Package: consrq (via r-universe)

November 25, 2024

Type Package

Title Constrained Quantile Regression

Version 1.0

Date 2024-11-20

Author Michail Tsagris [aut, cre]

Maintainer Michail Tsagris <mtsagris@uoc.gr>

Depends R (>= 4.0)

Suggests Rfast2, cols

Imports quantreg, Rfast

Description Constrained quantile regression is performed. One constraint is that all beta coefficients (including the constant) cannot be negative, they can be either 0 or strictly positive. Another constraint is that the beta coefficients lie within an interval. References: Koenker R. (2005) Quantile Regression, Cambridge University Press. <doi:10.1017/CBO9780511754098>.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2024-11-21 08:00:03 UTC

Config/pak/sysreqs make

Contents

`consrq-package` *Constrained Quantile Regression*

Description

Constrained quantile regression is performed. One constraint is that all beta coefficients (including the constant) cannot be negative. They can be either 0 or strictly positive. Another constraint is that the beta coefficients lie within an interval.

Details

Package:	consrq
Type:	Package
Version:	1.0
Date:	2024-11-20

Maintainers

Michail Tsagris <mtsagris@uoc.gr>.

Author(s)

Michail Tsagris <mtsagris@uoc.gr>

References

Koenker R. (2005) Quantile Regression, Cambridge University Press.

`Lower and upper bound constrained quantile regression` *Lower and upper bound constrained quantile regression*

Description

Lower and upper bound constrained quantile regression.

Usage

```
int.crq(y, x, tau = 0.5, lb, ub)
int.mcrq(y, x, tau = 0.5, lb, ub)
```

Arguments

y	For the int.crq() the response variable, a numerical vector with observations, but a matrix of response variables for the int.mcrq().
x	A matrix with independent variables, the design matrix.
tau	The quantile(s) to be estimated, a number strictly between 0 and 1. It a vector of values between 0 and 1; in this case an object of class "rqs" is returned containing among other things a matrix of coefficient estimates at the specified quantiles.
lb	A vector or a single value with the lower bound(s) in the coefficients.
ub	A vector or a single value with the upper bound(s) in the coefficients.

Details

This function performs quantile regression under the constraint that the beta coefficients lie within interval(s), i.e. $min \sum_{i=1}^{n} |y_i - \boldsymbol{x}_i^{\top} \boldsymbol{\beta}|$ such that $lb_j \leq \beta_j \leq ub_j$.

Value

A list including:

be	A numerical matrix with the constrained beta coefficients.
mae	A numerical vector with the mean absolute error(s).

Author(s)

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

See Also

prq, pcrq

Examples

```
x <- as.matrix( iris[1:50, 1:4] )
y <- rnorm(50)
int.crq(y, x, lb = -0.2, ub = 0.2)
```

`Positive and unit sum constrained quantile regression`
Positive and unit sum constrained quantile regression

Description

Positive and unit sum constrained quantile regression.

Usage

```
pcrq(y, x, tau = 0.5)
mpcrq(y, x, tau = 0.5)
```

Arguments

`y`	The response variable. For the pcrq() a numerical vector with observations, but for the mpcrq() a numerical matrix.
`x`	A matrix with independent variables, the design matrix.
`tau`	The quantile(s) to be estimated, a number strictly between 0 and 1. It a vector of values between 0 and 1; in this case an object of class "rqs" is returned containing among other things a matrix of coefficient estimates at the specified quantiles.

Details

The constraint is that all beta coefficients are positive and sum to 1. That is, i.e. $min \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i^{\top} \boldsymbol{\beta})^2$ such that $\beta_j \geq 0$ and $\sum_{j=1}^{d} \beta_j = 1$. The pcrq() function performs a single regression model, whereas the mpcrq() function performs a regression for each column of y. Each regression is independent of the others.

Value

A list including:

`be`	A numerical matrix with the positively constrained beta coefficients.
`mae`	A numerical vector with the mean absolute error.

Author(s)

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

See Also

prq

Examples

```
x <- as.matrix( iris[1:50, 1:4] )
y <- rnorm(50)
pcrq(y, x)
```

`Positively constrained quantile regression`
Positively constrained quantile regression

Description

Positively constrained quantile regression.

Usage

```
prq(y, x, tau = 0.5)
mprq(y, x, tau = 0.5)
```

Arguments

`y`	The response variable. For the prq() a numerical vector with observations, but for the mprq() a numerical matrix .
`x`	A matrix with independent variables, the design matrix.
`tau`	The quantile(s) to be estimated, a number strictly between 0 and 1. It a vector of values between 0 and 1; in this case an object of class "rqs" is returned containing among other things a matrix of coefficient estimates at the specified quantiles.

Details

The constraint is that all beta coefficients (including the constant) are non negative. That is, $min \sum_{i=1}^{n} |y_i - \boldsymbol{x}_i^{\top} \boldsymbol{\beta}|$ such that $\beta_j \geq 0$. The pls() function performs a single regression model, whereas the mpls() function performs a regression for each column of y. Each regression is independent of the others.

Value

A list including:

`be`	A numerical matrix with the positively constrained beta coefficients.
`mae`	A numerical vector with the mean absolute error(s).

Author(s)

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

See Also

pcrq

Examples

```
x <- as.matrix( iris[1:50, 1:4] )
y <- rnorm(50)
prq(y, x)
```

Index