Package 'stodom'

May 9, 2026

Title Estimating Consistent Tests for Stochastic Dominance

Version 0.0.1

Description Stochastic dominance tests help ranking different distributions. The package implements the consistent test for stochastic dominance by Barrett and Donald (2003) <doi:10.1111/1468-0262.00390>. Specifically, it implements Barrett and Donald's Kolmogorov-Smirnov type tests for first- and second-order stochastic dominance based on bootstrapping 2 and 1.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

Imports dplyr, tibble, ggplot2, pracma, tidyr

Suggests testthat

NeedsCompilation no

Author Sergei Schaub [aut, cre] (ORCID:
<https://orcid.org/0000-0001-8477-3737>),
Agroscope [cph]

Maintainer Sergei Schaub <sergei.schaub@agroscope.admin.ch>

Repository CRAN

Date/Publication 2024-02-02 12:40:02 UTC

Contents

`dif_ecdf_plot` *plot difference ecdfs*

Description

This function computes the values of the cumulative difference of two empirical cumulative distribution function and plots the values.

Usage

```
dif_ecdf_plot(data_1, data_2, bins_size)
```

Arguments

`data_1`	data 1.
`data_2`	data 2.
`bins_size`	bin size.

Details

This function computes the values of the cumulative difference of two empirical cumulative distribution function and plots the values. This relates two showing second-order stochastic dominance.

Value

The function returns a plot as a ggplot2 object.

Examples

```
# load stodom
require(stodom)

 data_a <- rnorm(500, 3, 2)
 data_b <- rnorm(500, 1, 2)

# plot cumulative difference between two ecdfs
dif_ecdf_plot(data_1 = data_a, data_2 = data_b, bins_size = 0.1)
```

ecdf_dat_g *values of two ecdf and their cumulative difference*

Description

This function computes the values of two empirical cumulative distribution function as well as their cumulative differences.

Usage

```
ecdf_dat_g(data_1, data_2, bins_size)
```

Arguments

data_1	data 1.
data_2	data 2.
bins_size	bin size.

Details

This function computes the values of two empirical cumulative distribution function as well as their cumulative differences.

Value

The function returns a data table.

Examples

```
# load stodom
require(stodom)

 data_a <- rnorm(500, 3, 2)
 data_b <- rnorm(500, 1, 2)

# compute the values of two ecdfs and their cumulative differences.
ecdf_dat_g(data_1 = data_a, data_2 = data_b, bins_size = 1)
```

`ecdf_plot` *plot ecdfs*

Description

This function computes the values of two empirical cumulative distribution function and plots the values.

Usage

```
ecdf_plot(data_1, data_2, bins_size)
```

Arguments

`data_1`	data 1.
`data_2`	data 2.
`bins_size`	bin size.

Details

This function computes the values of two empirical cumulative distribution function and plots the values.

Value

The function returns a plot as a ggplot2 object.

Examples

```
# load stodom
require(stodom)

 data_a <- rnorm(500, 3, 2)
 data_b <- rnorm(500, 1, 2)

# plot ecdfs
ecdf_plot(data_1 = data_a, data_2 = data_b, bins_size = 0.1)
```

`fo_stodom` *first-order stochastic dominance test*

Description

This function tests for first-order stochastic dominance.

Usage

```
fo_stodom(data_1, data_2, bins_size, n_draws, useed, variable_1, variable_2, type)
```

Arguments

`data_1`	data 1.
`data_2`	data 2.
`bins_size`	bin size.
`n_draws`	number of draws to compute p values (default = 500).
`useed`	user defined seed
`variable_1`	name of a (as a string); only for the output table (default = "a").
`variable_2`	name of b (as a string); only for the output table (default = "b").
`type`	type of bootstrapped test, bootstrapping 1 and 2 of Barrett and Donald (2003) are available (default = "boot2").

Details

This function computes the consistent test of first-order stochastic dominance following Barrett and Donald (2003). In detail, this function estimate their Kolmogorov-Smirnov type tests based on bootstrapping 2. The function was implemented as part of Schaub xxx

Value

The function returns a list object containing the p-values of two dominance tests (i.e., variable 1 vs. variable 1 and variable 2 vs. variable 1).

References

Barrett, G. F., & Donald, S. G. (2003). Consistent tests for stochastic dominance. Econometrica, 71(1), 71-104.

Schaub, S. & El Benni, N. (2024). How do price (risk) changes influence farmers' preference to reduce fertilizer application?

Examples

```
# load stodom
require(stodom)

 data_a <- rnorm(500, 3, 2)
 data_b <- rnorm(500, 1, 2)

# estimate first-order stochastic dominance
fo_stodom(data_1 = data_a, data_2 = data_b, n_draws = 100, useed = 1, bins_size = 1)
```

`so_stodom` *second-order stochastic dominance test*

Description

This function tests for second-order stochastic dominance.

Usage

```
so_stodom(data_1, data_2, bins_size, n_draws, useed, variable_1, variable_2, type)
```

Arguments

`data_1`	data 1.
`data_2`	data 2.
`bins_size`	bin size.
`n_draws`	number of draws to compute p values (default = 500).
`useed`	user defined seed
`variable_1`	name of a (as a string); only for the output table (default = "a").
`variable_2`	name of b (as a string); only for the output table (default = "b").
`type`	type of bootstrapped test, bootstrapping 1 and 2 of Barrett and Donald (2003) are available (default = "boot2").

Details

This function computes the consistent test of second-order stochastic dominance following Barrett and Donald (2003). In detail, this function estimate their Kolmogorov-Smirnov type tests based on bootstrapping 2. The function was implemented as part of Schaub xxx

Value

The function returns a list object containing the p-values of two dominance tests (i.e., variable 1 vs. variable 1 and variable 2 vs. variable 1).

References

Barrett, G. F., & Donald, S. G. (2003). Consistent tests for stochastic dominance. Econometrica, 71(1), 71-104.

Schaub, S. & El Benni, N. (2024). How do price (risk) changes influence farmers' preference to reduce fertilizer application?

Examples

```
# load stodom
require(stodom)

 data_a <- rnorm(500, 3, 2)
 data_b <- rnorm(500, 1, 2)

# estimate second-order stochastic dominance
so_stodom(data_1 = data_a, data_2 = data_b, n_draws = 100, useed = 1, bins_size = 1)
```

Index