Package ‘Compind’

November 20, 2023

Type Package

Title Composite Indicators Functions

Version 2.9.1

Date 2023-11-20

Author Francesco Vidoli, Elisa Fusco

Maintainer Francesco Vidoli <fvidoli@gmail.com>

Description A collection of functions to calculate Composite Indicators methods, focusing, in particular, on the normalisation and weighting-aggregation steps, as described in OECD Handbook on constructing composite indicators: methodology and user guide <https://www.oecd-ilibrary.org/economics/handbook-on-constructing-composite-indicators-methodology-and-user-guide_9789264043466-en>, 'Vidoli' and 'Fusco' and 'Mazziotta' <doi:10.1007/s11205-014-0710-y>, 'Mazziotta' and 'Pareto' (2016) <doi:10.1007/s11205-015-0998-2>, 'Van Puyenbroeck and 'Rogge' <doi:10.1016/j.ejor.2016.07.038> and other authors.

Depends R (>= 3.5.0), Benchmarking, psych, boot, lpSolve, spdep

Imports Hmisc, MASS, GPArotation, nonparaeff, smaa, np, FactoMineR

License GPL-3

Suggests R.rsp

VignetteBuilder R.rsp

NeedsCompilation no

Repository CRAN

Date/Publication 2023-11-20 15:30:02 UTC

R topics documented:

`Compind-package` *Composite Indicators - Compind*

Description

Compind package contains functions to enhance several approaches to the Composite Indicators (CIs) methods, focusing, in particular, on the normalisation and weighting-aggregation steps.

Author(s)

Francesco Vidoli, Elisa Fusco Maintainer: Francesco Vidoli <fvidoli@gmail.com>

References

Daraio, C., Simar, L. (2005) "*Introducing environmental variables in nonparametric frontier models: a probabilistic approach*", Journal of productivity analysis, 24(1), 93-121.

Fusco E. (2015) "*Enhancing non compensatory composite indicators: A directional proposal*", European Journal of Operational Research, 242(2), 620-630.

R. Lahdelma, P. Salminen (2001) "*SMAA-2: Stochastic multicriteria acceptability analysis for group decision making*", Operations Research, 49(3), pp. 444-454

OECD (2008) "*Handbook on constructing composite indicators: methodology and user guide*".

Mazziotta C., Mazziotta M., Pareto A., Vidoli F. (2010) "*La sintesi di indicatori territoriali di dotazione infrastrutturale: metodi di costruzione e procedure di ponderazione a confronto*", Rivista di Economia e Statistica del territorio, n.1.

Melyn W. and Moesen W.W. (1991) "*Towards a synthetic indicator of macroeconomic performance: unequal weighting when limited information is available*", Public Economic research Paper 17, CES, KU Leuven.

Van Puyenbroeck T. and Rogge N. (2017) "*Geometric mean quantity index numbers with Benefit-of-the-Doubt weights*", European Journal of Operational Research, 256(3), 1004-1014.

Rogge N., de Jaeger S. and Lavigne C. (2017) "*Waste Performance of NUTS 2-regions in the EU: A Conditional Directional Distance Benefit-of-the-Doubt Model*", Ecological Economics, vol.139, pp. 19-32.

Simar L., Vanhems A. (2012) "*Probabilistic characterization of directional distances and their robust versions*", Journal of Econometrics, 166(2), 342-354.

UNESCO (1974)"*Social indicators: problems of definition and of selection*", Paris.

Vidoli F., Fusco E., Mazziotta C. (2015) "*Non-compensability in composite indicators: a robust directional frontier method*", Social Indicators Research, 122(3), 635-652.

Vidoli F., Mazziotta C. (2013) "*Robust weighted composite indicators by means of frontier methods with an application to European infrastructure endowment*", Statistica Applicata, Italian Journal of Applied Statistics.

Zanella A., Camanho A.S. and Dias T.G. (2015) "*Undesirable outputs and weighting schemes in composite indicators based on data envelopment analysis*", European Journal of Operational Research, vol. 245(2), pp. 517-530.

`bandwidth_CI` *Multivariate mixed bandwidth selection for exogenous variables*

Description

A function for the selection of optimal multivariate mixed bandwidths for the kernel density estimation of continuous and discrete exogenous variables.

Usage

```
bandwidth_CI(x, indic_col, ngood, nbad, Q=NULL, Q_ord=NULL)
```

Arguments

`x`	A data frame containing simple indicators.
`indic_col`	Simple indicators column number.
`ngood`	The number of desirable outputs; it has to be greater than 0.
`nbad`	The number of undesirable outputs; it has to be greater than 0.
`Q`	A matrix containing continuous exogenous variables.
`Q_ord`	A matrix containing discrete exogenous variables.

Details

Author thanks Nicky Rogge for his help and for making available the original code of the bandwidth function.

Value

bandwidth	A matrix containing the optimal bandwidths for the exogenous variables indicate in Q and Q_ord.
ci_method	"bandwidth_CI

Author(s)

Fusco E., Rogge N.

Examples

```
data(EU_2020)
indic <- c("employ_2011", "gasemiss_2011","deprived_2011")
dat   <- EU_2020[-c(10,18),indic]
Q_GDP <- EU_2020[-c(10,18),"percGDP_2011"]

# Conditional robust BoD Constrained VWR
band = bandwidth_CI(dat, ngood=1, nbad=2, Q = Q_GDP)
```

BLI_2017 *Better Life Index 2017 indicators*

Description

Data related to BLI Edition 2017 (OECD, 2017) for all 38 OECD and non-OECD countries (Data extracted on: 19\02\2020).

For more info, please see https://data-explorer.oecd.org.

Usage

```
data(BLI_2017)
```

Format

BLI_2017 is a dataset with 38 observations and 12 indicators.

country OECD and non-OECD countries.

housing Housing.

income Income and wealth.

jobs Jobs and earnings.

community Community engagement.

education Education.

environment Environment quality.

civic Civic engagement.

health Health.

satisfaction Life satisfaction.

safety Personal security (safety).

worklife Work-Life balance.

Author(s)

Fusco E.

Examples

```
data(BLI_2017)
```

`ci_ampi` *Adjusted Mazziotta-Pareto Index (AMPI) method*

Description

Adjusted Mazziotta-Pareto Index (AMPI) is a non-compensatory composite index that allows to take into account the time dimension, too. The calculation part is similat to the MPI framework, but the standardization part make the scores obtained over the years comparable.

Usage

```
ci_ampi(x, indic_col, gp, time, polarity, penalty = "NEG")
```

Arguments

`x`	A data.frame containing simple indicators in a Long Data Format.
`indic_col`	Simple indicators column number.
`gp`	Goalposts; to facilitate the interpretation of results, the `goalposts` can be chosen so that 100 represents a reference value (e.g., the average in a given year).
`time`	The time variable (mandatory); if the analysis is carried out over a single year, it is necessary to create a constant variable (i.e. `dataframe@year <- 2014`).
`polarity`	Polarity vector: "POS" = positive, "NEG" = negative. The polarity of a individual indicator is the sign of the relationship between the indicator and the phenomenon to be measured (e.g., in a well-being index, "GDP per capita" has 'positive' polarity and "Unemployment rate" has 'negative' polarity).
`penalty`	Penalty direction; Use "NEG" (default) in case of 'increasing' or 'positive' composite index (e.g., well-being index)), "POS" in case of 'decreasing' or 'negative' composite index (e.g., poverty index).

Details

Author thanks Leonardo Alaimo for their help and for making available the original code of the AMPI function. Federico Roscioli for his integrations to the original code.

Value

An object of class "CI". This is a list containing the following elements:

`ci_ampi_est`	Composite indicator estimated values.
`ci_method`	Method used; for this function ci_method="ampi".
`ci_penalty`	Matrix containing penalties only.
`ci_norm`	List containing only the normalised indicators for each year.

Author(s)

Fusco E., Alaimo L., Giovagnoli C., Patelli L., F. Roscioli

References

Mazziotta, M., Pareto, A. (2013) "*A Non-compensatory Composite Index for Measuring Well-being over Time*", Cogito. Multidisciplinary Research Journal Vol. V, no. 4, pp. 93-104

Mazziotta, M., Pareto, A. (2016)."*On a Generalized Non-compensatory Composite Index for Measuring Socio-economic Phenomena*", Cogito. Social Indicators Research, Vol. 127, no. 3, pp. 983-1003

See Also

ci_bod, normalise_ci

Examples

```
data(EU_2020)

data_test = EU_2020[,c("employ_2010","employ_2011","finalenergy_2010","finalenergy_2011")]

EU_2020_long<-reshape(data_test,
               varying=c("employ_2010","employ_2011","finalenergy_2010","finalenergy_2011"),
                      direction="long",
                      idvar="geo",
                      sep="_")

CI <- ci_ampi(EU_2020_long,
               indic_col=c(2:3),
               gp=c(50, 100),
               time=EU_2020_long[,1],
               polarity= c("POS", "POS"),
               penalty="POS")
CI$ci_ampi_est
CI$ci_penalty
CI$ci_norm
```

`ci_bod` *Benefit of the Doubt approach (BoD)*

Description

Benefit of the Doubt approach (BoD) is the application of Data Envelopment Analysis (DEA) to the field of composite indicators. It was originally proposed by Melyn and Moesen (1991) to evaluate macroeconomic performance.

Usage

```
ci_bod(x,indic_col)
```

Arguments

`x`	A data.frame containing simple indicators.
`indic_col`	A numeric list indicating the positions of the simple indicators.

Value

An object of class "CI". This is a list containing the following elements:

`ci_bod_est`	Composite indicator estimated values.
`ci_method`	Method used; for this function ci_method="bod".
`ci_bod_weights`	Raw weights assigned to the simple indicators (Dual values - prices - in the dual DEA formulation).

Author(s)

Vidoli F.

References

OECD (2008) "*Handbook on constructing composite indicators: methodology and user guide*".

Melyn W. and Moesen W.W. (1991) "*Towards a synthetic indicator of macroeconomic performance: unequal weighting when limited information is available*", Public Economic research Paper 17, CES, KU Leuven.

Witte, K. D., Rogge, N. (2009) "*Accounting for exogenous influences in a benevolent performance evaluation of teachers*". Tech. rept. Working Paper Series ces0913, Katholieke Universiteit Leuven, Centrum voor Economische Studien.

See Also

`ci_bod_dir`,`ci_rbod`

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_bod(Indic)
 # validating BoD score
w = CI$ci_bod_weights
Indic[,1]*w[,1] + Indic[,2]*w[,2]


data(EU_NUTS1)
data_norm = normalise_ci(EU_NUTS1,c(2:3),polarity = c("POS","POS"), method=2)
CI = ci_bod(data_norm$ci_norm,c(1:2))
```

`ci_bod_constr` *Constrained Benefit of the Doubt approach (BoD)*

Description

The constrained Benefit of the Doubt function lets to introduce additional constraints to the weight variation in the optimization procedure so that all the weights obtained are greater than a lower value (low_w) and less than an upper value (up_w).

Usage

```
ci_bod_constr(x,indic_col,up_w,low_w)
```

Arguments

- `x` A data.frame containing simple indicators.
- `indic_col` A numeric list indicating the positions of the simple indicators.
- `up_w` Importance weights upper bound.
- `low_w` Importance weights lower bound.

Value

An object of class "CI". This is a list containing the following elements:

- `ci_bod_constr_est` Constrained composite indicator estimated values.
- `ci_method` Method used; for this function ci_method="bod_constrained".
- `ci_bod_constr_weights` Raw constrained weights assigned to the simple indicators.

Author(s)

Rogge N., Vidoli F.

References

Van Puyenbroeck T. and Rogge N. (2017) "*Geometric mean quantity index numbers with Benefit-of-the-Doubt weights*", European Journal of Operational Research, Volume 256, Issue 3, Pages 1004 - 1014.

See Also

ci_bod_dir,ci_bod

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_bod_constr(Indic,up_w=1,low_w=0.05)

data(EU_NUTS1)
data_norm = normalise_ci(EU_NUTS1,c(2:3),polarity = c("POS","POS"), method=2)
CI = ci_bod_constr(data_norm$ci_norm,c(1:2),up_w=1,low_w=0.05)
```

---

ci_bod_constr_bad — *Constrained Benefit of the Doubt approach (BoD) in presence of undesirable indicators*

Description

The constrained Benefit of the Doubt function introduces additional constraints to the weight variation in the optimization procedure (Constrained Virtual Weights Restriction) allowing to restrict the importance attached to a single indicator expressed in percentage terms, ranging between a lower and an upper bound (VWR); this function, furthermore, allows to calculate the composite indicator simultaneously in presence of undesirable (bad) and desirable (good) indicators allowing to impose a preference structure (ordVWR).

Usage

```
ci_bod_constr_bad(x, indic_col, ngood=1, nbad=1, low_w=0, pref=NULL)
```

Arguments

x	A data.frame containing simple indicators; the order is important: first columns must contain the desirable indicators, while second ones the undesirable indicators.
indic_col	A numeric list indicating the positions of the simple indicators.
ngood	The number of desirable outputs; it has to be greater than 0.
nbad	The number of undesirable outputs; it has to be greater than 0.
low_w	Importance weights lower bound.

pref The preference vector among indicators; For example if Indic1 is the most important, Indic2,Indic3 are more important than Indic4 and no preference judgment on Indic5 (= not included in the vector), the pref vector can be written as: c("Indic1", "Indic2","Indic3","Indic4")

Value

An object of class "CI". This is a list containing the following elements:

ci_bod_constr_bad_est
: Composite indicator estimated values.

ci_method
: Method used; for this function ci_method="bod_constr_bad".

ci_bod_constr_bad_weights
: Raw weights assigned to each simple indicator.

ci_bod_constr_bad_target
: Indicator target values.

Author(s)

Fusco E., Rogge N.

References

Rogge N., de Jaeger S. and Lavigne C. (2017) "*Waste Performance of NUTS 2-regions in the EU: A Conditional Directional Distance Benefit-of-the-Doubt Model*", Ecological Economics, vol.139, pp. 19-32.

Zanella A., Camanho A.S. and Dias T.G. (2015) "*Undesirable outputs and weighting schemes in composite indicators based on data envelopment analysis*", European Journal of Operational Research, vol. 245(2), pp. 517-530.

See Also

ci_bod_constr

Examples

```
data(EU_2020)
indic <- c("employ_2011", "percGDP_2011", "gasemiss_2011","deprived_2011")
dat <- EU_2020[-c(10,18),indic]

# BoD Constrained VWR
CI_BoD_C = ci_bod_constr_bad(dat, ngood=2, nbad=2, low_w=0.05, pref=NULL)
CI_BoD_C$ci_bod_constr_bad_est

# BoD Constrained ordVWR
importance <- c("gasemiss_2011","percGDP_2011","employ_2011")
CI_BoD_C = ci_bod_constr_bad(dat, ngood=2, nbad=2, low_w=0.05, pref=importance)
CI_BoD_C$ci_bod_constr_bad_est
```

ci_bod_dir *Directional Benefit of the Doubt (D-BoD) model*

Description

Directional Benefit of the Doubt (D-BoD) model enhance non-compensatory property by introducing directional penalties in a standard BoD model in order to consider the preference structure among simple indicators.

Usage

```
ci_bod_dir(x, indic_col, dir)
```

Arguments

x	A data.frame containing score of the simple indicators.
indic_col	Simple indicators column number.
dir	Main direction. For example you can set the average rates of substitution.

Value

An object of class "CI". This is a list containing the following elements:

ci_bod_dir_est	Composite indicator estimated values.
ci_method	Method used; for this function ci_method="bod_dir".

Author(s)

Vidoli F., Fusco E.

References

Fusco E. (2015) "*Enhancing non compensatory composite indicators: A directional proposal*", European Journal of Operational Research, 242(2), 620-630.

See Also

ci_bod, ci_rbod

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_bod_dir(Indic,dir=c(1,1))

data(EU_NUTS1)
data_norm = normalise_ci(EU_NUTS1,c(2:3),polarity = c("POS","POS"), method=2)
CI = ci_bod_dir(data_norm$ci_norm,c(1:2),dir=c(1,0.5))
```

ci_bod_mdir *Multi-directional Benefit of the Doubt approach (MDBoD)*

Description

Multi-directional Benefit of the Doubt (MDBoD) allows to introduce the non-compensability among simple indicators in a standard BOD in an objective manner: the preference structure, i.e., the direction, is determined directly from the data and is specific for each unit.

Usage

```
ci_bod_mdir(x,indic_col)
```

Arguments

`x`	A data.frame containing simple indicators.
`indic_col`	A numeric list indicating the positions of the simple indicators.

Value

An object of class "CI". This is a list containing the following elements:

`ci_bod_mdir_est`
: Composite indicator estimated values.

`ci_method`
: Method used; for this function ci_method="bod".

`ci_bod_mdir_spec`
: Simple indicators specific scores.

`ci_bod_mdir_dir`
: Directions for each simple indicator and unit.

Author(s)

Fusco E.

References

Fusco E. (2023) "*Potential improvements approach in composite indicators construction: the Multi-directional Benefit of the Doubt model*", Socio-Economic Planning Sciences, vol. 85, 101447

See Also

`ci_bod_dir`, `mea`

Examples

```
data(BLI_2017)
CI <- ci_bod_mdir(BLI_2017,c(2:12))
```

`ci_bod_var_w` *Variance weighted Benefit of the Doubt approach (BoD variance weighted)*

Description

Variance weighted Benefit of the Doubt approach (BoD variance weighted) is a particular form of BoD method with additional information in the optimization problem. In particular it has been added weight constraints (in form of an Assurance region type I (AR I)) endogenously determined in order to take into account the ratio of the vertical variability of each simple indicator relative to one another.

Usage

```
ci_bod_var_w(x,indic_col,boot_rep = 5000)
```

Arguments

`x`	A data.frame containing score of the simple indicators.
`indic_col`	Simple indicators column number.
`boot_rep`	The number of bootstrap replicates (default=5000) for the estimates of the non-parametric bootstrap (first order normal approximation) confidence intervals for the variances of the simple indicators.

Details

For more informations about the estimation of the confidence interval for the variances, please see function *boot.ci*, package *boot*.

Value

An object of class "CI". This is a list containing the following elements:

`ci_bod_var_w_est`	Composite indicator estimated values.
`ci_method`	Method used; for this function ci_method="bod_var_w".

Author(s)

Vidoli F.

References

Vidoli F., Mazziotta C. (2013) "*Robust weighted composite indicators by means of frontier methods with an application to European infrastructure endowment*", Statistica Applicata, Italian Journal of Applied Statistics.

See Also

ci_bod, ci_rbod

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_bod_var_w(Indic)
```

---

ci_factor — *Weighting method based on Factor Analysis*

Description

Factor analysis groups together collinear simple indicators to estimate a composite indicator that captures as much as possible of the information common to individual indicators.

Usage

```
ci_factor(x,indic_col,method="ONE",dim)
```

Arguments

x	A data.frame containing score of the simple indicators.
indic_col	Simple indicators column number.
method	If method = "ONE" (default) the composite indicator estimated values are equal to first component scores; if method = "ALL" the composite indicator estimated values are equal to component score multiplied by its proportion variance; if method = "CH" it can be choose the number of the component to take into account.
dim	Number of chosen component (if method = "CH", default is 3).

Value

An object of class "CI". This is a list containing the following elements:

ci_factor_est	Composite indicator estimated values.
loadings_fact	Variance explained by principal factors (in percentage terms).
ci_method	Method used; for this function ci_method="factor".

Author(s)

Vidoli F.

References

OECD (2008) "*Handbook on constructing composite indicators: methodology and user guide*".

See Also

ci_bod, ci_mpi

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_factor(Indic)

data(EU_NUTS1)
CI = ci_factor(EU_NUTS1,c(2:3), method="ALL")

data(EU_2020)
data_norm = normalise_ci(EU_2020,c(47:51),polarity = c("POS","POS","POS","POS","POS"), method=2)
CI3 = ci_factor(data_norm$ci_norm,c(1:5),method="CH", dim=3)
```

---

ci_factor_mixed — *Weighting method based on Factor analysis of mixed data (FAMD)*

Description

Factor analysis of mixed data (FAMD) can be seen as a principal component method dedicated to analyze a data set containing both quantitative and qualitative variables making possible to compute composite indicators taking into account continous, dummy, or factor variables

Usage

```
ci_factor_mixed(x,indic_col,method="ONE",dim)
```

Arguments

x	A data.frame containing score of the simple indicators.
indic_col	Simple indicators column number.
method	If method = "ONE" (default) the composite indicator estimated values are equal to first component scores; if method = "ALL" the composite indicator estimated values are equal to component score multiplied by its proportion variance; if method = "CH" it can be choose the number of the component to take into account.
dim	Number of chosen component (if method = "CH", default is 3).

Value

An object of class "CI". This is a list containing the following elements:

- `ci_factor_est` Composite indicator estimated values.
- `loadings_fact` Variance explained by principal factors (in percentage terms).
- `ci_method` Method used; for this function ci_method="factor_mixed".

Author(s)

Luis Carlos Castillo Tellez

See Also

`ci_bod`, `ci_factor`

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
i3 <- seq(0, 1, len = 100)
i3 = as.factor(ifelse(i3>0.5,1,0))
Indic = data.frame(i1, i2, i3)

CI  = ci_factor_mixed(Indic,c(1:3))
CI2 = ci_factor_mixed(Indic,c(1:3), method="ALL")
CI3 = ci_factor_mixed(Indic,c(1:3), method="CH", dim=2)
```

`ci_generalized_mean` *Weighting method based on generalized mean*

Description

Generalized means are a family of functions for aggregating sets of numbers (it include as special cases the Pythagorean means, arithmetic, geometric, and harmonic means). The generalized mean is also known as power mean or Holder mean.

Usage

```
ci_generalized_mean(x, indic_col, p, na.rm=TRUE)
```

Arguments

- `x` A data.frame containing simple indicators.
- `indic_col` Simple indicators column number.
- `p` Exponent p (real number).
- `na.rm` Remove NA values before processing; default is TRUE.

Value

An object of class "CI". This is a list containing the following elements:

`ci_generalized_mean_est`
: Composite indicator estimated values.

`ci_method`
: Method used; for this function ci_method="generalized_mean".

Note

The *generalized mean* with the exponent p can be espressed as:

$$M_p(I_1, \ldots, I_n) = \left(\frac{1}{n} \sum_{i=1}^{n} I_i^p \right)^{\frac{1}{p}}$$

Particular case are: $p = -\infty$: minimum, $p = -1$: harmonic mean, $p = 0$: geometric mean, $p = 1$: arithmetic mean, $p = 2$: root-mean-square and $p = \infty$: maximum.

Author(s)

Vidoli F.

See Also

`ci_geom_gen`, `ci_factor`

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_generalized_mean(Indic, p=-1) # harmonic mean

data(EU_NUTS1)
CI = ci_generalized_mean(EU_NUTS1,c(2:3),p=2) # geometric mean
```

`ci_geom_bod_intertemp` *Intertemporal analysis for geometric mean quantity index numbers*

Description

Intertemporal analysis for geometric mean quantity index numbers with Benefit-of-the-Doubt weights - see function `ci_bod_constr`.

Usage

```
ci_geom_bod_intertemp(x0,x1,indic_col,up_w,low_w,bench)
```

Arguments

x0	A data.frame containing simple indicators - time 0
x1	A data.frame containing simple indicators - time 1
indic_col	A numeric list indicating the positions of the simple indicators.
up_w	Weights upper bound.
low_w	Weights lower bound.
bench	Row number of the benchmark unit

Value

An object of class "CI". This is a list containing the following elements:

ci_geom_bod_intertemp_est
: A matrix containing the Overall Change (period t1 vs t0), the Change Effect (period t1 vs t0), the Benchmark Effect (period t1 vs t0) and Weight Effect (period t1 vs t0).

ci_method
: Method used; for this function ci_method="Intertemporal_effects_Geometric_BoD".

Author(s)

Rogge N., Vidoli F.

References

Van Puyenbroeck T. and Rogge N. (2017) "*Geometric mean quantity index numbers with Benefit-of-the-Doubt weights*", European Journal of Operational Research, Volume 256, Issue 3, Pages 1004 - 1014

See Also

ci_bod_constr,ci_bod

Examples

```
i1_t1 <- seq(0.3, 0.5, len = 100)
i2_t1 <- seq(0.3, 1, len = 100)
Indic_t1 = data.frame(i1_t1, i2_t1)

i1_t0 <- i1_t1 - rnorm (100, 0.2, 0.03)
i2_t0 <- i2_t1   - rnorm (100, 0.2, 0.03)
Indic_t0 = data.frame(i1_t0, i2_t0)

intertemp = ci_geom_bod_intertemp(Indic_t0,Indic_t1,c(1:2),up_w=0.95,low_w=0.05,1)
intertemp
```

ci_geom_gen *Generalized geometric mean quantity index numbers*

Description

This function use the geometric mean to aggregate the single indicators. Two weighting criteria has been implemented: EQUAL: equal weighting and BOD: Benefit-of-the-Doubt weights following the Puyenbroeck and Rogge (2017) approach.

Usage

```
ci_geom_gen(x,indic_col,meth,up_w,low_w,bench)
```

Arguments

x	A data.frame containing simple indicators.
indic_col	A numeric list indicating the positions of the simple indicators.
meth	"EQUAL" = Equal weighting set, "BOD" = Benefit-of-the-Doubt weighting set.
up_w	if meth="BOD"; upper bound of the weighting set.
low_w	if meth="BOD"; lower bound of the weighting set.
bench	Row number of the benchmark unit used to normalize the data.frame x.

Value

An object of class "CI". This is a list containing the following elements:

If meth = "EQUAL":

`ci_mean_geom_est` : Composite indicator estimated values.

`ci_method` : Method used; for this function ci_method="mean_geom".

If meth = "BOD":

`ci_geom_bod_est` : Constrained composite indicator estimated values.

`ci_geom_bod_weights` : Raw constrained weights assigned to the simple indicators.

`ci_method` : Method used; for this function ci_method="geometric_bod".

Author(s)

Rogge N., Vidoli F.

References

Van Puyenbroeck T. and Rogge N. (2017) "*Geometric mean quantity index numbers with Benefit-of-the-Doubt weights*", European Journal of Operational Research, Volume 256, Issue 3, Pages 1004 - 1014

See Also

ci_bod_dir,ci_bod

Examples

```
i1 <- seq(0.3, 1, len = 100) - rnorm (100, 0.1, 0.03)
i2 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.1, 0.03)
i3 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.1, 0.03)
Indic = data.frame(i1, i2,i3)

geom1 = ci_geom_gen(Indic,c(1:3),meth = "EQUAL")
geom1$ci_mean_geom_est
geom1$ci_method


geom2 = ci_geom_gen(Indic,c(1:3),meth = "BOD",0.7,0.3,100)
geom2$ci_geom_bod_est
geom2$ci_geom_bod_weights
```

---

ci_mean_min *Mean-Min Function*

Description

The Mean-Min Function (MMF) is an intermediate case between arithmetic mean, according to which no unbalance is penalized, and min function, according to which the penalization is maximum. It depends on two parameters that are respectively related to the intensity of penalization of unbalance (α) and intensity of complementarity (β) among indicators.

Usage

```
ci_mean_min(x, indic_col, alpha, beta)
```

Arguments

x	A data.frame containing simple indicators.
indic_col	Simple indicators column number.
alpha	The intensity of penalisation of unbalance among indicators, $0 \leq \alpha \leq 1$
beta	The intensity of complementarity among indicators, $\beta \geq 0$

Value

An object of class "CI". This is a list containing the following elements:

ci_mean_min_est	Composite indicator estimated values.
ci_method	Method used; for this function ci_method="mean_min".

Author(s)

Vidoli F.

References

Casadio Tarabusi, E., & Guarini, G. (2013) "*An unbalance adjustment method for development indicators*", Social indicators research, 112(1), 19-45.

See Also

ci_mpi, normalise_ci

Examples

```
data(EU_NUTS1)
data_norm = normalise_ci(EU_NUTS1,c(2:3),c("NEG","POS"),method=2)
CI = ci_mean_min(data_norm$ci_norm, alpha=0.5, beta=1)
```

---

ci_mpi *Mazziotta-Pareto Index (MPI) method*

Description

Mazziotta-Pareto Index (MPI) is a non-linear composite index method which transforms a set of individual indicators in standardized variables and summarizes them using an arithmetic mean adjusted by a "penalty" coefficient related to the variability of each unit (method of the coefficient of variation penalty).

Usage

```
ci_mpi(x, indic_col, penalty="POS")
```

Arguments

x	A data.frame containing simple indicators.
indic_col	Simple indicators column number.
penalty	Penalty direction; Use "POS" (default) in case of 'increasing' or 'positive' composite index (e.g., well-being index)), "NEG" in case of 'decreasing' or 'negative' composite index (e.g., poverty index).

Value

An object of class "CI". This is a list containing the following elements:

ci_mpi_est	Composite indicator estimated values.
ci_method	Method used; for this function ci_method="mpi".

Author(s)

Vidoli F.

References

De Muro P., Mazziotta M., Pareto A. (2011), "*Composite Indices of Development and Poverty: An Application to MDGs*", Social Indicators Research, Volume 104, Number 1, pp. 1-18.

See Also

ci_bod, normalise_ci

Examples

```
data(EU_NUTS1)

# Please, pay attention. MPI can be calculated only with two standardizations methods:
# Classic MPI - method=1, z.mean=100 and z.std=10
# Correct MPI - method=2
# For more info, please see references.

data_norm = normalise_ci(EU_NUTS1,c(2:3),c("NEG","POS"),method=1,z.mean=100, z.std=10)
CI = ci_mpi(data_norm$ci_norm, penalty="NEG")

data(EU_NUTS1)
CI = ci_mpi(EU_NUTS1,c(2:3),penalty="NEG")
```

---

ci_rbod *Robust Benefit of the Doubt approach (RBoD)*

Description

Robust Benefit of the Doubt approach (RBoD) is the robust version of the BoD method. It is based on the concept of the expected minimum input function of order-*m* so "*in place of looking for the lower boundary of the support of F, as was typically the case for the full-frontier (DEA or FDH), the order-m efficiency score can be viewed as the expectation of the maximal score, when compared to m units randomly drawn from the population of units presenting a greater level of simple indicators*", Daraio and Simar (2005).

Usage

```
ci_rbod(x,indic_col,M,B)
```

Arguments

x	A data.frame containing score of the simple indicators.
indic_col	Simple indicators column number.
M	The number of elements in each of the bootstrapped samples.
B	The number of bootstrap replicates.

Value

An object of class "CI". This is a list containing the following elements:

- `ci_rbod_est` Composite indicator estimated values.
- `ci_method` Method used; for this function ci_method="rbod".

Author(s)

Vidoli F.

References

Daraio, C., Simar, L. "*Introducing environmental variables in nonparametric frontier models: a probabilistic approach*", Journal of productivity analysis, 2005, 24(1), 93 - 121.

Vidoli F., Mazziotta C., "*Robust weighted composite indicators by means of frontier methods with an application to European infrastructure endowment*", Statistica Applicata, Italian Journal of Applied Statistics, 2013.

See Also

ci_bod

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_rbod(Indic,B=10)

data(EU_NUTS1)
data_norm = normalise_ci(EU_NUTS1,c(2:3),polarity = c("POS","POS"), method=2)
CI = ci_rbod(data_norm$ci_norm,c(1:2),M=10,B=20)
```

---

ci_rbod_constr_bad   *Robust constrained Benefit of the Doubt approach (BoD) in presence of undesirable indicators*

Description

The Robust constrained Benefit of the Doubt function introduces additional constraints to the weight variation in the optimization procedure (Constrained Virtual Weights Restriction) allowing to restrict the importance attached to a single indicator expressed in percentage terms, ranging between a lower and an upper bound (VWR); this function, furthermore, allows to calculate the composite indicator simultaneously in presence of undesirable (bad) and desirable (good) indicators allowing to impose a preference structure (ordVWR). This function is the robust version of the `ci_bod_constr_bad`: it is based on the concept of the expected minimum input function of order-*m* (Daraio and Simar, 2005) allowing to compare the unit under analysis against `M` peers by extracting `B` samples with replacement.

Usage

```
ci_rbod_constr_bad(x, indic_col, ngood=1, nbad=1, low_w=0, pref=NULL, M, B)
```

Arguments

`x`	A data.frame containing simple indicators.
`indic_col`	A numeric list indicating the positions of the simple indicators.
`ngood`	The number of desirable outputs; it has to be greater than 0.
`nbad`	The number of undesirable outputs; it has to be greater than 0.
`low_w`	Importance weights lower bound.
`pref`	The preference vector among indicators; For example if `Indic1` is the most important, `Indic2`,`Indic3` are more important than `Indic4` and no preference judgment on `Indic5` (= not included in the vector), the `pref` vector can be written as: `c("Indic1", "Indic2","Indic3","Indic4")`
`M`	The number of elements in each of the bootstrapped samples.
`B`	The number of bootstrap replicates.

Value

An object of class "CI". This is a list containing the following elements:

`ci_rbod_constr_bad_est`
: Composite indicator estimated values.

`ci_method`
: Method used; for this function ci_method="rbod_constr_bad".

`ci_rbod_constr_bad_weights`
: Raw weights assigned to each simple indicator.

`ci_rbod_constr_bad_target`
: Indicator target values.

Author(s)

Fusco E., Rogge N.

References

Rogge N., de Jaeger S. and Lavigne C. (2017) "*Waste Performance of NUTS 2-regions in the EU: A Conditional Directional Distance Benefit-of-the-Doubt Model*", Ecological Economics, vol.139, pp. 19-32.

Zanella A., Camanho A.S. and Dias T.G. (2015) "*Undesirable outputs and weighting schemes in composite indicators based on data envelopment analysis*", European Journal of Operational Research, vol. 245(2), pp. 517-530.

See Also

`ci_bod_constr`, `ci_bod_constr_bad`

Examples

```
data(EU_2020)
indic <- c("employ_2011", "percGDP_2011", "gasemiss_2011","deprived_2011")
dat <- EU_2020[-c(10,18),indic]

# Robust BoD Constrained VWR
CI_BoD_C = ci_rbod_constr_bad(dat, ngood=2, nbad=2, low_w=0.05, pref=NULL, M=10, B=50)
CI_BoD_C$ci_rbod_constr_bad_est

# Robust BoD Constrained ordVWR
importance <- c("gasemiss_2011","percGDP_2011","employ_2011")
CI_BoD_C = ci_rbod_constr_bad(dat, ngood=2, nbad=2, low_w=0.05, pref=importance, M=10, B=50)
CI_BoD_C$ci_rbod_constr_bad_est
```

`ci_rbod_constr_bad_Q` *Conditional robust constrained Benefit of the Doubt approach (BoD) in presence of undesirable indicators*

Description

The Conditional robust constrained Benefit of the Doubt function introduces additional constraints to the weight variation in the optimization procedure (Constrained Virtual Weights Restriction) allowing to restrict the importance attached to a single indicator expressed in percentage terms, ranging between a lower and an upper bound (VWR); this function, furthermore, allows to calculate the composite indicator simultaneously in presence of undesirable (bad) and desirable (good) indicators allowing to impose a preference structure (ordVWR). This function, in addition to being robust against outlier data (see `ci_rbod_constr_bad` function) allows to take into account external contextual continuous (`Q`) or/and ordinal (`Q_ord`) variables.

Usage

```
ci_rbod_constr_bad_Q(x, indic_col, ngood=1, nbad=1,
low_w=0, pref=NULL, M, B, Q=NULL, Q_ord=NULL, bandwidth)
```

Arguments

`x`	A data.frame containing simple indicators.
`indic_col`	A numeric list indicating the positions of the simple indicators.
`ngood`	The number of desirable outputs; it has to be greater than 0.
`nbad`	The number of undesirable outputs; it has to be greater than 0.
`low_w`	Importance weights lower bound.
`pref`	The preference vector among indicators; For example if `Indic1` is the most important, `Indic2`,`Indic3` are more important than `Indic4` and no preference judgment on `Indic5` (= not included in the vector), the `pref` vector can be written as: `c("Indic1", "Indic2","Indic3","Indic4")`
`M`	The number of elements in each of the bootstrapped samples.

B	The number of bootstrap replicates.
Q	A matrix containing continuous exogenous variables.
Q_ord	A matrix containing discrete exogenous variables.
bandwidth	Multivariate mixed bandwidth for exogenous variables; it can be calculated by bandwidth_CI function.

Value

An object of class "CI". This is a list containing the following elements:

ci_rbod_constr_bad_Q_est
: Composite indicator estimated values.

ci_method
: Method used; for this function ci_method="rbod_constr_bad_Q".

ci_rbod_constr_bad_Q_weights
: Raw weights assigned to each simple indicator.

ci_rbod_constr_bad_Q_target
: Indicator target values.

Author(s)

Fusco E., Rogge N.

References

Rogge N., de Jaeger S. and Lavigne C. (2017) "*Waste Performance of NUTS 2-regions in the EU: A Conditional Directional Distance Benefit-of-the-Doubt Model*", Ecological Economics, vol.139, pp. 19-32.

Zanella A., Camanho A.S. and Dias T.G. (2015) "*Undesirable outputs and weighting schemes in composite indicators based on data envelopment analysis*", European Journal of Operational Research, vol. 245(2), pp. 517-530.

See Also

ci_rbod_constr_bad, ci_bod_constr_bad

Examples

```
data(EU_2020)

indic <- c("employ_2011", "gasemiss_2011","deprived_2011")
dat   <- EU_2020[-c(10,18),indic]
Q_GDP <- EU_2020[-c(10,18),"percGDP_2011"]

# Conditional robust BoD Constrained VWR
band = bandwidth_CI(dat, ngood=1, nbad=2, Q = Q_GDP)

CI_BoD_C = ci_rbod_constr_bad_Q(dat,
                                ngood=1,
                                nbad=2,
```

```
                                low_w=0.05,
                                pref=NULL,
                                M=10,
                                B=50,
                                Q=Q_GDP,
                                bandwidth = band$bandwidth)
CI_BoD_C$ci_rbod_constr_bad_Q_est


# # Conditional robust BoD Constrained ordVWR
# import <- c("gasemiss_2011","employ_2011", "deprived_2011")
#
# CI_BoD_C2 = ci_rbod_constr_bad_Q(dat,
#                                 ngood=1,
#                                 nbad=2,
#                                 low_w=0.05,
#                                 pref=import,
#                                 M=10,
#                                 B=50,
#                                 Q=Q_GDP,
#                                 bandwidth = band$bandwidth)
# CI_BoD_C2$ci_rbod_constr_bad_Q_est
```

`ci_rbod_dir` *Directional Robust Benefit of the Doubt approach (D-RBoD)*

Description

Directional Robust Benefit of the Doubt approach (D-RBoD) is the directional robust version of the BoD method.

Usage

```
ci_rbod_dir(x,indic_col,M,B,dir)
```

Arguments

`x`	A data.frame containing score of the simple indicators.
`indic_col`	Simple indicators column number.
`M`	The number of elements in each of the bootstrapped samples.
`B`	The number of bootstap replicates.
`dir`	Main direction. For example you can set the average rates of substitution.

Value

An object of class "CI". This is a list containing the following elements:

`ci_rbod_dir_est`	Composite indicator estimated values.
`ci_method`	Method used; for this function ci_method="rbod_dir".

Author(s)

Fusco E., Vidoli F.

References

Daraio C., Simar L., "*Introducing environmental variables in nonparametric frontier models: a probabilistic approach*", Journal of productivity analysis, 2005, 24(1), 93 121.

Simar L., Vanhems A., "*Probabilistic characterization of directional distances and their robust versions*", Journal of Econometrics, 2012, 166(2), 342 354.

Vidoli F., Fusco E., Mazziotta C., "*Non-compensability in composite indicators: a robust directional frontier method*", Social Indicators Research, Springer Netherlands.

See Also

ci_bod, ci_rbod

Examples

```
data(EU_NUTS1)
data_norm = normalise_ci(EU_NUTS1,c(2:3),polarity = c("POS","POS"), method=2)
CI = ci_rbod_dir(data_norm$ci_norm, c(1:2), M = 25, B = 50, c(1,0.1))
```

---

ci_rbod_spatial — *Spatial robust Benefit of the Doubt approach (Sp-RBoD)*

Description

The Spatial robust Benefit of the Doubt approach (Sp-RBoD) method allows to take into account the spatial contextual condition into the robust Benefit of the Doubt method.

Usage

```
ci_rbod_spatial(x, indic_col, M=20, B=100, W)
```

Arguments

x	A data.frame containing score of the simple indicators.
indic_col	Simple indicators column number.
M	The number of elements in each of the bootstrapped samples; default is 20.
B	The number of bootstrap replicates; default is 100.
W	The spatial weights matrix. A square non-negative matrix with no NAs representing spatial weights; may be a matrix of class "sparseMatrix" (spdep package)

Value

An object of class "CI". This is a list containing the following elements:

`ci_rbod_spatial_est`
: Composite indicator estimated values.

`ci_method`
: Method used; for this function ci_method="rbod_spatial".

Author(s)

Fusco E., Vidoli F.

References

Fusco E., Vidoli F., Sahoo B.K. (2018) "*Spatial heterogeneity in composite indicator: a methodological proposal*", Omega, Vol. 77, pp. 1-14

See Also

ci_rbod

Examples

```
data(EU_NUTS1)

coord = EU_NUTS1[,c("Long","Lat")]
k<-knearneigh(as.matrix(coord), k=5)
k_nb<-knn2nb(k)
W_mat <-nb2mat(k_nb,style="W",zero.policy=TRUE)

CI = ci_rbod_spatial(EU_NUTS1,c(2:3),M=10,B=20, W=W_mat)
```

---

ci_smaa_constr — *Constrained stochastic multi-objective acceptability analysis (C-SMAA)*

Description

Stochastic multiobjective acceptability analysis (SMAA) is a multicriteria decision support technique for multiple decision makers based on exploring the weight space. Inaccurate or uncertain input data can be represented as probability distributions. In SMAA the decision makers need not express their preferences explicitly or implicitly; instead the technique analyses what kind of valuations would make each alternative the preferred one. The method produces for each alternative an acceptability index measuring the variety of different valuations that support that alternative, a central weight vector representing the typical valuations resulting in that decision, and a confidence factor measuring whether the input data is accurate enough for making an informed decision. (R Lahdelma, J. Hokkanen and P. Salminen, 1998); this function, in particular, allows to restricts the range of allowable weights within the SMAA analysis.

Usage

```
ci_smaa_constr(x,indic_col,rep, label, low_w=NULL)
```

Arguments

`x`	A data.frame containing simple indicators.
`indic_col`	A numeric list indicating the positions of the simple indicators.
`rep`	Number of samples.
`label`	A factor column useful to identify units.
`low_w`	Importance weights lower bound vector; default is NULL (for standard SMAA)

Details

Author thanks Giuliano Resce and Raffaele Lagravinese for their help and for making available the original code of the SMAA function.\ The lower bound vector must be set as a vector of the same size as the number of simple indicators; for example - in the presence of two indicators - if you want to constrain only one indicator, you must write: `low_w = c (0,0.2)`.

Value

An object of class "CI". This is a list containing the following elements:

`ci_smaa_constr_rank_freq`
: Frequence of the SMAA ranks based on the sampled alternatives' values. The rows represent the analysis units while the first column represents the number of times the unit was in first rank, the second one in second rank and so on.

`ci_smaa_constr_average_rank`
: The average rank.

`ci_smaa_constr_values`
: The alternative values based on a set of samples from the criteria values distribution and the samples set from the feasible weight space.

`ci_method`
: Method used; for this function ci_method="smaa_const".

Author(s)

Vidoli F.

References

R. Lahdelma, P. Salminen (2001) "*SMAA-2: Stochastic multicriteria acceptability analysis for group decision making*", Operations Research, 49(3), pp. 444-454

S. Greco, A. Ishizaka, B. Matarazzo and G. Torrisi (2017) "*Stochastic multi-attribute acceptability analysis (SMAA): an application to the ranking of Italian regions*", Regional Studies

R. Lagravinese, P. Liberati and G. Resce (2017) "*Exploring health outcomes by stochastic multi-objective acceptability analysis: an application to Italian regions*", Working Papers. Collection B: Regional and sectoral economics, 1703, Universidade de Vigo, GEN - Governance and Economics research Network.

See Also

ci_bod

Examples

```
# ----- Define a function for plotting a matrix ----- #
myImagePlot <- function(x, ...){
     min <- min(x)
     max <- max(x)
     yLabels <- rownames(x)
     xLabels <- colnames(x)
     title <-c()
  # check for additional function arguments
  if( length(list(...)) ){
    Lst <- list(...)
    if( !is.null(Lst$zlim) ){
       min <- Lst$zlim[1]
       max <- Lst$zlim[2]
    }
    if( !is.null(Lst$yLabels) ){
       yLabels <- c(Lst$yLabels)
    }
    if( !is.null(Lst$xLabels) ){
       xLabels <- c(Lst$xLabels)
    }
    if( !is.null(Lst$title) ){
       title <- Lst$title
    }
  }
# check for null values
if( is.null(xLabels) ){
   xLabels <- c(1:ncol(x))
}
if( is.null(yLabels) ){
   yLabels <- c(1:nrow(x))
}

layout(matrix(data=c(1,2), nrow=1, ncol=2), widths=c(4,1), heights=c(1,1))

 # Red and green range from 0 to 1 while Blue ranges from 1 to 0
 ColorRamp <- rgb( seq(0,1,length=256),  # Red
                   seq(0,1,length=256),  # Green
                   seq(1,0,length=256))  # Blue
 ColorLevels <- seq(min, max, length=length(ColorRamp))

 # Reverse Y axis
 reverse <- nrow(x) : 1
 yLabels <- yLabels[reverse]
 x <- x[reverse,]
```

```
 # Data Map
 par(mar = c(3,5,2.5,2))
 image(1:length(xLabels), 1:length(yLabels), t(x), col=ColorRamp, xlab="",
 ylab="", axes=FALSE, zlim=c(min,max))
 if( !is.null(title) ){
    title(main=title)
 }
axis(BELOW<-1, at=1:length(xLabels), labels=xLabels, cex.axis=0.7)
 axis(LEFT <-2, at=1:length(yLabels), labels=yLabels, las= HORIZONTAL<-1,
 cex.axis=0.7)

 # Color Scale
 par(mar = c(3,2.5,2.5,2))
 image(1, ColorLevels,
      matrix(data=ColorLevels, ncol=length(ColorLevels),nrow=1),
      col=ColorRamp,
      xlab="",ylab="",
      xaxt="n")

 layout(1)
}
# ----- END plot function ----- #


data(EU_NUTS1)

# Standard SMAA
test <- ci_smaa_constr(EU_NUTS1,c(2,3), rep=200, label = EU_NUTS1[,1])
# source("http://www.phaget4.org/R/myImagePlot.R")
# myImagePlot(test$ci_smaa_constr_rank_freq)
test$ci_smaa_constr_average_rank

# Constrained SMAA
test2 <- ci_smaa_constr(EU_NUTS1,c(2,3), rep=200, label = EU_NUTS1[,1], low_w=c(0.2,0.2) )
# myImagePlot(test2$ci_smaa_constr_rank_freq)
test2$ci_smaa_constr_average_rank
```

---

ci_wroclaw *Wroclaw Taxonomic Method*

Description

Wroclaw taxonomy method (also known as the dendric method), originally developed at the University of Wroclaw, is based on the distance from a theoretical unit characterized by the best performance for all indicators considered; the composite indicator is therefore based on the sum of euclidean distances from the ideal unit and normalized by a measure of variability of these distance (mean + 2*std).

Usage

```
ci_wroclaw(x,indic_col)
```

Arguments

x	A data.frame containing simple indicators.
indic_col	Simple indicators column number.

Details

Please pay attention that *ci_wroclaw_est* is the distance from the "ideal" unit; so, units with higher values for the simple indicators get lower values of composite indicator.

Value

An object of class "CI". This is a list containing the following elements:

ci_wroclaw_est	Composite indicator estimated values.
ci_method	Method used; for this function ci_method="wroclaw".

Author(s)

Vidoli F.

References

UNESCO, "*Social indicators: problems of definition and of selection*", Paris 1974.

Mazziotta C., Mazziotta M., Pareto A., Vidoli F., "*La sintesi di indicatori territoriali di dotazione infrastrutturale: metodi di costruzione e procedure di ponderazione a confronto*", Rivista di Economia e Statistica del territorio, n.1, 2010.

See Also

ci_bod, ci_mpi

Examples

```
i1 <- seq(0.3, 0.5, len = 100) - rnorm (100, 0.2, 0.03)
i2 <- seq(0.3, 1, len = 100)   - rnorm (100, 0.2, 0.03)
Indic = data.frame(i1, i2)
CI = ci_wroclaw(Indic)

data(EU_NUTS1)
CI = ci_wroclaw(EU_NUTS1,c(2:3))

data(EU_2020)
data_selez = EU_2020[,c(1,22,191)]
data_norm = normalise_ci(data_selez,c(2:3),c("POS","NEG"),method=3)
ci_wroclaw(data_norm$ci_norm,c(1:2))
```

`data_HPI` *Happy Planet Index 2017-2019 indicators*

Description

Data related to Happy Planet Index for 151 countries and the period 2017-2019.

For more info, please see https://happyplanetindex.org.

Usage

```
data(data_HPI)
```

Format

data_HPI is a dataset with 453 observations and 10 variables.

Country Country name

ISO ISO code

year Years 2017-2019

Continent Continent

Population Population (thousands)

Life_Expectancy Life Expectancy (years)

Ladder_of_life Ladder of life (Wellbeing) (0-10)

Ecological_Footprint Ecological Footprint (g ha)

HPI HPI

GDP_per_capita GDP per capita ($)

Author(s)

Fusco E.

References

https://happyplanetindex.org

Examples

```
data(data_HPI)
```

EU_2020 *Europe 2020 indicators*

Description

Europe 2020, a strategy for jobs and smart, sustainable and inclusive growth, is based on five EU headline targets which are currently measured by eight headline indicators, Headline indicators, Eurostat, year 1990-2012 (Last update: 21/11/2013).

For more info, please see https://ec.europa.eu/eurostat/en/web/products-statistics-in-focus/-/KS-SF-12-039.

Usage

```
data(EU_2020)
```

Format

EU_2020 is a dataset with 30 observations and 12 indicators (190 indicator per year).

geo EU-Member States including EU (28 countries) and EU (27 countries) row.

employXXXX Employment rate - age group 20-64, year XXXX (1992-2012).

perc_GDPXXXX Gross domestic expenditure on R&D (GERD), year XXXX (1990-2012).

gas_emissXXXX Greenhouse gas emissions - base year 1990, year XXXX (1990-2011).

share_renXXXX Share of renewable energy in gross final energy consumption, year XXXX (2004-2011).

prim_enerXXXX Primary energy consumption, year XXXX (1990-2011).

final_energyXXXX Final energy consumption, year XXXX (1990-2011).

final_energyXXXX Early leavers from education and training - Perc. of the population aged 18-24 with at most lower secondary education and not in further education or training, year XXXX (1992-2012).

tertiaryXXXX Tertiary educational attainment - age group 30-34, year XXXX (2000-2012).

risk_povertyXXXX People at risk of poverty or social exclusion - 1000 persons Perc. of total population, year XXXX (2004-2012).

low_workXXXX People living in households with very low work intensity - 1000 persons Perc. of total population, year XXXX (2004-2012).

risk_povertyXXXX People at risk of poverty after social transfers - 1000 persons Perc. of total population, year XXXX (2003-2012).

deprivedXXXX Severely materially deprived people - 1000 persons Perc. of total population, year XXXX (2003-2012).

Author(s)

Vidoli F.

Examples

```
data(EU_2020)
```

EU_NUTS1 *EU NUTS1 Transportation data*

Description

Eurostat regional transport statistics (reg_tran) data, year 2012.

Usage

```
data(EU_NUTS1)
```

Format

EU_NUTS1 is a dataset with 34 observations and two indicators describing transportation infrastructure endowment of the main (in terms of population and GDP) European NUTS1 regions: France, Germany, Italy, Spain (United Kingdom has been omitted, due to lack of data concerning railways).

roads Calculated as (2 * Motorways - Kilometres per 1000 km2 + Other roads - Kilometres per 1000 km2)/3

trains Calculated as (2 *Railway lines double+Electrified railway lines)/3

Author(s)

Vidoli F.

References

Vidoli F., Mazziotta C., "*Robust weighted composite indicators by means of frontier methods with an application to European infrastructure endowment*", Statistica Applicata, Italian Journal of Applied Statistics, 2013.

Examples

```
data(EU_NUTS1)
```

normalise_ci *Normalisation and polarity functions*

Description

This function lets to normalise simple indicators according to the polarity of each one.

Usage

```
normalise_ci(x, indic_col, polarity, method=1, z.mean=0, z.std=1, ties.method ="average")
```

Arguments

`x`	A data frame containing simple indicators.
`indic_col`	Simple indicators column number.
`method`	Normalisation methods:

- **1** (default) = standardization or z-scores using the following formulation:

$$z_{ij} = z.mean \pm \frac{x_{ij} - M_{x_j}}{S_{x_j}} \cdot z.std$$

where $\pm$ depends on *polarity* parameter and *z.mean* and *z.std* represent the shifting parameters.

- **2** = Min-max method using the following formulation:
if *polarity*="POS":
$$\frac{x - min(x)}{max(x) - min(x)}$$
if *polarity*="NEG":
$$\frac{max(x) - x}{max(x) - min(x)}$$
- **3** = Ranking method. If *polarity*="POS" ranking is increasing, while if *polarity*="NEG" ranking is decreasing.

`polarity`	Polarity vector: "POS" = positive, "NEG" = negative. The polarity of a individual indicator is the sign of the relationship between the indicator and the phenomenon to be measured (e.g., in a well-being index, "GDP per capita" has 'positive' polarity and "Unemployment rate" has 'negative' polarity).
`z.mean`	If method=1, Average shifting parameter. Default is 0.
`z.std`	If method=1, Standard deviation expansion parameter. Default is 1.
`ties.method`	If method=3, A character string specifying how ties are treated, see `rank` for details. Default is "average".

Value

`ci_norm`	A data.frame containing normalised score of the choosen simple indicators.
`norm_method`	Normalisation method used.

Author(s)

Vidoli F.

References

OECD, "*Handbook on constructing composite indicators: methodology and user guide*", 2008, pag.30.

See Also

ci_bod, ci_mpi

Examples

```
data(EU_NUTS1)

# Standard z-scores normalisation #
data_norm = normalise_ci(EU_NUTS1,c(2:3),c("NEG","POS"),method=1,z.mean=0, z.std=1)
summary(data_norm$ci_norm)

# Normalisation for MPI index #
data_norm = normalise_ci(EU_NUTS1,c(2:3),c("NEG","POS"),method=1,z.mean=100, z.std=10)
summary(data_norm$ci_norm)

data_norm = normalise_ci(EU_NUTS1,c(2:3),c("NEG","POS"),method=2)
summary(data_norm$ci_norm)
```

Index