich estimator. Furthermore, a simultaneous procedure of estimating regression parameters and their standard errors via an iterative updating procedure is implemented (iterative). See Kim, K. (2022) "Smoothed quantile regression for censored residual life", <arXiv:2205.00413>.

Author(s)

Maintainer: Kyu Hyun Kim <kyuhyunkim07@yonsei.ac.kr>

Authors:

- Sangwook Kang
- Sy Han Chiou

See Also

Useful links:

- <https://github.com/Kyuhyun07/qris>

export_Surv	<i>Surv function imported from survival</i>
-------------	---

Description

This function is imported from the survival package. See [Surv](#).

Value

An object of class Surv. There are methods for print, is.na, and subscripting survival objects. Surv objects are implemented as a matrix of 2 or 3 columns that has further attributes. These include the type (left censored, right censored, counting process, etc.) and labels for the states for multi-state objects. Any attributes of the input arguments are also preserved in inputAttributes. This may be useful for other packages that have attached further information to data items such as labels; none of the routines in the survival package make use of these values, however. In the case of is.Surv, a logical value TRUE if x inherits from class "Surv", otherwise an FALSE.

plot.qris	<i>Draw 95% confidence interval by a quantile regression estimator of residual lifetime from survival data</i>
-----------	--

Description

Draw 95% confidence interval by a quantile regression estimator of residual lifetime from survival data

Usage

```
## S3 method for class 'qris'
plot(
  x,
  t0s = NULL,
  Qs = NULL,
  nB = NULL,
  vari = NULL,
  byQs = FALSE,
  ggextra = NULL,
  ...
)
```

Arguments

<code>x</code>	is an "qris" object or a data.frame returned by <code>plot.qris</code> .
<code>t0s</code>	is a vector of range of t_0 to plot; when not specified, the default value is from 0 to presently defined t_0
<code>Qs</code>	is a vector of range of Q to plot; when not specified, the default value is from 5% to presently defined Q
<code>nB</code>	is the number of multiplier bootstrapping for standard error estimation.
<code>vari</code>	is a character string to choose variables to draw the regression coefficient.
<code>byQs</code>	put Q 's on x-axis; only used when both t_0 's and Q 's are specified.
<code>ggextra</code>	is a list that contains additional components to apply to the ggplot output. The ggplot2 library must be loaded in order to utilize this feature.
<code>...</code>	for future extension

Value

A list contains ggplot object and the information to generate it.

Examples

```
data(cancer, package = "survival")
lung2 <- subset(lung, select = c(time, status, age, sex))
## tidy up the data
lung2$status <- lung2$status - 1
lung2$sex <- lung2$sex - 1

library(qris)
fm <- Surv(time, status) ~ age + sex
fit <- qris(fm, data = lung2, t0 = 30, Q = 0.5, nB = 50, "nonsmooth", "fmb")

## Plot with default values; Qs <- 1:9 / 10 and t0s = fit2$para$t0 (in this case 30)
plot(fit)

## Plot with without 95% CI is much faster
plot(fit, nB = 0)

## Plot feature can update qris calls
fit <- plot(fit, Qs = 3:6 / 10, t0s = 1:6 * 10)

## Faster after updating the qris call
plot(fit, byQs = FALSE)
plot(fit, byQs = TRUE)

plot(fit, byQs = FALSE, vari = c("sex", "age"))
plot(fit, byQs = TRUE, vari = c("sex", "age"))

## Extra ggplot components
library(ggplot2)
plot(fit, byQs = FALSE, vari = c("sex", "age"), ggextra = theme(legend.position = "none"))
```

predict.qris

*Prediction for Quantile Regression Model Fitted on Residual life***Description**

Prediction based on fitted quantile regression model

Usage

```
## S3 method for class 'qris'
predict(object, newdata, ...)
```

Arguments

object	is a qris object
newdata	is a data frame for an optional new data to do predictions. If omitted, the fitted values based on the original data and fit will be returned.
...	for future extension

Value

A vector of prediction

qris

*Estimate a quantile regression estimator of residual lifetime from survival data***Description**

Using three estimation methods (1) L1-minimization(non-smooth estimating equation (2) Induced smoothing approach (smooth estimating equation (3) Iterative procedure with induced smoothing approach (smooth estimating equation

Usage

```
qris(
  formula,
  data,
  t0 = 0,
  Q = 0.5,
  nB = 100,
  method = c("smooth", "iterative", "nonsmooth"),
  se = c("fmb", "pmb"),
  init = c("rq", "noeffect"),
  verbose = FALSE,
  control = qris.control()
)
```

Arguments

<code>formula</code>	is a formula expression, of the form <code>response ~ predictors</code> . The response is a <code>Surv</code> object with right censoring.
<code>data</code>	is an optional <code>data.frame</code> in which to interpret the variables occurring in the formula.
<code>t0</code>	is the follow-up time (or basetime of analysis). The default followup time is set to 0.
<code>Q</code>	is the quantile. The default quantile is set to 0.5.
<code>nB</code>	is number of multiplier bootstrapping for V matrix estimation. The default number of bootstrapping is set to 100.
<code>method</code>	is an option for specifying the methods of parameters estimation; <code>smooth</code> is the default in which parameters estimates and their standard errors are obtained via induced smoothed estimating equations. <code>nonsmooth</code> uses a L1-minimization method for non-smooth object functions in coefficient estimation. <code>iterative</code> simultaneously estimates parameters and their standard errors based on the iterative updates for the parameter estimates.
<code>se</code>	is an option for specifying the methods of standard errors estimation; The available options are <code>pmb</code> and <code>fmb</code> . The <code>pmb</code> is the default option. It estimates the standard errors via partial multiplier bootstrapping and is only available when <code>method = "smooth"</code> or <code>method = "iterative"</code> . The <code>fmb</code> uses a full multiplier bootstrapping in standard errors estimation.
<code>init</code>	is an option for specifying the initial values of the parameters estimates. Available options are <code>rq</code> and <code>noeffect</code> . These options correspond to the estimates from the <code>quantreg::rq()</code> and a zero vector, respectively. Alternatively, user defined numerical vector is also allowed.
<code>verbose</code>	Shows computation status.
<code>control</code>	controls maximum number of iteration, tolerance of convergence and whether to display output for each iteration when <code>method = "iterative"</code> .

Value

An object of class "qris" contains model fitting results. The "qris" object is a list containing at least the following components:

coefficient a vector of point estimates

stderr a vector of standard error of point estimates

vcov a matrix of the estimated variance-covariance matrix

maxiter a number of iteration until convergence (only for iterative procedure)

Examples

```
## #####
## Simulated data
## #####
data.gen <- function(n) {
```

```

r0 <- .2 * sqrt(log(2))
r1 <- .1 * sqrt(log(2))
dat <- data.frame(censoring = runif(n, 0, 24.35),
                  Time0 = sqrt(-log(1 - runif(n))),
                  X = rbinom(n, 1, .5))
dat$Time0 <- ifelse(dat$X > 0, dat$Time0 / r1, dat$Time0 / r0)
dat$Time <- pmin(dat$Time0, dat$censoring)
dat$status <- 1 * (dat$Time0 < dat$censoring)
subset(dat, select = c(Time, status, X))
}

set.seed(1)
dat <- data.gen(200)
fm <- Surv(Time, status) ~ X
fit1 <- gris(fm, data = dat, t0 = 1, Q = 0.5, nB = 100, "smooth", "pmb", c(1,1))
fit2 <- gris(fm, data = dat, t0 = 1, Q = 0.5, nB = 100, "nonsmooth", "fmb", "rq")
fit3 <- gris(fm, data = dat, t0 = 1, Q = 0.5, nB = 100, "iterative", "fmb", "rq",
             control = gris.control(maxit = 20, tol = 1e-3, trace = TRUE))

summary(fit1)
summary(fit2)
summary(fit3)

## #####
## Real data application
## #####
data(cancer, package = "survival")
lung2 <- subset(lung, select = c(time, status, age, sex))
## tidy up the data
lung2$status <- lung2$status - 1
lung2$sex <- lung2$sex - 1

fm <- Surv(time, status) ~ age + sex
fit1 <- gris(fm, data = lung2, t0 = 0, Q = 0.5, nB = 100, "iterative", "pmb", "rq")
fit2 <- gris(fm, data = lung2, t0 = 30, Q = 0.5, nB = 100, "nonsmooth", "fmb", c(1, 0, 1))
fit3 <- gris(fm, data = lung2, t0 = 100, Q = 0.5, nB = 100, "smooth", "pmb", "rq")

summary(fit1)
summary(fit2)
summary(fit3)

plot(fit2, Qs = 4:6 / 10)

```

gris.control

Auxiliary for Controlling gris

Description

Auxiliary function as user interface for gris fitting.

Usage

```
qris.control(maxiter = 10, tol = 0.001, trace = FALSE)
```

Arguments

maxiter	max number of iteration.
tol	tolerance of convergence
trace	a binary variable, determine whether to save output for each iteration.

Details

When trace is TRUE, output for each iteration is printed to the screen.

Value

A list with the arguments as components.

See Also

[qris](#)

qris.extend	<i>Extend a "qris" object to a specified range of τ or t_0 values.</i>
-------------	---

Description

Extend a "qris" object to a specified range of τ or t_0 values.

Usage

```
qris.extend(x, t0s = NULL, Qs = NULL, nB = NULL, vari = NULL)
```

Arguments

x	is an qris object or a data.frame returned by plot.qris
t0s	is a vector of range of t_0 to plot; when not specified, the default value is from 0 to presently defined t_0
Qs	is a vector of range of Q to plot; when not specified, the default value is from 5% to presently defined Q
nB	is the number of multiplier bootstrapping for standard error estimation.
vari	is a character string to choose variables to draw the regression coefficient.

residuals.qris	<i>Residuals for Quantile Regression Model Fitted on Residual life</i>
----------------	--

Description

Residual based on fitted quantile regression model

Usage

```
## S3 method for class 'qris'  
residuals(object, newdata, ...)
```

Arguments

object	is a qris object
newdata	is a data frame for an optional new data to do predictions. If omitted, the fitted values based on the original data and fit will be returned.
...	for future extension

Value

A vector of residual

Index

`export_Surv`, [3](#)

`plot.qris`, [3](#)

`predict.qris`, [5](#)

`qris`, [5](#), [8](#)

`qris-package`, [2](#)

`qris-packages (qris-package)`, [2](#)

`qris.control`, [7](#)

`qris.extend`, [8](#)

`residuals.qris`, [9](#)

`Surv`, [3](#)

`Surv (export_Surv)`, [3](#)