Package ‘concstats’

May 15, 2022

Type Package

Title Market Structure, Concentration and Inequality Measures

Version 0.1.2

Description Based on individual market shares of all participants in a market or space or their respective sales figures, the package offers a set of different structural and concentration measures frequently - and not so frequently - used in research and in practice. Measures can be calculated in groups or individually. The calculated measure or the resulting vector in table format should help practitioners make more informed decisions. Methods used in this package are from:
1. Chang, E. J., Guerra, S. M., de Souza Penaloza, R. A. & Tabak, B. M. (2005) ``Banking concentration: the Brazilian case''.
2. Cobham, A. and A. Summer (2013). ``Is It All About the Tails? The Palma Measure of Income Inequality''.
3. Garcia Alba Idunate, P. (1994). ``Un Indice de dominancia para el analisis de la estructura de los mercados''.
4. Ginevicius, R. and S. Cirba (2009). ``Additive measurement of market concentration'' <doi:10.3846/1611-1699.2009.10.191-198>.
5. Herfindahl, O. C. (1950), ``Concentration in the steel industry (PhD thesis)''.
6. Hirschmann, A. O. (1945), ``National power and structure of foreign trade''.
7. Melnik, A., O. Shy, and R. Stenbacka (2008), ``Assessing market dominance'' <doi:10.1016/j.jebo.2008.03.010>.
8. Palma, J. G. (2006). ``Globalizing Inequality: 'Centrifugal' and 'Centripetal' Forces at Work''.
9. Shannon, C. E. (1948). ``A Mathematical Theory of Communication''.

License MIT + file LICENSE

URL https://github.com/schneiderpy/concstats,
https//schneiderpy.github.io/constats (website)

BugReports https://github.com/schneiderpy/concstats/issues

Depends R (>= 2.10)

Imports dplyr (>= 1.0.7), readr

Suggests rmarkdown, knitr, devtools, kableExtra, ggplot2, testthat (>= 3.0.0), covr

VignetteBuilder knitr, rmarkdown

Encoding UTF-8

Language en-US

LazyData true

RoxygenNote 7.1.2

Config/testthat/edition 3

NeedsCompilation no

Author Andreas Schneider [aut, cre] (<https://orcid.org/0000-0001-5630-1097>)

Maintainer Andreas Schneider <schneiderconsultingpy@gmail.com>

Repository CRAN

Date/Publication 2022-05-14 23:20:20 UTC

R topics documented:

`comp` *Concentration Measures*

Description

A set of different concentration measures.

Usage

```
comp(x, unbiased = FALSE, type = c("hhi", "hhi_d", "hhi_min", "dom",
 "sten", "all"),na.rm = TRUE)

hhi(x, unbiased = FALSE, na.rm = TRUE)

hhi_min(x, na.rm = TRUE)

hhi_d(x, na.rm = TRUE)

dom(x, na.rm = TRUE)

sten(x, na.rm = TRUE)

all_comp(x, na.rm = TRUE)
```

Arguments

x	A non-negative numeric vector.
unbiased	Logical. Argument specifying whether or not a finite sample correction should be applied. The default is FALSE.
type	A character string of the measure to be calculated, can be abbreviated with the first letter. Defaults to "hhi".
na.rm	A logical vector that indicates whether NA values should be excluded or not.

Details

comp is a wrapper for the proposed concentration measures hhi, hhi_d, hhi_min, dom, sten, all. If no measure is specified "hhi" will be the default.

hhi, hhi_min, hhi_d calculate the Herfindahl-Hirschman index, its minimum, and its dual, respectively. dom calculates a dominance index and sten calculates the Stenbacka index. The index indicates the market share of a dominant position.

All measures can be accessed individually. hhi, hhi_d, and dom can be calculated individually as a normalized measure changing the default setting to TRUE. all computes all measures in a one step procedure.

Value

prints the calculated measure

Note

the vector of market shares should be in a decimal form corresponding to total share of individual firms/units. The sum of the vector should sum up to 1. You can also use sales figures to compute the respective measure.

See Also

concstats, mstruct, inequ

Examples

```
# a vector of market shares
x <- c(0.35, 0.4, 0.05, 0.1, 0.06, 0.04)
# the Herfindahl-Hirschmann index of the vector
shares_hhi <- comp(x, type = "hhi")
# individual measure
shares_sten <- sten(x)
# complete group measures
shares_comp <- comp(x, type = "all")
```

concstats *A set of Market Structure, Concentration, and Inequality Measures*

Description

a convenience function which calculates a selected set of different market structure, inequality and concentration measures more or less commonly used, e.g. k-firm ratios, Entropy, HHI, Palma ratio, and others in a one step procedure to provide a first overview.

Usage

```
concstats(x, na.rm = TRUE)
```

Arguments

x	a non-negative numeric vector.
na.rm	a logical vector that indicates whether NA values should be excluded or not. If set to FALSE the computation yields NA.

Details

concstats computes a set of different and selected structural, inequality, and concentration measures in a one step procedure, however, all measures can be computed individually or in groups.

Value

returns a data frame of calculated measures .

Note

the vector of market shares should be in a decimal form corresponding to the total share of individual firms/units. The vector should sum up to 1.

See Also

mstruct, comp, inequ

Examples

```
# a vector of market shares
# share <- c(0.35, 0.4, 0.05, 0.1, 0.06, 0.04)
# a selected set of different structural, concentration, and inequality
# measures
# share_conc <- concstats(share)
```

`creditcoops` *Creditcoops*

Description

data set with 22 paired Paraguayan credit cooperatives (2016, 2018)

Usage

```
creditcoops
```

Format

A data frame with 44 rows and 5 variables:

`coop_id` double, ID of the credit cooperative

`year` integer, sample year

`total_loans` double, total loans granted (USD) per year and cooperative

`paired` integer, paires of cooperatives

`total_loans_log` double, the natural log of total loans

Note

real names of the cooperatives have been purposely omitted, but are available on request.

Author(s)

Andreas Schneider

Source

http://www.incoop.gov.py/v2/

Examples

```
data("creditcoops")
head(creditcoops)
```

inequ *Inequality and Diversity Measures*

Description

A set of different inequality and diversity measures.

Usage

```
inequ(x, unbiased = FALSE, type = c("entropy", "gini", "simpson", "palma",
 "grs", "all"), na.rm = TRUE)

entropy(x, unbiased = FALSE, na.rm = TRUE)

gini(x, unbiased = FALSE, na.rm = TRUE)

simpson(x, unbiased = FALSE, na.rm = TRUE)

palma(x, na.rm = TRUE)

grs(x, na.rm = TRUE)

all_inequ(x, na.rm = TRUE)
```

Arguments

`x`	a non-negative numeric vector.
`unbiased`	Logical. Argument specifying whether or not a finite sample correction should be applied. The default is FALSE.
`type`	a character string of the measure to be calculated, defaults to "entropy".
`na.rm`	a logical vector that indicates whether `NA` values should be excluded or not. If set to `FALSE` the computation yields `NA`.

Details

`inequ` is a wrapper for the proposed inequality measures `entropy`, `gini`, `simpson`, `palma`, `grs`, `all`. If no measure is specified, "entropy" is the default.

`entropy` returns the (Shannon) Entropy, `gini` is the Gini coefficient, `simpson` is the complement of the Herfindahl-Hirschmann Index. You can normalize each of these three measures by setting `unbiased = TRUE`

`Palma` measures the ratio of inequality (used with income inequality) of the top 10 percent to the bottom 40 percent. `grs` is an alternative inequality measure (Ginevicius, 2009), `all` returns all measures in a one step procedure.

Value

prints the calculated measure

Note

the non- negative vector might market shares of individual firms/units. In the latter case the vector should sum up to 1.

See Also

concstats, mstruct, comp

Examples

```
# a vector of market shares
share <- c(0.4, 0.2, 0.25, 0.1, 0.05)
# Calculate the Palma ratio
share_p <- inequ(share, type = "palma")
# Calculate the entropy measure directly
share_ent <- entropy(share, unbiased = TRUE)
# Calculate the group measures
share_inequ <- inequ(share, type = "all")
```

---

mstruct *Market Structure Measures*

Description

Set of different measures to reflect a given market structure.

Usage

```
mstruct(x, type = c("firm", "nrs_eq", "top", "top3", "top5", "all"),
 na.rm = TRUE)

firm(x, na.rm = TRUE)

nrs_eq(x, na.rm = TRUE)

top(x, na.rm = TRUE)

top3(x, na.rm = TRUE)

top5(x, na.rm = TRUE)

all_mstruct(x, na.rm = TRUE)
```

Arguments

x A non-negative numeric vector.

type A character string of the measure to be calculated, can be abbreviated with the first letter. Defaults to "firm".

na.rm A logical vector that indicates whether NA values should be excluded or not.

Details

mstruct is a wrapper for the proposed structural measures firm, returns the number of firms with a given market share, nrs_eq computes the numbers equivalent, top, top3, and top5 calculate the share of the top (top 3 and top 5) firm(s). all computes all measures in a one step procedure. All measures can be computed individually.

Value

returns the calculated measure

Note

the vector of market shares should be in a decimal form corresponding to total share of individual firms/units.The sum of the vector should sum up to 1.

See Also

concstats, comp, inequ

Examples

```
# a vector of market shares
share <- c(0.35, 0.4, 0.05, 0.1, 0.06, 0.04)
# the number of firms with market share
share_firm <- mstruct(share, type = ”firm”)
# Calculate top market share individually
share_top <- top(share)
# Calculate the market structure group measures
share_mstruct <- mstruct(share, type = ”all”)
```

Index