

# Package: ineqJD (via r-universe)

October 17, 2024

**Type** Package

**Title** Inequality Joint Decomposition

**Version** 1.0

**Date** 2019-09-03

**Author** Alberto Arcagni [aut, cre], Igor Valli [aut], Michele Zenga [ctb]

**Maintainer** Alberto Arcagni <alberto.arcagni@uniroma1.it>

**Description** Computes and decomposes Gini, Bonferroni and Zenga 2007 point and synthetic concentration indexes. Decompositions are intended: by sources, by subpopulations and by sources and subpopulations jointly. References, Zenga M. M.(2007) <doi:10.1400/209575> Zenga M. (2015) <doi:10.1400/246627> Zenga M., Valli I. (2017) <doi:10.26350/999999\_000005> Zenga M., Valli I. (2018) <doi:10.26350/999999\_000011>.

**License** GPL (>= 2)

**NeedsCompilation** no

**Date/Publication** 2019-09-20 07:30:02 UTC

**Repository** <https://albertoarcagni.r-universe.dev>

**RemoteUrl** <https://github.com/cran/ineqJD>

**RemoteRef** HEAD

**RemoteSha** 925ba5e3aafe8c8f4773b14c0bf7dfa29fed601

## Contents

bonferroni . . . . .	2
dataProcessing . . . . .	3
gini . . . . .	4
inequalityCurves . . . . .	5
plot.inequality_curves . . . . .	7
summary.decomposition . . . . .	9
zenga . . . . .	11

<b>Index</b>	<b>13</b>
--------------	-----------

---

`bonferroni`*Point and synthetic Bonferroni indexes*

---

**Description**

Computes the decomposition of the Bonferroni point inequality indexes of a statistical variable  $Y$  described in the object  $x$ .

**Usage**

```
bonferroni(x)
```

**Arguments**

$x$  An object of class "dataProcessed".  $x$  is usually the result of `dataProcessing` function. More details are given in the "Details" section and [dataProcessing](#) help page.

**Details**

`bonferroni` computes the decomposition of the Bonferroni point inequality indexes from the object  $x$  of class "dataProcessed".  $x$  is usually the result of `dataProcessing` function.

**Value**

`index` String denoting computed index.

`decomposition` Array containing the decompositions. The dimensions of `decomposition` are  $c(g, g, r, s)$  where  $g$  is the number of groups,  $r$  the number of different values of  $Y$  and  $s$  the number of sources.

$x$  Object of class `dataProcessed` passed as input.

**Author(s)**

Alberto Arcagni, Igor Valli.

**References**

Zenga M., Valli I. (2017). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Bonferroni Inequality Measures. *Statistics and Applications*, XV (2), pp. 83-120.

**See Also**

[gini](#) and [zenga](#) for other inequality indexes and [dataProcessing](#) for the class "dataProcessed".

**Examples**

```
G <- c(1, 2, 3, 1, 2, 3, 1, 1, 2, 3, 3, 3) # vector denoting group membership
X1 <- c(0, 0, 0, 500, 700, 300, 750, 1000, 500, 500, 500, 1000) # vector of the first source
X2 <- c(0, 0, 0, 500, 300, 700, 750, 500, 700, 700, 1000, 600) # vector of the second source
data <- data.frame(G, X1, X2) # no sample weights are considered
x <- dataProcessing( # data preparation
  units = data[, c('X1', 'X2')],
  groups = data[, 'G'],
)

decomposition <- bonferroni(x)
decomposition
```

dataProcessing

*Data Processing***Description**

Convert raw data to frequency distribution framework and returns cumulative sums.

**Usage**

```
dataProcessing(
  units,
  groups = rep("G1", nrow(as.matrix(units))),
  weights = rep(1, nrow(as.matrix(units)))
)
```

**Arguments**

units	Numeric vector of length $n$ or matrix of dimension $c(n, s)$ containing $s$ sources referred to $n$ statistical units
groups	Vector of length $n$ of group membership. If empty only one group is considered, otherwise the number of groups $g$ is defined by the number of distinct values or levels in this vector.
weights	Vector of $l$ weights of length $n$ . If empty uniform weights are considered.

**Details**

`dataProcessing` convert raw data in the frequency distribution framework with  $r$  distinct values of  $Y$ . In this way repeated values are removed as well as ordering issues. Moreover cumulative frequencies and cumulative sources values are evaluated in order to prepare data for inequality decompositions.

**Value**

yh	Vector of length $r$ of distinct values of $Y$ .
Ph1	Matrix of absolute cumulative frequencies of dimension $c(r, g)$ .
Qh1k	Array of cumulative sum of sources of dimension $c(r, g, s)$ .

**Author(s)**

Alberto Arcagni, Igor Valli

**References**

Zenga M.M. (2007). Inequality curve and inequality index based on the ratios between lower and upper means. *Statistica and Applicazioni*, V(1), 3-27

**Examples**

```
G <- c(1, 2, 3, 1, 2, 3, 1, 1, 2, 3, 3, 3) # vector denoting group membership
X1 <- c(0, 0, 0, 500, 700, 300, 750, 1000, 500, 500, 500, 1000) # vector of the first source
X2 <- c(0, 0, 0, 500, 300, 700, 750, 500, 700, 700, 1000,600) # vector of the second source
data <- data.frame(G, X1, X2) # no sample weights are considered

x <- dataProcessing(
  units = data[, c('X1', 'X2')],
  groups = data[, 'G'],
)
x
```

---

gini

*Point and synthetic Gini indexes*

---

**Description**

Computes point and synthetic Gini indexes on a variable  $Y$ .

**Usage**

```
gini(x)
```

**Arguments**

$x$  List containing: 'yh', the vector of unique values of the variable  $Y$  whose Bonferroni index is computed; 'Ph1', the matrix of absolute cumulative frequencies; 'Qh1k', the matrix of cumulative sums of  $Y$  or its sources.  $x$  is usually the result of `dataProcessing` function. More details are given in the "Details" section and [dataProcessing](#) help page.

**Details**

`gini` compute point and synthetic Gini indexes on a variable  $y$ , e.g. income, on a statistical population that could be partitioned in  $g$  subpopulations and could be considered as sum of  $c$  sources, e.g. income sources.

**Value**

index               String denoting computed index.  
 decomposition    array containing the decompositions.  
 x                   object usually of class dataProcessed passed as input.

**Author(s)**

Alberto Arcagni, Igor Valli.

**References**

Zenga M., Valli I. (2018). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Gini Indexes. *Statistics and Applications*, XVI (1).

**Examples**

```
G <- c(1, 2, 3, 1, 2, 3, 1, 1, 2, 3, 3, 3) # vector denoting group membership
X1 <- c(0, 0, 0, 500, 700, 300, 750, 1000, 500, 500, 500, 1000) # vector of the first source
X2 <- c(0, 0, 0, 500, 300, 700, 750, 500, 700, 700, 1000,600) # vector of the second source
data <- data.frame(G, X1, X2) # no sample weights are considered
x <- dataProcessing( # data preparation
  units = data[, c('X1', 'X2')],
  groups = data[, 'G'],
)

decomposition <- gini(x)
decomposition
```

---

inequalityCurves        *Inequality curves evaluation*

---

**Description**

Generates step-functions (see [stepfun](#)) representing inequality curves or sources/subpopulations point contributions in the decomposition x generated by functions [gini](#), [bonferroni](#) or [zenga](#).

**Usage**

```
inequalityCurves(x, ...)
## S3 method for class 'decomposition'
inequalityCurves(x, l = 1:dim(x$decomposition)[2], k = 1:dim(x$decomposition)[4], ...)
```

**Arguments**

x	An object of class <code>decomposition</code> output of functions <a href="#">gini</a> , <a href="#">bonferroni</a> , <a href="#">zenga</a> .
l	Vector of selected subpopulations. If only one subpopulation is selected the resulting step function provides its point contributions. More than one subpopulation can be selected and contributions are cumulated. If empty all subpopulations are considered and contributions are marginalized by subpopulations.
k	Vector of selected sources. If only one source is selected the resulting step function provides its point contributions. More than one source can be selected and contributions are cumulated. If empty all sources are considered and contributions are marginalized by sources.
...	Potentially further arguments (required by the generic).

**Details**

By default generates step functions representing inequality curves of the whole population. If arguments `l` and/or `k` are defined, results are step functions representing point contributions of the selected subpopulations and/or sources.

The class of the result is `"inequality_curves"` that is associated to the method [plot.inequality\\_curves](#) for graphical representations. Such derived class inherits the features of the `"stepfun"` class.

**Value**

An object of class `"inequality_curves"` inheriting the features of class `"stepfun"` with the following additional attributes:

<code>index</code>	String denoting computed index.
<code>min, max</code>	The range of values assumed by the function.
<code>groups</code>	Vector of names of the subpopulations partitioning the whole population .
<code>sources</code>	Vector of names of the all the sources that sum to the total variable $Y$ .
<code>selected_groups</code>	Vector of names of the selected subpopulations to evaluate the point contributions.
<code>selected_sources</code>	Vector of names of the selected sources to evaluate the point contributions.

**Author(s)**

Alberto Arcagni, Igor Valli

**References**

- Zenga M. M.(2007). Inequality Curve and Inequality Index based on the Ratios between lower and upper Means . *Statistica and Applicazioni*, V (1), 3-27.
- Zenga M. (2015) Joint decomposition by subpopulations and sources of the point and synthetic Zenga(2007) Index  $I(Y)$ . *Statistica and Applicazioni*, XIII (2), pp.163-195.
- Zenga M., Valli I. (2017). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Bonferroni Inequality Measures. *Statistics and Applications*, XV (2), pp. 83-120.

Zenga M., Valli I. (2018). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Gini Indexes. *Statistics and Applications*, XVI (1).

### See Also

See [gini](#), [bonferroni](#), [zenga](#) to obtain objects of class `decomposition` and see [plot.inequality\\_curves](#) for the graphical representation.

### Examples

```
G <- c(1, 2, 3, 1, 2, 3, 1, 1, 2, 3, 3, 3) # vector denoting group membership
X1 <- c(0, 0, 0, 500, 700, 300, 750, 1000, 500, 500, 500, 1000) # vector of the first source
X2 <- c(0, 0, 0, 500, 300, 700, 750, 500, 700, 700, 1000, 600) # vector of the second source
data <- data.frame(G, X1, X2) # no sample weights are considered
x <- dataProcessing( # data preparation
  units = data[, c('X1', 'X2')],
  groups = data[, 'G'],
)
decomposition <- zenga(x)

ic <- inequalityCurves(decomposition)
ic
contrib1 <- inequalityCurves(decomposition, l = 1)
contrib1
contrib12 <- inequalityCurves(decomposition, l = 1:2)
contrib12
```

---

plot.inequality\_curves

*Plot inequality curves*

---

### Description

Method of the generic [plot](#) for objects generated by the function [inequalityCurves](#).

### Usage

```
## S3 method for class 'inequality_curves'
plot(
  x,
  pch = 16,
  from = 0,
  to = 1,
  xlim = NULL,
  ylim = NULL,
  xaxs = "i",
  yaxs = "i",
  xlab = "p",
  ylab = NULL,
```

```

main = attributes(x)$index,
sub = paste0(
  "grp: ",
  paste(attributes(x)$groups, collapse = ", "),
  "; src: ",
  paste(attributes(x)$sources, collapse = ", ")
),
...
)

```

### Arguments

x	Object of class "decomposition".
pch	A vector of plotting characters or symbols: see <a href="#">points</a> .
from, to	The range over which the function will be plotted.
xlim	The x limits (x1, x2) of the plot. The default value, NULL, indicates that the range of the function should be used.
ylim	the y limits of the plot.
xaxs	The style of axis interval calculation to be used for the x-axis. See <a href="#">par</a> for details.
yaxs	The style of axis interval calculation to be used for the y-axis. See <a href="#">xaxs</a> above.
xlab	a title for the x axis: see <a href="#">title</a> .
ylab	a title for the y axis: see <a href="#">title</a> .
main	an overall title for the plot: see <a href="#">title</a> .
sub	a sub title for the plot: see <a href="#">title</a> .
...	Arguments to be passed to methods, such as <a href="#">graphical parameters</a> (see <a href="#">par</a> ). Many methods will accept the following arguments:

### Details

This method is a convenience wrapper for plotting inequality curves. Default values of the plot are modified in order to plot inequality curves in the unitary square. Moreover, the default value of the argument sub shows in the plot if the curve represents the whole inequality or a contribution.

### Value

The same output of the function [plot.stepfun](#), a list with two components:

t	abscissa (x) values, including the two outermost ones.
y	y values 'in between' the t[[]].

### Author(s)

Alberto Arcagni, Igor Valli



## References

Zenga M. M.(2007). Inequality Curve and Inequality Index based on the Ratios between lower and upper Means . Statistica and Applicazioni, V (1), 3-27.

Zenga M. (2015) Joint decomposition by subpopulations and sources of the point and synthetic Zenga(2007) Index I(Y). Statistica and Applicazioni, XIII (2), pp.163-195.

Zenga M., Valli I. (2017). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Bonferroni Inequality Measures. Statistics and Applications, XV (2), pp. 83-120.

Zenga M., Valli I. (2018). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Gini Indexes. Statistics and Applications, XVI (1).

## See Also

[plot](#), [graphical parameters](#), [par](#), [plot.stepfun](#), [inequalityCurves](#)

## Examples

```
G <- c(1, 2, 3, 1, 2, 3, 1, 1, 2, 3, 3, 3) # vector denoting group membership
X1 <- c(0, 0, 0, 500, 700, 300, 750, 1000, 500, 500, 500, 1000) # vector of the first source
X2 <- c(0, 0, 0, 500, 300, 700, 750, 500, 700, 700, 1000,600) # vector of the second source
data <- data.frame(G, X1, X2) # no sample weights are considered
x <- dataProcessing( # data preparation
  units = data[, c('X1', 'X2')],
  groups = data[, 'G'],
)
decomposition <- zenga(x)
ic <- inequalityCurves(decomposition)
contrib1 <- inequalityCurves(decomposition, l = 1)
contrib12 <- inequalityCurves(decomposition, l = 1:2)

plot(ic)
plot(contrib1, add = TRUE)
plot(contrib12, add = TRUE)
text(0.1, 1/6+0:2/3, labels = c("G1", "G2", "G3"))
```

---

summary.decomposition *Summarizing inequality decomposition*

---

## Description

summary method for class "decomposition".

## Usage

```
## S3 method for class 'decomposition'
summary(object, ...)
## S3 method for class 'summary.decomposition'
print(x, ...)
```

**Arguments**

object	An object of class "decomposition", usually, as result of a call to <a href="#">gini</a> , <a href="#">bonferroni</a> and <a href="#">zenga</a> .
x	rtrtrt
...	further arguments passed to or from other methods.

**Details**

summary.decomposition method use

**Value**

index	String denoting computed index.
joint	Array of joint decompositions by sources and subpopulations.
pairs	Matrix of decompositions by subpopulations.
within	Vector of within part to the overall inequality. It denotes the part of the overall inequality derived from the inequality inside each subpopulation.
between	Vector of between part to the overall inequality. It denotes the part of the overall inequality derived from the comparison between subpopulations.
groups	Vector of subpopulations contribution to the overall inequality.
groups_sources	Matrix of subpopulations contributions for each source to the overall inequality.
sources	Vector of sources contribution to the overall inequality.
synthetic	Scalar denoting the value of the synthetic index.

**Author(s)**

Alberto Arcagni, Igor Valli.

**References**

Zenga M. M.(2007). Inequality Curve and Inequality Index based on the Ratios between lower and upper Means . *Statistica and Applicazioni*, V (1), 3-27.

Zenga M. (2015) Joint decomposition by subpopulations and sources of the point and synthetic Zenga(2007) Index I(Y). *Statistica and Applicazioni*, XIII (2), pp.163-195.

Zenga M., Valli I. (2017). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Bonferroni Inequality Measures. *Statistics and Applications*, XV (2), pp. 83-120.

Zenga M., Valli I. (2018). Joint decomposition by Subpopulations and Sources of the Point and Synthetic Gini Indexes. *Statistics and Applications*, XVI (1).

**See Also**

[gini](#), [bonferroni](#), [zenga](#), [dataProcessing](#).

**Examples**

```
G <- c(1, 2, 3, 1, 2, 3, 1, 1, 2, 3, 3, 3) # vector denoting group membership
X1 <- c(0, 0, 0, 500, 700, 300, 750, 1000, 500, 500, 500, 1000) # vector of the first source
X2 <- c(0, 0, 0, 500, 300, 700, 750, 500, 700, 700, 1000, 600) # vector of the second source
data <- data.frame(G, X1, X2) # no sample weights are considered
x <- dataProcessing( # data preparation
  units = data[, c('X1', 'X2')],
  groups = data[, 'G'],
)
decomposition <- zenga(x)

summary(decomposition)
```

zenga

*Point and synthetic Zenga 2007 indexes***Description**

Computes point and synthetic Zenga 2007 indexes on a variable  $Y$ .

**Usage**

```
zenga(x)
```

**Arguments**

$x$  List containing: 'yh', the vector of unique values of the variable  $Y$  whose Bonferroni index is computed; 'Ph1', the matrix of absolute cumulative frequencies; 'Qh1k', the matrix of cumulative sums of  $y$  or its sources.  $x$  is usually the result of dataProcessing function. More details are given in the "Details" section and [dataProcessing](#) help page.

**Details**

zenga compute point and synthetic Zenga 2007 indexes on a variable  $y$ , e.g. income, on a statistical population that could be partitioned in  $g$  subpopulations and could be considered as sum of  $c$  sources, e.g. income sources.

**Value**

index String denoting computed index.  
 decomposition Array containing the decompositions.  
 $x$  Object usually of class dataProcessed passed as input.

**Author(s)**

Alberto Arcagni, Igor Valli

## References

Zenga M. M.(2007). Inequality Curve and Inequality Index based on the Ratios between lower and upper means . *Statistica and Applicazioni*, V (1), 3-27.

Zenga M. (2015) Joint decomposition by subpopulations and sources of the point and synthetic Zenga(2007) Index I(Y). *Statistica and Applicazioni*, XIII (2), pp.163-195.

## Examples

```
G <- c(1, 2, 3, 1, 2, 3, 1, 1, 2, 3, 3, 3) # vector denoting group membership
X1 <- c(0, 0, 0, 500, 700, 300, 750, 1000, 500, 500, 500, 1000) # vector of the first source
X2 <- c(0, 0, 0, 500, 300, 700, 750, 500, 700, 700, 1000,600) # vector of the second source
data <- data.frame(G, X1, X2) # no sample weights are considered
x <- dataProcessing( # data preparation
  units = data[, c('X1', 'X2')],
  groups = data[, 'G'],
)

decomposition <- zenga(x)
decomposition
```

# Index

- \* **Bonferroni concentration inequality point**
  - synthetic index decomposition**
  - bonferroni, 2
- \* **Frequency Distribution**
  - dataProcessing, 3
- \* **Gini concentration inequality point**
  - synthetic index decomposition**
  - gini, 4
- \* **Zenga concentration inequality point**
  - synthetic index decomposition**
  - zenga, 11

bonferroni, 2, 5–7, 10

dataProcessing, 2, 3, 4, 10, 11

gini, 2, 4, 5–7, 10

inequalityCurves, 5, 7, 9

par, 8, 9

plot, 7, 9

plot.inequality\_curves, 6, 7, 7

plot.stepfun, 8, 9

points, 8

print.dataProcessed (dataProcessing), 3

print.summary.decomposition  
(summary.decomposition), 9

stepfun, 5, 6

summary.decomposition, 9

title, 8

zenga, 2, 5–7, 10, 11