Package 'MTA'

September 25, 2017

Title Multiscalar Territorial Analysis

Version 0.1.1

Date 2017-09-25

Description Build multiscalar territorial analysis based on various contexts.

License GPL-3

URL https://github.com/riatelab/MTA/

BugReports https://github.com/riatelab/MTA/issues/

LazyData true

Depends R (>= 3.0)

Imports stats, sp, rgeos, igraph

Suggests cartography, knitr, rmarkdown, ineq, png, reshape2

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 6.0.1

NeedsCompilation no

Author Ronan Ysebaert [aut],
Nicolas Lambert [aut],
Timothée Giraud [aut, cre]

Maintainer Timothée Giraud <timothee.giraud@cnrs.fr>

Repository CRAN

Date/Publication 2017-09-25 12:35:17 UTC

R topics documented:

`cardist` *Time Distance Matrix Between Communes*

Description

Travel time between Grand Paris Metropole communes' centroids by car, in minutes.
Row names and column names match the DEPCOM field in com.

Source

The matrix is computed using the osrm package (https://cran.r-project.org/package=osrm).
Data (c) OpenStreetMap contributors, ODbL 1.0. http://www.openstreetmap.org/copyright
Routes: OSRM. http://project-osrm.org/

Examples

```
data(GrandParisMetropole)
cardist[1:10,1:10]
```

`com` *Grand Paris Metropole Communes Data*

Description

Data on the Grand Paris Metropole communes.

Format

A data frame with 150 rows and 10 variables:

DEPCOM Commune identifiers

LIBCOM Commune names

EPT EPT identifiers of the commune

LIBEPT EPT names of the commune

DEP Identifiers of the departement

INC Amount of income tax reference (in euros)

TH Number of tax households

Source

Direction générale des finances publiques, income tax 2014 (2013 incomes):
http://www.impots.gouv.fr/portal/dgi/public/statistiques.impot?espId=-4&pageId=stat_donnees_detaillees&sfid=4503
Atelier parisien d'urbanisme, Grand Paris communal composition (2015-12-17):
http://www.apur.org/article/composition-12-territoires-metropole-grand-paris

Examples

```
data(GrandParisMetropole)
head(com)
```

---

com.spdf *Grand Paris Metropole Communes SpatialPolygonsDataFrame*

Description

SpatialPolygonsDataFrame of the Grand Paris Metropole communes.

Format

DEPCOM Commune identifiers

Source

Institut national de l'information géographique et forestière (IGN), GEOFLA® 2015 v2.1 Communes France Métropolitaine:
http://professionnels.ign.fr/geofla
Atelier parisien d'urbanisme, Grand Paris communal composition (2015-12-17):
http://www.apur.org/article/composition-12-territoires-metropole-grand-paris

Examples

```
data(GrandParisMetropole)
sp::plot(com.spdf)
```

ept.spdf *Grand Paris Metropole EPTs SpatialPolygonsDataFrame*

Description

SpatialPolygonsDataFrame of Grand Paris Metropole EPTs. EPTs (Etablissements Publics Territoriaux) are groups of communes.

Format

EPT EPT identifiers

LIBEPT EPT names

Source

Atelier parisien d'urbanisme, Grand Paris communal composition (2015-12-17):
http://www.apur.org/article/composition-12-territoires-metropole-grand-paris

Examples

```
data(GrandParisMetropole)
sp::plot(ept.spdf)
```

gdev *General Deviation*

Description

This function computes the deviation between regional ratios and a ratio of reference. Each elementary unit's value will be compared to a global value.

Usage

```
gdev(x, var1, var2, type = "rel", ref = NULL)
```

Arguments

x	a data frame.
var1	name of the numerator variable in x.
var2	name of the denominator variable in x.
type	type of deviation; "rel" for relative deviation, "abs" for absolute deviation (see Details).
ref	ratio of reference; if NULL, the ratio of reference is the one of the whole study area (`sum(var1) / sum(var2)`).

Details

The relative global deviation is the ratio between var1/var2 and ref (`100 * (var1 / var2) / ref`). Values greater than 100 indicate that the unit ratio is greater than the ratio of reference. Values lower than 100 indicate that the unit ratio is lower than the ratio of reference.
The absolute global deviation is the amount of numerator that could be moved to obtain the ratio of reference on all units.

Value

A vector is returned.

Examples

```
# load data
data("GrandParisMetropole")
# compute absolute global deviation
com$gdevabs <- gdev(x = com, var1 = "INC", var2 = "TH", type = "abs")
# compute relative global deviation
com$gdevrel <- gdev(x = com, var1 = "INC", var2 = "TH", type = "rel")

# Deviations maps
if(require('cartography')){
  # set graphical parameters
  par(mar = c(0,0,1.2,0))
  # set breaks
  bks <- c(min(com$gdevrel),50,75,100,125,150,max(com$gdevrel))
  cols <- carto.pal(pal1 = "blue.pal", n1 = 3,
                    pal2 = "wine.pal", n2 = 3)
  # plot a choropleth map of the relative global deviation
  choroLayer(spdf = com.spdf, df = com, var = "gdevrel",
             legend.pos = "topleft",
             legend.title.txt = "Relative Deviation",
             breaks = bks, border = NA,
             col = cols)
  # add symbols proportional to the absolute general deviation
  com$sign <- ifelse(test = com$gdevabs<0, yes = "negative", no = "positive")
  propSymbolsTypoLayer(spdf = com.spdf, df = com, var = "gdevabs",var2 = "sign",
                       legend.var.pos = "left",legend.values.rnd = -2,
                       legend.var2.values.order = c("positive", "negative"),
                       legend.var.title.txt = "Absolute Deviation",
                       col = c("#ff000050","#0000ff50"),legend.var2.pos = "n",
                       legend.var.style = "e", inches = 0.2)
  # add EPT boundaries
  plot(ept.spdf, add=TRUE)
  # add a layout
  layoutLayer(title = "General Deviation (reference: Grand Paris Metropole)",
              sources = "GEOFLA® 2015 v2.1, Apur, impots.gouv.fr",
              north = TRUE,
              author = "MTA")
}
```

mas *Multiscalar Absolute Synthesis*

Description

This function sums the total amount of redistributions according to the three absolute deviations (global, territorial, spatial).

Usage

```
mas(x, var1, var2, ref = NULL, key, spdf, order = NULL, dist = NULL,
  mat = NULL, spdfid = NULL, xid = NULL)
```

Arguments

x	a data frame.
var1	name of the numerator variable in x.
var2	name of the denominator variable in x.
ref	ratio of reference; if NULL, the ratio of reference is the one of the whole study area (`sum(var1) / sum(var2)`).
key	aggregation key field.
spdf	a SpatialPolygonsDataFrame that matches x data frame.
order	contiguity order.
dist	distance threshold defining the contiguity. The cartesian distance between units centroids is used by default (see `gDistance`); use mat to apply different metrics.
mat	a distance matrix (road distance, travel time...) between x units. Row and column names must fit xid identifiers. (optional)
spdfid	identifier field in spdf, default to the first column of the spdf data frame. (optional)
xid	identifier field in x, default to the first column of x. (optional)

Value

A dataframe including the mass of numerator to redistribue to reach a perfect equilibrium according to the 3 contexts, expressed in numerator measure unit and as a share of the numerator mass.

Examples

```
data("GrandParisMetropole")
redistr <- mas(spdf = com.spdf,
                       x = com,
                       spdfid = "DEPCOM",
                       xid = "DEPCOM",
                       var1 = "INC",
                       var2 = "TH",
```

```
                        order = 2,
                        key = "EPT",
                        dist = NULL,
                        mat = NULL)
redistr
```

`mst` *Multiscalar Typology*

Description

This function compute a multiscalar typology according to the three relative deviations (general: G, territorial: T and spatial: S). The elementary units are classified in eight classes according to their three relative positions.

Usage

```
mst(x, var1, var2, ref = NULL, key, spdf, order = NULL, dist = NULL,
  mat = NULL, spdfid = NULL, xid = NULL, threshold, superior = FALSE)
```

Arguments

`x`	a dataframe.
`var1`	name of the numerator variable in x.
`var2`	name of the denominator variable in x.
`ref`	ratio of reference; if NULL, the ratio of reference is the one of the whole study area (`sum(var1) / sum(var2)`).
`key`	aggregation key field.
`spdf`	a SpatialPolygonsDataFrame that matches x data frame.
`order`	contiguity order.
`dist`	distance threshold defining the contiguity. The cartesian distance between units centroids is used by default (see `gDistance`); use mat to apply different metrics.
`mat`	a distance matrix (road distance, travel time...) between x units. Row and column names must fit xid identifiers. (optional)
`spdfid`	identifier field in spdf, default to the first column of the spdf data frame. (optional)
`xid`	identifier field in x, default to the first column of x. (optional)
`threshold`	defined to build the typology (100 is considered as the average).
`superior`	if TRUE, deviation values must be greater than threshold. If FALSE, deviation values must be lower than threshold.

Value

A dataframe including the initial dataset, the ratio, the 3 relative deviations (G, T and S) and the resulting typology.

Typology (which deviation is over/under the threshold):

- 0: none
- 1: G
- 2: T
- 3: G and T
- 4: S
- 5: G and S
- 6: T and S
- 7: G, T and S

Examples

```
data("GrandParisMetropole")
synthesis <- mst(spdf = com.spdf,
                         x = com,
                         spdfid = "DEPCOM",
                         xid = "DEPCOM",
                         var1 = "INC",
                         var2 = "TH",
                         dist = NULL,
                         key = "EPT",
                         order = 1,
                         mat = NULL,
                         threshold = 125,
                         superior = TRUE)

if(require('cartography')){
  par(mar = c(0,0,1.2,0))
  typoLayer(spdf = com.spdf, df = synthesis, var = "mst",
            border = "#D9D9D9",legend.values.order = 0:7,
            col = c("#f0f0f0", "#fdc785","#ffffab","#fba9b0",
                    "#addea6","#ffa100","#fff226","#e30020"),
            lwd = 0.25,
            legend.pos = "n")

  plot(ept.spdf,add=TRUE)

  colours <- c("#f0f0f0", "#fdc785","#ffffab","#fba9b0",
               "#addea6","#ffa100","#fff226","#e30020")

  rVal<-c(" .     .   . ",
          "[X]   .   . ",
          " .   [X]  . ",
          "[X] [X]  . ",
          " .    .   [X]",
```

```
            "[X]  .   [X]",
            " .   [X] [X]",
            "[X] [X] [X]")

  legendTypo(col = colours, categ = rVal,
             title.txt = "General, territorial and spatial\ndeviations above 125 %
             \n       G T S",
             nodata = FALSE)

  layoutLayer(title = "Multiscalar Typology",
              sources = "GEOFLA® 2015 v2.1, Apur, impots.gouv.fr",
              author = "MTA")
}
```

MTA *Multiscalar Territorial Analysis*

Description

Build multiscalar territorial analysis based on various contexts.
Main functions :

- gdev: general deviation between regional ratios and a ratio of reference.
- tdev: territorial deviation between regional ratios and ratios of an aggregated level.
- sdev: spatial deviation between regional ratios and ratios of neigborhing regions.
- mst: multiscalar typology based on the three deviations.
- mas: multiscalar absolute synthesis, total amount of redistributions based on the three deviations.

References

GRASLAND C., YSEBAERT R., ZANIN C., LAMBERT N., Spatial disparities in Europe (Chapter 4) in GLOERSEN E., DUBOIS A. (coord.), 2007, Regional disparities and cohesion: What Strategies for the future?, DG-IPOL – European Parliament.

---

sdev *Spatial Deviation*

Description

This function computes the deviation between regional ratios and local ratios. Local ratios are defined either by a contiguity order or by a distance measure between regions. Each elementary unit value will be compared to the value of its neighborhood.

Usage

```
sdev(x, var1, var2, type = "rel", spdf, order = NULL, dist = NULL,
  mat = NULL, spdfid = NULL, xid = NULL)
```

Arguments

x	a data frame.
var1	name of the numerator variable in x.
var2	name of the denominator variable in x.
type	type of deviation; "rel" for relative deviation, "abs" for absolute deviation (see Details).
spdf	a SpatialPolygonsDataFrame that matches x data frame.
order	contiguity order.
dist	distance threshold defining the contiguity. The cartesian distance between units centroids is used by default (see gDistance); use mat to apply different metrics.
mat	a distance matrix (road distance, travel time...) between x units. Row and column names must fit xid identifiers. (optional)
spdfid	identifier field in spdf, default to the first column of the spdf data frame. (optional)
xid	identifier field in x, default to the first column of x. (optional)

Details

The relative spatial deviation is the ratio between var1/var2 and var1/var2 in the specified neighborhoud. Values greater than 100 indicate that the unit ratio is greater than the ratio in its neighborhoud. Values lower than 100 indicate that the unit ratio is lower than the ratio in its neighborhoud.
The absolute spatial deviation is the amount of numerator that could be moved to obtain the same ratio in all units of its neighborhoud.

Value

A vector is returned.

Examples

```
# load data
data("GrandParisMetropole")
# compute absolute spatial deviation in a neighborhood defined by a contiguity
# order of 2.
com$sdevabs <- sdev(x = com, var1 = "INC", var2 = "TH",
                          type = "abs", spdf = com.spdf, order = 2)

# compute relative spatial deviation in a neighborhood defined within a distance
# of 5km between communes' centroids
com$sdevrel <- sdev(x = com, var1 = "INC", var2 = "TH", type = "rel",
                          spdf = com.spdf, dist = 5000)
```

```
# compute absolute spatial deviation in a neighborhood defined within a road
# travel time of 10 minutes by car
com$scardevabs <- sdev(x = com, var1 = "INC", var2 = "TH", type = "abs",
                           spdf = com.spdf, dist = 10, mat = cardist)
# compute relative spatial deviation in a neighborhood defined within a road
# travel time of 10 minutes by car
com$scardevrel <- sdev(x = com, var1 = "INC", var2 = "TH", type = "rel",
                           spdf = com.spdf, dist = 10, mat = cardist)

# map deviations
if(require('cartography')){
  # set graphical parameters
  par(mar = c(0,0,1.2,0))
  # set breaks
  bks <- c(min(com$scardevrel),50,75,100,125,150,max(com$scardevrel))
  bks <- sort(bks)
  # set colot palette
  cols <- carto.pal(pal1 = "blue.pal", n1 = 3,
                    pal2 = "wine.pal", n2 = 3)
  # plot a choropleth map of the relative spatial deviation
  choroLayer(spdf = com.spdf, df = com, var = "scardevrel",
             legend.pos = "topleft",
             legend.title.txt = "Relative Deviation",
             breaks = bks, border = NA,
             col = cols)
  # add symbols proportional to the absolute spatial deviation
  com$sign <- ifelse(test = com$scardevabs<0, yes = "negative", no = "positive")
  propSymbolsTypoLayer(spdf = com.spdf, df = com, var = "scardevabs",var2 = "sign",
                       legend.var.pos = "left",legend.values.rnd = -2,
                       legend.var2.values.order = c("positive", "negative"),
                       legend.var.title.txt = "Absolute Deviation",
                       col = c("#ff000050","#0000ff50"),legend.var2.pos = "n",
                       legend.var.style = "e", inches = 0.2)
  # add a layout
  layoutLayer(title = "Spatial Deviation (neighborhoud: 10 minutes by car)",
              sources = "GEOFLA® 2015 v2.1, impots.gouv.fr", north = TRUE,
              author = "MTA")
}
```

---

tdev *Territorial Deviation*

Description

This function computes the deviation between regional ratios and ratios of an aggregated level. Each elementary unit's value will be compared to the value of the aggregated level it belongs to.

Usage

```
tdev(x, var1, var2, type = "rel", key)
```

Arguments

x	a data frame.
var1	name of the numerator variable in x.
var2	name of the denominator variable in x.
type	type of deviation; "rel" for relative deviation, "abs" for absolute deviation (see Details).
key	aggregation key field.

Details

The relative territorial deviation is the ratio between var1/var2 and var1/var2 at the aggregated level. Values greater than 100 indicate that the unit ratio is greater than the ratio at the aggregated level. Values lower than 100 indicate that the unit ratio is lower than the ratio of the aggregated level.
The absolute territorial deviation is the amount of numerator that could be moved to obtain the ratio of the aggregated level on all belonging units.

Value

A vector is returned.

Examples

```
# load data
data("GrandParisMetropole")
# compute absolute territorial deviation (EPT level)
com$tdevabs <- tdev(x = com, var1 = "INC", var2 = "TH", type = "abs",
                            key = "EPT")
# compute relative territorial deviation (EPT level)
com$tdevrel <- tdev(x = com, var1 = "INC", var2 = "TH", type = "rel",
                            key = "EPT")

# map deviations
if(require('cartography')){
  # set graphical parameters
  par(mar = c(0,0,1.2,0))
  # set breaks
  bks <- c(min(com$tdevrel),75,100,125,150,max(com$tdevrel))
  # set colot palette
  cols <- carto.pal(pal1 = "blue.pal", n1 = 2,
                    pal2 = "wine.pal", n2 = 3)
  # plot a choropleth map of the relative territorial deviation
  choroLayer(spdf = com.spdf, df = com, var = "tdevrel",
             legend.pos = "topleft",
             breaks = bks, border = NA,
             col = cols)
  # add symbols proportional to the absolute territorial deviation
  com$sign <- ifelse(test = com$tdevabs<0, yes = "negative", no = "positive")
  propSymbolsTypoLayer(spdf = com.spdf, df = com, var = "tdevabs",var2 = "sign",
                       legend.var.pos = "left",legend.values.rnd = -2,
                       legend.var2.values.order = c("positive", "negative"),
```

```
                        legend.var.title.txt = "Absolute Deviation",
                        col = c("#ff000050","#0000ff50"),legend.var2.pos = "n",
                        legend.var.style = "e", inches = 0.2)
  # add EPT boundaries
  plot(ept.spdf, add=TRUE)
  # add a layout
  layoutLayer(title = "Territorial Deviation",
              sources = "GEOFLA® 2015 v2.1, Apur, impots.gouv.fr",
              author = "MTA")
}
```

Index