Package ‘jackstrap’

June 9, 2020

Type Package
Title Correcting Nonparametric Frontier Measurements for Outliers
Version 0.1.0
Author Kleber Morais de Sousa [aut, cre], Maria da Conceição Sampaio de Sousa [aut], Paulo Aguiar do Monte [aut]
Maintainer Kleber Morais de Sousa <kleberfinancas@gmail.com>
Description Provides method used to check whether data have outlier in efficiency measurement of big samples with data envelopment analysis (DEA). In this jackstrap method, the package provides two criteria to define outliers: heaviside and k-s test. The technique was developed by Sousa and Stosic (2005) "Technical Efficiency of the Brazilian Municipalities: Correcting Nonparametric Frontier Measurements for Outliers." <doi:10.1007/s11123-005-4702-4>.

Depends R (>= 2.15.1)
Imports fBasics, Benchmarking, dplyr, ggplot2, foreach, doParallel, reshape, tidyr, scales, parallel, graphics, plyr, rlang, utils
License GPL-3
Encoding UTF-8
LazyData TRUE
RoxygenNote 7.1.0
Suggests knitr, rmarkdown
VignetteBuilder knitr
Language en-US
NeedsCompilation no
Repository CRAN
Date/Publication 2020-06-09 15:10:02 UTC

R topics documented:

 hist_jack_ks ................................................................. 2
 hist_jack_step ................................................................. 3
hist_jack_ks

__Description__

Histogram with Jackstrap Efficiency Indicators: This function builds graphics with distributions of efficiency indicators without outliers and complete sample. The outliers are defined by K-S Test.

__Usage__

```r
hist_jack_ks(efficiency, model_hist_ks)
```

__Arguments__

- `efficiency` is the jackstrap object created by jackstrap function.
- `model_hist_ks` is the desired graphic model. There are four kinds: 1- Density Histogram of efficiency indicator with complete sample and without outliers by K-S test; 2 - Histogram of efficiency with complete sample and without outliers by K-S test; 3 - Histogram of efficiency without outliers by K-S test; 4 - Histogram of efficiency with complete sample.

__Value__

Return the plot with efficiency indicators with complete sample and/or without outliers by combination leverage level and K-S test;

__Examples__

```r
#Build charts with efficiency indicators with jackstrap method and K-S test criterion
hist_jack_ks(efficiency_ks, 1)
hist_jack_ks(efficiency_ks, 2)
hist_jack_ks(efficiency_ks, 3)
hist_jack_ks(efficiency_ks, 4)
```
**hist_jack_step**

*Histogram with Jackstrap Efficiency Indicators:* This function builds a graphic with indicator distributions without outliers and complete sample. The outliers are defined by heaviside step function method.

**Description**

Histogram with Jackstrap Efficiency Indicators: This function builds a graphic with indicator distributions without outliers and complete sample. The outliers are defined by heaviside step function method.

**Usage**

`hist_jack_step(efficiency, model_hist_step)`

**Arguments**

- `efficiency` is the jackstrap object created by jackstrap function.
- `model_hist_step` is the desired graphic model. There are four kinds: 1 - Density Histogram of efficiency indicators with complete sample and without outliers by heaviside step function; 2 - Histogram of efficiency with complete sample and without outliers by heaviside step function; 3 - Histogram of efficiency without outliers by heaviside step function; 4 - Histogram of efficiency with complete sample.

**Value**

Return the plot with efficiency indicators with complete sample and/or without outliers by heaviside step function;

**Examples**

```r
# Build charts with efficiency indicators with jackstrap method and heaviside criterion
hist_jack_step(efficiency, 1)
hist_jack_step(efficiency, 2)
hist_jack_step(efficiency, 3)
hist_jack_step(efficiency, 4)
```
Description

Jackstrap Method: Tool identifies outliers in Nonparametric Frontier. This function applies the developed technique by Sousa and Stosic (2005) Technical Efficiency of the Brazilian Municipalities: Correcting Nonparametric Frontier Measurements for Outliers.

Usage

```r
jackstrap(
data, ycolumn, xcolumn, bootstrap = 1000, perc_sample_bubble = 0.1, dea_method = "vrs", orientation_dea = "in", n_seed = NULL, repos = FALSE, num_cores = 1)
```

Arguments

data is the dataset with input and output used to measure efficiency; Dataset need to have this form: 1th column: name of DMU (string); 2th column: code of DMU (integer); n columns of output variables; n columns of input variables.
ycolumn is the quantity of y columns of dataset.
xcolumn is the quantity of x columns of dataset.
bootstrap is the quantity of applied resampling.
perc_sample_bubble is the percentage of sample in each bubble.
dea_method is the dea method: "crs" is DEA with constant returns to scale (CCR); "vrs" is DEA with variable returns to scale; and "fdh" is Free Disposal Hull (FDH) with variable returns to scale.
orientation_dea is the direction of the DEA: "in" for focus on inputs; and "out" for focus on outputs.
n_seed is the code as seed used to get new random samples.
repos identify if the resampling method is with reposition TRUE or not FALSE.
num_cores is the number of cores available to process.
jackstrap_ks

Value

Return the jackstrap object with information as follows: "mean_leverage" is leverage average for each DMU; "mean_gerald_leverage" is general average of leverage and step function threshold; "sum_leverage" is accrued leverage on all resampling for each DMU; "count_dmu" is amount of each DMU was selected by bootstrap. "count_dmu_zero" is amount of each DMU was selected by bootstrap but it did not influence in others. "ycolumn" is the number of output variables; "xcolumn" is the number of input variables; "perc_sample_bubble" is the percentage of sample used in each bubble; "dea_method" is the model used in DEA analysis; "orientation_dea" is the orientation of DEA; "bootstrap" is the amount of bubble used by jackstrap method; "type_obj" is type of object; "size_bubble" is the amount of DMU used in each bubble.

Examples

```r
# Examples with the municipalities data.
# Load package jackstrap
library(jackstrap)

# Load data example
municipalities <- jackstrap::municipalities

data_efficiency <- jackstrap::municipalities

efficiency <- jackstrap(data=municipalities, ycolumn=2, xcolumn=1, bootstrap=1000,
                        perc_sample_bubble=0.20, dea_method="vrs", orientation_dea="in",
                        n_seed = 2000, repos=FALSE, num_cores=4)
```

Description

Jackstrap KS Method: Tool identifies outliers in Nonparametric Frontier. This function applies the developed technique by Sousa and Stosic (2005) Technical Efficiency of the Brazilian Municipalities: Correcting Nonparametric Frontier Measurements for Outliers and to use the K-S test with criterion to define outliers.

Usage

`jackstrap_ks(data, jackstrap_obj, num_cores = 1, perc = 0.9)`
municipalities

Arguments

data is the dataset with input and output used to measure efficiency; Dataset need to have this form: 1th column: name of DMU (string); 2th column: code of DMU (integer); n columns of output variables; n columns of input variables.

jackstrap_obj is the object created by the function jackstrap.

num_cores is the number of cores available to process.

perc is the percentage of DMU analyzed by K-S test.

Value

Return the jackstrap object increased with informations as follows: "result_kstest_method" is p-values of K-S test obtained by removing sequentially one by one the high leverage DMU; "efficiency_ks_method" is efficiency indicators obtained by K-S test criterion.

Examples

#Command measures efficiency with jackstrap method and K-S test criterion
efficiency_ks <- jackstrap_ks (data=municipalities, jackstrap_obj=efficiency,
num_cores = 4)

municipalities

Dataset of Municipalities of Bahia state in Brazil

Description

Dataset of Municipalities of Bahia state in Brazil

Usage

municipalities

Format

A data frame with 489 rows (DMUs) and 3 variables (2 outputs and 1 inputs):

municipio string variable with descriptions of the each local governments
cod integer variable identifies each DMU for integer code
total_atend_amb_hosp_ab float variable with public health services in local governments (output)
total_diversid float variable with diversity of public services provide in local governments (output)
desp_saude float variable with public service expeditures in local governments (input)
Examples

```r
# Load data example
municipalities <- jackstrap::municipalities
```

Description

Plot Jackstrap KS: This function plots p-value of Kolmogorov-Smirnov Test in decreasing order of leverage.

Usage

```r
plot_jackstrap_ks(data_plot, model_plot)
```

Arguments

- `data_plot`: is the jackstrap object created by jackstrap function.
- `model_plot`: is the desired model. There are two models: 1 - The graphic shows the amount of removed DMU on x axis and p-value of K-S test on y axis; 2 - The graphic shows DMU code on x axis and p-value of K-S test on y axis.

Value

Return the plot with p-value of K-S test and removed DMU or DMU code.

Examples

```r
# Plot the dispersion chart with p value of K-S test and amount of DMU removed.
plot_jackstrap_ks(effic_ks, 1)
```
Summary Jackstrap: This function shows the main outcomes with outlier technique developed by Sousa and Stosic(2005).

Description

Summary Jackstrap: This function shows the main outcomes with outlier technique developed by Sousa and Stosic(2005).

Usage

summary_jackstrap(object_jackstrap, data)

Arguments

- object_jackstrap: is the jackstrap object created by jackstrap function.
- data: is the dataset of research.

Value

Return the data frame with information as follows: "outliers_by_step_func" are the outliers by heaviside step function criterion; "outliers_by_ks" are the outliers by K-S test; "dmu_efficiency_by_step_func" are DMUs evaluated as efficient by heaviside step function criterion; "dmu_inefficiency_by_step_func" are the DMUs evaluated as maximum inefficient by heaviside step function criterion; "dmu_efficiency_ks" are DMUs evaluated as efficient by K-S test criterion; "dmu_inefficiency_by_ks" are the DMUs evaluated as maximum inefficient by K-S test criterion.

Examples

```r
#Create object with the resume of efficiency measurement.
summary_efficiency <- summary_jackstrap(efficiency_ks, municipalities)
```
Index

*Topic datasets
  municipalities, 6

hist_jack_ks, 2
hist_jack_step, 3

jackstrap, 4
jackstrap_ks, 5

municipalities, 6
plot_jackstrap_ks, 7
summary_jackstrap, 8