Package ‘cols’

April 15, 2026

Type Package

Title Constrained Ordinary Least Squares

Version 1.7

Date 2026-04-15

Author Michail Tsagris [aut, cre]

Maintainer Michail Tsagris <mtsagris@uoc.gr>

Depends R (>= 4.0)

Imports nnsolve, quadprog, Rfast, Rfast2

Description Constrained ordinary least squares 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 sum of the beta coefficients equals a constant. References: Hansen, B. E. (2022). Econometrics, Princeton University Press. <ISBN:9780691235899>.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2026-04-15 18:30:02 UTC

Contents

`cols-package` *Constrained Ordinary Least Squares*

Description

Constrained ordinary least squares 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 sum of the beta coefficients equals a constant. References: Hansen, B.E. (2022). Econometrics, Princeton University Press.

Details

Package:	cols
Type:	Package
Version:	1.7
Date:	2026-04-15

Maintainers

Michail Tsagris <mtsagris@uoc.gr>.

Author(s)

Michail Tsagris <`mtsagris@uoc.gr`>

References

Hansen, B. E. (2022). Econometrics, Princeton University Press.

`Constrained least squares`
Constrained least squares

Description

Constrained least squares.

Usage

```
cls(y, x, R, ca)
mvcls(y, x, R, ca)
```

Arguments

y	The response variable. For the cls() a numerical vector with observations, but for the mvcls() a numerical matrix .
x	A matrix with independent variables, the design matrix.
R	The R vector that contains the values that will multiply the beta coefficients. See details and examples.
ca	The value of the constraint, $R^T \beta = c$. See details and examples.

Details

This is described in Chapter 8.2 of Hansen (2019). The idea is to inimise the sum of squares of the residuals under the constraint $R^\top \boldsymbol{\beta} = c$. As mentioned above, be careful with the input you give in the x matrix and the R vector. The cls() function performs a single regression model, whereas the mcls() 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 constrained beta coefficients.
mse	A numerical vector with the mean squared error.

Author(s)

Michail Tsagris.

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

References

Hansen, B. E. (2022). Econometrics, Princeton University Press.

See Also

pls, int.cls

Examples

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

`Lower and upper bound constrained least squares`
Constrained least squares

Description

Lower and upper bound constrained least squares

Usage

```
int.cls(y, x, lb, ub)
int.mcls(y, x, lb, ub)
```

Arguments

y	The response variable. For the int.cls() a numerical vector with observations, but for the int.mcls() a numerical matrix .
x	A matrix with independent variables, the design matrix.
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 least squares under the constraint that the beta coefficients lie within interval(s), i.e. $min \sum_{i=1}^n (y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})^2$ such that $lb_j \leq \beta_j \leq ub_j$.

Value

A list including:

be	A numerical matrix with the constrained beta coefficients.
mse	A numerical vector with the mean squared error.

Author(s)

Michail Tsagris.

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

See Also

pls

Examples

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

`Positive and unit sum constrained least squares`
Positive and unit sum constrained least squares

Description

Positive and unit sum constrained least squares.

Usage

```
pcls(y, x)
mpcls(y, x)
```

Arguments

`y`	The response variable. For the pcls() a numerical vector with observations, but for the mpcls() a numerical matrix.
`x`	A matrix with independent variables, the design matrix.

Details

The constraint is that all beta coefficients are positive and sum to 1. that is $min \sum_{i=1}^n (y_i - \boldsymbol{x}_i \top \boldsymbol{\beta})^2$ such that $0 \leq \beta_j \leq 1$ and $\sum_{j=1}^d \beta_j = 1$. The pcls() function performs a single regression model, whereas the mpcls() 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.
`mse`	A numerical vector with the mean squared error.

Author(s)

Michail Tsagris.

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

See Also

`pls`, `cls`, `mvpls`

Examples

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

`Positively constrained least squares`
Positively constrained least squares

Description

Positively constrained least squares.

Usage

```
pls(y, x)
mpls(y, x)
```

Arguments

`y`	The response variable. For the pls() a numerical vector with observations, but for the mpls() a numerical matrix .
`x`	A matrix with independent variables, the design matrix.

Details

The constraint is that all beta coefficients (including the constant) are non negative, i.e. $min \sum_{i=1}^n (y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})^2$ 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.
`mse`	A numerical vector with the mean squared error(s).

Author(s)

Michail Tsagris.

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

See Also

`cls`, `pcls`, `mvpls`

Examples

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

`Positively constrained least squares with a multivariate response`
Positively constrained least squares with a multivariate response

Description

Positively constrained least squares with a multivariate response.

Usage

```
mvpls(y, x)
```

Arguments

`y`	The response variables, a numerical matrix with observations.
`x`	A matrix with independent variables, the design matrix.

Details

The constraint is that all beta coefficients (including the constant) are positive, i.e. $min \sum_{i=1}^n (\boldsymbol{y}_i - \boldsymbol{x}_i \boldsymbol{\beta})^\top (\boldsymbol{y}_i - \boldsymbol{x}_i \boldsymbol{\beta})$ such that $\beta_{jk} \geq 0$.

Value

A list including:

`be`	The positively constrained beta coefficients.
`mse`	The mean squared error.

Author(s)

Michail Tsagris.

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

See Also

`cls`

Examples

```
y <- as.matrix( iris[, 1:2] )
x <- as.matrix( iris[, 3:4] )
mvpls(y, x)
```

Index