Package ‘deaR’

May 2, 2023

Type Package
Title Conventional and Fuzzy Data Envelopment Analysis
Version 1.4.1
Author Vicente Coll-Serrano, Vicente Bolos, Rafael Benitez Suarez <rabesua@uv.es>
Maintainer Vicente Bolos <vicente.bolos@uv.es>
License GPL
Encoding UTF-8
LazyData true
RoxygenNote 7.2.3
Depends R (>= 3.5)
Imports lpSolve, ggplot2, tidyr, plotly, igraph, writexl, dplyr, methods, gridExtra, grDevices
NeedsCompilation no
Repository CRAN
Date/Publication 2023-05-02 09:20:02 UTC

R topics documented:

Airlines ................................................................. 3
bootstrap_basic ...................................................... 4
Coelli_1998 .......................................................... 6
Coll_Blasco_2006 ..................................................... 7
cross_efficiency ...................................................... 8
cross_efficiency_fuzzy ............................................. 11
Departments .......................................................... 13
Doyle_Green_1994 .................................................... 14
Economy ............................................................... 15
### R topics documented:

- EconomyLong
- efficiencies
- efficiencies.dea
- efficiencies.dea_fuzzy
- eff_dmus
- Electric_plants
- extreme_efficient
- Fortune500
- Fried1993
- FuzzyExample
- Golany_Roll_1989
- Grifell_Lovell_1999
- Guo_Tanaka_2001
- Hotels
- Hua_Bian_2007
- is.dea
- is.deadata
- is.deadata_fuzzy
- is.dea_fuzzy
- is.friends
- Kao_Liu_2003
- lambdas
- Leon2003
- Libraries
- Lim_Zhu_2015
- make_deadata
- make_deadata_fuzzy
- make_malmquist
- malmquist_index
- maximal_friends
- modelfuzzy_guotanaka
- modelfuzzy_kao2003
- modelfuzzy_possible
- model_additive
- model_addmin
- model_addsupereff
- model_basic
- model_deaps
- model_fdh
- model_multiplier
- model_nonradial
- model_profit
- model_rdm
- model_sbmef
- model_sbmnsupereff
- model_supereff
- multipliers
- PFT1981
Airlines

**Data:** Coelli, Grifell-Tatje, and Perelman (2002).

**Description**

Data of 28 airlines with 2 outputs and 4 inputs.

**Usage**

data("Airlines")

**Format**

Data frame with 28 rows and 7 columns. Definition of outputs (Y) and inputs (X):

- **y1 = Pass**  Passenger-kilometers flown
- **y2 = Cargo**  Freight tonne-kilometers flown
- **x1 = Lab**  Labor (number of employees)
- **x2 = Fuel**  Fuel (millions of gallons)
- **x3 = Matl**  Other inputs (millions of U.S. dollar equivalent) consisting of operating and maintenance expenses excluding labor and fuel expenses
- **x4 = Cap**  Capital (sum of the maximum takeoff weights of all aircraft flown multiplied by the number of days flown)
**Author(s)**

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**Source**


**See Also**

`make_deadata`, `model_sbmeff`

**Examples**

```r
# Example. Replication of results in Aparicio et al. (2007).
data("Airlines")
data_example <- make_deadata(Airlines, 
  inputs = 4:7, 
  outputs = 2:3)
result <- model_sbmeff(data_example)
efficiencies(result)
result2 <- model_sbmeff(data_example, 
  kaizen = TRUE)
efficiencies(result2)
```

---

**bootstrap_basic**  
**Bootstrapping DEA**

**Description**

To bootstrap efficiency scores, deaR uses the algorithm proposed by Simar and Wilson (1998). For now, the function `bootstrap_basic` can only be used with basic DEA models.

**Usage**

```r
bootstrap_basic(datadea, 
  orientation = c("io", "oo"), 
  rts = c("crs", "vrs", "nirs", "ndrs", "grs"), 
  L = 1, 
  U = 1, 
  B = 2000, 
  h = NULL, 
  alpha = 0.05)
```
**Arguments**

- **datadea**: A deadata object with \( n \) DMUs, \( m \) inputs and \( s \) outputs.
- **orientation**: A string, equal to "io" (input oriented) or "oo" (output oriented).
- **rts**: A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
- **L**: Lower bound for the generalized returns to scale (grs).
- **U**: Upper bound for the generalized returns to scale (grs).
- **B**: Number of bootstrap iterations.
- **h**: Bandwidth of smoothing window. By default \( h = 0.014 \) (you can set \( h \) equal to any other value). The optimal bandwidth factor can also be calculated following the proposals of Silverman (1986) and Daraio y Simar (2007). So, \( h = "h_1" \) is the optimal \( h \) referred as "robust normal-reference rule" (Daraio and Simar, 2007 p.60), \( h = "h_2" \) is the value of \( h_1 \) but instead of the factor 1.06 with the factor 0.9, \( h = "h_3" \) is the value of \( h_1 \) adjusted for scale and sample size (Daraio and Simar, 2007 p.61), and \( h = "h_4" \) is the bandwidth provided by a Gaussian kernel density estimate.
- **alpha**: Between 0 and 1 (for confidence intervals).

**Author(s)**

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. Applied Economics.
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**References**


**Examples**

```r
# To replicate the results in Simar y Wilson (1998, p. 58) you have to
# set B=2000 (in the example B = 100 to save time)
data("Electric_plants")
data_example <- make_deadata(Electric_plants,
                           ni = 3,
                           no = 1)
result <- bootstrap_basic(datadea = data_example,
                 orientation = "io",
                 rts = "vrs",
                 B = 100)
result$score_bc
result$CI
```

**Description**

Data of five DMUs with two inputs and one output. Prices for inputs are available. Price for output is not from Coelli et al. (1998).

**Usage**

```r
data("Coelli_1998")
```

**Format**

Data frame with 6 rows and 5 columns. Definition of inputs (X) and outputs (Y):

- **Input1** Input 1
- **Input2** Input 2
- **Output** Output
- **Price_input1** Price input 1
- **Price_input2** Price input 2
- **Price_output** Price output
Author(s)

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

**Vicente Bolos** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benitez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

`make_deadata`

Examples

```r
# Example. Replication of results in Coelli et al. (1998, p.166).
# Cost efficiency model.
data("Coelli_1994")
data_prices <- t(Coelli_1998[, 5:6])
data_example <- make_deadata(Coelli_1998,
dmus = 1,
ni = 2,
no = 1)
result <- model_profit(data_example,
price_input = data_prices,
rts = "crs",
restricted_optimal = FALSE)
# notice that the option by default is restricted_optimal=TRUE
efficiencies(result)
```

Coll_Blasco_2006

Data: Coll and Blasco (2006).

Description

Data of six authorized dealers with two inputs and two outputs.

Usage

data("Coll_Blasco_2006")
cross_efficiency

Format

Data frame with 6 rows and 5 columns. Definition of inputs (X) and outputs (Y):

\[ x_1 = \text{Employees} \quad \text{Number of employees} \]
\[ x_2 = \text{Capital} \quad \text{Impairment of assets} \]
\[ y_1 = \text{Vehicles} \quad \text{Number of vehicles sold} \]
\[ y_2 = \text{Orders} \quad \text{Number of orders received at the garage} \]

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

make_deadata

Examples

# Example. How to read data with deaR
data("Coll_Blasco_2006")
data_example <- make_deadata(Coll_Blasco_2006,
    dmus = 1,
    ni = 2,
    no = 2)

cross_efficiency

Cross efficiency analysis

Description

Computes arbitrary, benevolent and aggressive formulations of cross-efficiency under any returns-to-scale. Doyle and Green (1994) present three alternatives ways of formulating the secondary goal (which will minimize or maximize the other DMUs’ cross-efficiencies in some way). Methods II and III are implemented in deaR with any returns-to-scale. The maverick index is also calculated.
cross_efficiency

Usage

cross_efficiency(datadea,
    dmu_eval = NULL,
    dmu_ref = NULL,
    epsilon = 0,
    orientation = c("io", "oo"),
    rts = c("crs", "vrs", "nirs", "ndrs", "grs"),
    L = 1,
    U = 1,
    selfapp = TRUE,
    correction = FALSE,
    M2 = TRUE,
    M3 = TRUE)

Arguments

datadea An object of class dea or deadata. If it is of class dea it must have been obtained with some of the multiplier DEA models.
dmu_eval A numeric vector. Only the multipliers of DMUs in dmu_eval are computed. If NULL (default), all DMUs are considered.dmu_ref A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.epsilon Numeric, multipliers must be >= epsilon.orientation A string, equal to "io" (input-oriented) or "oo" (output-oriented).rts A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).L Lower bound for the generalized returns to scale (grs).U Upper bound for the generalized returns to scale (grs).selfapp Logical. If it is TRUE, self-appraisal is included in the average scores of A and e.correction Logical. If it is TRUE, a correction is applied in the "vrs" input-oriented model in order to avoid negative cross-efficiencies, according to Lim & Zhu (2015).M2 Logical. If it is TRUE, it computes Method II for aggressive/benevolent estimations.M3 Logical. If it is TRUE, it computes Method III for aggressive/benevolent estimations.

Note

(1) We can obtain negative cross-efficiency in the input-oriented DEA model under no constant returns-to-scale. However, the same does not happen in the case of the output-oriented VRS DEA model. For this reason, the proposal of Lim and Zhu (2015) is implemented in deaR to calculate the input-oriented cross-efficiency model under no constant returns-to-scale.

(2) The multiplier model can have alternate optimal solutions (see note 1 in model_multiplier). So, depending on the optimal weights selected we can obtain different cross-efficiency scores.
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* *Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References


See Also

*model_multiplier, cross_efficiency_fuzzy*

Examples

```r
# Example 1.
# Arbitrary formulation. Input-oriented model under constant returns-to-scale.
data("Golany_Roll_1989")
data_example <- make_deadata(datadea = Golany_Roll_1989,
                          inputs = 2:4,
                          outputs = 5:6)
result <- cross_efficiency(data_example,
                          orientation = "io",
                          rts = "crs",
                          selfapp = TRUE)
result$Arbitrary$cross_eff
result$Arbitrary$e

# Example 2.
# Benevolent formulation (method II). Input-oriented.
data("Golany_Roll_1989")
data_example <- make_deadata(datadea = Golany_Roll_1989,
                          inputs = 2:4,
                          outputs = 5:6)
result <- cross_efficiency(data_example,
                          orientation = "io",
                          selfapp = TRUE)
result$M2_ben$cross_eff
```
result$M2_ben$e

# Example 3.
# Benevolent formulation (method III). Input-oriented.
data("Golany_Roll_1989")
data_example <- make_deadata(datadea = Golany_Roll_1989,
                            inputs = 2:4,
                            outputs = 5:6)
result <- cross_efficiency(data_example,
                            orientation = "io",
                            selfapp = TRUE)
result$M3_ben$cross_eff
result$M3_ben$e

result$Arbitrary$cross_eff
result$Arbitrary$e

# Example 5.
# Arbitrary formulation. Input-oriented model under vrs returns-to-scale.
data("Lim_Zhu_2015")
data_example <- make_deadata(Lim_Zhu_2015,
                            ni = 1,
                            no = 5)
cross <- cross_efficiency(data_example,
                            epsilon = 0,
                            orientation = "io",
                            rts = "vrs",
                            selfapp = TRUE,
                            M2 = FALSE,
                            M3 = FALSE)
cross$Arbitrary$e

cross_efficiency_fuzzy

Cross efficiency fuzzy analysis

Description

Computes the cross-efficiency fuzzy tables from DEA fuzzy data or a Guo-Tanaka DEA model solution. The (crisp) relative efficiencies for the case $h = 1$ are obtained from the CCR model (model_multiplier).
cross_efficiency_fuzzy

Usage

cross_efficiency_fuzzy(datadea,  
    orientation = c("io", "oo"),  
    h = 1,  
    selfapp = TRUE)

Arguments

datadea: An object of class dea_fuzzy or deadata_fuzzy. If it is of class dea_fuzzy it must have been obtained with modelfuzzy_guotanaka.

orientation: A string, equal to "io" (input-oriented) or "oo" (output-oriented).

h: A numeric vector with the h-levels (in [0,1]).

selfapp: Logical. If it is TRUE, self-appraisal is included in the average scores of A and e.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics*.

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References


Examples

data("Guo_Tanaka_2001")
data(dea <- make_deadata_fuzzy(datadea = Guo_Tanaka_2001,  
    inputs.mL = 2:3,  
    inputs.dL = 4:5,  
    outputs.mL = 6:7,  
    outputs.dL = 8:9))

result <- cross_efficiency_fuzzy(datadea = dea,  
    h = seq(0, 1, 0.2))
Description

Data from 20 University accounting departments in the UK.

Usage

data("Departments")

Format

Data frame with 20 rows and 11 columns. Definition of inputs (X) and outputs (Y):

- **x1 = Staff** Average Full Time Academic Staff 82/3-84/5
- **x2 = Salaries** 1984-5 Salaries Academics and Related (in pounds)
- **x3 = Other_Exp** 1984-5 Other Expenses (in pounds)
- **y1 = Undergrad** Average Number Undergraduates 82/3-84/5
- **y2 = Research_post** Research Postgraduates
- **y3 = Taught_post** Taught Postgraduates
- **y4 = Res_co_income** Research council income (in pounds)
- **y5 = Other_res_income** Other research income (in pounds)
- **y6 = Other_income** Other income (in pounds)
- **y7 = Publications** Number of publications

Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* *Applied Economics.*
- **Vicente Bolos** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benitez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

- `make_deadata`
- `model_basic`
Examples

# Example.
# Replication of results DEA1 in Tomkins and Green (1988)
data("Departments")
# Calculate Total income
Departments$Total_income <- Departments[, 5] + Departments[, 6] + Departments[, 7]
data_example <- make_deadata(Departments,
    inputs = 9,
    outputs = c(2, 3, 4, 12))
result <- model_basic(data_example,
    orientation = "io",
    rts = "crs")
efficiencies(result) # Table 3 (p.156)
references(result) # Table 3 (p.157)

---

Data: Doyle and Green (1994).

Description

Data adapted from Tomkins and Green (1988). 13 DMUs using 3 inputs to produce 2 outputs.

Usage

data("Doyle_Green_1994")

Format

Data frame with 13 rows and 6 columns. Definition of inputs (X) and outputs (Y):

- **y1 = Undergraduate**  Number of undergraduates
- **y2 = Postgraduates**  Number of postgraduates (taught and research)
- **y3 = Research_income**  Research and other income
- **y4 = Publications**  Number of publications
- **x1 = Salaries**  Salaries of academic and related staff
- **x2 = Other_expenses**  Other expenses

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics*.

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)
Source


See Also

make_deadata, model_multiplier, cross_efficiency

Examples

# Example.
data("Doyle_Green_1994")
data_example <- make_deadata(datadea = Doyle_Green_1994,
dmus = 1,
inputs = 6:7,
outputs = 2:5)
result <- cross_efficiency(data_example,
orientation = "io",
selfapp = TRUE)
result$Arbitrary$cross_eff
result$Arbitrary$e
# Aggressive using method II
result$M2_agg$cross_eff
# Aggressive using method III
result$M3_agg$cross_eff

---


Description


Usage

data("Economy")

Format

Data frame with 31 rows and 16 columns. Definition of inputs (X) and outputs (Y):

\[ x1 = \text{Capital} \quad \text{Total assets (in 100 million RMB)} \]
\[ x2 = \text{Labor} \quad \text{Annual average employed persons (in 10000 persons)} \]
\[ y1 = \text{GIOV} \quad \text{Gross industrial output value (in 100 million RMB)} \]
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

`make_malmquist`, `malmquist_index`

Examples

```r
# Example. Data in wide format.
# Replication of results in Wang and Lan (2011, p. 2768)
data("Economy")
data_example <- make_malmquist(Economy,
                               nper = 5,
                               arrangement = "horizontal",
                               ni = 2,
                               no = 1)
result <- malmquist_index(data_example)
```

Description


Usage

```r
data("EconomyLong")
```

Format

Data frame with 155 rows and 5 columns. Definition of inputs (X) and outputs (Y):

- **x1 = Capital**  Total assets (in 100 million RMB)
- **x2 = Labor** Annual average employed persons (in 10000 persons)
- **y1 = GIOV** Gross industrial output value (in 100 million RMB)
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolos (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benitez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

Source


See Also

make_malmquist, malmquist_index

Examples

# Example. Data in long format.
# Replication of results in Wang and Lan (2011, p. 2768)
data("EconomyLong")
data_example <- make_malmquist(EconomyLong,
                           percol = 2,
                           arrangement = "vertical",
                           ni = 2,
                           no = 1)
result <- malmquist_index(data_example)

efficiencies

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract the scores (optimal objective values) of the evaluated DMUs from a conventional, fuzzy or stochastic DEA solution. Note that these scores may not always be interpreted as efficiencies.</td>
</tr>
</tbody>
</table>

Usage

efficiencies(x, ...)

Arguments

x An object of class dea, dea_fuzzy or dea_stoch.
... ignored.
# Description

Extract the scores (optimal objective values) of the evaluated DMUs from a conventional DEA solution. Note that these scores may not always be interpreted as efficiencies.

## Usage

```r
## S3 method for class 'dea'
efficiencies(x, ...)
```

## Arguments

- `x` Object of class `dea` obtained with some of the conventional DEA model_\* functions.
- `...` Other options (for compatibility reasons).

## Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*
  
University of Valencia (Spain)

## References


## Examples

```r
# Replication results model DEA1 in Tomkins and Green (1988)
data("Departments")
# Calculate Total income
Departments$Total_income <- Departments[, 5] + Departments[, 6] + Departments[, 7]
data_DEA1 <- make_deadata(Departments,
                           inputs = 9,
                           outputs = c(2, 3, 4, 12))
result <- model_basic(data_DEA1,
                      orientation = "io",
                      rts = "crs")
efficiencies(result) # Table 3 (p.156)
```
Efficiencies.dea_fuzzy

Efficiencies

Description

Extract the scores (optimal objective values) of the evaluated DMUs from a fuzzy DEA solution. Note that these scores may not always be interpreted as efficiencies.

Usage

```r
## S3 method for class 'dea_fuzzy'
efficiencies(x, ...)
```

Arguments

- `x`: Object of class `dea_fuzzy` obtained with some of the fuzzy DEA model `fuzzy_*` functions.
- `...`: Other options (for compatibility).

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References


Examples

```r
# Replication of results in Boscá, Liern, Sala and Martínez (2011, p.125)
data("Leon2003")
data_example <- make_data_fuzzy(dataea = Leon2003, 
  inputs.mL = 2, 
  inputs.dL = 3, 
  outputs.mL = 4, 
  outputs.dL = 5)
result <- modelfuzzy_kaoliu(data_example, 
  kaoliu_modelname = "basic", 
  alpha = seq(0, 1, by = 0.1), 
  orientation = "io",

```
eff_dmus = efficiencies(result)

---

**eff_dmus**

**Efficient DMUs.**

**Description**

Returns the efficient DMUs evaluated in a dea class object.

**Usage**

```r
eff_dmus(deasol, tol = 1e-04)
```

**Arguments**

- `deasol`: An object of class "dea" obtained by a DEA model function.
- `tol`: Numeric. Absolute tolerance for numeric comparisons in efficiency scores. By default, it is 1e-4.

**Value**

A numeric vector containing which DMUs has been evaluated as efficient. This vector is empty if there is not any efficient DMU.

**Note**

If `maxslack` is FALSE, the slacks computed in the first stage are supposed to be the max slacks.

**Author(s)**

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**Examples**

```r
dataFortune <- make_deadata(Fortune500, ni = 3, no = 2)
ccrFortune <- model_basic(dataFortune)
eff_dmus(ccrFortune)
```

Description

Data of 19 coal-fired steam-electric generating plants operating in Illinois in 1978. Each plant uses 3 inputs to produce 1 output.

Usage

data("Electric_plants")

Format

Data frame with 18 rows and 5 columns. Definition of inputs (X) and outputs (Y):

x1 = Labor  Labor average annual employment
x2 = Fuel   Fuel $10^{10}$ Btu
x3 = Capital Capital MW (fixed input)
y1 = Output Output $10^6$ Kwh

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolos (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benitez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

Source


See Also

make_deadata, model_basic
Examples

data("Electric_plants")
data_example <- make_data(Electric_plants,
    dmus = 1,
    ni = 3,
    no = 1)
result <- model_basic(data_example,
    orientation = "io",
    rts = "vrs")
efficiencies(result)

---

extreme_efficient

**Extreme efficient DMUs**

Description

Find a set of extreme efficient DMUs from a deodata object.

Usage

```r
extreme_efficient(datadea,
dmu_ref = NULL,
rts = c("crs", "vrs", "nirs", "ndrs"),
tol = 1e-6)
```

Arguments

- **datadea**: A deodata object with n DMUs, m inputs and s outputs.
- **dmu_ref**: A numeric vector containing which DMUs are the evaluation reference set, i.e. the cluster of DMUs from which we want to find a extreme efficient DMUs subset. If NULL (default), all DMUs are considered.
- **rts**: A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing) or "ndrs" (non-decreasing).
- **tol**: Numeric, a tolerance margin for checking efficiency. It is 1e-6 by default.

Value

A numeric vector representing a extreme efficient subset of DMUs.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)
Fortune500

References

Examples

data("PFT1981")
datadea <- make_deadata(PFT1981,
  ni = 5,
  no = 3)
# We find a extreme efficient subset from a cluster formed by the first 20 DMUs
result <- extreme_efficient(datadea = datadea,
  dmu_ref = 1:20)

Fortune500

Data: Zhu (2014).

Description
This dataset consists of 15 firms from the Fortune 500 list 1995 (https://fortune.com/ranking/fortune500/) with 3 inputs and 2 outputs.

Usage
data("Fortune500")

Format
Data frame with 15 rows and 6 columns. Definition of inputs (X) and outputs (Y):

x1 = Assets Assets (millions of dollars)

x2 = Equity Equity (millions of dollars)

x3 = Employees Number of employees

y1 = Revenue Revenue (millions of dollars)

y2 = Profit Profit (millions of dollars)

Author(s)
Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolos (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benitez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)
Fried1993

Source

See Also
make_deadata, model_multiplier

Examples

```r
data("Fortune500")
data_Fortune <- make_deadata(datadea = Fortune500,
  dmus = 1,
  inputs = 2:4,
  outputs = 5:6)
result <- model_multiplier(data_Fortune,
  epsilon = 1e-6,
  orientation = "io",
  rts = "crs")
# results for General Motors and Ford Motor are not shown # by deaR because the solution is infeasible
efficiencies(result)
multipliers(result)
```

Fried1993


Description

Data of 11 DMUs with two inputs and one output.

Usage

data("Fried1993")

Format

Data frame with 11 rows and 4 columns. Definition of inputs (X) and outputs (Y):

- **x1** Input 1
- **x2** Input 2
- **y1** Output 1
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

make_deadata, model_basic

Examples

```r
# Example. Replication of results in Ali and (1993, p.143).
data("Fried1993")
data_example <- make_deadata(Fried1993,
   ni = 2,
   no = 1)
result <- model_basic(data_example,
   orientation = "oo",
   rts = "vrs")
efficiencies(result)
targets(result)
```

FuzzyExample

*Data: Fuzzy data reading example.*

Description

Synthetic dataset of 5 DMUs with 3 inputs and 3 outputs containing fuzzy and crisp data.

Usage

data("FuzzyExample")
Format

Data frame with 5 rows and 15 columns.

DMU  DMU names
Input1.mL  First Input (crisp numbers)
Input2.mL  Second Input (left centers)
Input2.mR  Second Input (right centers)
Input2.dL  Second Input (left radii)
Input2.dR  Second Input (right radii)
Input3.mL  Third Input (centers)
Input3.dL  Third Input (radii)
Output1.mL  First Output (crisp numbers)
Output2.mL  Second Output (left centers)
Output2.mR  Second Output (right centers)
Output2.dL  Second Output (radii)
Output3.mL  Third Output (centers)
Output3.dL  Third Output (left radii)
Output3.dR  Third Output (right radii)

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolos (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benitez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

See Also

make_deadata_fuzzy

Examples

# Example. Reading the data.
data("FuzzyExample")
datafuzzy <- make_deadata_fuzzy(FuzzyExample,
  inputs.mL = c(2, 3, 7),
  inputs.mR = c(NA, 4, NA),
  inputs.dL = c(NA, 5, 8),
  inputs.dR = c(NA, 6, NA),
  outputs.mL = c(9, 10 , 13),
  outputs.mR = c(NA, 11, NA),
  outputs.dL = c(NA, 12, 14),
  outputs.dR = c(NA, NA, 15))
Description

Data of 13 DMUs using 3 inputs to produce 2 outputs.

Usage

data("Golany_Roll_1989")

Format

Data frame with 13 rows and 6 columns. Definition of inputs (X) and outputs (Y):

- **x1**: Input 1
- **x2**: Input 2
- **x3**: Input 3
- **y1**: Output 1
- **y2**: Output 2

Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*
- **Vicente Bolos** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benitez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

- `make_deadata`, `model_multiplier`, `cross_efficiency`

Examples

```r
# Example.
data("Golany_Roll_1989")
data_example <- make_deadata(datadea = Golany_Roll_1989,
                             dmus = 1,
                             inputs = 2:4,
                             outputs = 5:6)
```
result <- cross_efficiency(data_example,
    orientation = "io",
    selfapp = TRUE)
result$Arbitrary$cross_eff
result$Arbitrary$e

---

**Grifell_Lovell_1999**


**Description**

Data of 8 DMUs producing 1 output (Y) by using 1 input (X) for two periods of time.

**Usage**

data("Grifell_Lovell_1999")

**Format**

Data frame with 16 rows and 4 columns. Definition of inputs (X) and outputs (Y):

- X Input
- Y Output

**Author(s)**

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**Source**


**See Also**

`make_malmquist`, `malmquist_index`
Examples

# Example. Replication of results in Grifell-Tatjé and Lovell (1999, p. 100).
data("Grifell_Lovell_1999")
data_example <- make_malmquist(Grifell_Lovell_1999,
  percol = 1,
  dmus = 2,
  inputs = 3,
  outputs = 4,
  arrangement = "vertical")

result_fgnz <- malmquist_index(data_example,
  orientation = "oo",
  rts = "vrs",
  type1 = "cont",
  type2 = "fgnz")

result_fgnz$mi

Description

Data of 5 DMUs with two symmetric triangular fuzzy inputs, Xj = (xj, alphaj), and two symmetric triangular fuzzy outputs, Yj = (yj, betaj).

Usage

data("Guo_Tanaka_2001")

Format

Data frame with 5 rows and 9 columns. Definition of fuzzy inputs (X) and fuzzy outputs (Y):

- x1 Input 1
- x2 Input 2
- alpha1 spread vector Input 1
- alpha2 spread vector Input 2
- y1 Output 1
- y2 Output 2
- beta1 spread vector Output 1
- beta2 spread vector Output 2
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

*make_deadata_fuzzy*, *modelfuzzy_guotanaka*, *cross_efficiency_fuzzy*

Examples

```r
data("Guo_Tanaka_2001")
data_example <- make_deadata_fuzzy(Guo_Tanaka_2001,
dmus = 1,
inputs.mL = 2:3,
inputs.dL = 4:5,
outputs.mL = 6:7,
outputs.dL = 8:9)
result <- modelfuzzy_guotanaka(data_example,
    h = seq(0, 1, by = 0.1),
    orientation = "io")
efficiencies(result)
```

---

**Hotels**

*Data: Wu, Tsai and Zhou (2011).*

Description

This dataset consists of 23 four- and five-plum ITHs in Taipei in 2006. Authors consider 4 inputs and 3 outputs.

Usage

```r
data("Hotels")
```
Hotels

Format

Data frame with 23 rows and 8 columns. Definition of inputs (X) and outputs (Y):

- **x1 = Employees**  Total number of employees
- **x2 = Guest_rooms** Total number of guest rooms
- **x3 = Area_F&B** Total area of F&B departments (in 36 square-feet)
- **x4 = Operating_cost** Total operating cost (in NT$)
- **y1 = Room_revenue** Room revenues (in NT$)
- **y2 = F&B_revenue** F&B revenues (in NT$)
- **y3 = Other_revenue** Other revenues (in NT$)

Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* Applied Economics.
- **Vicente Bolos** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benitez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

- `make_deadata`, `model_nonradial`

Examples

```r
# Example. Replication of results in Wu, Tsai and Zhou (2011)
data("Hotels")
data_hotels <- make_deadata(Hotels, 
                           dmus = 1, 
                           inputs = 2:5, 
                           outputs = 6:8) 
result <- model_nonradial(data_hotels, 
                           orientation = "oo", 
                           rts = "vrs")
efficiencies(result)
```
Data: Hua and Bian (2007).

Description

Data of 30 DMUs with two desirable inputs, two desirable outputs and one undesirable output.

Usage

data("Hua_Bian_2007")

Format

Data frame with 30 rows and 6 columns. Definition of inputs (X) and outputs (Y):

\[
\begin{align*}
  x_1 &= \text{D-Input1} & \text{Desirable Input 1} \\
  x_2 &= \text{D-Input2} & \text{Desirable Input 2} \\
  y_1 &= \text{D-Output1} & \text{Desirable Output 1} \\
  y_2 &= \text{D-Output2} & \text{Desirable Output 2} \\
  y_3 &= \text{UD-Output1} & \text{Undesirable Output 1}
\end{align*}
\]

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics*.

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

*make_deadata*, *model_basic*
Examples

# Example. Replication of results in Hua and Bian (2007).
data("Hua_Bian_2007")
# The third output is an undesirable output
data_example <- make_deadata(Hua_Bian_2007,
   ni = 2,
   no = 3,
   ud_outputs = 3)

# Translation parameter (vtrans_o) is set to 1500
result <- model_basic(data_example,
   orientation = "oo",
   rts = "vrs",
   vtrans_o = 1500)
eff <- efficiencies(result)
1 / eff # results M5 in Table 6-5 (p.119)

is.dea

de class check.

Description

Checks whether an R object is of dea class or not.

Usage

is.dea(x)

Arguments

x

Any R object.

Value

Returns TRUE if its argument is a dea object (that is, has "dea" amongst its classes) and FALSE otherwise.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)
is.deadata  

deadata class check.

Description

Checks whether an R object is of deadata class or not.

Usage

is.deadata(x)

Arguments

x Any R object.

Value

Returns TRUE if its argument is a deadata object (that is, has "deadata" amongst its classes) and FALSE otherwise.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

is.deadata_fuzzy  
deadata_fuzzy class check.

Description

Checks whether an R object is of deadata_fuzzy class or not.

Usage

is.deadata_fuzzy(x)

Arguments

x Any R object.
is.dea_fuzzy

dea_fuzzy class check.

Description
Checks whether an R object is of dea_fuzzy class or not.

Usage
is.dea_fuzzy(x)

Arguments
x Any R object.

Value
Returns TRUE if its argument is a dea_fuzzy object (that is, has "dea_fuzzy" amongst its classes) and FALSE otherwise.

Author(s)
Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.
Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics
Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics
University of Valencia (Spain)
is.friends

Friends check.

Description
Checks whether a subset of DMUs is friends or not, according to Tone (2010).

Usage

```r
is.friends(datadea,
  dmu_eval = NULL,
  dmu_ref = NULL,
  rts = c("crs", "vrs", "nirs", "ndrs"),
  tol = 1e-6)
```

Arguments

data.dea The data, including \( n \) DMUs, \( m \) inputs and \( s \) outputs.
dmu_eval A numeric vector containing the subset of DMUs to be checked. If NULL (default), all DMUs are considered.
dmu_ref A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
rts A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing) or "ndrs" (non-decreasing).
tol Numeric, a tolerance margin for checking efficiency. It is 1e-6 by default.

Value

Returns TRUE if dmu_eval is friends of dmu_ref, and FALSE otherwise.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

References


See Also

maximal_friends, model_sbmeff
Examples

data("PFT1981")
datad <- make_deadata(PFT1981,
    ni = 5,
    no = 3)
subset1 <- c(15, 16, 17, 19) # Subset of DMUs to be checked
result1 <- is.friends(datad = datad,
    dmu_eval = subset1,
    dmu_ref = 1:20) # We only consider a cluster formed by the first 20 DMUs
subset2 <- c(15, 16, 17, 20) # Another subset of DMUs to be checked
result2 <- is.friends(datad = datad,
    dmu_eval = subset2,
    dmu_ref = 1:20) # We only consider a cluster formed by the first 20 DMUs

Description


Usage

data("Kao_Liu_2003")

Format

Data frame with 24 rows and 11 columns. Definition of fuzzy inputs (X) and fuzzy outputs (Y):

x1 = Patronage It is a weighted sum of the standardized scores of faculty, graduate students, undergraduate students, and extension students in the range of 0 and 1.

y1 = Collections Books, serials, microforms, audiovisual works, and database.

y2 = Personnel Classified staff, unclassified staff, and student assistants.

y3 = Expenditures Capital expenditure, operating expenditure, and special expenditure.

y4 = Buildings Area and seats

y5 = Services Operating hours, attendance, circulation, communication channels, range of services, amount of services, etc.

beta3_l lower spread vector Expenditures
beta3_u upper spread vector Expenditures
beta5_l lower spread vector Services
beta5_u upper spread vector Services
Note
There are three observations that are missing: expenditures of Library 24 and services of Library 22 and Library 23. Kao and Liu (2000b) represent the expenditures of Library 24 by the triangular fuzzy number $Y = (0.11; 0.41; 1.0)$. The services of Library 22 and Library 23 are expressed by a same triangular fuzzy number $Y = (0.41; 0.69; 1.0)$.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source

See Also

*make_deadata_fuzzy, model_basic*

Examples

```r
# Example. Replication of results in Kao and Liu (2003, p.152)
data_example <- make_deadata_fuzzy(Kao_Liu_2003,
  dmus = 1,
  inputs.mL = 2,
  outputs.mL = 3:7,
  outputs.dL = c(NA, NA, 8, NA, 10),
  outputs.dR = c(NA, NA, 9, NA, 11))
result <- modelfuzzy_kaoliu(data_example,
  kaoliu_modelname = "basic",
  orientation = "oo",
  rts = "vrs",
  alpha = 0)
eff <- efficiencies(result)
eff
```

<table>
<thead>
<tr>
<th>lambdas</th>
<th>Lambda</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Description

Extract the lambdas of the DMUs from a dea or dea_fuzzy solution.
Leon2003

Usage

`lambdas(deasol)`

Arguments

`deasol` Object of class `dea` or `dea_fuzzy` obtained with some of the DEA model functions.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics*.

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Examples

```r
data("Coll_Blasco_2006")
data_example <- make_deadata(Coll_Blasco_2006, ni = 2, no = 2)
result <- model_multiplier(data_example, orientation = "io", rts = "crs")
lambdas(result)
```

Leon2003  

*Data: Leon, Liern, Ruiz and Sirvent (2003).*

Description

Data of 8 DMUs with one symmetric triangular fuzzy inputs: $X_j = (x_j, \alpha_j)$, and one symmetric triangular fuzzy outputs: $Y_j = (y_j, \beta_j)$.

Usage

```r
data("Leon2003")
```

Format

Data frame with 8 rows and 5 columns. Definition of fuzzy inputs (X) and fuzzy outputs (Y):

- **x1** Input 1
- **alpha1** spread vector Input 1
- **y1** Output 1
- **beta1** spread vector Output 1
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

`make_deadata_fuzzy`, `modelfuzzy_possibilistic`, `cross_efficiency_fuzzy`, `modelfuzzy_guotanaka`

Examples

```r
# Example. Replication of results in Leon et. al (2003, p. 416)
data("Leon2003")
data_example <- make_deadata_fuzzy(Leon2003, 
dmus = 1, 
inputs.mL = 2, 
inputs.dL = 3, 
outputs.mL = 4, 
outputs.dL = 5)
result <- modelfuzzy_possibilistic(data_example, 
h = seq(0, 1, by = 0.1), 
orientation = "io", 
rts = "vrs")
efficiencies(result)
```

Libraries

Data: Cooper, Seiford and Tone (2007).

Description

Data for 23 public libraries of the Tokyo Metropolitan Area in 1986.

Usage

data("Libraries")
Libraries

Format

Data frame with 23 rows and 7 columns. Definition of inputs (X) and outputs (Y):

\[ x_1 = \text{AREA} \quad \text{Floor area (unit=1000 m}^2\text{)} \]
\[ x_2 = \text{BOOKS} \quad \text{Number of books (unit=1000)} \]
\[ x_3 = \text{STAFF} \quad \text{Staff} \]
\[ x_4 = \text{POPULATION} \quad \text{Population (unit=1000)} \]
\[ y_1 = \text{REGISTERED} \quad \text{Registered residents (unit=1000)} \]
\[ y_2 = \text{BORROWED} \quad \text{Borrowed books (unit=1000)} \]

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics*.

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

make_deadata, model_basic

Examples

```
# Example 1. Non-controllable input (POPULATION).
# Replication of results in Cooper, Seiford and Tone (2007, p.221)
data(Libraries)
data_example <- make_deadata(Libraries,
  dmus = 1,
  inputs = 2:5,
  nc_inputs = 4,
  outputs = 6:7)
result <- model_basic(data_example,
  orientation = "io",
  rts = "crs")
efficiencies(result)
targets(result)
```

```
# Example 2. Non-discretionary input (POPULATION).
data(Libraries)
data_example2 <- make_deadata(Libraries,
  dmus = 1,
  inputs = 2:5,
  nc_inputs = 4,
  outputs = 6:7)
result2 <- model_basic(data_example2,
  orientation = "io",
  rts = "crs")
efficiencies(result2)
targets(result2)
```
result2 <- model_basic(data_example2, 
  orientation="io", 
  rts="crs")

efficiencies(result2)
targets(result2)

**Description**

Data of 37 R&D project proposal relating to the Turkish iron and steel industry. Authors consider one input and five outputs.

**Usage**

data("Lim_Zhu_2015")

**Format**

Data frame with 37 rows and 7 columns. Definition of inputs (X) and outputs (Y):

- **x1 = Budget**  Budget
- **y1 = Indirect_economic**  Indirect economic contribution
- **y2 = Direct_economic**  Direct economic contribution
- **y3 = Technical**  Technical contribution
- **y4 = Social**  Social contribution
- **y5 = Scientific**  Scientific contribution

**Author(s)**

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**Source**

Description

This function creates, from a data frame, a deadata structure, which is as list with fields input, output, dmunames, nc_inputs, nc_outputs, nd_inputs, nd_outputs.

Usage

make_deadata(datadea = NULL, 
  ni = NULL, 
  no = NULL, 
  dmus = 1, 
  inputs = NULL, 
  outputs = NULL, 
  nc_inputs = NULL, 
  nc_outputs = NULL, 
  nd_inputs = NULL, 
  nd_outputs = NULL, 
  ud_inputs = NULL, 
  ud_outputs = NULL)
Arguments

- **datadea**: Data frame with DEA data.
- **ni**: Number of inputs, if inputs are in columns 2:(ni + 1) (if DMUs are in the first column) or 1:ni (no DMUs column).
- **no**: Number of outputs, if outputs are in columns (ni + 2):(ni + no + 1) (if DMUs are in the first column) or (ni + 1):(ni + no) (no DMUs column). If not specified, DMUs are in the first column.
- **dmos**: Column (number or name) of DMUs (optional). By default, it is the first column. If there is not any DMU column, then it must be NULL.
- **inputs**: Columns (numbers or names) of inputs (optional). It prevails over ni. Alternatively to datadea, it can be a matrix with the inputs (DMUs in columns). In this case, DMUs names are taken from the columns names.
- **outputs**: Columns (numbers or names) of outputs (optional). It prevails over no. Alternatively to datadea, it can be a matrix with the outputs (DMUs in columns).
- **nc_inputs**: A numeric vector containing the indices of non-controllable inputs.
- **nc_outputs**: A numeric vector containing the indices of non-controllable outputs.
- **nd_inputs**: A numeric vector containing the indices of non-discretionary inputs.
- **nd_outputs**: A numeric vector containing the indices of non-discretionary outputs.
- **ud_inputs**: A numeric vector containing the indices of undesirable (good) inputs.
- **ud_outputs**: A numeric vector containing the indices of undesirable (bad) outputs.

Value

An object of class deadata

Author(s)

- Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics*.
- Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Examples

```r
data("Coll_Blasco_2006")
data_example <- make_deadata(datadea = Coll_Blasco_2006, 
ni = 2,
no = 2)
# This is the same as:
data_example <- make_deadata(Coll_Blasco_2006, 
inputs = 2:3,
outputs = 4:5)
# And the same as:
dmunames <- c("A", "B", "C", "D", "E", "F")```
make_deadata_fuzzy

Description

This function creates, from a data frame, a deadata_fuzzy structure, which is as list with fields `input`, `output` and `dmunames`. At the same time, `input` and `output` are lists with fields `mL`, `mR`, `dL` and `dR`.

Usage

```R
make_deadata_fuzzy(datadea,
  dmus = 1,
  inputs.mL = NULL,
  inputs.mR = NULL,
  inputs.dL = NULL,
  inputs.dR = NULL,
  outputs.mL = NULL,
  outputs.mR = NULL,
```
make_deadata_fuzzy outputs.dL = NULL,
outputs.dR = NULL,
nc_inputs = NULL,
nc_outputs = NULL,
nd_inputs = NULL,
nd_outputs = NULL,
ud_inputs = NULL,
ud_outputs = NULL)

Arguments

datadea Data frame with DEA data.
dmus Column (number or name) of DMUs (optional). By default, it is the first column. If there is not any DMU column, then it must be NULL.

inputs.mL Where are (columns) the mL (left centers) of trapezoidal fuzzy inputs in datadea. If an input is triangular or crisp, we put the column where the centers or the crisp values are, respectively. Alternatively to datadea, inputs.mL can be a matrix of size (number of inputs x number of DMUs) with the mL of trapezoidal fuzzy inputs, the centers of triangular inputs, and the crisp values of crisp inputs. In this case, DMUs names are taken from the columns names.

inputs.mR Where are (columns) the mR (right centers) of trapezoidal fuzzy inputs in datadea. If an input is triangular or crisp, we put NA. Alternatively to datadea, inputs.mR can be a matrix of size (number of inputs x number of DMUs) with the mR of trapezoidal fuzzy inputs, the centers of triangular inputs, and the crisp values of crisp inputs. If all inputs are triangular or crisp, then inputs.mR must be NULL (default) or equal to inputs.mL.

inputs.dL Where are (columns) the dL (left radii) of trapezoidal and triangular fuzzy inputs in datadea. If an input is symmetric, we put the column where the radii are. If an input is rectangular or crisp, we put NA. Alternatively to datadea, inputs.dL can be a matrix of size (number of inputs x number of DMUs) with the dL of trapezoidal and triangular fuzzy inputs. If an input is rectangular or crisp, its radius is zero. If all inputs are rectangular or crisp, then inputs.dL must be NULL (default) or a zero matrix.

inputs.dR Where are (columns) the dR (right radii) of trapezoidal and triangular fuzzy inputs in datadea. If an input is symmetric, rectangular or crisp, we put NA. Alternatively to datadea, inputs.dR can be a matrix of size (number of inputs x number of DMUs) with the dR of trapezoidal and triangular fuzzy inputs. If an input is rectangular or crisp, its radius is zero. If all inputs are symmetric, rectangular or crisp, then inputs.dR must be NULL (default) or equal to inputs.dL.

outputs.mL Analogous to inputs.mL, but relating to outputs.
outputs.mR Analogous to inputs.mR, but relating to outputs.
outputs.dL Analogous to inputs.dL, but relating to outputs.
outputs.dR Analogous to inputs.dR, but relating to outputs.
make_malmquist

Description
This function creates, from a data frame, a list of deadata objects.

Usage
make_malmquist(datadea,  
  nper = NULL,  
  percol = NULL,  
  arrangement = c("horizontal", "vertical"),  
  ...)
Arguments

- **datadea**: Data frame with DEA data.
- **nper**: Number of time periods (with dataset in wide format).
- **percol**: Column of time period (with dataset in long format).
- **arrangement**: Horizontal with data in wide format. Vertical with data in long format.
- **...**: Other options to be passed to the `make_deadata` function.

Value

An object of class `deadata`

Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Examples

```r
# Example 1. If you have a dataset in wide format.
data("Economy")
data_example <- make_malmquist(datadea = Economy,
                                  nper = 5,
                                  arrangement = "horizontal",
                                  ni = 2,
                                  no = 1)

# This is the same as:
data_example <- make_malmquist(datadea = Economy,
                                  nper = 5,
                                  arrangement = "horizontal",
                                  inputs = 2:3,
                                  outputs = 4)

# Example 2. If you have a dataset in long format.
data("EconomyLong")
data_example2 <- make_malmquist(EconomyLong,
                                 percol = 2,
                                 arrangement = "vertical",
                                 inputs = 3:4,
                                 outputs = 5)
```
**malmquist_index**

**Malmquist index**

**Description**

This function calculates the input/output oriented Malmquist productivity index under constant or variable returns-to-scale.

**Usage**

```r
malmquist_index(datadealist, 
    dmu_eval = NULL, 
    dmu_ref = NULL, 
    orientation = c("io", "oo"), 
    rts = c("crs", "vrs"), 
    type1 = c("cont", "seq", "glob"), 
    type2 = c("fgnz", "rd", "gl", "bias"), 
    tc_vrs = FALSE, 
    vtrans_i = NULL, 
    vtrans_o = NULL)
```

**Arguments**

- **datadealist**: A list with the data (deadata objects) at different times, including DMUs, inputs and outputs.
- **dmu_eval**: A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
- **dmu_ref**: A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
- **orientation**: A string, equal to "io" (input oriented) or "oo" (output oriented).
- **rts**: A string, determining the type of returns to scale, equal to "crs" (constant) or "vrs" (variable).
- **type1**: A string, equal to "cont" (contemporary), "seq" (sequential) or "glob" (global).
- **type2**: A string, equal to "fgnz" (Fare et al. 1994), "rd" (Ray and Desli 1997), "gl" (generalized) or "bias" (biased).
- **tc_vrs**: Logical. If it is FALSE, it computes the vrs bias malmquist index by using the technical change under crs (Fare and Grosskopf 1996). Otherwise, it uses the technical change under vrs.
- **vtrans_i**: Numeric vector of translation for undesirable inputs in non-directional basic models. If vtrans_i[i] is NA, then it applies the "max + 1" translation to the i-th undesirable input. If vtrans_i is a constant, then it applies the same translation to all undesirable inputs. If vtrans_i is NULL, then it applies the "max + 1" translation to all undesirable inputs.
- **vtrans_o**: Numeric vector of translation for undesirable outputs in non-directional basic models, analogous to vtrans_i, but applied to outputs.
**Value**

A numeric list with Malmquist index and other parameters.

**Note**

In the results: EC = Efficiency Change, PTEC = Pure Technical Efficiency Change, SEC = Scale Efficiency Change, TC = Technological Change, MI = Malmquist Index

**Author(s)**

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* Applied Economics.

**Vicente Bolos** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benitez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**References**


Examples

# Example 1. With dataset in wide format.
# Replication of results in Wang and Lan (2011, p. 2768)
data("Economy")
data_example <- make_malmquist(datadea = Economy,
    nper = 5,
    arrangement = "horizontal",
    ni = 2,
    no = 1)
result <- malmquist_index(data_example, orientation = "io")
mi <- result$mi
effch <- result$ec
techno <- result$tc

# Example 2. With dataset in long format.
# Replication of results in Wang and Lan (2011, p. 2768)
data("EconomyLong")
data_example2 <- make_malmquist(EconomyLong,
    percol = 2,
    arrangement = "vertical",
    inputs = 3:4,
    outputs = 5)
result2 <- malmquist_index(data_example2, orientation = "io")
mi2 <- result2$mi
effch2 <- result2$ec
techno2 <- result2$tc

# Example 3. Replication of results in Grifell-Tatje and Lovell (1999, p. 100).
data("Grifell_Lovell_1999")
data_example <- make_malmquist(Grifell_Lovell_1999,
    percol = 1,
    dmus = 2,
    inputs = 3,
    outputs = 4,
    arrangement = "vertical")
result_fgnz <- malmquist_index(data_example,
    orientation = "oo",
    rts = "vrs",
    type1 = "cont",
    type2 = "fgnz")
mi_fgnz <- result_fgnz$mi

result_rd <- malmquist_index(data_example,
    orientation = "oo",
    rts = "vrs",
    type1 = "cont",
    type2 = "rd")
mi_rd <- result_rd$mi

result_gl <- malmquist_index(data_example,
    orientation = "oo",
    rts = "vrs",
maximal_friends

```r
type1 = "cont",
type2 = "gl"

mi_gl <- result_gl$mi
```

### maximal_friends

**Maximal friends of a set of DMUs.**

#### Description

Finds the maximal friends subsets of a given set of DMUs, according to Tone (2010). It uses an ascending algorithm in order to find directly maximal subsets.

#### Usage

```r
maximal_friends(datadea, 
    dmu_ref = NULL, 
    rts = c("crs", "vrs", "nirs", "ndrs"),
    tol = 1e-6,
    silent = FALSE)
```

#### Arguments

- **datadea**: A deadata object with n DMUs, m inputs and s outputs.
- **dmu_ref**: A numeric vector containing which DMUs are the evaluation reference set, i.e. the cluster of DMUs from which we want to find maximal friends. If NULL (default), all DMUs are considered.
- **rts**: A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing) or "ndrs" (non-decreasing).
- **tol**: Numeric, a tolerance margin for checking efficiency. It is 1e-6 by default.
- **silent**: Logical, if FALSE (default) steps are printed.

#### Value

A list with numeric vectors representing maximal friends subsets of DMUs.

#### Author(s)

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

**Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)
modelfuzzy_guotanaka

References


See Also

is.friends, model_sbmeff

Examples

```r
## Not run:
data("PFT1981")
datadea <- make_deadata(PFT1981,
                        ni = 5,
                        no = 3)
# We find maximal friends of a cluster formed by the first 20 DMUs
result <- maximal_friends(datadea = datadea,
                          dmu_ref = 1:20)

## End(Not run)
```

modelfuzzy_guotanaka  Fuzzy DEA model

Description

Solve the Fuzzy input-oriented and output-oriented DEA model proposed by Guo and Tanaka (2001) under constant returns to scale. In deaR is implemented the LP problem given by the model (16) in Guo and Tanaka (2001, p.155). The fuzzy efficiencies are calculated according to equations in (17) (Guo and Tanaka, 2001, p.155). The (crisp) relative efficiencies and multipliers for the case \( h = 1 \) are obtained from the CCR model (model_multiplier).

Usage

```r
modelfuzzy_guotanaka(datadea,
                      dmu_eval = NULL,
                      dmu_ref = NULL,
                      orientation = c("io", "oo"),
                      h = 1)
```

Arguments

- `datadea`: A deadata_fuzzy object, including DMUs, inputs and outputs.
- `dmu_eval`: A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
- `dmu_ref`: A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
orientation A string, equal to "io" (input oriented) or "oo" (output oriented).
h A numeric vector with the h-levels (in [0,1]).

Value
An object of class deadata_fuzzy.

Note
The optimal solution of model (16) is not unique.

Author(s)
Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.
Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics
Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics
University of Valencia (Spain)

References

See Also
model_basic, model_multiplier, modelfuzzy_kaoliu, modelfuzzy_possibilistic, cross_efficiency_fuzzy

Examples
# Example 1.
# Replication results in Guo and Tanaka (2001, p. 159).
# In deaR is implemented the LP problem given by the model 16 in Guo and Tanaka (2001, p. 155).
# The fuzzy efficiencies are calculated according to equations in (17) (Guo and Tanaka, 2001,p.155).
data("Guo_Tanaka_2001")
data_example <- make_deadata_fuzzy(Guo_Tanaka_2001,
inputs.mL = 2:3,
inputs.dL = 4:5,
outputs.mL = 6:7,
outputs.dL = 8:9)
result <- modelfuzzy_guotanaka(data_example,
h = c(0, 0.5, 0.75, 1),
modelfuzzy_kaoliu

Fuzzy DEA model.

Description
Solve the fuzzy DEA model by Kao and Liu (2000)

Usage

modelfuzzy_kaoliu(datadea, 
dmu_eval = NULL, 
kaoliu_modelname = c("basic", "additive", "addsupereff", 
"deaps", "fdh", "multiplier", "nonradial", "profit", 
"rdm", "sbmeff", "sbmsupereff", "supereff"), 
alpha = 1, 
...) 

Arguments

datadea A deadata_fuzzy object, including DMUs, inputs and outputs.
dmu_eval A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
kaoliu_modelname a string containing the name of the model.
alpha A numeric vector with the alpha-cuts (in [0,1]). If alpha>1, it determines the number of alpha-cuts, equispatially distributed in [0,1].
... dmu_ref, orientation, rts and other model parameters.

Value
An object of class deadata_fuzzy.
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

References


See Also

model_basic, model_multiplier, modelfuzzy_possibilistic, modelfuzzy_guotanaka

Examples

# Example 1.
# Replication of results in Boscá, Liern, Sala and Martínez (2011, p.125)
data("Leon2003")
data_example <- make_deadata_fuzzy(datadea = Leon2003,
inputs.mL = 2,
inputs.dL = 3,
outputs.mL = 4,
outputs.dL = 5)
result <- modelfuzzy_kaoliu(data_example,
kaoliu_modelname = "basic",
alpha = seq(0, 1, by = 0.1),
orientation = "io",
rts = "vrs")
efficiencies(result)

# Example 2.
modelfuzzy_possibilistic

Possibilistic Fuzzy DEA model.

Description
Solve the possibilistic fuzzy DEA model proposed by León et al (2003).

Usage
modelfuzzy_possibilistic(datadea, 
dmu_eval = NULL, 
poss_modelname = c("basic"), 
h = 1, 
...)

Arguments
datadea A deadata_fuzzy object, including DMUs, inputs and outputs.
dmu_eval A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
poss_modelname a string containing the name of the model.
h A numeric vector with the h-levels in [0,1].
... dmu_ref, orientation, rts and other model parameters.

Value
An object of class deadata_fuzzy.
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References


See Also

`model_basic, modelfuzzy_kaoliu, modelfuzzy_guotanaka`

Examples

```r
# Replication of results in Leon et. al (2003, p. 416)
data("Leon2003")
data_example <- make_deadata_fuzzy(Leon2003, 
  inputs.mL = 2, 
  inputs.dL = 3, 
  outputs.mL = 4, 
  outputs.dL = 5)
result <- modelfuzzy_possibilistic(data_example, 
  h = seq(0, 1, by = 0.1), 
  orientation = "io", 
  rts = "vrs")
efficiencies(result)
```

---

**model_additive**

Additive DEA model.
Description

Solve the additive model of Charnes et. al (1985). With the current version of deaR, it is possible to solve input-oriented, output-oriented, and non-oriented additive model under constant and non-constant returns to scale.

Besides, the user can set weights for the input slacks and/or output slacks. So, it is also possible to solve weighted additive models. For example: Measure of Inefficiency Proportions (MIP), Range Adjusted Measure (RAM), etc.

Usage

```r
model_additive(datadea,
    dmu_eval = NULL,
    dmu_ref = NULL,
    orientation = NULL,
    weight_slack_i = 1,
    weight_slack_o = 1,
    rts = c("crs", "vrs", "nirs", "ndrs", "grs"),
    L = 1,
    U = 1,
    compute_target = TRUE,
    returnlp = FALSE,
    ...)
```

Arguments

- `datadea` A `deadata` object with `n` DMUs, `m` inputs and `s` outputs.
- `dmu_eval` A numeric vector containing which DMUs have to be evaluated. If `NULL` (default), all DMUs are considered.
- `dmu_ref` A numeric vector containing which DMUs are the evaluation reference set. If `NULL` (default), all DMUs are considered.
- `orientation` This parameter is either `NULL` (default) or a string, equal to "io" (input-oriented) or "oo" (output-oriented). It is used to modify the weight slacks. If input-oriented, `weight_slack_o` are taken 0. If output-oriented, `weight_slack_i` are taken 0.
- `weight_slack_i` A value, vector of length `m`, or matrix `m` x `ne` (where `ne` is the length of `dmu_eval`) with the weights of the input slacks. If 0, output-oriented.
- `weight_slack_o` A value, vector of length `s`, or matrix `s` x `ne` (where `ne` is the length of `dmu_eval`) with the weights of the output slacks. If 0, input-oriented.
- `rts` A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
- `L` Lower bound for the generalized returns to scale (grs).
- `U` Upper bound for the generalized returns to scale (grs).
- `compute_target` Logical. If it is `TRUE`, it computes targets.
model_additive

returnlp Logical. If it is TRUE, it returns the linear problems (objective function and constraints).

... Ignored, for compatibility issues.

Note

In this model, the efficiency score is the sum of the slacks. Therefore, a DMU is efficient when the objective value (objval) is zero.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

References


See Also

model_addsupereff

Examples

# Example 1.
# Replication of results in Charnes et al (1994, p. 27)
x <- c(2, 3, 6, 9, 5, 4, 10)
y <- c(2, 5, 7, 8, 3, 1, 7)
data_example <- data.frame(dmus = letters[1:7], x, y)
data_example <- make_deadata(data_example,
   ni = 1,
   no = 1)
result <- model_additive(data_example,
   rts = "vrs")
efficiencies(result)
slacks(result)
lambdas(result)
# Example 2.
# Measure of Inefficiency Proportions (MIP).
x <- c(2, 3, 6, 9, 5, 4, 10)
y <- c(2, 5, 7, 8, 3, 1, 7)
data_example <- data.frame(dmus = letters[1:7], x, y)
data_example <- make_deadata(data_example, 
    ni = 1, 
    no = 1)
result2 <- model_additive(data_example, 
    rts = "vrs", 
    weight_slack_i = 1 / data_example["input"], 
    weight_slack_o = 1 / data_example["output"])
slacks(result2)

# Example 3.
# Range Adjusted Measure of Inefficiencies (RAM).
x <- c(2, 3, 6, 9, 5, 4, 10)
y <- c(2, 5, 7, 8, 3, 1, 7)
data_example <- data.frame(dmus = letters[1:7], x, y)
data_example <- make_deadata(data_example, 
    ni = 1, 
    no = 1)
range_i <- apply(data_example["input"], 1, max) - 
    apply(data_example["input"], 1, min)
range_o <- apply(data_example["output"], 1, max) - 
    apply(data_example["output"], 1, min)
w_range_i <- 1 / (range_i * (dim(data_example["input"])[1] + 
                     dim(data_example["output"])[1]))
w_range_o <- 1 / (range_o * (dim(data_example["input"])[1] + 
                     dim(data_example["output"])[1]))
result3 <- model_additive(data_example, 
    rts = "vrs", 
    weight_slack_i = w_range_i, 
    weight_slack_o = w_range_o)
slacks(result3)

---

**model_addmin**

Additive-min DEA model.

**Description**
Solve the weighted version of the additive-min (mADD) model of Aparicio et al (2007) with different returns to scale. For non constant returns to scale, a modification given by Zhu et al. (2018) is done.

**Usage**

```r
model_addmin(datadea, 
    dmu_eval = NULL,
```
model_addmin

dmu_ref = NULL,
orientation = NULL,
weight_slack_i = 1,
weight_slack_o = 1,
rts = c("crs", "vrs", "nirs", "ndrs"),
method = c("mf", "milp"),
extreff = NULL,
M_d = NULL,
M_lambda = 1e3,
maxfr = NULL,
tol = 1e-6,
silent = TRUE,
compute_target = TRUE,
check_target = FALSE,
returnlp = FALSE,
...)

Arguments

datadea  A deadata object with n DMUs, m inputs and s outputs.
dmu_eval  A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
dmu_ref  A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
orientation  This parameter is either NULL (default) or a string, equal to "io" (input-oriented) or "oo" (output-oriented). It is used to modify the weight slacks. If input-oriented, weight_slack_o are taken 0. If output-oriented, weight_slack_i are taken 0.
weight_slack_i  A value, vector of length m, or matrix m x ne (where ne is the length of dmu_eval) with the weights of the input slacks. If 0, output-oriented.
weight_slack_o  A value, vector of length s, or matrix s x ne (where ne is the length of dmu_eval) with the weights of the output slacks. If 0, input-oriented.
rts  A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing) or "ndrs" (non-decreasing). Under non-increasing or non-decreasing returns to scale, you may set check_target = TRUE because methods are not reliable. Generalized returns to scale are not available.
method  A string with the method: "mf" (default) for maximal friends, or "milp" for the mixed integer linear program of Aparicio et al. (2007). MILP method is faster but very problematic numerically.
extreff  A vector with the extreme efficient DMUs for "milp" method, as it is returned by function extreme_efficient. If NULL (default) this vector is computed internally.
M_d  Numeric, a big positive quantity for "milp" method. It is an upper bound for auxiliary variables named "d" in Aparicio (2007). If NULL (default), it is estimated automatically. A very big value can produce catastrophic cancellations. If the results are not correct or the solver hangs, try to change its value.
**Note**

In this model, the efficiency score is the sum of the slacks. Therefore, a DMU is efficient when the objective value (objval) is zero.

**Author(s)**

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

**Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**References**


**See Also**

`model_additive, extreme_efficient, maximal_friends`
**Examples**

```r
# Example 1.
data("Airlines")
datadea <- make_deadata(Airlines,
                        inputs = 4:7,
                        outputs = 2:3)
result <- model_addmin(datadea = datadea,
                        method = "milp")
targets(result)
```

```r
## Not run:
# Example 2. Directional model with Additive-min model in second stage
data("Airlines")
datadea <- make_deadata(Airlines,
                        inputs = 4:7,
                        outputs = 2:3)
resdir <- model_basic(datadea = datadea,
                        orientation = "dir",
                        maxslack = FALSE)
proj_input <- targets(resdir)[[1]] + slacks(resdir)[[1]]
proj_output <- targets(resdir)[[2]] - slacks(resdir)[[2]]
nd <- ncol(datadea$dmunames) # Number of DMUs
maxfr <- maximal_friends(datadea = datadea)
for (i in 1:nd) {
  datadea2 <- datadea
  datadea2$input[, i] <- proj_input[i, ]
  datadea2$output[, i] <- proj_output[i, ]
  DMUaux <- model_addmin(datadea = datadea2,
                        method = "mf",
                        maxfr = maxfr,
                        dmu_eval = i)$DMU[[1]]
  resdir$DMU[[i]]$slack_input <- DMUaux$slack_input
  resdir$DMU[[i]]$slack_output <- DMUaux$slack_output
  resdir$DMU[[i]]$target_input <- DMUaux$target_input
  resdir$DMU[[i]]$target_output <- DMUaux$target_output
}
targets(resdir)
```

```r
## End(Not run)
```

---

**model_addsupereff**  
Additive super-efficiency DEA model.

---

**Description**

Solve the additive super-efficiency model proposed by Du, Liang and Zhu (2010). It is an extension of the SBM super-efficiency to the additive DEA model.
Usage

```r
test_model <- model_addsupereff(
  datadea,
  dmu_eval = NULL,
  dmu_ref = NULL,
  orientation = NULL,
  weight_slack_i = NULL,
  weight_slack_o = NULL,
  rts = c("crs", "vrs", "nirs", "ndrs", "grs"),
  L = 1,
  U = 1,
  compute_target = TRUE,
  returnlp = FALSE,
  ...
)
```

Arguments

- `datadea`: A deadata object with n DMUs, m inputs and s outputs.
- `dmu_eval`: A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
- `dmu_ref`: A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
- `orientation`: This parameter is either NULL (default) or a string, equal to "io" (input-oriented) or "oo" (output-oriented). It is used to modify the weight slacks. If input-oriented, `weight_slack_o` are taken 0. If output-oriented, `weight_slack_i` are taken 0.
- `weight_slack_i`: A value, vector of length m, or matrix m x ne (where ne is the length of `dmu_eval`) with the weights of the input super-slacks (t_input). If 0, output-oriented. If `weight_slack_i` is the matrix of the inverses of inputs of DMUS in `dmu_eval` (default), the model is unit invariant.
- `weight_slack_o`: A value, vector of length s, or matrix s x ne (where ne is the length of `dmu_eval`) with the weights of the output super-slacks (t_output). If 0, input-oriented. If `weight_slack_o` is the matrix of the inverses of outputs of DMUS in `dmu_eval` (default), the model is unit invariant.
- `rts`: A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
- `L`: Lower bound for the generalized returns to scale (grs).
- `U`: Upper bound for the generalized returns to scale (grs).
- `compute_target`: Logical. If it is TRUE, it computes targets, projections and slacks.
- `returnlp`: Logical. If it is TRUE, it returns the linear problems (objective function and constraints).
- `...`: Ignored, for compatibility issues.
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References


See Also

*model_additive, model_supereff, model_sbmsupereff*

Examples

# Replication of results in Du, Liang and Zhu (2010, Table 6, p.696)
data("Power_plants")
Power_plants <- make_deadata(Power_plants,
                           ni = 4,
                           no = 2)
result <- model_addsupereff(Power_plants,
                           rts = "crs")
efficiencies(result)

---

**model_basic**

*Basic (radial and directional) DEA model.*

Description

It solves input and output oriented, along with directional, basic DEA models (envelopment form) under constant (CCR model), variable (BCC model), non-increasing, non-decreasing or generalized returns to scale. By default, models are solved in a two-stage process (slacks are maximized).

You can use the `model_basic` function to solve directional DEA models by choosing orientation = "dir".

The `model_basic` function allows to treat with non-discretional, non-controllable and undesirable inputs/outputs.
**Usage**

```r
model_basic(datadea, 
  dmu_eval = NULL, 
  dmu_ref = NULL, 
  orientation = c("io", "oo", "dir"), 
  dir_input = NULL, 
  dir_output = NULL, 
  rts = c("crs", "vrs", "nirs", "ndrs", "grs"), 
  L = 1, 
  U = 1, 
  maxslack = TRUE, 
  weight_slack_i = 1, 
  weight_slack_o = 1, 
  vtrans_i = NULL, 
  vtrans_o = NULL, 
  compute_target = TRUE, 
  compute_multiplier = FALSE, 
  returnlp = FALSE, 
  silent_ud = FALSE, 
  ...)```

**Arguments**

- `datadea` A `deadata` object with n DMUs, m inputs and s outputs.
- `dmu_eval` A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
- `dmu_ref` A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
- `orientation` A string, equal to "io" (input oriented), "oo" (output oriented), or "dir" (directional).
- `dir_input` A value, vector of length m, or matrix m x ne (where ne is the length of `dmu_eval`) with the input directions. If `dir_input` == input matrix (of DMUS in `dmu_eval`) and `dir_output` == 0, it is equivalent to input oriented (beta = 1 - efficiency). If `dir_input` is omitted, input matrix (of DMUS in `dmu_eval`) is assigned.
- `dir_output` A value, vector of length s, or matrix s x ne (where ne is the length of `dmu_eval`) with the output directions. If `dir_input` == 0 and `dir_output` == output matrix (of DMUS in `dmu_eval`), it is equivalent to output oriented (beta = efficiency - 1). If `dir_output` is omitted, output matrix (of DMUS in `dmu_eval`) is assigned.
- `rts` A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
- `L` Lower bound for the generalized returns to scale (grs).
- `U` Upper bound for the generalized returns to scale (grs).
- `maxslack` Logical. If it is TRUE, it computes the max slack solution.
model_basic

weight_slack_i  A value, vector of length m, or matrix m x ne (where ne is the length of dmu_eval) with the weights of the input slacks for the max slack solution.

weight_slack_o  A value, vector of length s, or matrix s x ne (where ne is the length of dmu_eval) with the weights of the output slacks for the max slack solution.

vtrans_i  Numeric vector of translation for undesirable inputs with non-directional orientation. If vtrans_i[i] is NA, then it applies the "max + 1" translation to the i-th undesirable input. If vtrans_i is a constant, then it applies the same translation to all undesirable inputs. If vtrans_i is NULL, then it applies the "max + 1" translation to all undesirable inputs.

vtrans_o  Numeric vector of translation for undesirable outputs with non-directional orientation, analogous to vtrans_i, but applied to outputs.

compute_target  Logical. If it is TRUE, it computes targets of the max slack solution.

compute_multiplier  Logical. If it is TRUE, it computes multipliers (dual solution) when orientation is "io" or "oo".

returnlp  Logical. If it is TRUE, it returns the linear problems (objective function and constraints) of stage 1.

silent_ud  Logical. For internal use, to avoid multiple warnings in the execution of malmquist_index function with undesirable variables.

...  Ignored, for compatibility issues.

Note

(1) Model proposed by Seiford and Zhu (2002) is applied for undesirable inputs/outputs and non-directional orientation (i.e., input or output oriented). You should select "vrs" returns to scale (BCC model) in order to maintain translation invariance. If deaR detects that you are not specifying rts = "vrs", it makes the change to "vrs" automatically.

(2) With undesirable inputs and non-directional orientation use input-oriented BCC model, and with undesirable outputs and non-directional orientation use output-oriented BCC model. Alternatively, you can also treat the undesirable outputs as inputs and then apply the input-oriented BCC model (similarly with undesirable inputs).

(3) Model proposed by Fare and Grosskopf (2004) is applied for undesirable inputs/outputs and directional orientation.

(4) With orientation = "dir" (directional distance function model), efficient DMUs are those for which beta = 0.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)
References


Undesirable inputs/outputs:


Non-discretionary/Non-controllable inputs/outputs:


Directional DEA model:


See Also

model_multiplier, model_supereff

Examples

# Example 1. Basic DEA model with desirable inputs/outputs.
# Replication of results in Charnes, Cooper and Rhodes (1981).
data("PFT1981")
# Selecting DMUs in Program Follow Through (PFT)
PFT <- PFT1981[1:49, ]
PFT <- make_deadata(PFT, inputs = 2:6,
```r
outputs = 7:9 )
eval_pft <- model_basic(PFT,
  orientation = "io",
  rts = "crs")

eff <- efficiencies(eval_pft)
s <- slacks(eval_pft)
lamb <- lambdas(eval_pft)
tar <- targets(eval_pft)
ref <- references(eval_pft)
returns <- rts(eval_pft)

# Example 2. Basic DEA model with undesirable outputs.
# Replication of results in Hua and Bian (2007).
data("Hua_Bian_2007")
# The third output is an undesirable output.
data_example <- make_deadata(Hua_Bian_2007,
  ni = 2,
  no = 3,
  ud_outputs = 3)
# Translation parameter (vtrans_o) is set to 1500
result <- model_basic(data_example,
  orientation = "oo",
  rts = "vrs",
  vtrans_o = 1500)
eff <- efficiencies(result)
1 / eff # results M5 in Table 6-5 (p.119)

# Example 3. Basic DEA model with non-discretionary (fixed) inputs.
# Replication of results in Ruggiero (2007).
data("Ruggiero2007")
# The second input is a non-discretionary input.
datadea <- make_deadata(Ruggiero2007,
  ni = 2,
  no = 1,
  nd_inputs = 2)
result <- model_basic(datadea,
  orientation = "io",
  rts = "crs")
efficiencies(result)
```

---

**Preference Structure DEA model.**

**Description**

With this non-radial DEA model (Zhu, 1996), the user can specify the preference input (or output) weights that reflect the relative degree of desirability of the adjustments of the current input (or output) levels.
usage

```r
model_deaps(datadea, 
  dmu_eval = NULL, 
  dmu_ref = NULL, 
  weight_eff = 1, 
  orientation = c("io", "oo"), 
  rts = c("crs", "vrs", "nirs", "ndrs", "grs"), 
  L = 1, 
  U = 1, 
  restricted_eff = TRUE, 
  maxslack = TRUE, 
  weight_slack = 1, 
  compute_target = TRUE, 
  returnlp = FALSE, 
  ...
)
```

Arguments

datadea A `deadata` object, including \( n \) DMUs, \( m \) inputs and \( s \) outputs.
dmu_eval A numeric vector containing which DMUs have to be evaluated. If `NULL` (default), all DMUs are considered.
dmu_ref A numeric vector containing which DMUs are the evaluation reference set. If `NULL` (default), all DMUs are considered.
weight_eff Preference weights. If input-oriented, it is a value, vector of length \( m \), or matrix \( m \times ne \) (where \( ne \) is the length of `dmu_eval`) with the weights applied to the input efficiencies. If output-oriented, it is a value, vector of length \( s \), or matrix \( s \times ne \) with the weights applied to the output efficiencies.
orientation A string, equal to "io" (input-oriented) or "oo" (output-oriented).
rts A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
L Lower bound for the generalized returns to scale (grs).
U Upper bound for the generalized returns to scale (grs).
restricted_eff Logical. If it is `TRUE`, the efficiencies are restricted to be <=1 (input-oriented) or >=1 (output-oriented).
maxslack Logical. If it is `TRUE`, it computes the max slack solution.
weight_slack If input-oriented, it is a value, vector of length \( s \), or matrix \( s \times ne \) with the weights of the output slacks for the max slack solution. If output-oriented, it is a value, vector of length \( m \), or matrix \( m \times ne \) with the weights of the input slacks for the max slack solution.
compute_target Logical. If it is `TRUE`, it computes targets of the max slack solution.
returnlp Logical. If it is `TRUE`, it returns the linear problems (objective function and constraints) of stage 1.
... Ignored, for compatibility issues.
Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

References


See Also

model_nonradial, model_profit, model_sbmeff

Examples

data("Fortune500")
data_deaps <- make_deadata(datadea = Fortune500,
ni = 3,
no = 2)
result <- model_deaps(data_deaps,
weight_eff = c(1, 2, 3),
orientation = "io",
rts = "vrs")

efficiencies(result)

model_fdh

Free disposal hull (FDH) model.

Description

FDH model allows the free disposability to construct the production possibility set. The central feature of the FDH model is the lack of convexity for its production possibility set (Thrall, 1999).

Usage

model_fdh(datadea,
fdh_modelname = c("basic"),
...)

Arguments

- `datadea` A deadata object, including DMUs, inputs and outputs.
- `fdh_modelname` A string containing the name of the model to apply FDH. For now, only "basic" is available.
- `...` `dmu_eval`, `dmu_ref`, `orientation` and other model parameters. Parameters like `rts`, `max_slack` and `returnlp` are ignored.

Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics*.
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References


Examples

```r
# Example 1. FDH input-oriented.
# Replication of results in Sanei and Mamizadeh Chatghayeb (2013)
data("Supply_Chain")
data_fdh1 <- make_deadata(Supply_Chain,
                           inputs = 2:4,
                           outputs = 5:6)
result <- model_fdh(data_fdh1) # by default orientation = "io"
efficiencies(result)

# Example 2. FDH output-oriented.
# Replication of results in Sanei and Mamizadeh Chatghayeb (2013)
data("Supply_Chain")
data_fdh2 <- make_deadata(Supply_Chain,
                           inputs = 5:6,
                           outputs = 7:8)
result2 <- model_fdh(data_fdh2,
                      orientation = "oo")
efficiencies(result2)
```
model_multiplier  Multiplier DEA model

Description

Solve input-oriented and output-oriented basic DEA models (multiplicative form) under constant (CCR DEA model), variable (BCC DEA model), non-increasing, non-decreasing or generalized returns to scale. It does not take into account non-controllable, non-discretionary or undesirable inputs/outputs.

Usage

model_multiplier(datadea, 
    dmu_eval = NULL, 
    dmu_ref = NULL, 
    epsilon = 0, 
    orientation = c("io", "oo"), 
    rts = c("crs", "vrs", "nirs", "ndrs", "grs"), 
    L = 1, 
    U = 1, 
    returnlp = FALSE, 
    compute_lambda = TRUE, 
    ...) 

Arguments

datadea  A deadata object, including DMUs, inputs and outputs.
dmu_eval  A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
dmu_ref  A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
epsilon  Numeric, multipliers must be >= epsilon.
orIENTATION  A string, equal to "io" (input-oriented) or "oo" (output-oriented).
rts  A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
L  Lower bound for the generalized returns to scale (grs).
U  Upper bound for the generalized returns to scale (grs).
returnlp  Logical. If it is TRUE, it returns the linear problems (objective function and constraints).
compute_lambda  Logical. If it is TRUE, it computes the dual problem and lambdas.
...  Ignored, for compatibility issues.
Note

(1) Very important with the multiplier model: "The optimal weights for an efficient DMU need not be unique" (Cooper, Seiford and Tone, 2007:31). "Usually, the optimal weights for inefficient DMUs are unique, the exception being when the line of the DMU is parallel to one of the boundaries of the feasible region" (Cooper, Seiford and Tone, 2007:32).

(2) The measure of technical input (or output) efficiency obtained by using multiplier DEA models is better the smaller the value of epsilon.

(3) Epsilon is usually set equal to 10^-6. However, if epsilon is not set correctly, the multiplier model can be infeasible (Zhu, 2014:49).

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

References


See Also

model_basic, cross_efficiency

Examples

# Example 1.
# Replication of results in Golany and Roll (1989).
data("Golany_Roll_1989")
data_example <- make_deadata(datadea = Golany_Roll_1989[1:10, ],
ininputs = 2:4,
### model_nonradial

**Non-radial DEA model.**

**Description**

Non-radial DEA model allows for non-proportional reductions in each input or augmentations in each output.

**Usage**

```r
model_nonradial(datadea,
    dmu_eval = NULL,
    dmu_ref = NULL,
    orientation = c("io", "oo"),
    rts = c("crs", "vrs", "nirs", "ndrs", "grs"),
    L = 1,
    U = 1,
    maxslack = TRUE,
    weight_slack = 1,
    compute_target = TRUE,
    returnlp = FALSE,
    ...)
```

Arguments

- **datadea**: A deodata object, including \( n \) DMUs, \( m \) inputs and \( s \) outputs.
- **dmu_eval**: A numeric vector containing which DMUs have to be evaluated. If `NULL` (default), all DMUs are considered.
- **dmu_ref**: A numeric vector containing which DMUs are the evaluation reference set. If `NULL` (default), all DMUs are considered.
- **orientation**: A string, equal to "io" (input-oriented) or "oo" (output-oriented).
- **rts**: A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
- **L**: Lower bound for the generalized returns to scale (grs).
- **U**: Upper bound for the generalized returns to scale (grs).
- **maxslack**: Logical. If it is `TRUE`, it computes the max slack solution.
- **weight_slack**: If input-oriented, it is a value, vector of length \( s \), or matrix \( s \times \text{ne} \) (where \( \text{ne} \) is the length of \( \text{dmu_eval} \)) with the weights of the output slacks for the max slack solution. If output-oriented, it is a value, vector of length \( m \), or matrix \( m \times \text{ne} \) with the weights of the input slacks for the max slack solution.
- **compute_target**: Logical. If it is `TRUE`, it computes targets of the max slack solution.
- **returnlp**: Logical. If it is `TRUE`, it returns the linear problems (objective function and constraints) of stage 1.
- **...**: Ignored, for compatibility issues.

Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References

model_profit

Profit efficiency DEA model.

Description

Cost, revenue and profit efficiency DEA models.

Usage

model_profit(datadea, 
  dmu_eval = NULL, 
  dmu_ref = NULL, 
  price_input = NULL, 
  price_output = NULL, 
  rts = c("crs", "vrs", "nirs", "ndrs", "grs"), 
  L = 1, 
  U = 1, 
  restricted_optimal = TRUE, 
  returnlp = FALSE, 
  ...) 

Arguments

datadea A deadata object, including n DMUs, m inputs and s outputs.
dmu_eval A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
dmu_ref A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
price_input Unit prices of inputs for cost or profit efficiency models. It is a value, vector of length m, or matrix m x ne (where ne is the length of dmu_eval).

See Also

model_deaps, model_profit, model_sbmeff

Examples

# Replication of results in Wu, Tsai and Zhou (2011)
data("Hotels")
data_hotels <- make_deadata(Hotels,  
  inputs = 2:5, 
  outputs = 6:8)
result <- model_nonradial(data_hotels,  
  orientation = "oo",  
  rts = "vrs")
efficiencies(result)
**model_profit**

- **price_output**: Unit prices of outputs for revenue or profit efficiency models. It is a value, vector of length $s$, or matrix $s \times ne$.

- **rts**: A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).

- **L**: Lower bound for the generalized returns to scale (grs).

- **U**: Upper bound for the generalized returns to scale (grs).

- **restricted_optimal**: Logical. If it is `TRUE`, the optimal inputs are restricted to be $\leq$ inputs (for cost efficiency models) or the optimal outputs are restricted to be $\geq$ outputs (for revenue efficiency models).

- **returnlp**: Logical. If it is `TRUE`, it returns the linear problems (objective function and constraints) of stage 1.

- **...**: Ignored, for compatibility issues.

**Author(s)**

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**References**


**See Also**

- `model_deaps`
- `model_nonradial`
- `model_sbmeff`

**Examples**

```r
# Example 1. Replication of results in Coelli et al. (1998, p.166).
# Cost efficiency model.
data("Coelli_1998")
# Selection of prices: input_prices is the transpose where the prices for inputs are.
input_prices <- t(Coelli_1998[, 5:6])
data_example1 <- make_deadata(Coelli_1998,
ni = 2,
no = 1)
result1 <- model_profit(data_example1,
price_input = input_prices,
rts = "crs",
restricted_optimal = FALSE)
# notice that the option by default is restricted_optimal = TRUE
```
# Example 2. Revenue efficiency model.
data("Coelli_1998")
# Selection of prices for output: output_prices is the transpose where the prices for outputs are.
output_prices <- t(Coelli_1998[, 7])
data_example2 <- make_deadata(Coelli_1998,
  ni = 2,
  no = 1)
result2 <- model_profit(data_example2,
  price_output = output_prices,
  rts = "crs",
  restricted_optimal = FALSE)
# notice that the option by default is restricted_optimal = TRUE
efficiencies(result2)

# Example 3. Profit efficiency model.
data("Coelli_1998")
# Selection of prices for inputs and outputs: input_prices and output_prices are
# the transpose where the prices (for inputs and outputs) are.
input_prices <- t(Coelli_1998[, 5:6])
output_prices <- t(Coelli_1998[, 7])
data_example3 <- make_deadata(Coelli_1998,
  ni = 2,
  no = 1)
result3 <- model_profit(data_example3,
  price_input = input_prices,
  price_output = output_prices,
  rts = "crs",
  restricted_optimal = FALSE)
# notice that the option by default is restricted_optimal = TRUE
efficiencies(result3)

---

**model_rdm**  
*Range directional model.*

**Description**
Range directional model from Portela et al. (2004).

**Usage**

```r
data = model_rdm(data,  
dmu_eval = NULL,  
dmu_ref = NULL,  
orientation = c("no", "io", "oo"),  
irdm = FALSE,  
maxslack = TRUE,  
weight_slack_i = 1,
```
model_rdm

weight_slack_o = 1,
compute_target = TRUE,
returnlp = FALSE,
...)

Arguments

datadea A deodata object, including n DMUs, m inputs and s outputs.
dmu_eval A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
dmu_ref A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
orientation A string, equal to "no" (non-oriented), "io" (input oriented), or "oo" (output oriented).
irdm Logical. If it is TRUE, it applies the IRDM (inverse range directional model).
maxslack Logical. If it is TRUE, it computes the max slack solution.
weight_slack_i A value, vector of length m, or matrix m x ne (where ne is the length of dmu_eval) with the weights of the input slacks for the max slack solution.
weight_slack_o A value, vector of length s, or matrix s x ne (where ne is the length of dmu_eval) with the weights of the output slacks for the max slack solution.
compute_target Logical. If it is TRUE, it computes targets of the max slack solution.
returnlp Logical. If it is TRUE, it returns the linear problems (objective function and constraints) of stage 1.
... Ignored, for compatibility issues.

Note

Undesirable inputs/outputs are treated as negative inputs/outputs in this model.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References

model_sbmeff

Slack based measure (SBM) of efficiency model.

Description

Calculate the SBM model proposed by Tone (2001).

Usage

model_sbmeff(datadea, 
dmu_eval = NULL, 
dmu_ref = NULL, 
weight_input = 1, 
weight_output = 1, 
orientation = c("no", "io", "oo"), 
rts = c("crs", "vrs", "nirs", "ndrs", "grs"), 
L = 1, 
U = 1, 
kaizen = FALSE, 
maxfr = NULL, 
tol = 1e-6, 
silent = FALSE, 
compute_target = TRUE, 
returnlp = FALSE, 
...)

Arguments

datadea A deadata object with n DMUs, m inputs and s outputs.
dmu_eval A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.
dmu_ref A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.
weight_input A value, vector of length m, or matrix m x ne (where ne is the length of dmu_eval) with weights to inputs corresponding to the relative importance of items.
weight_output A value, vector of length m, or matrix m x ne (where ne is the length of dmu_eval) with weights to outputs corresponding to the relative importance of items.
orientation A string, equal to "no" (non-oriented), "io" (input-oriented) or "oo" (output-oriented).
rts A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).
L Lower bound for the generalized returns to scale (grs).
U Upper bound for the generalized returns to scale (grs).
model_sbmeff

- **kaizen**: Logical. If TRUE, the kaizen version of SBM (Tone 2010), also known as SBM-Max, is computed.
- **maxfr**: A list with the maximal friends sets, as it is returned by function `maximal_friends`. If NULL (default) this list is computed internally.
- **tol**: Numeric, a tolerance margin for checking efficiency (only for the kaizen version).
- **silent**: Logical. If FALSE (default) it prints all the messages from function `maximal_friends`.
- **compute_target**: Logical. If it is TRUE, it computes targets.
- **returnlp**: Logical. If it is TRUE, it returns the linear problems (objective function and constraints). If kaizen is TRUE it is ignored.

... Other options (currently not implemented)

**Author(s)**

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**References**


**See Also**

- `model_nonradial`, `model_deaps`, `model_profit`, `model_sbmsupereff`

**Examples**

```r
# Example 1. Replication of results in Tone (2001, p.505)
data("Tone2001")
data_example <- make_deadata(Tone2001,
   ni = 2,
   no = 2)
result_SBM <- model_sbmeff(data_example,
   orientation = "no",
   ... Other options (currently not implemented)
```
model_sbmsupereff

Slack based measure of superefficiency model

Description

Slack based measure of superefficiency model (Tone 2002) with \( n \) DMUs, \( m \) inputs and \( s \) outputs.

Usage

model_sbmsupereff(datadea,
   dmu_eval = NULL,
   dmu_ref = NULL,
   weight_input = 1,
   weight_output = 1,
   orientation = c("no", "io", "oo"),
   rts = c("crs", "vrs", "nirs", "ndrs", "grs"),
   L = 1,
   U = 1,
   compute_target = TRUE,
model_sbmsupereff

```r
compute_rho = FALSE,
kaizen = FALSE,
silent = FALSE,
returnlp = FALSE)
```

### Arguments

- **data**
  A deadata object, including DMUs, inputs and outputs.

- **dmu_eval**
  A numeric vector containing which DMUs have to be evaluated. If `NULL` (default), all DMUs are considered.

- **dmu_ref**
  A numeric vector containing which DMUs are the evaluation reference set. If `NULL` (default), all DMUs are considered.

- **weight_input**
  A value, vector of length m, or matrix m x ne (where ne is the length of `dmu_eval`) with weights to inputs corresponding to the relative importance of items.

- **weight_output**
  A value, vector of length m, or matrix m x ne (where ne is the length of `dmu_eval`) with weights to outputs corresponding to the relative importance of items.

- **orientation**
  A string, equal to "no" (non-oriented), "io" (input-oriented) or "oo" (output-oriented).

- **rts**
  A string, determining the type of returns to scale, equal to "crs" (constant), "vrs" (variable), "nirs" (non-increasing), "ndrs" (non-decreasing) or "grs" (generalized).

- **L**
  Lower bound for the generalized returns to scale (grs).

- **U**
  Upper bound for the generalized returns to scale (grs).

- **compute_target**
  Logical. If it is `TRUE`, it computes targets, superslacks (t_input and t_output) and slacks.

- **compute_rho**
  Logical. If it is `TRUE`, it computes the SBM efficiency score (applying `model_sbmeff`) of the DMU (`project_input`, `project_output`).

- **kaizen**
  Logical. If `TRUE`, the kaizen version of SBM (Tone 2010), also known as SBM-Max, is computed for the efficiency score of the DMU (`project_input`, `project_output`).

- **silent**
  Logical. If `FALSE` (default) it prints all the messages from function `maximal_friends`.

- **returnlp**
  Logical. If it is `TRUE`, it returns the linear problems (objective function and constraints).

### Author(s)

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* *Applied Economics.*

**Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)
References


See Also

model_sbmeff, model_supereff, model_addsupereff

Examples

# Replication of results in Tone(2002, p.39)
data("Power_plants")
data_example <- make_deadata(Power_plants,
    ni = 4,
    no = 2)
result <- model_sbmsupereff(data_example,
    orientation = "io",
    rts = "crs")
efficiencies(result)
slacks(result)$slack_input
references(result)

model_supereff  
Radial super-efficiency basic DEA model

Description

Solve Andersen and Petersen radial Super-efficiency DEA model.

Usage

model_supereff(datadea,
    dmu_eval = NULL,
    dmu_ref = NULL,
    supereff_modelname = c("basic"),
    ...)

model_supereff

Arguments

**datadea**  
An object of class deadata.

**dmu_eval**  
A numeric vector containing which DMUs have to be evaluated. If NULL (default), all DMUs are considered.

**dmu_ref**  
A numeric vector containing which DMUs are the evaluation reference set. If NULL (default), all DMUs are considered.

**supereff_modelname**  
A string containing the name of the radial model to apply super-efficiency.

...  
orientation, rts and other model parameters.

Note

(1) Radial super-efficiency model under variable (vrs, nirs, ndrs, grs) returns to scale can be infeasible for certain DMUs. See example 2.

(2) DMUs with infeasible solution are not shown in the results.

Author(s)

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

**Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

References


See Also

model_basic, model_sbmsupereff, model_addsupereff

Examples

# Example 1.  
# Replication of results in Tone (2002, p.38)
data("Power_plants")
data_example <- make_deadata(Power_plants,  
ni = 4,  
no = 2)
result <- model_supereff(data_example,  
orientation = "io",  
rts = "crs")
eff <- efficiencies(result)
# Example 2.
# Results of Super-efficiency with vrs returns to scale show infeasibility solutions
# for DMUs D4 and D6 (these DMUs are not shown in deaR results).
data("Power_plants")
data_example2 <- make_deadata(Power_plants,
   ni = 4,
   no = 2)
result2 <- model_supereff(data_example2,
   orientation = "io",
   rts = "vrs")
eff2 <- efficiencies(result2)

---

### Description

Extract the multipliers of the DMUs from a dea or dea_fuzzy solution.

### Usage

`multipliers(deasol)`

### Arguments

- **deasol**: Object of class `dea` or `dea_fuzzy` obtained with some of the DEA model functions.

### Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* Applied Economics.
- **Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

### Examples

data("Coll_Blasco_2006")
data_example <- make_deadata(Coll_Blasco_2006,
   ni = 2,
   no = 2)
result <- model_multiplier(data_example,
   orientation = "io",
   rts = "crs")
multipliers(result)
Description

Data from Project Follow Through (PTF) in public school education. There are 49 DMUs (school sites) in PFT and 21 DMUs in Non-Follow Through (NFT). Authors consider 3 outputs (Y) and 5 inputs (X).

Usage

data("PFT1981")

Format

Data frame with 70 rows and 10 columns. Definition of inputs (X) and outputs (Y):

Y1 = Reading  Total Reading Scores (as measured by the Metropolitan Achievement Test).
Y2 = Math  Total Math Scores (total mathematics score by the Metropolitan Achievement Test).
Y3 = Coopersmith  Total Coopersmith Scores (Coopersmith self-esteem inventory, intended as a measure of self-esteem).
X1 = Education  Education level of mother (as measured in terms of percentage of high school graduates among female parents).
X2 = Occupation  Occupation Index (highest occupation of a family member according to a pre-arranged rating scale).
X3 = Parental  Parental Visit Index (representing the number of visits to the school site).
X4 = Counseling  Counseling Index (parent counselling index calculated from data on time spent with child on school-related topics such as reading together, etc.).
X5 = Teachers  Number of Teachers (number of teachers at a given site).
Program  PFT or NFT.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.
Vicente Bolos (<vicente.bolos@uv.es>). Department of Business Mathematics
Rafael Benitez (<rafael.suarez@uv.es>). Department of Business Mathematics
University of Valencia (Spain)

Source

See Also

make_deadata, model_basic

Examples

# Example 1. Replication of results in Charnes, Cooper and Rhodes (1981)
data("PFT1981")
# selecting DMUs in Project Follow Through (PFT)
PFT <- PFT1981[1:49,]
PFT <- make_deadata(PFT,
  dmus = 1,
  inputs = 2:6,
  outputs = 7:9)
eval_pft <- model_basic(PFT,
  orientation = "io",
  rts = "crs")
eff_pft <- efficiencies(eval_pft)

# Example 2. Replication of results in Charnes, Cooper and Rhodes (1981)
data("PFT1981")
# selecting DMUs in Non-Follow Through (NFT)
NFT <- PFT1981[50:70,]
NFT <- make_deadata(NFT,
  dmus = 1,
  inputs = 2:6,
  outputs = 7:9)
eval_nft <- model_basic(NFT,
  orientation = "io",
  rts = "crs")
eff_nft <- efficiencies(eval_nft)

plot.dea

Plot for DEA models.

Description

Plot some attribute of a DEA model.

Usage

## S3 method for class 'dea'
plot(x, tol = 1e-04, showPlots = TRUE, ...)

Arguments

x
  An object of class dea obtained by a DEA model function.

tol
  Numeric. Absolute tolerance for numeric comparisons. By default, it is 1e-4.
showPlots Logical. When TRUE (default) the plots are shown one by one. When it is FALSE the plots are not shown and are returned by the function (invisibly) as a list.

... Ignored, for compatibility issues.

Value

Depending on the model, it returns some plots.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

References


Examples

data_example <- make_deadata(datadea = Fortune500,  
    inputs = 2:4,  
    outputs = 5:6)

result <- model_basic(data_example)
plot(result)

plot.dea_fuzzy

Plot fuzzy DEA models.

Description

Plot some attributes of a fuzzy DEA model (Guo-Tanaka, Kao-Liu and possibilistic models).

Usage

## S3 method for class 'dea_fuzzy'
plot(x, showPlots = TRUE, ...)
**Arguments**

- **x**: An object of class `dea_fuzzy` obtained by a fuzzy DEA model function.
- **showPlots**: Logical. When TRUE (default) the plots are shown one by one. When it is FALSE the plots are not shown and are returned by the function (invisibly) as a list.
- **...**: Ignored, for compatibility issues.

**Value**

Depending on the model, it returns some plots.

**Author(s)**

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics*.

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**References**


---

**Power_plants**

**Data: Tone (2002).**

**Description**

This dataset consists of six power plants with 4 inputs (X) and 2 outputs (Y).

**Usage**

```r
data("Power_plants")
```

**Format**

Data frame with 15 rows and 7 columns. Definition of inputs (X) and outputs (Y):

- **x1**: Manpower required
- **x2**: Construction costs in millions of dollars
- **x3**: Annual maintenance costs in millions of dollars
- **x4**: Number of villages to be evacuated
- **y1**: Power generated in megawatts
- **y2**: Safety level
Power_plants

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. Applied Economics.

Vicente Bolos (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benitez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

make_deadata, model_supereff, model_sbmsupereff

Examples

# Example 1. Radial super-efficiency model.
# Replication of results in Tone (2002)
data("Power_plants")
data_example <- make_deadata(Power_plants,
ni = 4,
no = 2)
result <- model_supereff(data_example,
orientation = "io",
rts = "crs")
eff <- efficiencies(result)
eff

# Example 2. SBM super-efficiency model.
data("Power_plants")
data_example <- make_deadata(Power_plants,
ni = 4,
no = 2)
result2 <- model_sbmsupereff(data_example,
orientation = "io",
rts = "crs")
efficiencies(result2)
slacks(result2)$input
references(result2)
### print.deadata

**deadata class print method**

#### Description

Print method for `deadata` class.

#### Usage

```r
## S3 method for class 'deadata'
print(x, ...)
```

#### Arguments

- **x**: A `deadata` object (as returned by `make_deadata` function).
- **...**: For compatibility issues.

#### Author(s)

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

**Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

### print.deadata_fuzzy

**deadata_fuzzy class print method**

#### Description

Print method for `deadata_fuzzy` class.

#### Usage

```r
## S3 method for class 'deadata_fuzzy'
print(x, ...)
```

#### Arguments

- **x**: A `deadata_fuzzy` object (as returned by `make_deadata_fuzzy` function).
- **...**: For compatibility issues.
**read_data**

**Description**

This function is deprecated. Use `make_deadata` instead.

**Usage**

```r
read_data(datadea = NULL, 
ni = NULL, 
no = NULL, 
dmus = 1, 
inputs = NULL, 
outputs = NULL, 
nc_inputs = NULL, 
nc_outputs = NULL, 
nd_inputs = NULL, 
nd_outputs = NULL, 
ud_inputs = NULL, 
ud_outputs = NULL)
```

**Arguments**

- `datadea` Data frame with DEA data.
- `ni` Number of inputs, if inputs are in columns 2:(ni + 1) (if DMUs are in the first column) or 1:ni (no DMUs column).
- `no` Number of outputs, if outputs are in columns (ni + 2):(ni + no + 1) (if DMUs are in the first column) or (ni + 1):(ni + no) (no DMUs column). If not specified, DMUs are in the first column.
- `dmus` Column (number or name) of DMUs (optional). By default, it is the first column. If there is not any DMU column, then it must be NULL.
- `inputs` Columns (numbers or names) of inputs (optional). It prevails over `ni`. Alternatively to `datadea`, it can be a matrix with the inputs (DMUs in columns). In this case, DMUs names are taken from the columns names.
- `outputs` Columns (numbers or names) of outputs (optional). It prevails over `no`. Alternatively to `datadea`, it can be a matrix with the outputs (DMUs in columns).
nc_inputs A numeric vector containing the indices of non-controllable inputs.
nc_outputs A numeric vector containing the indices of non-controllable outputs.
nd_inputs A numeric vector containing the indices of non-discretionary inputs.
nd_outputs A numeric vector containing the indices of non-discretionary outputs.
ud_inputs A numeric vector containing the indices of undesirable (good) inputs.
ud_outputs A numeric vector containing the indices of undesirable (bad) outputs.

Description

This function is deprecated. Use make_deadata_fuzzy instead.

Usage

read_data_fuzzy(datadea, 
dmus = 1,
inputs.mL = NULL,
inputs.mR = NULL,
inputs.dL = NULL,
inputs.dR = NULL,
outputs.mL = NULL,
outputs.mR = NULL,
outputs.dL = NULL,
outputs.dR = NULL,
nc_inputs = NULL,
nc_outputs = NULL,
nd_inputs = NULL,
nd_outputs = NULL,
ud_inputs = NULL,
ud_outputs = NULL)

Arguments

datadea Data frame with DEA data.
dmus Column (number or name) of DMUs (optional). By default, it is the first column. If there is not any DMU column, then it must be NULL.
inputs.mL Where are (columns) the mL (left centers) of trapezoidal fuzzy inputs in datadea. If an input is triangular or crisp, we put the column where the centers or the crisp values are, respectively. Alternatively to datadea, inputs.mL can be a matrix of size (number of inputs x number of DMUs) with the mL of trapezoidal fuzzy inputs, the centers of triangular inputs, and the crisp values of crisp inputs. In this case, DMUs names are taken from the columns names.
inputs.mR Where are (columns) the mR (right centers) of trapezoidal fuzzy inputs in datadea. If an input is triangular or crisp, we put NA. Alternatively to datadea, inputs.mR can be a matrix of size (number of inputs x number of DMUs) with the mR of trapezoidal fuzzy inputs, the centers of triangular inputs, and the crisp values of crisp inputs. If all inputs are triangular or crisp, then inputs.mR must be NULL (default) or equal to inputs.mL.

inputs.dL Where are (columns) the dL (left radii) of trapezoidal and triangular fuzzy inputs in datadea. If an input is symmetric, we put the column where the radii are. If an input is rectangular or crisp, we put NA. Alternatively to datadea, inputs.dL can be a matrix of size (number of inputs x number of DMUs) with the dL of trapezoidal and triangular fuzzy inputs. If an input is rectangular or crisp, its radius is zero. If all inputs are rectangular or crisp, then inputs.dL must be NULL (default) or a zero matrix.

inputs.dR Where are (columns) the dR (right radii) of trapezoidal and triangular fuzzy inputs in datadea. If an input is symmetric, rectangular or crisp, we put NA. Alternatively to datadea, inputs.dR can be a matrix of size (number of inputs x number of DMUs) with the dR of trapezoidal and triangular fuzzy inputs. If an input is rectangular or crisp, its radius is zero. If all inputs are symmetric, rectangular or crisp, then inputs.dR must be NULL (default) or equal to inputs.dL.

outputs.mL Analogous to inputs.mL, but relating to outputs.
outputs.mR Analogous to inputs.mR, but relating to outputs.
outputs.dL Analogous to inputs.dL, but relating to outputs.
outputs.dR Analogous to inputs.dR, but relating to outputs.
nc_inputs A numeric vector containing the indices of non-controllable inputs.
nc_outputs A numeric vector containing the indices of non-controllable outputs.
nd_inputs A numeric vector containing the indices of non-discretionary inputs.
nd_outputs A numeric vector containing the indices of non-discretionary outputs.
ud_inputs A numeric vector containing the indices of undesirable (good) inputs.
ud_outputs A numeric vector containing the indices of undesirable (bad) outputs.

Description
This function is deprecated. Use make_malmquist instead.

Usage
read_malmquist(datadea,
              nper = NULL,
              percol = NULL,
              arrangement = c("horizontal", "vertical"),
              ...)
Arguments

datadea  Data frame with DEA data.
nper     Number of time periods (with dataset in wide format).
percol   Column of time period (with dataset in long format).
arrangement Horizontal with data in wide format. Vertical with data in long format.
...      Other options to be passed to the make_deadata function.

Description

Extract the reference set for each DMU (inefficient DMUs and efficient DMUs that are combination of other efficient DMUs) from a DEA model solution.

Usage

references(deasol,  
thr = 1e-4)

Arguments

deasol  Object of class dea obtained with some of the DEA model functions.
thr     Tolerance threshold (for avoiding miss detection of efficient DMUs due to round off errors)

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

Examples

# Replication results model DEA1 in Tomkins and Green (1988).
data("Departments")
# Calculate Total income
Departments$Total_income <- Departments[, 5] + Departments[, 6] + Departments[, 7]
data_DEA1 <- make_deadata(Departments,
                          inputs = 9,
                          outputs = c(2, 3, 4, 12))
result <- model_basic(data_DEA1,
                      orientation = "io",
                      thr = 1e-4)
rts = "crs")

references(result) # Table 3 (p.157)

<table>
<thead>
<tr>
<th>rts</th>
<th>Returns to scale</th>
</tr>
</thead>
</table>

Description

Extract the returns to scale.

Usage

```r
rts(deamodel,
   thr = 1e-4)
```

Arguments

deamodel: Object of class dea obtained with some of the DEA model functions.

thr: Threshold for the tolerance for considering something equal to 1. Defaults to 1e-4.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Examples

```r
data("Coll_Blasco_2006")
data_example <- make_deadata(Coll_Blasco_2006,
   ni = 2,
   no = 2)
result <- model_basic(data_example,
   orientation = "io",
   rts ="crs")

rts(result)
```
Description

Simulated data of 35 DMUs with two inputs and one output.

Usage

data("Ruggiero2007")

Format

Data frame with 35 rows and 4 columns. Definition of inputs (X) and outputs (Y):

- \( x_1 \) Input 1
- \( x_2 \) Input 2
- \( y_1 \) Output 1

Author(s)

- **Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*
- **Vicente Bolos** (<vicente.bolos@uv.es>). *Department of Business Mathematics*
- **Rafael Benitez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

- `make_deadata`, `model_basic`

Examples

```r
# Example. Replication of results in Ruggiero (2007).
data("Ruggiero2007")
# the second input is a non-discretionary input
datadea <- make_deadata(Ruggiero2007,
                       ni = 2,
                       no = 1,
                       nd_inputs = 2)
result <- model_basic(datadea,
```
slacks

```r
orientation = "io",
rt = "crs"

efficiencies(result)
slacks(result)
```

---

**slacks**

**Slacks**

---

**Description**

Extract the slacks of the DMUs from a dea or dea_fuzzy solution.

**Usage**

```r
slacks(deasol)
```

**Arguments**

- `deasol`: Object of class dea or dea_fuzzy obtained with some of the DEA model functions.

**Author(s)**

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2).* Applied Economics.

**Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**Examples**

```r
data("Coll_Blasco_2006")
data_example <- make_deadata(Coll_Blasco_2006,
   ni = 2,
   no = 2)
result <- model_multiplier(data_example,
   orientation = "io",
   rts = "crs")
slacks(result)
```
Summary conventional DEA models.

Description

Summary of the results obtained by a conventional DEA model.

Usage

```r
## S3 method for class 'dea'
summary(object, exportExcel = FALSE, filename = NULL, returnList = FALSE, ...)
```

Arguments

- `object`: An object of class "dea" obtained by a DEA model function.
- `exportExcel`: Logical value. If TRUE (FALSE by default) the results are also exported to an Excel file.
- `filename`: Character string. Absolute file name (including path) of the exported Excel file. If NULL, then the file name will be "ResultsDEA" + timestamp.xlsx.
- `returnList`: Logical value. If TRUE then the results are given as a list of data frames. If FALSE (default) all the data frames are merged into a single data frame.
- `...`: Ignored. Used for compatibility issues.

Value

Depending on the model it returns a single data.frame containing: efficiencies, slacks, lambdas, targets, references or a list of data.frames with the cross-efficiencies computed with different methods (Arbitrary, Method II or Method III (see CITA)) or, in case the model is a Malmquist index, a single data.frame with the coefficients for the different periods.

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (<vicente.bolos@uv.es>). Department of Business Mathematics

Rafael Benítez (<rafael.suarez@uv.es>). Department of Business Mathematics

University of Valencia (Spain)

References

Examples

data("PFT1981")
# Selecting DMUs in Program Follow Through (PFT)
PFT <- PFT1981[1:49, ]
PFT <- make_deadata(PFT,
   inputs = 2:6,
   outputs = 7:9 )
eval_pft <- model_basic(PFT,
   orientation = "io",
   rts = "crs")
summary(eval_pft)

summary.dea_fuzzy  Summary Fuzzy DEA models.

Description

Summary of the results obtained by a fuzzy DEA model.

Usage

## S3 method for class 'dea_fuzzy'
summary(object, ..., exportExcel = FALSE, filename = NULL, returnList = FALSE)

Arguments

object  An object of class "dea_fuzzy" obtained with a fuzzy DEA model function
   (modelfuzzy_guotanaka, modelfuzzy_kaoliu, modelfuzzy_possibilistic).
...
   Extra options.
exportExcel  Logical value. If TRUE (FALSE by default) the results are also exported to an
   Excel file.
filename  Character string. Absolute file name (including path) of the exported Excel file.
   If NULL, then the file name will be "ResultsDEA" + timestamp.xlsx.
returnList  Logical value. If TRUE then the results are given as a list of data frames. If
   FALSE (default) all the data frames are merged into a single data frame.

Value

If the model is that from Guo and Tanaka (modelfuzzy_guotanaka), it returns a data.frame with
   columns: DMU, alpha cuts and efficiencies. For the possibilistic model (modelfuzzy_possibilistic)
   it returns a data.frame with columns: DMU, alpha-cuts, efficiencies and the corresponding lambda
   values For the Kao-Liu model (modelfuzzy_kaoliu), the result may depend on the crisp sub-model
   used. It will contain a data.frame with the efficiencies (if any), the slacks and superslacks (if any),
   the lambda values and the targets.

If exportExcel is TRUE, then an Excel file will be created containing as many sheets as necessary
   depending on the variables returned.
Supply Chain

Author(s)

Vicente Coll-Serrano (vicente.coll@uv.es). Quantitative Methods for Measuring Culture (MC2). Applied Economics.

Vicente Bolós (vicente.bolos@uv.es). Department of Business Mathematics

Rafael Benítez (rafael.suarez@uv.es). Department of Business Mathematics

University of Valencia (Spain)

References


Examples

data("Leon2003")
data_example <- make_deadata_fuzzy(Leon2003,
    inputs.mL = 2,
    inputs.dL = 3,
    outputs.mL = 4,
    outputs.dL = 5)

result <- modelfuzzy_possibilistic(data_example,
    h = seq(0, 1, by = 0.1),
    orientation = "io",
    rts = "vrs")

summary(result)

Supply Chain

Data: Sanei and Mamizadeh Chatghayeb (2013).

Description

Data of 17 supply chain (buyer-supplier relationship in manufacturing).

Usage

data("Supply_Chain")

Format

Data frame with 17 rows and 8 columns. Definition of inputs (X) and outputs (Y):

X1 to X3 Inputs of buyers
I1 to I2 Outputs of buyers, Inputs of suppliers
Y1 to Y2 Outputs of suppliers
targets

Author(s)

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

**Vicente Bolos** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benitez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

Source


See Also

`make_deadata`, `model_fdh`

Examples

```
# Example. FDH input-oriented.
# Replication of results in Sanei and Mamizadeh Chatghayeb (2013)
data("Supply_Chain")
data_fdh1 <- make_deadata(Supply_Chain,
dmus = 1,
inputs = 2:4,
outputs = 5:6)
# by default orientation = "io"
result <- model_fdh(data_fdh1)
efficiencies(result)
```

### targets

<table>
<thead>
<tr>
<th>Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract the targets of the DMUs from a dea or dea_fuzzy solution.</td>
</tr>
</tbody>
</table>

#### Usage

```
targets(deasol)
```

#### Arguments

- `deasol` Object of class dea or dea_fuzzy obtained with some of the DEA model functions.
Tone2001

Data: Tone (2001).

Description

Data of 5 DMUs producing 2 outputs by using 2 inputs

Usage

data("Tone2001")

Format

Data frame with 5 rows and 5 columns. Definition of inputs (X) and outputs (Y):

x1 Input1
x2 Input2
y1 Output1
y2 Output2

Author(s)

Vicente Coll-Serrano (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2)*. *Applied Economics.*

Vicente Bolós (<vicente.bolos@uv.es>). *Department of Business Mathematics*

Rafael Benítez (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)
Source

See Also
make_deadata, model_sbmeff

Examples
```r
# Example. Replication of results in Tone (2001, p. 505)
data("Tone2003")
data_example <- make_deadata(Tone2003, ni = 2, no = 2)
result <- model_sbmeff(data_example, orientation = "no", rts = "crs")
efficiencies(result)
slacks(result)
```

Description
Data of 9 DMUs producing 2 outputs, being second output undesirable, by using 1 input.

Usage
data("Tone2003")

Format
Data frame with 9 rows and 4 columns. Definition of inputs (X) and outputs (Y):

- **x** Input
- **yg** Output1 ("good" output)
- **yb** Output2 (undesirable "bad" output)

Author(s)
Vicente Coll-Serrano (<vicente.coll@uv.es>). Quantitative Methods for Measuring Culture (MC2). Applied Economics.
Vicente Bolos (<vicente.bolos@uv.es>). Department of Business Mathematics
Rafael Benitez (<rafael.suarez@uv.es>). Department of Business Mathematics
University of Valencia (Spain)
undesirable_basic

Source

See Also
make_deadata, model_sbmeff

Examples

# Example. Replication of results in Tone (2003), pp 10-11.
data("Tone2003")
data_example <- make_deadata(Tone2003,
ni = 1,
no = 2,
ud_outputs = 2)
result <- model_sbmeff(data_example,
rts = "vrs")
efficiencies(result)
targets(result)

undesirable_basic

Undesirable inputs and outputs for basic DEA model.

Description
This function transforms a deadata or deadata_fuzzy class with undesirable inputs/outputs according to Seiford and Zhu (2002). Onwards, it is recommended to use a DEA model with variable returns to scale (vrs).

Usage
undesirable_basic(datadea,
vtrans_i = NULL,
vtrans_o = NULL)

Arguments
datadea A deadata object, including DMUs, inputs and outputs.
vtrans_i Numeric vector of translation for undesirable inputs. If vtrans_i[i] is NA, then it applies the "max + 1" translation to the i-th undesirable input. If vtrans_i is a constant, then it applies the same translation to all undesirable inputs. If vtrans_i is NULL, then it applies the "max + 1" translation to all undesirable inputs.
vtrans_o Numeric vector of translation for undesirable outputs, analogous to vtrans_i, but applied to outputs.
**Value**

An list with the transformed object of class deadata or deadata_fuzzy and the corresponding translation vectors vtrans_i and vtrans_o.

**Author(s)**

**Vicente Coll-Serrano** (<vicente.coll@uv.es>). *Quantitative Methods for Measuring Culture (MC2). Applied Economics.*

**Vicente Bolós** (<vicente.bolos@uv.es>). *Department of Business Mathematics*

**Rafael Benítez** (<rafael.suarez@uv.es>). *Department of Business Mathematics*

University of Valencia (Spain)

**References**


**Examples**

```r
data("Hua_Bian_2007")
data_example <- make_deadata(Hua_Bian_2007,
i = 2,
o = 3,
ud_outputs = 3)
result <- model_basic(data_example,
orientation = "oo",
rts = "vrs")
```

Index

* datasets

Airlines, 3
Coelli_1998, 6
Coll_Blasco_2006, 7
Departments, 13
Doyle_Green_1994, 14
Economy, 15
EconomyLong, 16
Electric_plants, 21
Fortune500, 23
Fried1993, 24
FuzzyExample, 25
Golany_Roll_1989, 27
Grifell_Lovell_1999, 28
Guo_Tanaka_2001, 29
Hotels, 30
Hua_Bian_2007, 32
Kao_Liu_2003, 37
Leon2003, 39
Libraries, 40
Lim_Zhu_2015, 42
PFT1981, 89
Power_plants, 92
Ruggiero2007, 100
Supply_Chain, 104
Tone2001, 106
Tone2003, 107

Airlines, 3
bootstrap_basic, 4

Coelli_1998, 6
Coll_Blasco_2006, 7
cross_efficiency, 8, 15, 27, 43, 75
cross_efficiency_fuzzy, 10, 11, 30, 40, 54

Departments, 13
Doyle_Green_1994, 14
Economy, 15

EconomyLong, 16
eff_dmus, 20
efficiencies, 17
efficiencies.dea, 18
efficiencies.dea_fuzzy, 19
Electric_plants, 21
extreme_efficient, 22, 63

Fortune500, 23
Fried1993, 24
FuzzyExample, 25

Golany_Roll_1989, 27
Grifell_Lovell_1999, 28
Guo_Tanaka_2001, 29

Hotels, 30
Hua_Bian_2007, 32

is.dea, 33
is.dea_fuzzy, 35
is.deadata, 34
is.deadata_fuzzy, 34
is.friends, 36, 53

Kao_Liu_2003, 37

lambda, 38
Leon2003, 39
Libraries, 40
Lim_Zhu_2015, 42

make_deadata, 4, 7, 8, 13, 15, 21, 24, 25, 27,
31, 32, 41, 43, 45, 90, 93, 100, 105, 107, 108

make_deadata_fuzzy, 26, 30, 38, 40, 45
make_malmquist, 16, 17, 28, 49
malmquist_index, 16, 17, 28, 49
maximal_friends, 36, 52, 63
model_additive, 58, 63, 66
model_addmin, 61

110
model_addsupereff, 60, 64, 86, 87
model_basic, 13, 21, 25, 32, 38, 41, 54, 56, 58, 66, 75, 87, 90, 100
model_deaps, 70, 78, 79, 83
model_fdh, 72, 105
model_multiplier, 10, 15, 24, 27, 43, 54, 56, 69, 74
model_nonradial, 31, 72, 76, 79, 83
model_profit, 72, 78, 78, 83
model_rdm, 80
model_sbmeff, 4, 36, 53, 72, 78, 79, 82, 86, 107, 108
model_sbmsupereff, 66, 83, 84, 87, 93
model_supereff, 66, 69, 86, 86, 93
modelfuzzy_guotanaka, 30, 40, 53, 56, 58
modelfuzzy_kaoliu, 54, 55, 58
modelfuzzy_possibilistic, 40, 54, 56, 57
multipliers, 88

PFT1981, 89
plot.dea, 90
plot.dea_fuzzy, 91
Power_plants, 92
print.deadata, 94
print.deadata_fuzzy, 94

read_data, 95
read_data_fuzzy, 96
read_malmquist, 97
references, 98
rts, 99
Ruggiero2007, 100

slacks, 101
summary.dea, 102
summary.dea_fuzzy, 103
Supply_Chain, 104

targets, 105
Tone2001, 106
Tone2003, 107

undesirable_basic, 108