Package 'compindexR'

November 26, 2023

Type Package

Title Calculates Composite Index

Version 0.1.3

Description It uses the first-order sensitivity index to measure whether the weights assigned by the creator of the composite indicator match the actual importance of the variables. Moreover, the variance inflation factor is used to reduce the set of correlated variables. In the case of a discrepancy between the importance and the assigned weight, the script determines weights that allow adjustment of the weights to the intended impact of variables. If the optimised weights are unable to reflect the desired importance, the highly correlated variables are reduced, taking into account variance inflation factor. The final outcome of the script is the calculated value of the composite indicator based on optimal weights and a reduced set of variables, and the linear ordering of the analysed objects.

License GPL (>= 3)

Encoding UTF-8

URL https://github.com/olgnaydn/compindexR

BugReports https://github.com/olgnaydn/compindexR/issues

Depends R (>= 4.0.0), car (>= 3.1.0), pracma(>= 2.3.8), dplyr(>= 1.0.7), NlcOptim(>= 0.6)

RoxygenNote 7.1.1

NeedsCompilation no

Author Olgun Aydin [cre] (<https://orcid.org/0000-0002-7090-0931>), Marta Kuc-Czarnecka [aut] (<https://orcid.org/0000-0003-2970-9980>), Michal Bernard Pietrzak [aut] (<https://orcid.org/0000-0002-9263-4478>)

Maintainer Olgun Aydin <olgun.aydin@pg.edu.pl>

Repository CRAN

Date/Publication 2023-11-26 00:10:02 UTC

R topics documented:

`calc_average` *Calculate averages*

Description

Calculate different types of averages

Usage

```
calc_average(x, avg_type = "simple")
```

Arguments

`x`	A Dataframe
`avg_type`	Choosing average type. So far "simple", "geometric" and "harmonic" average are availableç

Value

A data frame

Examples

```
x <- data.frame(rnorm(20),rnorm(20),rnorm(20),rnorm(20))
calc_average(x,avg_type = "simple")
```

`calc_compindex` *Calculating composite indicator automatically step by step*

Description

Calculates composite indicator by excluding the least significant variable at each step.

Usage

```
calc_compindex(
  x,
  avg_type = "simple",
  scaling_method = "min-max",
  vif_based_calc = FALSE,
  si_diff = 0.05
)
```

Arguments

`x`	A Dataframe
`avg_type`	Choosing average type. So far "simple", "geometric" and "harmonic" average are available
`scaling_method`	Scaling method selection. So far "min-max" and "standardization" are available
`vif_based_calc`	If TRUE, variable with highest VIF is removed at each step. Default value is FALSE
`si_diff`	Tolerance for normalized Si calculation. Can be between 0 and 1

Value

A list of lists

Examples

```
x <- data.frame(rnorm(20),rnorm(20),rnorm(20),rnorm(20))
calc_compindex(x, avg_type = "simple",
scaling_method = "min-max",
vif_based_calc = FALSE,
si_diff = 0.1)
```

ci_optimizer *Optmization algorithm based on fmincon*

Description

Optmization algorithm based on fmincon

Usage

```
ci_optimizer(x)
```

Arguments

x	A Dataframe

Value

A data frame

Examples

```
x <- data.frame(rnorm(20),rnorm(20),rnorm(20),rnorm(20))
ci_optimizer(x)
```

scaling *Normalization and standardization techniques*

Description

Normalization and standardization techniques

Usage

```
scaling(x, method = "min-max")
```

Arguments

x	A Dataframe
method	Standardization or normalization technique. So far "min-max" and "standardization" are available

Value

A data frame

Examples

```
x <- data.frame(rnorm(20),rnorm(20),rnorm(20),rnorm(20))
scaling(x,method = "min-max")
```

`si_linear` *Calculate Si using linear method*

Description

Calculate Si using linear method

Usage

```
si_linear(x, avg_type = "simple")
```

Arguments

x	A Dataframe
avg_type	Choosing average type. So far "simple", "geometric" and "harmonic" average are availableç

Value

A data frame

Examples

```
x <- data.frame(rnorm(20),rnorm(20),rnorm(20),rnorm(20))
si_linear(x,avg_type = "simple")
```

`si_linear_exc` *Calculate Si using linear method by excluding Xi*

Description

Calculate Si using linear method by excluding Xi at each iteration while calculating Si

Usage

```
si_linear_exc(x, avg_type = "simple")
```

Arguments

x	A Dataframe
avg_type	Choosing average type. So far "simple", "geometric" and "harmonic" average are availableç

Value

A data frame

Examples

```
x <- data.frame(rnorm(20),rnorm(20),rnorm(20),rnorm(20))
si_linear_exc(x,avg_type = "simple")
```

`si_linear_exc_vif` *Calculate Si using linear method by excluding Xi using VIF*

Description

Calculate Si using linear method by excluding Xi using VIF

Usage

```
si_linear_exc_vif(x, avg_type = "simple", vif_threshold = 4.5)
```

Arguments

`x`	A Dataframe
`avg_type`	Choosing average type. So far "simple", "geometric" and "harmonic" average are availableç
`vif_threshold`	Threshold for VIF. Based on this threshold variables from input data (x) are excluded for the calculations.

Value

A data frame

Examples

```
x <- data.frame(rnorm(20),rnorm(20),rnorm(20),rnorm(20))
si_linear_exc_vif(x,avg_type = "simple", vif_threshold = 4.5)
```

Index