Package 'Blaunet'

May 10, 2020

Type Package

Title Calculate and Analyze Blau Status for (Covert) Organizations

Depends R (>= 3.0.0)

Imports gWidgets, gWidgetsRGtk2, RGtk2, cairoDevice, plot3D, plot3Drgl, rgl, network, sna, ergm, statnet.common, haven, foreign

Version 2.0.9

Author Cheng Wang*, Michael Genkin*, George Berry, Liyuan Chen, Matthew Brashears *Both authors contributed equally to this work and their names are randomly ordered

Maintainer Cheng Wang <chengwang@wayne.edu>

Description An integrated set of tools to calculate and analyze Blau statuses quantifying social distance between individuals belonging to (covert) organizations. Relational (network) data can be incorporated for additional analyses. This project is supported by Defense Threat Reduction Agency (DTRA) Grant HDTRA-10-1-0043.

License GPL-3

URL https://www.facebook.com/groups/425015561030239/

Repository CRAN

Date/Publication 2020-05-10 14:40:06 UTC

NeedsCompilation no

R topics documented:

`Blaunet-package` *Calculate and Analyze Blau statuses for measuring social distance*

Description

An integrated set of tools to calculate and analyze Blau statuses quantifying social distance between individuals belonging to organizations. Blaunet works by considering individuals placed in an abstract social space, defined by various user-provided variables. Organizations recruit from an area (or "niche") in social space. Blaunet analyzes where organizations recruit from, the positions users occupy in this competitive environment, and substantial features of the connections between individuals.

Details

Package:	Blaunet
Type:	Package
Version:	2.0.9
Date:	2020-05-10
Depends:	R (>= 3.0.0)
Imports:	gWidgets, gWidgetsRGtk2, RGtk2, cairoDevice, plot3D, plot3Drgl, rgl, network, sna, ergm, statnet.common, haven,foreign
License:	GPL-3

1. Purpose of software: The purpose of the software is to perform Blau status analysis, which is an analytic framework that allows one to describe how organizational competition affects micro-level processes. It relies on abstractly conceptualizing individuals as inhabiting a k-dimensional euclidean space formed by k variables.

2. Defintions

a. Blau Space: A k-dimensional space, where socio-demographic characteristics such as age, years of education, or income are used to structure the dimensions of the space. These characteristics are referred to as Blau parameters. Individuals are then populated in this space based on their attributes on the specified demographic dimensions. Each person is represented by a set of k coordinates, which correspond to the person's attributes on the Blau parameters.

b. Niche: The area of the Blau space from which organizations recruit members. The niches are calculated based on the Blau parameters of individuals already in the organization. Organizations compete with each other both by recruiting members already inhabiting their niches and by expanding their niches. The extent to which niches overlap reflects competition between organizations for members. The niche for an organization is a hypercube in k-dimensional space.

c. Blau Status: Denotes a person's (or a relationship's) position in the organizational competitive ecology relative to the competing niches. Blau statuses that characterize a person's position in the

ecology are called nodal blau statuses. Computing nodal statuses only requires membership information. Blau statuses that characterize a relationship's position in the ecology are called dyadic blau statuses. Computing dyadic statuses requires not only membership, but also sociometric network information.

d. Ecology: A relatively bound system where organizations compete for members. The system is bound by spatial or other considerations. For example, a school where extracurricular clubs compete for members could represent a single ecology. A second school in another town may form a second ecology, as the second schools clubs do not recruit at the first school. A town where different voluntary organizations compete for members can also represent a single ecology. By assumption, organizations can only compete within an ecology. For instance, an Elks group can compete with a church group so long as the two draw members from the same community. However, much like the boundary specification problem (Laumann, Marsden, Prensky 1983) in social networks, ecologies are difficult to precisely define.

3. Data Structure: Several types of information are necessary or helpful to specific the Blau statuses of individuals: **i.)** Individual ID: this variable identifies the individuals in the dataset

ii.) Ecology ID: indicates how individuals are partitioned into ecologies (e.g. schools, cities, etc.).

iii.) Blau Parameters: continuous demographic characteristics of individuals that the researcher finds relevant to determining similarity among individuals (e.g. age, SES, BMI). This set of variables is used to construct the niches that structure Blau space.

iv.) Organizational Memberships: the membership of individuals in specific organizations (e.g. charitable organizations, religious groups, fraternal organizations). This set of variables is used to construct the niches that individuals occupy.

v.) Network data: the sociometric network information of individuals. This information is used to compute dyadic blau statuses, although is not necessary for other status computations.

vi.) Weights (optional): for data that was sampled from a population and where Blau parameters need to be adjusted by weights.

4. Note: We've made every effort to make sure that that the program works correctly. However, we advise caution as we are unable to test on every type of dataset. If anything looks anomalous, please let us know at the maintenance email address below.

Author(s)

Cheng Wang*, 'chengwang@wayne.edu'

Michael Genkin*, 'mgenkin@smu.edu.sg'

George Berry, 'geb97@cornell.edu'

Liyuan Chen,

Matthew E. Brashears, 'brasheam@mailbox.sc.edu'

*Both authors contributed equally to this work and their names are randomly ordered

Maintainer: Cheng Wang <'chengwang@wayne.edu'>

References

McPherson, J. Miller. (1983). "Ecology of Affiliation". American Sociological Review, 48(4), 519-532.

McPherson, J. Miller. (2004). "A Blau Space Primer: Prolegomenon to an Ecology of Affiliation". Industrial and Corporate Change, 13(1), 263-280.

Genkin, Michael, Cheng Wang, George Berry, and Matthew E. Brashears. (2018). "Blaunet: An R-based Graphical User Interface Package to Analyze Blau Space". PLoS ONE, 13(10), e0204990.

active *Quick summary of blau object.*

Description

Identifies measures that have been computed.

Usage

```
active(blauObj)
```

Arguments

blauObj	An object of class blau initialized with the function blau.

Details

Since measures computed with many functions are stored in the blau object itself, it is helpful to quickly see which elements are active.

Value

A string indicating the active elements.

Examples

```
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
b <- niches(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions
active(b) #will produce command line output
```

blau *Converts raw data into an object for Blau status analysis*

Description

Converts a matrix of organization memberships and demographic dimensions, along with (optionally) an edgelist or adjacency matrix, into an object of class blau. Automatically detects organization membership and demographic columns.

Usage

```
blau(
  square.data, graph = NULL, directed.el = FALSE, node.ids = NULL,
  weights = NULL, ecology.ids = NULL, exclude = NULL, dimensions = NULL,
  memberships = NULL, complete.cases = FALSE
  )
```

Arguments

square.data	R matrix or data.frame object that must contain demographic and membership information (in columns). May also include columns of individual or ecology identifiers, weights, or a primary membership column.
graph	A named edgelist or adjacency matrix. This is required for computing measures which incorporate network information. Relies on the network object from package network.
directed.el	Defaults to FALSE. Used only to indicate whether an edgelist passed to this function should be treated as directed or undirected. Not necessary if passing an adjacency matrix.
node.ids	Indicates the column which holds node (individual) identifiers. May be the column number or column name.
weights	A column with weights for each observation. May be the column number or column name. In the absence of specification, all weights are assumed to be equal (and are set to 1). Weights are used in a weighted standard deviation calculation.
ecology.ids	Indicates the column which holds ecology identifiers. May be the column number or column name.
exclude	A way to manually exclude columns from automatic incorporation as membership or demographic columns. May be a vector of column names as strings, or a vector of column numbers. Useful for larger datasets where the vast majority of columns are included.
dimensions	Indicates the columns which hold Blau parameter information. May be a vector of column names as strings, or a vector of column numbers. In the absence of specification, all non-binary columns that are not used for other purposes will be assumed to be demographic variables.
memberships	Indicates the columns which hold organizational membership information. May be a vector of column names as strings, or a vector of column numbers. In the absence of specification, all binary columns that are not used for other purposes will be assumed to be membership indicators.
complete.cases	Defaults to FALSE. A boolean setting indicating whether all rows with at least one missing value should be dropped before proceeding with calculations. Otherwise, the program will be "greedy" about determining niche boundaries, using as much information as possible.

Details

The most common raw data format is a .csv file that contains ecology identifiers, node identifiers, blau parameters, and memberships, among other variables. **Note:** Unless told otherwise, this function will automatically assume binary columns are memberships and non-binary columns are parameters. Manual specification of memberships or parameters will overwrite this auto-detection. This makes it easy to work with relatively large data without specificing dozens of columns.

The vast majority of configuration takes place when calling the blau function. As such, it is essential that the user understand how choices made here impact the operation of subsequent functions. The easiest way to get started is to determine which of the four optional parameters—node identifiers (node.ids), ecology identifiers (ecology.ids), weights (weights), and relational data (graph)—are present in your dataset and will be used for analysis. These should be specified by indicating their locations with the appropriate function argument, and the blau function will automatically assume all other columns are either membership or demographic columns. If there are columns to

be excluded from analysis, they can be specified with the exclude parameter. This type of setup is appropriate for the vast majority of datasets.

It is important to remember that any information incorporated into the `blau` object through this function will be used when calling subsequent functions. For instance, if your analysis does not require weights, but they are provided in the dataset, they should be explicitly excluded with the `exclude` argument.

If ecology identifiers are provided, all subsequent analyses will automatically proceed on a by-ecology level (unless specified explicitly in subsequent functions).

With network information, the most important consideration is that node identifiers are properly indicated and may be matched up with node identifications provided with the `node.ids` parameter. Adjacency matrix or edgelist input formats are both converted to an `network` object. The preferred format is a named edgelist (two columns, with node names in each row indicating an edge).

If node names are `numeric` rather than `character`, they should still be specified in the input function with `node.ids` and a network should correctly indicate node identifiers.

If complete.cases is `FALSE` (the default option), we automatically use as much information as possible to compute niche boundaries. For example, an individual may have missing Blau parameter information for a certain dimension. Under the default settings, we still utilize the user's other demographic information to compute niche boundaries. If compleCases is specified as `TRUE`, then only observations with no missing values along all elements in the input matrix will be utilized in determining boundaries.

Value

`obj`	An object of class `blau`. Object contains several elements that may be accessed with the $ operator.

Examples

```
##simple example
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
##blau object will store whatever you compute
b <- niches(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions
##see active elements
print(b)
##compute global Blau statuses
b <- nodal.global(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions

##assume we don't care about the 'ideo' column
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp', exclude='ideo')
##compute niches like before
b <- niches(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions

##assume we only want income and educ parameters. Note the "c".
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp', dimensions=c('income', 'educ'))

##example with relational data
data(BSANet)
square.data <- BSANet$square.data
el <- BSANet$el

b <- blau(square.data, node.ids = 'person', ecology.ids = 'city', graph = el)
#compute dyadic statuses
```

```
b <- dyadic(b, dev.range = rep(1.5, 3)) # 3 is the number of dimensions
```

---

blaunetgui *Run Blau status analysis in graphic user interface (GUI).*

Description

Load graphic user interface (GUI) for Blaunet package.

Usage

```
blaunetgui()
```

Details

blaunetgui has the following five main capabilities, it can:

1) identify a list of possible salient dimensions;

2) calculate, plot, and analyze niches for social entities by measuring the social distance along the salient dimensions between individuals affiliated with them;

3) capture niche dynamics cross-sectionally by calculating the intensity of exploitation from the carrying capacity and the membership rate;

4) analyze the niche movement longitudinally by estimating the predicted niche movement equations; and

5) generate Blau bubbles for individuals, thereby allowing the study of interpersonal influence of similar others even with no or limited network information.

Examples

```
##Not Run
##blaunetgui()
```

BSANet *Example dataset for relational measures.*

Description

The dataset is a practice dataset created to illustrate the dyadic blau status functions of Blaunet.

Usage

```
data(BSANet)
```

Format

The format is:
List of 3
$ square.data:'data.frame': 10 obs. of 7 variables:
..$ person : Factor w/ 10 levels "Amir","Aude",..: 6 5 8 1 7 4 10 2 9 3
..$ city : Factor w/ 2 levels "New York","San Francisco": 1 1 1 1 1 2 2 2 2 2
..$ age : int [1:10] 24 38 58 47 33 23 28 22 29 41
..$ income : int [1:10] 5 1 2 4 4 0 6 1 7 5
..$ educ : int [1:10] 7 3 7 2 4 7 0 4 5 7
..$ grp_conservative: int [1:10] 1 0 1 0 0 1 1 1 1 1
..$ grp_liberal : int [1:10] 1 1 0 0 0 0 1 1 1 1
$ el :'data.frame': 8 obs. of 2 variables:
..$ V1: Factor w/ 6 levels "Amir","Aude",..: 4 4 5 1 3 3 2 6
..$ V2: Factor w/ 6 levels "Aude","Bruce",..: 3 5 3 4 6 1 6 2
$ adj :'data.frame': 10 obs. of 10 variables:
..$ John : int [1:10] 0 1 1 0 0 0 0 0 0 0
..$ Emma: int [1:10] 1 0 1 0 0 0 0 0 0 0
..$ Mary : int [1:10] 1 1 0 0 0 0 0 0 0 0
..$ Amir : int [1:10] 0 0 0 0 1 0 0 0 0 0
..$ Mark : int [1:10] 0 0 0 1 0 0 0 0 0 0
..$ Bryan: int [1:10] 0 0 0 0 0 0 1 1 0 0
..$ Wendy: int [1:10] 0 0 0 0 0 1 0 1 0 0
..$ Aude : int [1:10] 0 0 0 0 0 1 1 0 0 0
..$ Mona : int [1:10] 0 0 0 0 0 0 0 0 0 1
..$ Bruce: int [1:10] 0 0 0 0 0 0 0 0 1 0

Details

This dataset is a small dataset containing 10 individuals in two non-overlapping locations (as cities), created solely to illustrate the functions of the Blaunet package. Affiliation variables contains membership in a liberal or conservative organization (or both). Demographic data contains variables of age, education, and income. Individual-level network data is also available in both adjacency matrix and edge list formats.

Source

Data was created by the Blaunet team. All data is fictional and was created with a random number generator.

Examples

```
data(BSANet)
```

`dyadic` *Computes dyadic Blau status measures.*

Description

Computes six dyadic measures (computed for each edge in the edgelist). They are Euclidean Distance, Mahalanobis Distance, Co-nichers, Co-outsiders, Straddlers, Spanners.

Usage

```
dyadic(blauObj, dev.range, ecologies.off = FALSE, m.dist = TRUE)
```

Arguments

blauObj	An object of class blau initialized with the function blau. Individuals will automatically be placed in niches with the function niches if this has not been done manually.
dev.range	When creating niches, indicates standard deviation around the mean in each dimension to include in niche. A larger value will make niches larger and therefore include more individuals.
ecologies.off	Defaults to FALSE. If set to TRUE, treats all individuals as in the same ecology, even if ecology identifiers (ecology.ids) have been specified. Will call the niches function and overwrite its output even if it has been manually called by the user.
m.dist	Defaults to FALSE. If TRUE, Mahalanobis distance will be computed in addition to Euclidean distance.

Details

All measures are at the edge-level of analysis and characterize ties/dyads rather than individuals/nodes.

Euclidean Distance: is the euclidean distance between two nodes in the edge

Mahalanobis Distance: is euclidean distance standardized by the unit of measurement

Co-nichers: are ties that are located within the same niche. A 0 indicates that both nodes of the tie do not share any niche in common. A 1 indicates that both nodes of the tie belong to only one common niche. A number of 2 or more indicates that the nodes of the tie share more than 1 common niche.

Co-outsiders: is an indicator variable that denotes ties where both nodes do not belong to any niche.

Straddlers: is an indicator variable that denotes ties where one node belongs to at least one niche and the other node does not belong to any niche.

Spanners: is an indicator variable that denotes ties where one node belongs to one niche and the other node belongs to a different niche.

Value

Returns a matrix stored in object$dyadic that contains eight columns. The first two contain the ego and alter for the edge, and the next six contain the dyadic measures.

Examples

```
data(BSANet)
square.data <- BSANet$square.data
el <- BSANet$el #edgelist
adj <- BSANet$adj #adjacency matrix

b <- blau(square.data, node.ids = 'person', ecology.ids = 'city', graph = el)
#implicitly a directed edgelist
#will automatically compute niches
b <- dyadic(b, dev.range = rep(1.5, 3)) # 3 is the number of dimensions
```

```
#adjacency matricies are also OK
b <- blau(square.data, node.ids = 'person', ecology.ids = 'city', graph = adj)
#will automatically compute niches
b <- dyadic(b, dev.range = rep(1.5, 3)) # 3 is the number of dimensions
```

`ecology.summary` *Summarizes the distribution of individuals across niches.*

Description

Outputs a niche-by-niche matrix for each ecology where each cell, [A,B], corresponds to the number of individuals who are in both niche A and niche B. Diagonal elements, [A,A], contain the number of individuals exclusively in ecology A.

Usage

```
ecology.summary(blauObj, percent = FALSE)
```

Arguments

`blauObj`	An object of class blau initialized with the function `blau`. The function niches must have been called previously by the user.
`percent`	Defaults to `FALSE`. If `TRUE`, expresses output in terms of percentages (relative to the total number of people in an ecology).

Details

The purpose of this function is to give the analyst a sense of the structure of the ecology. At one extreme the niches may be completely overlapping and at the other the niches may be completely separate. The former condition suggests fierce competition, whereas the later suggests strong localization. A situation in the middle indicates a particularly interesting competitive situation.

Value

Returns an object of class `data.frame` that contains a niche-by-niche matrix for each ecology. Ecology and niche identifiers are presented in the first two columns, followed by the niche-by-niche matrix. Matrices are stacked vertically.

Examples

```
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
b <- niches(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions
ecology.summary <- ecology.summary(b)
```

export.dyadic *Outputs dyadic level measures computed by the user*

Description

Takes any output from dyadic blau functions (dyadic) and presents it in one matrix for further analysis outside of Blaunet or output to disk.

Usage

```
export.dyadic(blauObj)
```

Arguments

`blauObj`

Details

This function is useful to manipulate the computed dyadic Blau statuses and analyze them outside the Blaunet program, either within R or by exporting them to another statistical package.

Value

Returns an object of class data.frame that includes all output from function dyadic. The user must manually call the `dyadic` function prior to calling `export.dyadic`.

Examples

```
data(BSANet)
square.data <- BSANet$square.data
el <- BSANet$el
b <- blau(square.data, node.ids = 'person', ecology.ids = 'city', graph = el)
b <- dyadic(b, dev.range = rep(1.5, 3)) # 3 is the number of dimensions
export.dyadic(b)
```

export.nodal *Outputs all node-level measures computed by the user*

Description

Takes any output from nodal functions (`niches`, `nodal.local`, `nodal.global`, or `nodal.network`) and presents it in one matrix for further analysis outside of Blaunet or output to disk.

Usage

```
export.nodal(blauObj, niches = TRUE)
```

Arguments

blauObj	An object of class blau initialized with the function blau. User must have called at least one of the following functions: niches, nodal.local, nodal.global, or nodal.network. Otherwise, there is nothing to export.
niches	Defaults to TRUE. If TRUE, includes the full matrix of individuals located in niches as part of the output. This information may not always be desirable for output, and setting this to FALSE will prevent the full matrix of individuals located in niches from being included.

Details

This function is useful to manipulate the computed nodal Blau statuses and analyze them outside the Blaunet program, either within R or by exporting them to another statistical package.

Value

Returns an object of class data.frame that includes all output from niches, nodal.local, nodal.global, or nodal.network previously computed by the user. The user must manually call one or more of these four functions prior to calling export.nodal. Columns are labeled with the name of the measure. Row names are node names provided with the blau function.

Examples

```
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
#blau object will store whatever you compute
b <- nodal.global(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions
b <- nodal.local(b, dev.range = rep(1.5, 10), focal.niche = 'grppta')
# 10 is the number of dimensions in the command line above
export.nodal(b) #will export global and local
export.nodal(b, niches = FALSE) #suppress niche export
```

---

gss74_87 *Example attribute dataset from General Social Survey (GSS).*

Description

The attribute dataset contains 13,865 respondents surveyed between 1974 and 1987 and their attribute and group affiliation information. The attribute data includes education level and occupational prestige for constructing Blau space, along with group affiliation data in 16 types of organizations

Usage

```
data(gss74_87)
```

Source

This dataset isexcerpted from the General Social Survey, 1972-2012, available from both the NORC website(http://gss.norc.org/get-the-data/) and the Interuniversity Consortium for Political and Social Research (ICPSR) website (http://www.icpsr.umich.edu/icpsrweb/NACJD/studies/34802).

Examples

```
data(gss74_87)
```

`niche.summary` *Summarizes information on each membership organization*

Description

Aggregates information on each organization and returns five summary measures—the number of individuals in the organization, the number of individuals in the organization's niche, the number in the organization but not in the niche, the number of individuals exclusively in the niche, and the number of individuals who overlap with other niches.

Usage

```
niche.summary(blauObj)
```

Arguments

`blauObj`	An object of class blau initialized with the function `blau`. The function `niches` must have been called previously by the user.

Details

The purpose of this function is to get a better sense of how the niches are organized within each ecology.

The rows represent niches. The number of rows corresponds to the number of niches.

Computes five measures:

OrgMem: computes how many people are in each of the organizations that structure the niche.

NicheMem: computes to how many people are in each of the niches

NichExc: computes how many people are exclusive to that niche and only to that niche (not in any other niche);

NicheOvr: computes how many people are in 2 or more niches

MemExc: computes how many people are in the organization but not in the organization's niche. This happens because they are outside the demographic boundaries of the set standard deviations that are used to define the niche.

Value

Returns an object of class `data.frame` that contains the five summary measures as columns, along with two columns identifying the ecology and the niche corresponding to the information provided.

Examples

```
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
b <- niches(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions
niche.summary <- niche.summary(b)
```

niches *Locate individuals in organizational niches in Blau space*

Description

Takes an object created with function blau and locates individuals in organizational niches. Automatically handles multiple ecologies by performing niche calculations separately for each ecology.

Usage

```
niches(blauObj, dev.range, ecologies.off = FALSE)
```

Arguments

blauObj	An object created with the function blau.
dev.range	When creating niches, indicates standard deviation around the mean in each dimension to include in niche. A larger value will make niches larger and therefore include more individuals.
ecologies.off	Defaults to FALSE. If set to TRUE, treats all individuals as in the same ecology, even if ecology identifiers (ecology.ids) have been specified.

Details

Creates niches for each organization within each ecology. Niches are hypercubes in euclidean space that define the area a member-seeking organization is likely to recruit members from. Consider a given organization: its members have various traits such as age, income, and work experience. We find the mean point in n-dimensional space for an organization. For each dimension, we extend dev.range*SD out from this mean in both positive and negative directions. Doing this for each dimension defines the niche for the given organization within the given ecology.

Once we define niche dimensions, we examine the demographic information for each individual and indicate whether the individual is inside the niche. This information is stored in the matrix blauObject$isInNiche.

Each ecology may have different niche boundaries for a given organization. This heterogeneity is important for examining how different organizations recruit in different ecologies (e.g. cities, schools, etc.).

Value

obj	Adds topbounds, lowbounds, and isInNiche to the Blau object.

Examples

```
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
#compute niches
b0 <- niches(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions
#change std. dev used for calculating niches
b1 <- niches(b, dev.range = rep(1.0, 10)) # 10 is the number of dimensions
#treat everyone as members of one ecology
b2 <- niches(b, dev.range = rep(1.0, 10), ecologies.off = TRUE) # 10 is the number of dimensions
```

nodal.global *Compute Blau statues for individuals across all niches*

Description

Computes three measures—total number of organizations the individual occupies, total number of niches a person occupies, and a string indicating the niches an individual occupies.

Usage

```
nodal.global(blauObj, dev.range, ecologies.off = FALSE)
```

Arguments

blauObj	An object of class blau initialized with the function blau. Individuals will automatically be placed in niches with the function niches if this has not been done manually.
dev.range	When creating niches, indicates standard deviation around the mean in each dimension to include in niche. A larger value will make niches larger and therefore include more individuals.
ecologies.off	Defaults to FALSE. If set to TRUE, treats all individuals as in the same ecology, even if ecology identifiers (ecology.ids) have been specified. Will call the niches function and overwrite its output even if it has been manually called by the user.

Details

The three measures computed provide information on each individual across all niches.

TotalOrgs: Total number of organizations the person is in. The lower bound is 0 and the upper bound is the maximum number of organizations in the ecology.

Nicher: Provides information on how many niches the person is in or how many organizations are competing for that individual (irrespective of actual membership). The value of 0 indicates that a person in an outsider (is not in any niche). The value of 1 indicates that the person is an exclusive nicher, suggesting an organizational monopoly. A value of 2 or more indicates that the person is a manifolder or belongs to multiple niches and is a potential recruit for multiple organizations.

Niches: a string indicating which niches the individual belongs to and containing niche column numbers separated by spaces. This provides an at-a-glance format and may be easily split up with string parsing functions.

Value

A matrix object stored in object$nodalGlobal that contains the three measures in three columns. Row names are node names provided with the blau function.

Examples

```
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
#will automatically call niches
b0 <- nodal.global(b, dev.range = rep(1.5, 10)) # 10 is the number of dimensions
```

```
#treat all individuals as in same ecology
b1 <- nodal.global(b, dev.range = rep(1.5, 10), ecologies.off = TRUE)
# 10 is the number of dimensions in the command line above
```

`nodal.local` *Computes Blau statuses for individuals with respect to a primary membership*

Description

Computes two measures: whether an individual is within the niche of the primary membership specified, and whether the individual is a member of the primary membership but outside of that membership's niche.

Usage

```
nodal.local(blauObj, focal.niche = NULL, dev.range, ecologies.off = FALSE)
```

Arguments

`blauObj`	An object of class blau initialized with the function blau. Individuals will automatically be placed in niches with the function `niches` if this has not been done manually.
`focal.niche`	Specifies a focial niche by name of the organization.
`dev.range`	When creating niches, indicates standard deviation around the mean in each dimension to include in niche. A larger value will make niches larger and therefore include more individuals.
`ecologies.off`	Defaults to `FALSE`. If set to `TRUE`, treats all individuals as in the same ecology, even if ecology identifiers (`ecology.ids`) have been specified. Will call the niches function and overwrite its output even if it has been manually called by the user.

Details

This provides information about the focal membership for each individual.

FocNicher: Provides information on how many niches a person is in including the focal niche (it may also be accessed manually by `object$isInNiche`). A value of 0 indicates a person is not in any niche. A value of 1 indicates the person is only in the focal niche. A value of more than 1 indicates the person is in more than the focal niche. ((it may be more intuitive if we use two columns: in focal niche and total # niches))

MemNotNiche: indicates whether an individual is a member of the focal membership but outside of the focal niche. This indicates that an individual is atypical compared to other group members.

Value

A matrix stored in `object$nodalLocal` with two columns, each containing one of the two measures. Row names are node names provided with the `blau` function.

Examples

```
data(TwoCities)
b <- blau(TwoCities, node.ids = 'respID', ecology.ids = 'samp')
#will automatically compute niches
b0 <- nodal.local(b, dev.range = rep(1.5, 10), focal.niche = 'grppta')
# 10 is the number of dimensions in the command line above
```

`nodal.network` *Computes nodal spanners*

Description

Computes whether an individual spans (has connections in) another niche and the number of other niches each individual spans to.

Usage

```
nodal.network(blauObj, dev.range, ecologies.off = FALSE)
```

Arguments

`blauObj`	An object of class blau initialized with the function `blau`. Individuals will automatically be placed in niches with the function `niches` if this has not been done manually.
`dev.range`	When creating niches, indicates standard deviation around the mean in each dimension to include in niche. A larger value will make niches larger and therefore include more individuals.
`ecologies.off`	Defaults to `FALSE`. If set to `TRUE`, treats all individuals as in the same ecology, even if ecology identifiers (`ecology.ids`) have been specified. Will call the `niches` function and overwrite its output even if it has been manually called by the user.

Details

This function requires sociometric network data and identifies ties that bridge niches. The preferred format is a named edgelist, although an adjacency matrix with properly named rows and columns is also acceptable. Network information will be matched to organization and demographic information by these names.

For individuals `i` and `j` in niches `alpha` and `beta`, if individual `i` is in niche `alpha` and individual `j` is in niche `beta`, and `i` and `j` have an edge between them, then `i` spans to niche `beta` and `j` spans to niche `alpha`. However, if `i` is in both niche `alpha` and niche `beta`, `i` would not span to `beta` because `i` is already in niche `beta`.

Value

A matrix stored in `object$nodalNetwork` with two columns holding the two measures. Row names are node names provided with the `blau` function.

Examples

```
data(BSANet)
el <- BSANet$el
square.data <- BSANet$square.data
b <- blau(square.data, node.ids = 'person', ecology.ids = 'city', graph = el)
#will automatically compute niches
b <- nodal.network(b, dev.range = rep(1.5, 3)) # 3 is the number of dimensions
```

---

schlattr *Example attribute dataset from Add Health.*

Description

The attribute dataset contains 100 respondents with 20 individual characteristics and group affiliation data in 5 kinds of organizations.

Usage

```
data(schlattr)
```

Source

The attribute data of 100 respondents are randomly drawn from the wave I public use data of National Longitudinal Study of Adolescent to Adult Health (Add Health), available from the Interuniversity Consortium for Political and Social Research (ICPSR) website, data number 21600.

Examples

```
data(schlattr)
```

---

schlnet *Example network dataset simulated based on the attribute dataset from Add Health.*

Description

The network dataset contains an adjacent matrix of 100 individuals.

Usage

```
data(schlnet)
```

Source

The network dataset is not real. It is simulated with an Exponential family random graph (ERG) model based on the attribute dataset from Add Health

Examples

```
data(schlnet)
```

TwoCities *Example social capital dataset*

Description

The dataset contains 1,008 individuals from two cities in the United States: Bismarck, North Dakota and Grand Rapids, Michigan. The attributes include 4 individual characteristics for constructing Blau parameters, and the group affiliation data include membership in 18 voluntary organizations. There are also six other assorted variables.

Usage

```
data(TwoCities)
```

Format

A data frame with 1008 observations on the following 30 variables.

`respID` a numeric vector

`samp` a numeric vector

`age` a numeric vector

`income` a numeric vector

`educ` a numeric vector

`wrktime` a numeric vector

`grprel` a numeric vector

`grpsport` a numeric vector

`grpyouth` a numeric vector

`grppta` a numeric vector

`grpvet` a numeric vector

`grpnei` a numeric vector

`grpeld` a numeric vector

`grpsoc` a numeric vector

`grplab` a numeric vector

`grpprof` a numeric vector

`grpfrat` a numeric vector

`grpeth` a numeric vector

`grppol` a numeric vector

`grpart` a numeric vector

`grphob` a numeric vector

`grpself` a numeric vector

`grpwww` a numeric vector

`grpothr` a numeric vector

`gender` a numeric vector

`race` a numeric vector

`ideo` a numeric vector

`trust` a numeric vector

`friends` a numeric vector

`divrsity` a numeric vector

Source

The TwoCities dataset excerpted, with permission, from the Social Capital Benchmark Survey, which was collected by Professor Robert D. Putnam of the Saguaro Seminar: Civic Engagement in America, a project of the John F. Kennedy School of Government at Harvard University and numerous community foundations nationwide, and made available through the Roper Center for Public Opinion Research.

Examples

```
data(TwoCities)
```

Index