Package 'ineq'

May 8, 2026

Version 0.2-13

Date 2014-07-21

Title Measuring Inequality, Concentration, and Poverty

Description
Inequality, concentration, and poverty measures. Lorenz curves (empirical and theoretical).

Depends R (>= 2.10.0)

Imports stats, graphics, grDevices

License GPL-2 | GPL-3

Author Achim Zeileis [aut, cre],
Christian Kleiber [ctb]

Maintainer Achim Zeileis <Achim.Zeileis@R-project.org>

NeedsCompilation no

Repository CRAN

Date/Publication 2014-07-21 20:10:45

Contents

conc *Concentration Measures*

Description

computes the concentration within a vector according to the specified concentration measure

Usage

```
conc(x, parameter = NULL, type = c("Herfindahl", "Rosenbluth"), na.rm = TRUE)

Herfindahl(x, parameter = 1, na.rm = TRUE)
Rosenbluth(x, na.rm = TRUE)
```

Arguments

x	a vector containing non-negative elements
parameter	parameter of the concentration measure (if set to NULL the default parameter of the respective measure is used)
type	character string giving the measure used to compute concentration. must be one of the strings in the default argument (the first character is sufficient). defaults to "Herfindahl".
na.rm	logical. Should missing values (NAs) be removed prior to computations? If set to FALSE the computations yield NA.

Details

conc is just a wrapper for the concentration measures of Herfindahl and Rosenbluth (Hall / Tiedemann / Rosenbluth). If parameter is set to NULL the default from the respective function is used.

Value

the value of the concentration measure

References

F A Cowell: Measurement of Inequality, 2000, in A B Atkinson / F Bourguignon (Eds): Handbook of Income Distribution, Amsterdam,

F A Cowell: Measuring Inequality, 1995 Prentice Hall/Harvester Wheatshef,

M Hall / N Tidemann: Measures of Concentration, 1967, JASA 62, 162-168.

See Also

ineq, pov

Examples

```
# generate vector (of sales)
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
# compute Herfindahl coefficient with parameter 1
conc(x)
# compute coefficient of Hall/Tiedemann/Rosenbluth
conc(x, type="Rosenbluth")
```

`Ilocos` *Income Metadata from Ilocos, Philippines*

Description

Income metadata from surveys conducted by the Philippines' National Statistics Office.

Usage

```
data(Ilocos)
```

Format

A data frame with 632 observations of 8 variables.

income total income of household,

sex sex of household head (`"male"` or `"female"`),

family.size family size (sometimes averaged over two semesters),

urbanity factor with levels `"rural"` and `"urban"`,

province factor indicating the particular province,

AP.income total household income during the APIS,

AP.family.size family size during APIS,

AP.weight APIS survey weight for each household.

Details

The data contains household income and metadata in one of the sixteen regions of the Philippines called Ilocos. The data comes from two of the NSO's surveys: the 1997 Family and Income and Expenditure Survey and the 1998 Annual Poverty Indicators Survey (APIS).

Since the APIS only has a six month reference period, the original data were rescaled using an adjustment factor from the quarterly GDP figures that can be obtained from the major sectors.

Source

National Statistics Office, Philippines: http://www.census.gov.ph/, where also the whole data set may be obtained.

ineq *Inequality Measures*

Description

computes the inequality within a vector according to the specified inequality measure

Usage

```
ineq(x, parameter = NULL, type = c("Gini", "RS", "Atkinson", "Theil", "Kolm", "var",
     "square.var", "entropy"), na.rm = TRUE)

Gini(x, corr = FALSE, na.rm = TRUE)
RS(x, na.rm = TRUE)
Atkinson(x, parameter = 0.5, na.rm = TRUE)
Theil(x, parameter = 0, na.rm = TRUE)
Kolm(x, parameter = 1, na.rm = TRUE)
var.coeff(x, square = FALSE, na.rm = TRUE)
entropy(x, parameter = 0.5, na.rm = TRUE)
```

Arguments

x	a vector containing at least non-negative elements
parameter	parameter of the inequality measure (if set to NULL the default parameter of the respective measure is used)
type	character string giving the measure used to compute inequality. must be one of the strings in the default argument (the first character is sufficient). defaults to "Gini".
corr	logical. Argument of the function Gini specifying whether or not a finite sample correction should be applied.
square	logical. Argument of the function var.coeff, for details see below.
na.rm	logical. Should missing values (NAs) be removed prior to computations? If set to FALSE the computations yield NA.

Details

ineq is just a wrapper for the inequality measures Gini, RS, Atkinson, Theil, Kolm,var.coeff, entropy. If parameter is set to NULL the default from the respective function is used.

Gini is the Gini coefficient, RS is the the Ricci-Schutz coefficient (also called Pietra's measure), Atkinson gives Atkinson's measure and Kolm computes Kolm's measure.

If the parameter in Theil is 0 Theil's entropy measure is computed, for every other value Theil's second measure is computed.

ineq(x, type="var") and var.coeff(x) respectively compute the coefficient of variation, while ineq(x,type="square.var") and var.coeff(x, square=TRUE) compute the squared coefficient of variation.

entropy computes the generalized entropy, which is for parameter 1 equal to Theil's entropy coefficient and for parameter 0 equal to the second measure of Theil.

Value

the value of the inequality measure

References

F A Cowell: Measurement of Inequality, 2000, in A B Atkinson / F Bourguignon (Eds): Handbook of Income Distribution, Amsterdam,

F A Cowell: Measuring Inequality, 1995 Prentice Hall/Harvester Wheatshef,

Marshall / Olkin: Inequalities: Theory of Majorization and Its Applications, New York 1979 (Academic Press).

See Also

conc, pov

Examples

```
# generate vector (of incomes)
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
# compute Gini coefficient
ineq(x)
# compute Atkinson coefficient with parameter=0.5
ineq(x, parameter=0.5, type="Atkinson")
```

Lasym *Lorenz Asymmetry Coefficient*

Description

Coefficient of asymmetry in the Lorenz curve.

Usage

```
Lasym(x, n = rep(1, length(x)), interval = FALSE, na.rm = TRUE)
```

Arguments

x	a vector containing non-negative elements.
n	a vector of frequencies, must be same length as x.
interval	logical. In the case where there are observations exactly equal to the mean, either an interval of asymmetry coefficients can be returned or their midpoint.
na.rm	logical. Should missing values (NAs) be removed prior to computations? If set to FALSE the computations yield NA.

Details

Damgaard and Weinter (2000) have suggested an additional measure for comparing inequality in in distributions (specifically for describing plant size or fecundity distributions) to accompany the Lorenz curve and Gini coefficient. It assesses the asymmetry in the Lorenz curve of the distributions.

References

C Damgaard, J Weiner: Describing Inequality in Plant Size or Fecundity, 2000. Ecology 81(4), 1139–1142.

See Also

Lc, Gini

Examples

```
## Examples from Damgaard & Weiner (2000)

## Figure 2
x <- rep(c(50/9, 50), c(9, 1))
y <- rep(c(2, 18), c(5, 5))
plot(table(x))
plot(table(y))

## statistics
mean(x)
mean(y)
Gini(x, corr = TRUE)
Gini(y, corr = TRUE)
Lasym(x)
Lasym(y)

## Figure 3
plot(Lc(x))
lines(Lc(y), col = "slategray")
abline(1, -1, lty = 2)
```

Lc *Lorenz Curve*

Description

Computes the (empirical) ordinary and generalized Lorenz curve of a vector x

Usage

```
Lc(x, n = rep(1,length(x)), plot = FALSE)
```

Arguments

x	a vector containing non-negative elements.
n	a vector of frequencies, must be same length as x.
plot	logical. If TRUE the empirical Lorenz curve will be plotted.

Details

Lc(x) computes the empirical ordinary Lorenz curve of x as well as the generalized Lorenz curve (= ordinary Lorenz curve * mean(x)). The result can be interpreted like this: p*100 percent have L(p)*100 percent of x.

If n is changed to anything but the default x is interpreted as a vector of class means and n as a vector of class frequencies: in this case Lc will compute the minimal Lorenz curve (= no inequality within each group). A maximal curve can be computed with Lc.mehran.

Value

A list of class "Lc" with the following components:

p	vector of percentages
L	vector with values of the ordinary Lorenz curve
L.general	vector with values of the generalized Lorenz curve

References

B C Arnold: Majorization and the Lorenz Order: A Brief Introduction, 1987, Springer,

F A Cowell: Measurement of Inequality, 2000, in A B Atkinson / F Bourguignon (Eds): Handbook of Income Distribution, Amsterdam,

F A Cowell: Measuring Inequality, 1995 Prentice Hall/Harvester Wheatshef.

See Also

plot.Lc, Lc.mehran, plot.theorLc

Examples

```
## Load and attach income (and metadata) set from Ilocos, Philippines
data(Ilocos)
attach(Ilocos)
## extract and rescale income for the provinces "Pangasinan" und "La Union"
income.p <- income[province=="Pangasinan"]/10000
income.u <- income[province=="La Union"]/10000

## compute the Lorenz curves
Lc.p <- Lc(income.p)
Lc.u <- Lc(income.u)
## it can be seen the the inequality in La Union is higher than in
## Pangasinan because the respective Lorenz curve takes smaller values.
plot(Lc.p)
```

```
lines(Lc.u, col=2)
## the picture becomes even clearer with generalized Lorenz curves
plot(Lc.p, general=TRUE)
lines(Lc.u, general=TRUE, col=2)
## inequality measures emphasize these results, e.g. Atkinson's measure
ineq(income.p, type="Atkinson")
ineq(income.u, type="Atkinson")
## or Theil's entropy measure
ineq(income.p, type="Theil", parameter=0)
ineq(income.u, type="Theil", parameter=0)



# income distribution of the USA in 1968 (in 10 classes)
# x vector of class means, n vector of class frequencies
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
n <- c(482, 825, 722, 690, 661, 760, 745, 2140, 1911, 1024)

# compute minimal Lorenz curve (= no inequality in each group)
Lc.min <- Lc(x, n=n)
# compute maximal Lorenz curve (limits of Mehran)
Lc.max <- Lc.mehran(x,n)
# plot both Lorenz curves in one plot
plot(Lc.min)
lines(Lc.max, col=4)

# add the theoretic Lorenz curve of a Lognormal-distribution with variance 0.78
lines(Lc.lognorm, parameter=0.78)
# add the theoretic Lorenz curve of a Dagum-distribution
lines(Lc.dagum, parameter=c(3.4,2.6))
```

`Lc.mehran` *Mehran Bounds For Lorenz Curves*

Description

Computes the Mehran bounds for a Lorenz curve of grouped data

Usage

```
Lc.mehran(x,n)
```

Arguments

`x`	vector of class means.
`n`	vector of class frequencies.

Value

An object of class `"Lc"`, but containing only `p` and `L`.

References

F Mehran: Bounds on the Gini Index Based on Observed Points of the Lorenz Curve, 1975, JASA 70, 64-66.

See Also

Lc, plot.Lc, plot.theorLc

Examples

```
# income distribution of the USA in 1968 (in 10 classes)
# x vector of class means, n vector of class frequencies
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
n <- c(482, 825, 722, 690, 661, 760, 745, 2140, 1911, 1024)

# compute minimal Lorenz curve (= no inequality in each group)
Lc.min <- Lc(x, n=n)
# compute maximal Lorenz curve (limits of Mehran)
Lc.max <- Lc.mehran(x,n)
# plot both Lorenz curves in one plot
plot(Lc.min)
lines(Lc.max, col=4)

# add the theoretic Lorenz curve of a Lognormal-distribution with variance 0.78
lines(Lc.lognorm, parameter=0.78)
# add the theoretic Lorenz curve of a Dagum-distribution
lines(Lc.dagum, parameter=c(3.4,2.6))
```

---

major *Majorization*

Description

tests whether a vector x majorizes another vector y

Usage

```
major(x,y)
```

Arguments

x, y	vectors containing non-negative elements (with same length and same mean)

Details

even if x and y are comparable (i.e. have same length and same mean) it is possible that neither x majorizes y nor y majorizes x.

Value

logical. TRUE if x majorizes y (x >=(M) y), FALSE if not.

References

Marshall / Olkin: Inequalities: Theory of Majorization and Its Applications, New York 1979 (Academic Press)

See Also

Lc

Examples

```
# generate vectors (of incomes)
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
y <- c(841, 2063, 2445, 3438, 4437, 5401, 6392, 8304, 11304, 21961)
# test whether x majorizes y (TRUE, because y is result of
# Pigou-Dalton-transfers)
major(x,y)
```

Pen *Pen's Parade*

Description

plots Pen's Parade of a vector x

Usage

```
Pen(x, n = rep(1, length(x)), group = NULL,
  scaled = TRUE, abline = TRUE, add = FALSE, segments = NULL,
  main = "Pen's Parade", ylab = NULL, xlab = NULL,
  col = NULL, lwd = NULL, las = 1, fill = NULL, ...)
```

Arguments

x	a vector containing non-negative elements.
n	a vector of frequencies or weights, must be same length as x.
group	a factor coding different groups, must be same length as x. See also details.
scaled	logical. Should Pen's parade be divided by mean(x)?
abline	logical. Should a horizontal line for the mean be drawn?
add	logical. Should the plot be added to an existing plot?
segments	logical. Should histogram-like segments be drawn?
col	a (vector of) color(s) for drawing the curve.

`fill`	a (vector of) color(s) for filling the area under the curve.
`xlab`, `ylab`	axis labels. Suitable defaults depending on `scaled` and `n` are chosen.
`main`, `lwd`, `las`, `...`	further high-level `plot` parameters.

Details

Pen's Parade is basically the inverse distribution function (standardized by `mean(x)`).

`Pen` allows for fine control of the layout—the graphical parameters `col` and `fill` can be vectorized if histogram-like segments are drawn (`segments = TRUE`)—but implements several heuristics in choosing its default plotting parameters. If a grouping factor `group` is given, the default is to draw segments with a grey-shaded filling. If no fill color is used, the default is to draw a thick blue curve. But as all of these are just defaults, they can of course easily be changed. See also the examples.

References

F A Cowell: Measurement of Inequality, 2000, in A B Atkinson / F Bourguignon (Eds): Handbook of Income Distribution, Amsterdam,

F A Cowell: Measuring Inequality, 1995 Prentice Hall/Harvester Wheatshef,

J Pen: Income Distribution, 1971, Harmondsworth: Allen Lane.

See Also

`Lc`, `plot.Lc`

Examples

```
# load and attach Philippine income data
data(Ilocos)
attach(Ilocos)
# plot Pen's Parade of income
Pen(income)
Pen(income, fill = hsv(0.1, 0.3, 1))

# income distribution of the USA in 1968 (in 10 classes)
# x vector of class means, n vector of class frequencies
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
n <- c(482, 825, 722, 690, 661, 760, 745, 2140, 1911, 1024)
Pen(x, n = n)
# create artificial grouping variable
myfac <- factor(c(1, 1, 1, 2, 2, 2, 3, 3, 3, 3))
Pen(x, n = n, group = myfac)
```

plot.Lc *Plot Lorenz Curve*

Description

plotting method for objects of class "Lc" (Lorenz curves)

Usage

```
## S3 method for class 'Lc'
plot(x, general=FALSE, lwd=2, xlab="p", ylab="L(p)",
    main="Lorenz curve", las=1, ...)
```

Arguments

x	an object of class "Lc"
general	logical. If TRUE the generalized Lorenz curve will be plotted
lwd, xlab, ylab, main, las, ...	high-level plot function parameters.

References

B C Arnold: Majorization and the Lorenz Order: A Brief Introduction, 1987, Springer,

F A Cowell: Measurement of Inequality, 2000, in A B Atkinson / F Bourguignon (Eds): Handbook of Income Distribution, Amsterdam,

F A Cowell: Measuring Inequality, 1995 Prentice Hall/Harvester Wheatshef.

See Also

Lc, Lc.mehran, plot.theorLc

Examples

```
## Load and attach income (and metadata) set from Ilocos, Philippines
data(Ilocos)
attach(Ilocos)
## extract and rescale income for the provinces "Pangasinan" und "La Union"
income.p <- income[province=="Pangasinan"]/10000
income.u <- income[province=="La Union"]/10000
## compute the Lorenz curves
Lc.p <- Lc(income.p)
Lc.u <- Lc(income.u)
## plot both Lorenz curves
plot(Lc.p)
lines(Lc.u, col=2)
```

plot.theorLc *Plot Theoretical Lorenz Curves*

Description

Plotting method for objects of class "theorLc" (theoretical Lorenz curves)

Usage

```
## S3 method for class 'theorLc'
plot(x, parameter=NULL, xlab="p", ylab="L(p)", lwd=2, las=1, ...)
```

Arguments

x	an object of class "theorLc"
parameter	vector containing parameters of the distributions. If x was generated by the function theorLc the parameters are already fixed and have to be set to NULL.
xlab, ylab, lwd, las, ...	high-level plot function parameters.

References

C Dagum: Income Distribution Models, 1983, in: Johnson / Kotz (Eds): Encyclopedia of Statistical Sciences Vol.4, 27-34.

J B McDonald: Some generalized functions for the size distribution of income, 1984, Econometrica 52, 647-664.

See Also

Lc, plot.Lc

Examples

```
# income distribution of the USA in 1968 (in 10 classes)
# x vector of class means, n vector of class frequencies
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
n <- c(482, 825, 722, 690, 661, 760, 745, 2140, 1911, 1024)

# compute minimal Lorenz curve (= no inequality in each group)
Lc.min <- Lc(x, n=n)
# compute maximal Lorenz curve (limits of Mehran)
Lc.max <- Lc.mehran(x,n)
# plot both Lorenz curves in one plot
plot(Lc.min)
lines(Lc.max, col=4)

# add the theoretic Lorenz curve of a Lognormal-distribution with variance 0.78
lines(Lc.lognorm, parameter=0.78)
```

```
# add the theoretic Lorenz curve of a Dagum-distribution
lines(Lc.dagum, parameter=c(3.4,2.6))
```

---

pov *Poverty Measures*

Description

computes the poverty of an (income) vector according to the specified poverty measure

Usage

```
pov(x, k, parameter = NULL, type = c("Watts", "Sen", "SST", "Foster"), na.rm = TRUE)

Watts(x, k, na.rm = TRUE)
Sen(x, k, na.rm = TRUE)
SST(x, k, na.rm = TRUE)
Foster(x, k, parameter = 1, na.rm = TRUE)
```

Arguments

x	a vector containing at least non-negative elements
k	a constant giving the absolute poverty line
parameter	parameter of the poverty measure (if set to NULL the default parameter of the respective measure is used)
type	character string giving the measure used to compute poverty coefficient must be one of the strings in the default argument. Defaults to "Watts".
na.rm	logical. Should missing values (NAs) be removed prior to computations? If set to FALSE the computations yield NA.

Details

pov is just a wrapper for the poverty measures of Watts, Sen, SST, and Foster (Foster / Greer / Thorbecke). If parameter is set to NULL the default from the respective function is used.

Foster gives for parameter 1 the headcount ratio and for parameter 2 the poverty gap ratio.

Value

the value of the poverty measure

References

Foster, J. E. (1984). On Economic Poverty: A Survey of Aggregate Measures. *Advances in Econometrics*, **3**, 215–251.

Shorroks, A. F. (1995). Revisiting the Sen Poverty Index. *Econometrica*, **63**(5), 1225–1230.

Zheng, B. (1997). Aggregate Poverty Measures. *Journal of Economic Surveys*, **11**, 123–162.

See Also

ineq, conc

Examples

```
# generate vectors (of incomes)
x <- c(541, 1463, 2445, 3438, 4437, 5401, 6392, 8304, 11904, 22261)
y <- c(841, 2063, 2445, 3438, 4437, 5401, 6392, 8304, 11304, 21961)
# compute Watts index with poverty line 2000
pov(x, 2000)
pov(y, 2000)
# compute headcount ratio with poverty line 2000
pov(x, 2000, parameter=1, type="Foster")
pov(y, 2000, parameter=1, type="Foster")
```

---

theorLc *Theoretical Lorenz Curves*

Description

Theoretical Lorenz curves of income distributions

Usage

```
theorLc(type=c("Singh-Maddala","Dagum","lognorm","Pareto","exponential"), parameter=0)

Lc.dagum(p, parameter=c(2,2))
Lc.singh(p, parameter=c(2,2))
Lc.pareto(p, parameter=2)
Lc.lognorm(p, parameter=1)
Lc.exp(p)
```

Arguments

type	character string giving the income distribution. Must be one of the strings in the default argument (the first character is sufficient). Defaults to "Singh-Maddala".
parameter	vector containing parameter(s) of the distributions.
p	vector with elements from [0,1].

Details

Lc.dagum, Lc.singh, Lc.pareto, Lc.lognorm, Lc.exp are theoretical Lorenz curves of income distributions. They are functions of class "theorLc" with plot- and a lines- method, so that they can be added into an existing Lorenz curve plot.

theorLc returns a function of class "theorLc", that is a one of the above theoretical Lorenz curves with fixed parameters.

`Lc.dagum` is the Lorenz curve of the Dagum distribution (2 parameters), `Lc.singh` the one of the Singh-Maddala distribution (2 parameters), `Lc.pareto` the one of the Pareto distribution (1 parameter), `Lc.lognorm` the one of the Lognormal distribution (1 parameter) and `Lc.exp` the Lorenz curve of the exponential distribution (no parameter).

Value

A function of class `"theorLc"` or its value at `p` respectively.

References

C Dagum: Income Distribution Models, 1983, in: Johnson / Kotz (Eds): Encyclopedia of Statistical Sciences Vol.4, 27-34.

J B McDonald: Some generalized functions for the size distribution of income, 1984, Econometrica 52, 647-664.

See Also

`Lc`, `plot.Lc`, `plot.theorLc`

Examples

```
## Load and attach income (and metadata) set from Ilocos, Philippines
data(Ilocos)
attach(Ilocos)
## extract income for the province "Pangasinan"
income.p <- income[province=="Pangasinan"]

## plot empirical Lorenz curve and add theoretical Lorenz curve of
## a lognormal distribution with an estimate of the standard
## deviation parameter
Lc.p <- Lc(income.p)
plot(Lc.p)
lines(Lc.lognorm, parameter=sd(log(income.p)), col=4)



# vector of percentages
p <- (1:10)*0.1
# compute values of theoretic Lorenz curve of a Dagum-distribution
Lc.dagum(p, parameter=c(3.4,2.6))
# or
mydagum <- theorLc(type="Dagum", parameter=c(3.4,2.6))
mydagum(p)
```

Index