Package 'MLID'

July 21, 2025

Type Package

Title Multilevel Index of Dissimilarity

Version 1.0.1

Author Richard Harris [aut],
Dewi Owen [ctb]

Maintainer Richard Harris <rich.harris@bris.ac.uk>

Description Tools and functions to fit a multilevel index of dissimilarity.

Depends R (>= 3.3.0)

URL https://github.com/profrichharris/MLID

BugReports https://github.com/profrichharris/MLID/issues

License GPL-3

Encoding UTF-8

LazyData true

Imports nlme (>= 3.1.128), lme4 (>= 1.1.12), methods

RoxygenNote 6.0.1

Suggests raster (>= 2.5.8), sp (>= 1.2.3), knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2017-03-06 00:18:37

Contents

`aggdata` *Aggregated population counts by ethnic group*

Description

Ethnicity data for England and Wales from the 2011 Census. Each row in the data set gives population counts for Lower Level Super Output Area (LSOA). The census geography is hierarchical: LSOAs group into Middle Level Super Output Areas (MSOAs), which group into Local Authority Districts (LADs) and then into Government Regions (RGNs). These groupings are included in the data.

Usage

```
aggdata
```

Format

A `data.frame` with 19 columns:

- LSOA, the Census ID for the Output Area
- Persons, the residential population count for the OA
- columns 3-16, the number of people White British, Irish, of an other White ethnicity, of a mixed ethnicity, Indian, Pakistani, Bangladeshi, Chinese, of an other Asian ethnicity, Black African, Black Caribbean, of an other Black ethnicity, Arab or of an other ethnicity.
- columns 17-19, the ID codes for the higher-level geographies: MSOAs, LADs and RGNs

Source

Office for National Statistics; National Records of Scotland; Northern Ireland Statistics and Research Agency (2016): 2011 Census aggregate data. UK Data Service (Edition: June 2016). DOI: http://dx.doi.org/10.5257/census/aggregate-2011-1. This information is licensed under the terms of the Open Government Licence http://www.nationalarchives.gov.uk/doc/open-government-licence/version/3. The LSOA, MSOA, LAD and RGN codes are from http://bit.ly/2lGMdkE and are supplied under the Open Government Licence: Contains National Statistics data. Crown copyright and database right 2017.

See Also

ethnicities

`catplot` *Caterpillar plot*

Description

Draws a series of caterpillar plots, showing the residuals from the multilevel model at each level and the estimates of their confidence interval

Usage

```
catplot(confint, ann = TRUE, grid = FALSE)
```

Arguments

`confint`	an object containing the output from function `confint.index`
`ann`	default is TRUE. If set to false, suppresses the automatic annotation of residuals on the plots with a confidence interval that does not overlap with any other
`grid`	arrange the plots in a grid? (Default is TRUE)

Details

A caterpillar plot is a visual way of looking at the variance of the residuals at each level of a multilevel model. It can be used to see which places are contributing most to the Index of Dissimilarity net of the effects of other scales.

To aid the interpretability of the plots, the residuals are scaled by the standard error of the residuals from the OLS estimate of the index. Additionally, to avoid over-plotting only a maximum of 50 residuals are shown on each plot. These are the 10 highest and lowest ranked residuals and then a sample of 30 from the remaining residuals, chosen as the ones with values that differ most from the residuals that precede them by ranking. In this way, the plots aim to preserve the tails of the ranked distribution as well as the most important break points inbetween.

When `ann = TRUE` (the default) some outliers are labelled and the percentage of the total variance due to each level is included. These will not add up to 100 `catplot` is a wrapper to `plot.confintindex`

See Also

confint.index id

Examples

```
## Not run:
data(aggdata)
index <- id(aggdata, vars = c("Bangladeshi", "WhiteBrit"),
levels = c("MSOA","LAD","RGN"))
ci <- confint(index)
catplot(ci, grid = TRUE)
# Plots for all levels above the base level

## End(Not run)
```

---

checkerboard *Checkerboard*

Description

A demonstration of how the Multilevel Index of Dissimilarity measures spatial clustering as well as unevenness

Usage

```
checkerboard()
```

Details

A criticism of the standard Index of Dissimilarity (ID) is that it only measures one of the two principal dimensions of segregation - unevenness but not spatial clustering. Because of this, very different spatial patterns of segregation can generate the same ID score but the ID is unable to distinguish between them.

In contrast, the multilevel index can detect the differences because different patterns (scales) of segregation change the percentage of the variance due to each level. The demonstation illustrates this using the classic example of a checkerboard. The examples show how the percentage of the total variance (labelled Pvariance) moves up the hierarchy with the increase in spatial clustering at greater geographical cases. However, the ID is always the same.

The 'stray' cell in examples 2-4 is to allow the model to be fitted. With it the model correctly identifies that some of the variation remains at the base level)

See Also

id

confint.index *Confidence intervals for the multilevel index*

Description

Calculates the confidence intervals for the residuals of the multilevel index at each level. These can then be visualised in a caterpillar plot.

Usage

```
## S3 method for class 'index'
confint(object, parm, level = 0.95, ...)
```

Arguments

object	an object of class index: a multilevel index created using function id
parm	NA
level	the confidence level required
...	other arguments

Details

confint.index is a wrapper to lme4::ranef(mlm, condVar = TRUE) and is used to calculate the confidence intervals for the locations and regions at each of the higher levels of the model. In this way, places with an usually high (or low) share of population group Y with respect to population group X can be identified, net of the effects of other levels of the model. The width of the confidence interval is adjusted for a test of difference between two means (see Statistical Rules of Thumb by Gerald van Belle, 2011, eq 2.18). A 95 per cent confidence interval, for example, extends to 1.39 times the standard error around the mean and not 1.96.

Value

an object of class confint, a list of length equal to the number of levels in the index where each part of the list is a data frame giving the confidence interval for the location

See Also

catplot id ranef

Examples

```
## Not run:
data(aggdata)
index <- id(aggdata, vars = c("Bangladeshi", "WhiteBrit"),
levels = c("MSOA","LAD","RGN"))
ci <- confint(index)
catplot(ci)

## End(Not run)
```

effect *Consider the effect of particular places upon the ID*

Description

Evaluates the effect on the index of the named `places` under three different scenarios.

Usage

```
effect(object, places)
```

Arguments

`object`	an object of class `index`: a multilevel index created using function `id`
`places`	a character vector containing the names of the places in any of the higher-level geographies for which the evaluation will be made

Details

The three different scenarios considered are:

1. if the effects (the estimated residuals from the multilevel model) are set to zero for the named higher-level `places`;
2. if the shares of the two population groups are equal everywhere except within the named `places`; and
3. if all but the neighbourhoods within the named `places` are omitted from the data and the index then recalculated using only those that remain.

The evaluation also includes:

- the impact of the chosen places upon the overall ID, where a value over 100 indicates a group of neighbourhoods with a disproportionately high (in the same way that `impacts` calculates it)
- an R-squared value. This is the proportion of the total variation in (Y - X) that is due to the chosen places, where (Y - X) are the differences between the share of population group Y and the share of population group X that are observed in each neighbourhood.

See `vignette("MLID")` for further details

Value

an object, primarily a list containing the evaluated values

See Also

`id`

Harris R (2017) Fitting a multilevel index of segregation in R: using the MLID package http://rpubs.com/profrichharris/MLID

Examples

```
## Not run:
data(aggdata)
index <- id(aggdata, vars = c("Bangladeshi", "WhiteBrit"),
levels = c("MSOA","LAD","RGN"))
ci <- confint(index)
catplot(ci)
# Note Tower Hamlets and Newham. Obtain the predictions for them:
effect(index, "Tower Hamlets")
effect(index, "Newham")
effect(index, c("Tower Hamlets","Newham"))

## End(Not run)
```

---

ethnicities *Population counts by ethnic group*

Description

Ethnicity data for England and Wales from the 2011 Census. Each row in the data set gives population counts for a small area census Output Area (OA). The census geography is hierarchical: OAs group into Lower Level Super Output Areas (LSOAs), which group into Middle Level Super Output Areas (MSOAs), which group into Local Authority Districts (LADs) and then into Government Regions (RGNs). These groupings are included in the data.

Usage

```
ethnicities
```

Format

A `data.frame` with 20 columns:

- OA, the Census ID for the Output Area
- Persons, the residential population count for the OA
- columns 3-16, the number of people White British, Irish, of an other White ethnicity, of a mixed ethnicity, Indian, Pakistani, Bangladeshi, Chinese, of an other Asian ethnicity, Black African, Black Caribbean, of an other Black ethnicity, Arab or of an other ethnicity.
- columns 17-20, the ID codes for the higher-level geographies: LSOAs, MSOAs, LADs and RGNs

Source

Office for National Statistics; National Records of Scotland; Northern Ireland Statistics and Research Agency (2016): 2011 Census aggregate data. UK Data Service (Edition: June 2016). DOI: http://dx.doi.org/10.5257/census/aggregate-2011-1. This information is licensed under the terms of the Open Government Licence http://www.nationalarchives.gov.uk/doc/

open-government-licence/version/3. The LSOA, MSOA, LAD and RGN codes are from http://bit.ly/2lGMdkE and are supplied under the Open Government Licence: Contains National Statistics data. Crown copyright and database right 2017.

See Also

aggdata

`head.impacts` *Return the highest impact scores for each higher level area*

Description

Returns the first parts of the set of impact calculations, ordered by the impact score

Usage

```
## S3 method for class 'impacts'
head(x, n = 5, ...)
```

Arguments

`x`	an object of class `impacts` generated by the function `impacts`
`n`	a single integer giving the number of rows to show. Defaults to 5.
`...`	other arguments

See Also

impacts

`holdback` *Holdback scores*

Description

Calculates the holdback scores for a multilevel index of dissimilarity

Usage

```
holdback(mlm)
```

Arguments

`mlm`	an object of class `lmerMod` generated by the `lme4` package

Details

For the index of dissimilarity (ID), the residuals are the differences between the share of the Y population and the share of the X population per neighbourhood. For the multilevel index, the residuals are estimated at and partitioned between each level of the model. The holdback scores take each level in the model in turn and set the residuals (the effects) at that level to zero, then recalculating the ID on that basis and recording the percentage change in the original value that occurs. The holdback scores are calculated automatically as part of the function `id` and can be viewed through `print(index)`, where `index` is the object returned by the function, or as `attr(index, "holdback")`.

Value

a numeric vector containing the holdback scores

See Also

`id` `print.index`

`id` *(Multilevel) index of dissimilarity*

Description

Returns either the standard index of dissimilarity (ID) or its multilevel equivalent

Usage

```
id(data, vars, levels = NA, expected = FALSE, nsims = 100, omit = NULL)
```

Arguments

`data`	a data frame with `ncol(data) >= 2`. Each row of the data represents a neighbourhood or some other areal unit for which counts of population have been made.
`vars`	a character or numeric vector of length 2 or 3 giving either the names or columns positions of the variables in `data` in the following order: 1. the number of population group Y in each neighbourhood 2. the number of population group X in each neighbourhood 3. (optional) The total population in each neighbourhood
`levels`	a character or numeric vector of minimum length 1 identifying either the names or columns positions of the variables in `data` that record to which higher-level grouping each lower-lower level unit belongs. If `levels = NA`, the default, then only the standard index of dissimilarity is calculated.
`expected`	a logical scaler. Should the expected value of the ID under randomisation be calculated? Requires a measure of the total population in each neighbourhood. If omitted from `vars` that total will be calculated as `sum(X + Y)`.

nsims	a vector, the number of random draws to be used for calculating the expected value. Default is 100.
omit	(optional) a character vector containing the names of places to search for in the data and to omit from the calculations

Details

If Y is the number of population group Y living in each neighbourhood and X is the number of population group X then id measures how unevenly distributed are the two groups relative to one another and is a measure of segregation. In addition, for geographically hierarchichal data, scale effects may be explored to examine the scale of geographical clustering.

The method works by treating the calculation of the ID as a regression problem: if Y is recalculated as the share per neighbourhood of the total count of population group Y (i.e. `Y <- Y / sum(Y)`) and X is recalculated in the same way for X, then fitting `ols <- lm(Y ~ 0, offset = X)` generates a set of residuals, `e <- residuals(ols)` where each residual is the difference in the share of Y and the share of X per neighbourhood, and the sum of the absolute of those residuals can be used to obtain the id: `id <- 0.5 * sum(abs(e))`.

The advantage of calculating the ID in this way is that it can be extended to consider geographic hierarchies, where neighbourhoods at the base level can be grouped into larger regions at the next level, and so forth. Then, for the multilevel index, the residuals are estimated at and partitioned between each level of the model *net* of the other levels, allowing scale effects to be explored.

`print(index)` displays the ID value, the expected value of the ID under randomisation (NA if not calculated), and, for a multilevel model, the percentage share of the total variance due to each level (a measure of the geographical scale of segregation: see the examples given by `checkerboard`) and the holdback scores - see `holdback`

Value

an object of class `index`. This is a value between zero and one where 0 implies no segreation, and 1 means 'complete segregation' - wherever group Y is located, X is not (and vice versa). If `expected = TRUE` the expected value under randomisation also is given. In addition, the object contains the following attributes:

- `attr(x, "ols")` an object of class `lm`. The OLS regression used to calculate the ID. Useful for identifying significant residuals (see Example below)
- `attr(x, "vars")` the names of Y and X in `data`
- `attr(x, "data")` a data frame with the population counts for Y and X

and also, for a multilevel model,

- `attr(index, "mlm")` an object of class `lmerMod`. Fitted using `lmer`
- `attr(index, "variance")` the percentage of the total variance due to each level of the model. This indicates the scale at which the segregation is most prominent
- `attr(index, "holdback")` records the percentage change in the ID that occurs if, at each level, its contribution to the ID *net* of other levels is heldback (set to zero)

See Also

checkerboard print.index holdback residuals.index lmer sumup

Harris R (2017) Fitting a multilevel index of segregation in R: using the MLID package http://rpubs.com/profrichharris/MLID

Harris R (2017) Measuring the scales of segregation: Looking at the residential separation of White British and other school children in England using a multilevel index of dissimilarity http://bit.ly/2lQ4r0n

Examples

```
data(ethnicities)
head(ethnicities)
# Calculate the standard index value
id(ethnicities, vars = c("Bangladeshi", "WhiteBrit"))

## Not run:
# Calculate also the expected value under randomisation
id(ethnicities, vars = c("Bangladeshi", "WhiteBrit"), expected = TRUE)
# will generate a warning because the total population per neighbourhood
# has not been specified
id(ethnicities, vars = c("Bangladeshi", "WhiteBrit", "Persons"),
expected = TRUE)
# The expected value is a high percentage of the actual value so
# aggregate it into a higher level geography...
aggdata <- sumup(ethnicities, sumby = "LSOA", drop = "OA")
head(aggdata)

# Multilevel models
id(aggdata, vars = c("Bangladeshi", "WhiteBrit"),
levels = c("MSOA","LAD","RGN"))
id(aggdata, vars = c("Bangladeshi", "WhiteBrit"),
levels = c("MSOA","LAD","RGN"), omit = c("Tower Hamlets", "Newham"))

## End(Not run)
```

---

impacts *Impact calculations*

Description

Calculates the total contribution to the index of dissimilarity of neighbourhoods grouped by regions or other higher-level geographies

Usage

```
impacts(data, vars, levels, omit = NULL)
```

Arguments

`data`	a data frame with `ncol(data) >= 2`. Each row of the data represents a neighbourhood or some other areal unit for which counts of population have been made.
`vars`	a character or numeric vector of length 2 or 3 giving either the names or columns positions of the variables in `data` in the following order 1. the number of population group Y in each neighbourhood 2. the number of population group X in each neighbourhood
`levels`	a character or numeric vector of minimum length 1 identifying either the names or columns positions of the variables in `data` that record to which higher-level grouping each lower-lower level unit belongs
`omit`	(optional) a character vector containing the names of places to search for in the data and to omit from the calculations

Details

When the index of dissimilarity (ID) is estimated as a regression model the residuals from that model are the differences between the share of population group Y and the share of population group X that are observed in each neighbourhood. The `impacts` function summaries those differences by higher-level geographies to consider which places or regions have the neighbourhoods that contribute most to the ID. The measures are useful for understanding where the seperations of the two population groups are greatest. However, to look at scale effects, where the effect of each level *net* of the other levels is wanted, fit a multilevel index using function `id`.

Value

A list of data.frames, each containing the impact calculations for the higher-level geographies. The variables are

- `pcntID` The total contribution of the neighbourhoods within the region to the overall ID score, expressed as a percentage
- `pcntN` The number of neighbourhoods within the region, expressed as a percentage of the total number in `data`
- `impact` The ratio of `pcntID` to `pcntN` multiplied by 100. Values over 100 indicate a group of neighbourhoods that have a disproportionately high impact on the ID
- `scldMean` The average difference between the share of the Y population and the share of the X population, scaled by the standard error of the differences for the whole data set (to give a z-value). Positive values mean that, on average, the region has a greater share of the Y population than the X. Negative values mean it has less.
- `scldSD` A measure of how much the differences between the shares of the two populations vary within the region. It is the standard deviation of those differences scaled by the standard error for the whole data set. Higher values indicate greater variability within the region.
- `scldMin` The minimum difference between the share of the Y population and the share of the X for neighbourhoods within the region, scaled by the standard error
- `scldMax` The maximum difference between the share of the Y population and the share of the X for neighbourhoods within the region, scaled by the standard error

- pNYgtrNX The percentage of neighbourhoods within the region where the count of population group Y (as opposed to the share) is greater than the count of population group X

Examples

```
data(aggdata)
impx <- impacts(aggdata, c("Bangladeshi", "WhiteBrit"), c("LAD","RGN"))
head(impx)
# sorted by impact score
# For $RGN London has the greatest impact on the ID
# The 'excess' share of the Bangladeshi population is not especially
# significant (see scldMean) but there is a lot of variation between
# neighbourhoods (see scldSD)
# For $LAD note the impacts of Tower Hamlets and Newham
```

MLID *MLID: a package for calculating a multilevel index of dissimilarity*

Description

Tools for fitting a multilevel index of dissimilarity to geographically hierarchical data. The results measure the two principal dimensions of segregation, unevenness and clustering. The amount of segregation, scale effects and the impact of particular places on the overall index can be assessed. The package development was funded partly under the ESRC's Urban Big Data Centre http://ubdc.ac.uk/, grant ES/L011921/1.

See Also

Harris R (2017) Measuring the scales of segregation: Looking at the residential separation of White British and other school children in England using a multilevel index of dissimilarity http://bit.ly/2lQ4r0n

---

plot.confintindex *Plot the confidence intervals for the multilevel residuals*

Description

Plots the confidence intervals to produce a caterpillar plot

Usage

```
## S3 method for class 'confintindex'
plot(x, ann = TRUE, grid = TRUE, ...)
```

Arguments

x	an object containing the output from function confint.index
ann	add annotation to the plot?
grid	arrange the plots in a grid? (Default is TRUE)
...	other arguments

See Also

catplot

---

print.fxindex *Print values*

Description

Prints predicted changes to the index of dissimilarity under various scenarios

Usage

```
## S3 method for class 'fxindex'
print(x, ...)
```

Arguments

x	output from effect
...	other arguments

---

print.impacts *Print values*

Description

Prints the impact values

Usage

```
## S3 method for class 'impacts'
print(x, ...)
```

Arguments

x	output from impacts
...	other arguments

print.index *Print values*

Description

Prints output from the single or multi-level index of dissimilarity

Usage

```
## S3 method for class 'index'
print(x, ...)
```

Arguments

x	output from id
...	other arguments

See Also

id holdback

residuals.index *Extract Model Residuals*

Description

For the index of dissimilarity (ID), the residuals are the differences between the share of the Y population and the share of the X population per neighbourhood. For the multilevel index, the residuals are estimated at and partitioned between each level of the model.

Usage

```
## S3 method for class 'index'
residuals(object, ...)
```

Arguments

object	an object of class index
...	other arguments

Value

a numeric vector of matrix containing the residuals

See Also

rstandard.index rstudent.index

Examples

```
data(aggdata)
index <- id(aggdata, vars = c("Bangladeshi", "WhiteBrit"))
# The ID can be derived from the residuals
0.5 * sum(abs(residuals(index)))
# which is the same as
index[1]

# Extract the standardized and look for regions where the share of the
# Bangladeshi population is unusualy high with respect to the White British
# resids <- rstandard(index)
# table(aggdata$RGN[resids > 2.58])

# Residuals for a multilevel index
index <- id(aggdata, vars = c("Bangladeshi", "WhiteBrit"),
levels = c("MSOA","LAD","RGN"))
resids <- residuals(index)
head(resids)
# Again, the ID can be derived from the residuals
0.5 * sum(abs(rowSums(resids)))

# Looking at the residuals, the London effect is different from other regions
sort(tapply(resids[,4], aggdata$RGN, mean))

# At the local authority scale it is Tower Hamlets and Newham
# (both in London) that have the highest share of the Bangladeshi population
# with respect to the White British:
tail(sort(tapply(resids[,3], aggdata$LAD, mean)),5)
```

---

rstandard.index *The Standardised resdiduals for the single-level Index of Dissimilarity*

Description

Calculates the standardised residuals for the single-level index

Usage

```
## S3 method for class 'index'
rstandard(model, ...)
```

Arguments

model	an object of class index generated by the function id
...	other arguments

Details

The residuals are the differences between the share of the Y population and the share of the X population per neighbourhood. A positive residual occurs where the share of the Y population exceeds the share of the X population, and a negative residual where the opposite. The standardised residuals can help to identify 'extreme' differences.

---

rstudent.index *The Studentised resdiduals for the single-level Index of Dissimilarity*

Description

Calculates the studentised residuals for the single-level index

Usage

```
## S3 method for class 'index'
rstudent(model, ...)
```

Arguments

model	an object of class index generated by the function id
...	other arguments

Details

The residuals are the differences between the share of the Y population and the share of the X population per neighbourhood. A positive residual occurs where the share of the Y population exceeds the share of the X population, and a negative residual where the opposite. The studentised residuals can help to identify 'extreme' differences.

---

sumup *Sum the data up into higher level groups*

Description

Aggregates the data into higher level groups by calculating the sum of all the numeric data columns by group

Usage

```
sumup(data, sumby, drop = NA)
```

Arguments

data	a data frame with ncol(data) >= 2. Each row of the data represents a neighbourhood or some other areal unit for which counts of population have been made.
sumby	a character or numeric vector of length 1 identifying either the name or columns position of the variables in data that records the higher-level group into which the data will be aggregated (summed)
drop	a character or numeric vector identifying any variables to be dropped from the aggregated data, such as lower-level names and identifiers

Details

Sometimes a population group is too few in number sensibly to be analysed at the smallest area scale. An indication of this is when the expected value under randomisation of the index of dissimilarity is a large fraction of the observed value. In this case, the data can be aggregated into higher level units, summing the population counts. Aggregating the data also can be used to explore how the index changes with the scale of analysis.

Value

a data frame containing the aggregated data

See Also

id

Examples

```
## Not run:
data(ethnicities)
head(ethnicities)
id(ethnicities, vars = c("Arab","Other","Persons"), expected = TRUE)
# the expected value is very high relative to the ID
aggdata <- sumup(ethnicities, sumby = "LSOA", drop = "OA")
head(aggdata)
id(aggdata, vars=c("Arab","Other","Persons"), expected = TRUE)
# Note the sensitivity of the ID to the scale of analysis

## End(Not run)
data(aggdata)
head(aggdata)
moreagg <- sumup(ethnicities, sumby = "MSOA", drop = "LSOA")
head(moreagg)
```

`tail.impacts` *Return the lowest impact scores for each higher level area*

Description

Returns the last parts of the set of impact calculations, ordered by the impact score

Usage

```
## S3 method for class 'impacts'
tail(x, n = 5, ...)
```

Arguments

`x`	an object of class `impacts` generated by the function `impacts`
`n`	a single integer giving the number of rows to show. Defaults to 5.
`...`	other arguments

See Also

`impacts`

Index