Package 'Compositionalscsmr'

September 12, 2026

Type Package

Title Simplical-Simplicial Spatial Median Regression for Compositional Data

Version 1.0

Date 2026-09-02

Author Michail Tsagris [aut, cre]

Maintainer Michail Tsagris <mtsagris@uoc.gr>

Depends R (>= 4.0)

Imports cluster, Compositional, graphics, MASS, Matrix, quadprog, rangen, Rfast, stats

Suggests Rfast2

Description Simplicial-simplicial regression is performed via the simplicially constrained spatial median regression model. The regression coefficients are constrained to be non-negative and sum to 1. For the spatial median regression the iteratively reweighted least squares algorithm is adopted, where quadratic programming is used to impose the constraints.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2026-09-12 12:50:07 UTC

Contents

`Compositionalscsmr-package`
Simplical-Simplicial Spatial Median Regression for Compositional Data

---

Description

Simplical-Simplicial Spatial Median Regression for Compositional Data.

Details

Package:	Compositionalscsmr
Type:	Package
Version:	1.0
Date:	2026-09-02

Maintainers

Michail Tsagris <mtsagris@uoc.gr>.

Author(s)

Michail Tsagris <mtsagris@uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data.

`Cross-validation for the alpha-SCSMR model`
Cross-validation for the $alpha$-SCSMR model

Description

Cross-validation for the $alpha$-SCSMR model.

Usage

```
cv.ascsmr(y, x, a = seq(0.1, 1, by = 0.1), tol = 1e-6, eps = 1e-6,
nfolds = 10, folds = NULL, seed = NULL)
```

Arguments

`y`	A numerical matrix with the simplicial response data. Zero values are allowed.
`x`	A matrix with the simplicial predictor variables. Zero values are allowed.
`a`	A vector or a single number of values of the α-parameter. This has to be different from zero, and it can take negative values if there are no zeros in the simplicial response (y).
`tol`	The tolerance value to terminate the IRLS algortihm.
`eps`	A small value to prevent or safeguard agains no solution of the quadratic programming.
`nfolds`	The number of folds for the K-fold cross-validation, set to 10 by default.
`folds`	If you have the list with the folds supply it here. You can also leave it NULL and it will create folds.
`seed`	You can specify your own seed number here or leave it NULL.

Details

The K-fold cross-validation is performed in order to select the optimal value for α of the α-SCSMR model.

Value

A list including:

`runtime`	The runtime of the cross-validation procedure.
`kl`	The Kullback-Leibler divergence for every value of α.
`js`	The Jensen-Shannon divergence for every value of α.

Author(s)

Michail Tsagris.

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

References

Tsagris. M. (2025). Constrained least squares simplicial-simplicial regression. Statistics and Computing, 35(27).

See Also

`ascsmr`

Examples

```
y <- as.matrix(iris[, 1:4])
x <- matrix( rgamma(450, 3, 5), ncol = 3)
x <- x / rowSums(x)
mod <- cv.ascsmr(y, x, nfolds = 5)
```

`Permutation linear independence test in the SCSMR model`
Permutation linear independence test in the SCSMR model

Description

Permutation linear independence test in the SCSMR model.

Usage

```
scsmr.indeptest(y, x, tol = 1e-6, eps = 1e-6, R = 999)
```

Arguments

`y`	A matrix with the compositional data (dependent variable). Zero values are allowed.
`x`	A matrix with the compositional predictors. Zero values are allowed.
`tol`	The tolerance value to terminate the IRLS algortihm.
`eps`	A small value to prevent or safeguard agains no solution of the quadratic programming.
`R`	The number of permutations to perform.

Details

Permutation independence test in the constrained spatial median regression for compositional responses and predictors is performed. The observed test statistic is the square root of the of the sum of squares of the errors computed by `scsmr`. Then, the rows of X are permuted B times and each time the constrained spatial median regression is performed and the sum is again computed. The p-value is then computed in the usual way.

Value

The p-value for the test of independence between Y and X.

Author(s)

Michail Tsagris.

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

References

Tsagris. M. (2025). Constrained least squares simplicial-simplicial regression. Statistics and Computing, 35(27).

See Also

`scsmr`, `ascsmr`

Examples

```
y <- as.matrix(iris[, 1:4])
x <- matrix( rgamma(450, 3, 5), ncol = 3)
x <- x / rowSums(x)
scsmr.indeptest(y, x, R = 99)
```

`Ternary diagram with confidence region for the matrix of coefficients of the SCSMR model`

Ternary diagram with confidence region for the matrix of coefficients of the SCSMR model

---

Description

Ternary diagram with confidence region for the matrix of coefficients of the SCSMR model.

Usage

```
ternary.coefcr(y, x, conf = 0.95, R = 1000, dg = FALSE, hg = FALSE)
```

Arguments

`y`	A matrix with the response compositional data.
`x`	A matrix with the predictor compositional data.
`conf`	The confidence level, by default this is set to 0.95.
`R`	Number of bootstrap replicates to run.
`dg`	Do you want diagonal grid lines to appear? If yes, set this TRUE.
`hg`	Do you want horizontal grid lines to appear? If yes, set this TRUE.

Details

This function runs the SCLS or the TFLR model and constructs confidence regions for the estimated matrix of regression coefficients using non-parametric bootstrap.

Value

A ternary plot of the estimated matrix of coefficients of the SCLS or of the TFLR model, and their associated confidence regions.

Author(s)

Michail Tsagris.

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

References

Tsagris. M. (2025). Constrained least squares simplicial-simplicial regression. Statistics and Computing, 35(27).

See Also

ternary.coef, scsmr

Examples

```
y <- as.matrix(iris[, 1:3])
y <- y / rowSums(y)
x <- matrix( rgamma(450, 3, 5), ncol = 3)
```

`Ternary diagram with the coefficients of the SCSMR model`

Ternary diagram with the coefficients of the SCSMR model

Description

Ternary diagram with the coefficients of the SCSMR model.

Usage

```
ternary.coef(B, dg = FALSE, hg = FALSE, colour = NULL)
```

Arguments

B	A matrix with the coefficients of the scsmr function. See examples for this.
dg	Do you want diagonal grid lines to appear? If yes, set this TRUE.
hg	Do you want horizontal grid lines to appear? If yes, set this TRUE.
colour	If you want the points to appear in different colour put a vector with the colour numbers or colours.

Details

Ternary plot of the coefficients of the scsmr function.

Value

A ternary plot of the coefficients of the scsmr function.

Author(s)

Michail Tsagris.

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

References

Tsagris. M. (2025). Constrained least squares simplicial-simplicial regression. Statistics and Computing, 35(27).

See Also

ternary.coefcr, scsmr

Examples

```
y <- as.matrix(iris[, 1:3])
y <- y / rowSums(y)
x <- matrix( rgamma(450, 3, 5), ncol = 3)
mod <- scsmr(y, x)
ternary.coef(mod$be)
```

The alpha-SCSMR model *The α-SCSMR model for compositional responses and predictors*

Description

The α-SCSMR model for compositional responses and predictors.

Usage

```
ascsmr(y, x, a = seq(0.1, 1, by = 0.1), tol = 1e-6, eps = 1e-6, xnew)
```

Arguments

y	A matrix with the compositional data (dependent variable). Zero values are allowed.
x	A matrix with the compositional predictors. Zero values are allowed.
a	A vector or a single number of values of the α-parameter. This has to be different from zero, and it can take negative values if there are no zeros in the simplicial response (y).
tol	The tolerance value to terminate the IRLS algortihm.
eps	A small value to prevent or safeguard agains no solution of the quadratic programming.
xnew	The new data for which predictions will be made.

Details

This is an extension of the SCSMR model that includes the α-transformation and is intended solely for prediction purposes.

Value

A list with matrices containing the predicted simplicial response values, one matrix for each value of α.

Author(s)

Michail Tsagris.

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

References

Tsagris. M. (2025). Constrained least squares simplicial-simplicial regression. Statistics and Computing, 35(27).

See Also

scsmr, cv.ascsmr

Examples

```
set.seed(1234)
y <- as.matrix(iris[, 1:4])
x <- matrix( rgamma(450, 3, 5), ncol = 3)
x <- x / rowSums(x)
mod <- ascsmr(y, x, xnew = x)
mod
```

The SCSMR model *Simplicial constrained spatial median regressio (SCSMR) for compositional responses and predictors*

Description

Simplicial constrained spatial median regressio (SCSMR) for compositional responses and predictors.

Usage

```
scsmr(y, x, xnew = NULL, tol = 1e-06, eps = 1e-6)
```

Arguments

y	A matrix with the compositional data (dependent variable). Zero values are allowed.
x	A matrix with the compositional predictors. Zero values are allowed.
xnew	If you have new data use it, otherwise leave it NULL.
tol	The tolerance value to terminate the IRLS algortihm.
eps	A small value to prevent or safeguard agains no solution of the quadratic programming.

Details

The function performs spatial median regression where the beta coefficients are constained to be positive and sum to 1. The inspiration came from the SCLS model of Tsagris (2025). The function uses quadratic programming (package quadprog) and the solution is fast.

The SCSMR model adopts the same link as TFLR and SCLS between the simplicial response and predictor variables, but only this time the elements of $\mathbf{B}$ are estimated by minimizing the L_1 norm of the residuals

$$\min_{\mathbf{B}} \sum_{k=1}^{n} \|\mathbf{Y}_k - \mathbf{X}_k\mathbf{B}\|_2.$$

The first option is the strategy implemented in the **Rfast** package. The function `spatmed.reg()` employs an iterative algorithm similar to that for the spatial median. The iteratively reweighted least squares (IRLS) algorithm works as follows. Beginning with an initial estimate of the regression coefficients using least squares, $\mathbf{B}^0 \leftarrow (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y}$, the algorithm iterates the following steps until convergence:

1. $\mathbf{e}_k \leftarrow \mathbf{Y}_k - \mathbf{X}_k\mathbf{B}^{(t)}$, $k = 1, \ldots, n$
2. $w_k \leftarrow \sqrt{\sum_{j=1}^{d} \mathbf{e}_{kj}^2}$
3. $\mathbf{Z}_k = \mathbf{X}_k / w_k$ if $w_k \neq 0$ and $\mathbf{Z}_k = 0$ otherwise
4. $\mathbf{B}^{(t+1)} \leftarrow (\mathbf{Z}^\top\mathbf{X})^{-1}\mathbf{Z}^\top\mathbf{Y}$

We adopt the above steps to the simplicial setting by changing Step 4, and instead of the classical least squares we fit a weighted SCLS model. At iteration t of the IRLS algorithm, we need to solve the following constrained weighted least squares problem, which is solved via quadratic programming (QP).

$$\min_{\mathbf{B}} \quad \sum_{k=1}^{n} w_k^{(t)} (Y_k^{(t)} - \mathbf{Z}_k^\top \mathbf{B})^2$$

subject to

$$\sum_{j=1}^{D_p} B_{ij} = 1, \quad i = 1, \ldots, D_r$$

and

$$0 \leq B_{ij} \leq 1, \ \forall i, j.$$

Value

A list including:

`norm`	The mean of the square root of the sum of squares.
`be`	The beta coefficients.
`est`	The fitted of xnew if xnew is not NULL.

Author(s)

Michail Tsagris.

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

References

Tsagris. M. (2025). Constrained least squares simplicial-simplicial regression. Statistics and Computing, 35(27).

See Also

ascsmr scsmr.indeptest

Examples

```
y <- as.matrix(iris[, 1:4])
x <- matrix( rgamma(450, 3, 5), ncol = 3)
x <- x / rowSums(x)
mod <- scsmr(y, x)
mod
```

Index