Package 'cartograflow'

May 10, 2019

Title Filtering Matrix for Flow Mapping

Version 1.0.0

Description
Functions to prepare and filter an origin-destination matrix for thematic flow mapping purposes. This comes after Bahoken, Francoise (2016), Mapping flow matrix a contribution, PhD in Geography - Territorial sciences. See Bahoken (2017) <doi:10.4000/netcom.2565>.

Depends R (>= 3.4.0)

License GPL-3

URL https://halshs.archives-ouvertes.fr/tel-01273776 ,
http://journals.openedition.org/netcom/2565 ,
https://mappemonde-archive.mgm.fr/num44/articles/art14404.html

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests knitr, rmarkdown, testthat, cartography

VignetteBuilder knitr

Imports dplyr, g.data, graphics, ggplot2, maptools, plotly, rgdal, rgeos, reshape2, rlang, sp, utils

NeedsCompilation no

Author Sylvain Blondeau [aut, cre], Francoise Bahoken [aut]

Maintainer Sylvain Blondeau <blondeau.sylvain@yahoo.fr>

Repository CRAN

Date/Publication 2019-05-10 15:50:09 UTC

R topics documented:

`cartograflow` *Cartograflow Package*

Description

This package is designed to create the so-called flowmaps, by filtering origin-destination (OD) matrix. It is based on different functions that are mainly used to prepare the flow dataset. The spatial objects processing are those of sp or sf and the mapping elements are often those of Cartography except particular cases.

Details

To learn more about cartograflow, see the vignette cartograflow.html

Main functions :

- flowanalysis flowcarre flowcontig flowdist flowgini flowjointure flowmap flowreduct flowstructmat flowcontig

---

`flowanalysis` *Computation of a global concentration criterion of flows values or features*

Description

`Computation of a global selection criterion for thresholding flow information and/or features before m`
To be use after flowgini and before flowmap.

Usage

```
flowanalysis(tab, critflow, critlink, result)
```

Arguments

tab	flow dataset from flowgini
critflow	level of flow significativity. See Details.
critlink	level of features density. See Details.
result	resulting filtering criterion value. See Details.

Details

-critflow = desired level of flow's information significativity (e.g. 80 of the total information ;
-critlink = desired level of flow's features density (e.g. 20 features that represents "more significant information.

-result="density" returns the desired level of features density as a -result = "significativity" returns the level of flow significativity as a

References

Bahoken Françoise, 2016,« La cartographie d'une sélection globale de flux, entre 'significativité' et 'densité' », Netcom Online, 30-3/4 | 2016, Online since 23 March 2017, connection on 05 May 2019. URL : http://journals.openedition.org/netcom/2565 ; DOI : 10.4000/netcom.2565

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                  lib.loc = NULL, mustWork = TRUE)

#1/4: Computes Gini's coefficent
tab_gini<-flowgini(flows,format="L",origin="i",dest="j",valflow="Fij",
          bkg,code="EPT_NUM",lorenz.plot = FALSE)
### [1] Gini's coefficent = 73.16 %

#2/4: Plot Lorenz curve
flowgini(tab_gini,format="L",origin="i",dest="j",valflow="ydata",
          bkg,code="EPT_NUM",lorenz.plot = TRUE)

#3/4: Compute critflow filtering parameter
#critflow = 0.8 #selected criterion
flowanalysis(tab_gini,critflow = 0.8,result = "signif")
### [1] "threshold =  11238  ---  flows =  80 % ---  links =  22.94 %"

#4/4: Plot the flowmap

flowmap(flows,format="L",bkg,code = "EPT_NUM",filter = TRUE,
         threshold = 11238,taille = 8,a.head = 1,a.length = 0.11,
         a.angle = 30,a.col="#3f4247")
```

`flowcarre` *Create a square matrice from geographical ID*

Description

Create a closed and square matrice from a list of geographic ID

Usage

```
flowcarre(liste, tab, origin, dest, valflow, empty.sq, format, diagonale)
```

Arguments

`liste`	list or all the geographical ID as a single vector flow dataset
`tab`	the non squared input flow dataset with three column : origin, destination, flow value
`origin`	origin place
`dest`	destination place
`valflow`	flow value
`empty.sq`	TRUE : to allows to have an matrice empty with only the ID of background map ; FALSE or missing
`format`	the squared flow dataset output format. See Details.
`diagonale`	See Details.

Details

- format is "M" for matrice format

- format is "L" for long format

- diagonal is "TRUE" to zero the main diagonal

Examples

```
library(cartograflow)
data(flowdata)
var1<-geoid
var2<-flows

#1/2 Compute an empty square matrice with ID code, and sets the value to zero
#Example for matrice format (same procedure for the long format)

mat<-flowcarre(var1,var2,origin="i",dest="j",valflow="Fij",
               format="M",empty.sq=TRUE)

#2/2 Fill in the matrice with external flow values
mat<-flowcarre(var1,var2,origin="i",dest="j",valflow="Fij",
               format="M",empty.sq=FALSE)
```

```
#Square a matrice and zero the main diagonal
mat<-flowcarre(var1,var2,origin="i",dest="j",valflow="Fij",format="M",
               empty.sq=FALSE,diagonale = FALSE)
```

---

flowcontig *Computes an ordinal distance matrices based on geographical background*

Description

From a geographical background, compute an ordinal distance matrice based on a k-contiguity. The result is a neighbourhood graph that can be used for filtering flow values beor flow mapping (flowmap)

Usage

```
flowcontig(fdc, code, ordre)
```

Arguments

fdc	is the map background file (ie. a shapefile of polygons)
code	identifiant
ordre	number of borders to cross between origin and destination place. See details.

Details

Contiguity is in terms of the (k=1,2,4) number of spatial boundaries to be crossed between a place of origin and a place of destination -ordre=1 is when the flow have to cross only 1 boundary
-ordre=2 is when the origin-destinations places are distant from 2 borders
-ordre=4 is when the origin-destinations places are distant from 4 borders

Value

a (k) contiguity matrice with the (k) contiguity measures

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                  lib.loc = NULL, mustWork = TRUE)
graph_ckij_1<-flowcontig(bkg,"EPT_NUM",ordre =1)

flowmap(graph_ckij_1,format="L",bkg,"EPT_NUM",
        filter = TRUE, taille = 0.5)
```

`flowdist` *Compute continous distance matrix from geographical background*

Description

From a geographical background computes (and threshold) a distance matrix.

Usage

```
flowdist(tab, dist.method, result)
```

Arguments

`tab`	the input flow dataset.
`dist.method`	euclidian calculation
`result`	take the vallue "flowdist" or "dist" allows to parameter the resulting distance dataset (flows filtered by a distance criterion or not)

Details

– result = "dist" is the resulting tab of the distance
– result = "flowdist" with all the calculated parameters

Value

(1) A flowdata set with continuous euclidian distances calculations, see dist.method parameter

(2) A flowdata set with movement from euclidian distances calculations.

(3) A flowmap filtered by a global distance criterion.

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                  lib.loc = NULL, mustWork = TRUE)
tab<-flowjointure(flows,bkg,"EPT_NUM")
#Format long with only origin, destination and distance parameters:
tab.distance<-flowdist(tab, dist.method = "euclidian",result = "dist")
#Format long with with all parameters: coordinates, distance, mouvement
tab.distance<-flowdist(tab, dist.method = "euclidian",result = "flowdist")
```

`flowgini` *Analysis of flow concentration (Gini coefficient)*

Description

Calculates Gini coefficient, plot Lorenz curve and threshold the matrice according to a global concentration criterion for mapping flow intensity or flow density).
To be use before flowanalysis

Usage

```
flowgini(tab, origin, dest, valflow, format, fdc, code, lorenz.plot)
```

Arguments

`tab`	dataset is a matrice or long format
`origin`	origin place to be used with the long format
`dest`	destination place to be used with the list format
`valflow`	to be used with the list format
`format`	is a variable that identify the data : matrice or long format
`fdc`	is the map background file, ie. a shapefile.
`code`	is the map background IDs code
`lorenz.plot`	allows to plot the Lorenz curve associate to the gini coefficient

Details

flowgini(...,lorenz.plot = TRUE) for ploting Lorenz curve from cumulated flows and links.

Value

plot Lorenz curve for the cumulated flow and links : flowgini(...,gini.plot = TRUE),warning : the function must be not assign a variable

value of the Gini's coefficent and the table : table<-flowgini(...,missing(gini.plot) or gini.plot = FALSE)

References

Bahoken Françoise, 2016,« La cartographie d'une sélection globale de flux, entre 'significativité' et 'densité' », Netcom Online, 30-3/4 | 2016, Online since 23 March 2017, connection on 05 May 2019. URL : http://journals.openedition.org/netcom/2565 ; DOI : 10.4000/netcom.2565.
Grasland Claude, 2014, "Flows analysis carto", unpublished R functions.

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                  lib.loc = NULL, mustWork = TRUE)
#Computes Gini's coefficent
tab_gini<-flowgini(flows,format="L",origin="i",dest="j",valflow="Fij",
          bkg,code="EPT_NUM",lorenz.plot = FALSE)
#Plot Lorenz curve
flowgini(tab_gini,format="L",origin="i",dest="j",valflow="ydata",
          bkg,code="EPT_NUM",lorenz.plot = TRUE)
#See \link{flowanalysis} for viewing the tab_gini table
```

`flowjointure` *Create a spatial join with flow*

Description

Create an attribute spatial join between a flow dataset table and a map background

Usage

```
flowjointure(tab, fdc, code)
```

Arguments

`tab`	the input flow dataset table in long format
`fdc`	the map background file, ie. a shapefile
`code`	the ID of the spatial units in the map background

Value

Resulting jointure table between flow dataset and map background

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                   lib.loc = NULL, mustWork = TRUE)
tabflow<-flowjointure(flows,bkg,"EPT_NUM")
```

`flowmap` *Mapping a flow matrix origin-destination*

Description

Mapping a flow matrix origin-destination

Usage

```
flowmap(tab, format, fdc, code, filter, threshold, taille, a.head,
  a.length, a.angle, a.col)
```

Arguments

`tab`	the input flow dataset .csv
`format`	the flow dataset format : M=matrice or L=long.
`fdc`	the geographical background file .shp
`code`	is the column with the spatial units ID
`filter`	allows you to filter (or not) the flow dataset. See details
`threshold`	is the value of the threshold criterion used to filter the values. The default is 1.
`taille`	is a graphical parameter for modifying the width of the feature
`a.head`	integer code, determining the kind of arrows to be drawn. See Details
`a.length`	length of the edges of the arrow head (in inches).
`a.angle`	angle from the shaft of the arrow to the edge of the arrow head.
`a.col`	color of the arrows

Details

The flow dataset must be converted to a dataframe for optimal performance (troubles remains with tibble format)

If filter = FALSE, all the matrice values are plot [(n*(n-1)] cells, i.e. all links out of the main diagonal. If filter = TRUE only non-zero values are plotted, i.e. existing links with or without threshold. The default threshold is set to 1.

a.head is for applying an arrow or not – code=0 : the link has no head - no arrow – code=1 : an arrow is draw at (x0[i], y0[i]). – code=2 : an arrow is draw at (x1[j], y1[j]) – code=3 : an arrow is draw at both nodes.

Value

a matrix or a list with the correct tabflow ID code

The resulting flowmap

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                  lib.loc = NULL, mustWork = TRUE)
flowmap(flows,format="L",bkg,code = "EPT_NUM",filter = TRUE,
         threshold = 20,taille = 5,a.head = 1,a.length = 0.05)
```

`flowreduct` *Flow matrice reduction according to another matrix*

Description

Reduces a flow dataset according to an external matrix (eg. distance travelled) Computes geographical movements (by weighting a flow dataset according to a distance criterion)

Usage

```
flowreduct(tab, tab.metric, metric, select, d)
```

Arguments

`tab`	is the input flowdata set.
`tab.metric`	is the table of distance (continuous dataset) or contiguity (ordinal dataset)
`metric`	See Details.
`select`	is the continuous distance thresholding parameter. See Details.
`d`	distance thresholds criterion.

Details

– Metric is 'continous" for distance as euclidian, maximum, manhattan, etc.
See flowdist for computing neighbourhood ordinal distance matrix.

select ="dmin" for reducing flow dataset to values that are up or equal to the dmin distance parameter ;
select ="dmax" for reducing flow dataset to values that are less or equal to the dmax distance.

– Metric is 'ordinal' for neighbourhood ordinal distance so-called k contiguity.
See flowcontig for computing continuous distance matrix

Value

A flow dataset with continuous euclidian distances calculations

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                  lib.loc = NULL, mustWork = TRUE)
tab<-flowjointure(flows,bkg,"EPT_NUM")

#Example for reducing a flow matrice with a distance matrice, in long format (i,j, distance)
##1/2: Computes the matrice distances
tab.distance<-flowdist(tab, dist.method = "euclidian",result = "dist")
##2/2: Reduce the flow matrice
tab.flow<-flowreduct(tab,tab.distance, metric = "continous",
                     select = "dmax", #maximum distance travelled criterion
                     d = 8567) #maximum distance value
```

---

flows *MOBPRO: professional commuting flows in 2015*

Description

Statistical dataset in .csv: Extraction of the french national census : "Mobilités professionnelles en 2015 : - Base flux de mobilité - for the Greater Paris area.
Citation: INSEE - RP MOBPRO, 2015.
Variable (i) is the place of origin of the flow.
Variable (j) is the place of destination of the flow.
Variable (Fij) is the flow value for an (i,j) couple of origin-destination places.
Variable (count) is the frequency of the (i,j) couple of places.

Source

https://www.insee.fr/fr/statistiques/fichier/3566008/rp2015_mobpro_txt.zip

---

flowstructmat *Structuring a matrix*

Description

Fixes an ID shift in the flow matrix (to bo use with flowjointure if necessary and flowtabmat)

Usage

```
flowstructmat(z)
```

Arguments

z	The flow dataset is in the matrice format where the first column is filled with the ID

Value

A flowdataset with an usable format

Examples

```
library(cartograflow)
data(flowdata)

dim(mat_ex) # dimension fo the original matrice
### 10 11  # first colum is fill with the ID

tab<-flowstructmat(mat_ex)
dim(tab)
## 10 10 # dimension fo the resulting matrice
```

---

flowtabmat *Changing the format of a flow dataset*

Description

Transform a flow dataset from long to matrice format, and vice versa

Usage

```
flowtabmat(tab, matlist)
```

Arguments

tab	flow dasaset, in matrice or long format
matlist	choose matrice or long as the result format. See Details.

Details

– From long to matrice format [n*n]: matlist="M";
– From matrice to long format [i,j,Fij]: matlist="L".

Value

flow data in matrice or long format

Examples

```
library(cartograflow)
data(flowdata)
#1: From long to matrix format (n*m)
matFlow<-flowtabmat(flows,matlist="M")
#2: From matrix to long format [i,j,Fij]
listflow<-flowtabmat(matFlow,matlist="L")
```

`flowtype` *Compute flowdata types*

Description

Compute gross and net flows from initial asymetric flow values

Usage

```
flowtype(tab, format, x)
```

Arguments

`tab`	the input flow dataset
`format`	specify the flow dataset format : M = square matrix [n*n] or L=lng [i,j,data]
`x`	enter the computation type : "flux", "transpose", "bivolum" and "bisold".

Details

The matrice must be squared (if not, see flowcarre). This function compute for all (i,j) involved in an asymetric flow matrix (Fij) several matrix. - x = "flux" for initial flow (Fij)
- x = "transpose" for reverse flow value (Fji)
- x = "bivolum" for bilateral gross flow Vij=(Fij+Fji)
- x = "bisold" for bilateral net flow Sij=(Fij-Fji)

References

Bahoken Francoise, 2016, L'approche cartographique de la décomposition des matrices de flux, Mappemonde, Revue sur l'image géographique et les formes du territoire, number 116, URL : https://mappemonde-archive.mgm.fr/num44/articles/art14404.html

Examples

```
library(cartograflow)
data(flowdata)
bkg<- system.file("shape/MGP_TER.shp", package="cartograflow",
                  lib.loc = NULL, mustWork = TRUE)

##1a:Computes flowtypes: Matrice format
```

```
matflow<-flowtabmat(flows,matlist = "M")
m<-flowtype(matflow,format ="M",x="flux")
m<-flowtype(matflow,format ="M",x="transpose")
m<-flowtype(matflow,format ="M",x="bivolum")
m<-flowtype(matflow,format ="M",x="bisold")

##1b:Computes flowtypes: Long format
list<-flowtabmat(matflow,matlist="L")
colnames(list)<-c("i","j","Fij")
l_all<-flowtype(list,format ="L",x="all")
l_sold<-flowtype(list,format ="L",x="bisold")


#2:flowmapping: example of bisold
flowmap(l_sold,format = "L",bkg,code = "EPT_NUM",
         filter= TRUE,threshold= 20, taille = 5)
```

---

geoid *Geographical ID*

Description

One column dataframe in.csv.
Variable (COD_GEO_EPT) is the geographical code of the territory
citation : APUR, 2018

Source

https://www.insee.fr/fr/statistiques/fichier/3566008/rp2015_mobpro_txt.zip

---

mat_ex *Example of a small flow data matrix*

Description

Example of a small data, in format matrice

Index