Package ‘netCoin’

February 4, 2020

Type Package
Version 1.1.24
Date 2020-02-03
Title Interactive Analytic Networks
Description Create interactive analytic networks. It joins the data analysis power of R to obtain coincidences, co-occurrences and correlations, and the visualization libraries of 'JavaScript' in one package.
License CC BY-NC-SA 4.0
Depends R (>= 3.0.0)
Imports igraph (>= 1.0.1), Matrix (>= 1.2-4), haven (>= 1.1.0)
Suggests knitr, rmarkdown, lavaan, shiny
VignetteBuilder knitr
NeedsCompilation no
Author Modesto Escobar <modesto@usal.es>,
David Barrios <metal@usal.es>,
Carlos Prieto <cprietos@usal.es>,
Luis Martinez-Uribe <lmartinez@usal.es>.
Maintainer Modesto Escobar <modesto@usal.es>
Repository CRAN
Date/Publication 2020-02-03 23:10:05 UTC

R topics documented:

  netCoin-package .................................................. 2
  allNet ............................................................. 4
  asNodes ............................................................ 6
  barCoin ............................................................. 7
  coin ................................................................. 9
  coocur .............................................................. 10
  dice ................................................................. 11
  dichotomize ......................................................... 12
netCoin-package

The netCoin package.

Description

Create interactive networked coincidences. It joins the data analysis power of \( \mathbb{R} \) to study coincidences and the visualization libraries of JavaScript in one package.

Details

Coincidence analysis detects what events, characters, objects, attributes, or characteristics tend to occur together within certain limits.

These given limits are call scenarios (\( S \)) and are considered to be the units of analysis, and as such they have to be placed in the rows of a matrix or data.frame.

In each \( i \) scenario, a series of \( J \) events \( X_j \), which are to be represented as dichotomous variables \( X_j \) in columns, may occur (1) or may not occur (0). Scenarios and events constitute an incidence matrix (\( I \)).

<table>
<thead>
<tr>
<th>netCoin-package</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>distant</td>
<td>14</td>
</tr>
<tr>
<td>edgeList</td>
<td>15</td>
</tr>
<tr>
<td>ess</td>
<td>16</td>
</tr>
<tr>
<td>events</td>
<td>17</td>
</tr>
<tr>
<td>expectedList</td>
<td>18</td>
</tr>
<tr>
<td>families</td>
<td>18</td>
</tr>
<tr>
<td>finches</td>
<td>19</td>
</tr>
<tr>
<td>fromIgraph</td>
<td>20</td>
</tr>
<tr>
<td>Galapagos</td>
<td>21</td>
</tr>
<tr>
<td>glmCoin</td>
<td>22</td>
</tr>
<tr>
<td>incTime</td>
<td>23</td>
</tr>
<tr>
<td>layoutCircle</td>
<td>23</td>
</tr>
<tr>
<td>layoutGrid</td>
<td>24</td>
</tr>
<tr>
<td>links</td>
<td>25</td>
</tr>
<tr>
<td>lower</td>
<td>26</td>
</tr>
<tr>
<td>mobileEdges</td>
<td>27</td>
</tr>
<tr>
<td>multigraphCreate</td>
<td>28</td>
</tr>
<tr>
<td>netCoin</td>
<td>29</td>
</tr>
<tr>
<td>netCorr</td>
<td>30</td>
</tr>
<tr>
<td>pathCoin</td>
<td>31</td>
</tr>
<tr>
<td>propCoin</td>
<td>32</td>
</tr>
<tr>
<td>savePajek</td>
<td>33</td>
</tr>
<tr>
<td>shinyCoin</td>
<td>34</td>
</tr>
<tr>
<td>sim</td>
<td>35</td>
</tr>
<tr>
<td>sociologists</td>
<td>36</td>
</tr>
<tr>
<td>surCoin</td>
<td>37</td>
</tr>
<tr>
<td>timeCoin</td>
<td>38</td>
</tr>
<tr>
<td>telgraph</td>
<td>39</td>
</tr>
</tbody>
</table>

Index 42
Incidence matrix

\[
\begin{array}{cccccc}
X_1 & X_2 & X_3 & \ldots & X_J \\
S_1 & 0 & 1 & 0 & \ldots & 1 \\
S_2 & 1 & 0 & 1 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
S_n & 1 & 1 & 0 & \ldots & 1 \\
\end{array}
\]

From this incidences matrix, a coincidence (C) matrix can be obtained with the function `coin`. In this matrix the main diagonal represents frequencies of \(X_j\), while the others elements are number of coincidences between two events.

Coincidence matrix

\[
\begin{array}{cccccc}
X_1 & X_2 & X_3 & \ldots & X_J \\
X_1 & 2 & 1 & 1 & \ldots & 1 \\
X_2 & 1 & 2 & 0 & \ldots & 2 \\
X_3 & 1 & 0 & 1 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
X_J & 1 & 2 & 0 & \ldots & 2 \\
\end{array}
\]

Once there is a `coin` object, a similarity matrix can be obtained. Similarity matrices available in netCoin are:

- Matching (m), Rogers \& Tanimoto (t) Gower (g) Sneath (s) and Anderberg (and).
- Jaccard (j), dice (d), antiDice (a), Ochiai (o) and Kulczynski (k).
- Hamann (ham), Yule (y), Pearson (p), odds ratio (od) and Russell (r).

Other measures that can be obtained from `coin` are:

- Relative frequencies (x), conditional frequencies (i) coincidence degree (cc) and probable degree of coincidence (cp).
- Haberman (h) and Z value of Haberman (z)

To obtain similarity and other measures matrices, the function `sim` elaborates a list of them.

Similarity matrix

\[
\begin{array}{cccccc}
X_1 & X_2 & X_3 & \ldots & X_J \\
X_1 & 1.73 & -.87 & .87 & \ldots & -.87 \\
X_2 & -.87 & 1.73 & -1.73 & \ldots & 1.73 \\
X_3 & .87 & -1.73 & 1.73 & \ldots & -1.73 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
X_J & -.87 & 1.73 & -1.73 & \ldots & 1.73 \\
\end{array}
\]

`edgelist` makes a collection of edges composed by a list of similarity measures whenever a cri-
allNet (generally p(Z)<.50) is met.

**Edge list**

<table>
<thead>
<tr>
<th>source</th>
<th>target</th>
<th>Haberman</th>
<th>P(z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X1</td>
<td>X3</td>
<td>0.8660254</td>
</tr>
<tr>
<td>2</td>
<td>X2</td>
<td>X4</td>
<td>1.7320508</td>
</tr>
</tbody>
</table>

In order to make a graph, two data frames are needed: a nodes data frames with names and other nodes attributes (see `asNodes`) and an edge data frame (see `edgeList`). For more information go to [netCoin](https://sociocav.usal.es/blog/modesto-escobar/).

**Author**

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

**References**


---

**allNet**

*Networked coincidences from incidences data.*

**Description**

allNet produces a network object of coincidences from a data frame or a matrix with dichotomous values.

**Usage**

allNet(incidences, weight = NULL, subsample = FALSE, pairwise = FALSE, minimum=1, maximum = nrow(incidences), sort = FALSE, decreasing = TRUE, frequency = FALSE, percentages = TRUE, procedures = "Haberman", criteria = "Z", Bonferroni = FALSE, support = -Inf, minL = -Inf, maxL = Inf, directed = FALSE, diagonal = FALSE, sortL = NULL, decreasingL = TRUE, igraph = FALSE, dir=NULL, ...)

**Arguments**

- **incidences**: an incidence matrix or data frame with only 0/1 variables.
- **weight**: a vector of weights. Optimal for data framed tables.
- **subsample**: restrict the analysis to scenarios with at least one event.
- **pairwise**: Pairwise mode of handling missing values if TRUE. Listwise by default.
- **minimum**: minimum frequency to be considered.
- **maximum**: maximum frequency to be considered.
- **sort**: sort the coincidence matrix according to frequency of events.
- **decreasing**: decreasing or increasing sort of the matrix.
- **frequency**: a logical value true if frequencies are to be shown. Default = FALSE.
- **percentages**: a logical value true if percentages are to be shown. Default = TRUE.
- **procedures**: a vector of statistics of similarity. See below.
- **criteria**: statistic to be use for selection criteria.
- **Bonferroni**: Bonferroni criterium of the signification test.
- **support**: minimum value of the frequency of the coincidence to be edged.
- **minL**: minimum value of the statistic to include the edge in the list.
- **maxL**: maximum value of the statistic to include the edge in the list.
- **directed**: includes same edges only once.
- **diagonal**: includes auto-links.
- **sortL**: sort the list according to the values of a statistic. See below.
- **decreasingL**: order in a decreasing way.
- **igraph**: Produces an igraph object instead of a netCoin object if TRUE.
- **dir**: a "character" string representing the directory where the web files will be saved.
- **...**: Any netCoin argument.

**Details**

Possible measures in procedures are

- Frequencies (f), Relative frequencies (x), Conditional frequencies (i), Coincidence degree (cc), Probable degree (cp),
- Expected (e), Confidence interval (con)
- Matching (m), Rogers & Tanimoto (t), Gower (g), Sneath (s), Anderberg (and),
- Jaccard (j), Dice (d), antiDice (a), Ochiai (o), Kulczynski (k),
- Hamann (ham), Yule (y), Pearson (p), odds ratio (od), Rusell (r),
- Haberman (h), Z value of Haberman (z),
- Hypergeometric p greater value (hyp).
- Convert a matrix into an edge list (shape).
asNodes

Value

This function creates a netCoin object (or igraph) and, if stated, a folder in the computer with an
HTML document named index.html which contains the produced graph. This file can be directly
opened with your browser and sent to a web server to work properly.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See
https://sociocav.usal.es/blog/modesto-escobar/

Examples

# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women",
                        "Man; Man", "Undet.; Women; Man"))
data <- dichotomize(frame, "A", sep = "; ")[2:4]
allNet(data) # network object

---

asNodes

Nodes data frame.

Description

Nodes data frame from either an edge list or a coin object.

Usage

asNodes(C, frequency = TRUE, percentages = FALSE, language = c("en", "es", "ca"))

Arguments

C has to be an edge list or, better, a coin object.
frequency add frequency of nodes
percentages add nodes percentages
language a character vector (es=spanish; en=english; ca=catalan).

Value

A data frame with nodes' names and their frequency and/or percentages if the input is a coin object

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See
https://sociocav.usal.es/blog/modesto-escobar/
Examples

# From a random incidence matrix I(25X4)
I <- matrix(rbinom(100, 1, .5), nrow = 25, ncol = 4,
dimnames = list(NULL, c("A", "B", "C", "D")))
C <- coin(I)
asNodes(C)

Description

barCoin produces a barCoin object.

Usage

barCoin(data, variables = colnames(data), commonlabel = NULL,
dichotomies = c("_all","_none"), valueDicho = 1, weight = NULL,
subsample = FALSE, sort = NULL, decreasing = TRUE, nodes = NULL,
name = NULL, select = NULL, scalebar = FALSE, note = NULL,
label = NULL, text = NULL, color = NULL, defaultColor = "#1f77b4",
expected = FALSE, confidence = FALSE, level = .95, significance = FALSE,
minimum = 1, maximum = nrow(data), percentages = FALSE,
criteria = c("Z","hyp"), Bonferroni = FALSE,
support = 1, minL = -Inf, maxL = 1,
language = c("en","es","ca"), cex = 1.0, dir = NULL)

Arguments

data a data frame
variables a vector of variables included in the previous data frame
commonlabel a vector of variables whose names are to be included in nodes labels
dichotomies a vector of dichotomous variables to appear as just one categorie
valueDicho value to be selected for dichotomous variables. Default is 1
weight a vector of weights. Optimal for data.framed tables.
subsample restrict the analysis to scenarios with at least one event.
sort name of the vector in the nodes data frame to order the graph.
decreasing decreasing or increasing sort of the graph order.
nodes a data frame with at least two vectors of names and incidences.
name name of the vector with names in the nodes data frame.
select Name of the event (in nodes name column) to start the visualization.
scalebar Should the bars fill the screen height? Default = FALSE.
note lower title of the graph.
label  name of the vector with labels in the nodes data frame.
text   name of the vector with html text in the nodes data frame.
color  name of the vector with color variable in the nodes data frame.
defaultColor a character vector giving a valid html color.
expected name of the vector with expected coincidences in the links data frame.
confidence name of the vector with confidence interval in the links data frame.
level   confidence level
significance name of the vector with significance in the links data frame.
minimum minimum frequency to be considered.
maximum maximum frequency to be considered.
percentages a logical value true if percentages are to be shown. Default = TRUE.
criteria statistic to be use for selection criteria.
Bonferroni Bonferroni criterium of the signification test.
support minimum value of the frequency of the coincidence to be edged.
minL    minimum value of the statistic to include the edge in the list.
maxL    maximum value of the statistic to include the edge in the list.
language a character vector (es=spanish; en=english; ca=catalan).
cex     number indicating the amount by which plotting text should be scaled relative to the default. Default = 1.
dir     a "character" string representing the directory where the web files will be saved.

Value

Object of class barCoin.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women", "Man; Man", "Undet.; Women; Man"))
data <- dichotomize(frame, "A", sep = "; ")[2:4]
barCoin(data,dichotomies="_all")
barCoin(data,dichotomies="_all",confidence=TRUE,percentages=TRUE)
Description

A coincidence object consists of a list with two elements: 1) the number of scenarios ($n$), and 2) a coincidence matrix of events, whose main diagonal figures are the frequency of events and outside this diagonal there are conjoint frequencies of these events ($f$)

Usage

```
coin(incidences, minimum = 1, maximum = nrow(incidences),
     sort = FALSE, decreasing = TRUE,
     total = FALSE, subsample = FALSE,
     weight = NULL, pairwise = FALSE)
```

Arguments

- `incidences`: an incidence matrix or data frame with only 0/1 variables
- `minimum`: minimum frequency to be considered
- `maximum`: maximum frequency to be considered
- `sort`: sort the coincidence matrix according to frequency of events
- `decreasing`: decreasing or increasing sort of the matrix
- `total`: add one first row and column with total
- `subsample`: restrict the analysis to scenarios with at least one event
- `weight`: a vector of weights. Optimal for data.framed tables
- `pairwise`: Pairwise mode of handling missing values if TRUE. Listwise by default.

Details

Produce a matrix of coincidences from a matrix of incidences.

Value

An object of `coin` class

- `n`: Number of scenarios (rows of the incidence matrix)
- `f`: Coincidence matrix

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/
Examples

```r
## Random incidence matrix: 25 scenarios, 4 events.
I <- matrix(rbinom(100, 1, .5), nrow = 25, ncol = 4,
            dimnames = list(NULL, c("A", "B", "C", "D")))
coin(I, sort = TRUE)

## Hair by Eye by Sex table from M. Friendly (2000)
data(HairEyeColor)
H<-as.data.frame(HairEyeColor)
W<-H$Freq
I<-dichotomize(H,c("Hair","Eye","Sex"),add=FALSE)
coin(I,w=W)
```

## Coocurrence matrix.

Description

A coocurrence object consists of a matrix with the number of occurrences in its main diagonal and the number of cooccurrences outside this diagonal. Besides, this object has two attributes: 1) \( n \) is the total of the sum of the occurrences in each row.2) \( m \) is the sum of the maximum number of occurrences in each row.

Usage

```r
coocur(occurrences, minimum = 1, maximum = Inf,
       sort = FALSE, decreasing = TRUE)
```

Arguments

- `occurrences`: an occurrence matrix or data frame
- `minimum`: minimum frequency to be considered
- `maximum`: maximum frequency to be considered
- `sort`: sort the coincidence matrix according to frequency of events
- `decreasing`: decreasing or increasing sort of the matrix

Details

Produce a matrix of cooccurrences from a matrix of occurrences.

Value

An object of cooc class with a coocurrence matrix. It has two attributes:

- \( n \): Total sum of occurrences
- \( m \): Sum of maximum occurrences in each row of the occurrence matrix
**Author(s)**
Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See [https://sociocav.usal.es/blog/modesto-escobar/](https://sociocav.usal.es/blog/modesto-escobar/)

**Examples**
```r
## Tossing two coins five times.
D<-data.frame(Head=c(2,1,1,0,2),Tail=c(0,1,1,2,0))
coocur(D)
```

---

**Description**
Data frame with events as result.

**Usage**
```r
data("dice")
```

**Format**
A data frame with 100 observations (scenarios) on the following 11 variables (events):
- `dice`: a numeric vector, representing dice results
- `1`: a dichotomous vector of the elemental event "1"
- `2`: a dichotomous vector of the elemental event "2"
- `3`: a dichotomous vector of the elemental event "3"
- `4`: a dichotomous vector of the elemental event "4"
- `5`: a dichotomous vector of the elemental event "5"
- `6`: a dichotomous vector of the elemental event "6"
- `odd`: a dichotomous vector of odd events
- `even`: a dichotomous vector of even events
- `small`: a dichotomous vector of small number events
- `large`: a dichotomous vector of large number events

**Source**
Random extraction via `sample(1:6,100,replace=TRUE)`

**References**
See `events`.
Examples

data(dice)
head(dice,10)

dichotomize

Dichotomize.

Description

This converts factor(s) or character(s) column(s) of a data frame into a set of dichotomous columns. Their names will correspond to the labels or text of every category.

Usage

dichotomize(data, variables,
    sep = "", min = 1, length = 0, values = NULL,
    sparse = FALSE, add = TRUE, sort = TRUE, nas = "None")

Arguments

data a data frame with a factor or textual column which can be simple (only one value for each scenario) or multiple if components are delimited with a separator.
variables vector of column names that have to be converted into dichotomous vectors.
sep vector of characters used to divide columns with multiple events. If this separator is "", every unique cell of every column is converted into a dichotomous data frame’s column.
min convert to dichotomous vectors only label or text that has a frequency less or equal to the value of this parameter. If the value of min is between 0 and 1, its value is interpreted as a percentage
length maximum number of dichotomous columns generated for every variable
values vector of labels or texts selected to their conversion to dichotomous columns
sparse produce a sparse matrix instead of a data.frame
add add the new columns to the input data.frame
sort order the new columns by their frequencies
nas variable name to convert the NA values of the set of variables

Value

A data frame composed by the original plus the added dichotomous columns.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca, and Luis Martinez Uribe, Fundacion Juan March. See https://sociocav.usal.es/blog/modesto-escobar/
Examples

# A character column
frame1 <- data.frame(A = c("Man", "Women", "Man", "Undet."))
dichotomize(frame1, "A", sep = "; ")

# A character column (with separator)
frame2 <- data.frame(A = c("Man; Women", "Women; Women", "Man; Man", "Undet.; Women; Man"))
dichotomize(frame2, "A", sep = "; ")

# A character column and another factor column (same separator)
frame3 <- data.frame(A = c("Man; Women", "Women; Women", "Man; Man", "Undet.; Women; Man"),
                     C = factor(c(1:4), labels = c("Paris", "New York", "London; New York", "<NA>")))
dichotomize(frame3, c("A", "C"), sep = "; ")

# A set of simple character or factor (same levels) variables.
# In this case, you must use "C" separator.
frame4 <- data.frame(A = c("Man", "Women", "Man", "Undet", NA),
                     B = c("Women", "Women", "Man", "Women", NA),
                     C = c(NA, NA, NA, "Man", NA))
dichotomize(frame4, c("A", "B", "C"), sep="C")

distant

Distance matrix.

Description

Convert a similarity matrix into a distance matrix.

Usage

distant(s, t = FALSE)

Arguments

s a similarity matrix

t return the same matrix if t=FALSE

Details

For better results, use the parameter distance in sim function.

Value

A distance matrix.
Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

# From a random incidence matrix I(25X4)
I <- matrix(rbinom(100, 1, .5), nrow = 25, ncol = 4,
         dimnames = list(NULL, c("A", "B", "C", "D")))
J <- sim(I, "Jaccard")
distant(J, t = TRUE)
#Same results
sim(I, "Jaccard", distance = TRUE)

edgeList

Description

Convert a coincidence/similarity/distance matrix into an edge list form.

Usage

edgeList(data, procedures="Haberman",
         criteria="Z", level=.95, Bonferroni=FALSE,
         min=-Inf, max=Inf, support=-Inf, directed=FALSE,
         diagonal=FALSE, sort=NULL, decreasing=TRUE, pairwise=FALSE)

Arguments

data a coin object, let's say an R matrix with frequencies and an attribute (n) giving
the number of scenarios. In case of change of shape, data should be a matrix.
procedures a vector of statistics of similarity. See below.
criteria statistic to be use for selection criteria.
level confidence level
Bonferroni Bonferroni criterium of the signification test.
min minimum value of the statistic to include the edge in the list.
max maximum value of the statistic to include the edge in the list.
support minimum value of the frequency of the coincidence to be edged
directed includes same edges only once.
diagonal includes auto-links
sort sort the list according to the values of a statistic. See below
decreasing order in a decreasing way.
pairwise Pairwise mode of handling missing values if TRUE. Listwise by default.
Details

Possible measures in procedures are

- Frequencies (f), Relative frequencies (x), Conditional frequencies (i), Coincidence degree (cc), Probable degree (cp),
- Expected (e), Confidence interval (con)
- Matching (m), Rogers & Tanimoto (t), Gower (g), Sneath (s), Anderberg (and),
- Jaccard (j), Dice (d), antiDice (a), Ochiai (o), Kulczynski (k),
- Hamann (ham), Yule (y), Pearson (p), odds ratio (od), Rusell (r),
- Haberman (h), Z value of Haberman (z),
- Hypergeometric p greater value (hyp).
- Convert a matrix into an edge list (shape).

Value

A data frame in which the two first columns are source and target. The rest of the columns are the different statistics explicited in funcs parameter.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

# From a random incidence matrix I(25X4)
I<-matrix(rbinom(100,1,.5),nrow=25,ncol=4, dimnames=list(NULL,c("A","B","C","D")))
C<-coin(I)
edgelist(C)

Data: European Social Survey, Round-8.
**Format**

A data frame with 1000 cases (respondents) and 5 variables:

- **Gender**: Gender (factor vector): Female, Male.
- **Age**: Age (recoded factor vector): 15-29, 30-30, 40-49, 50-59, 60-69, 70 and +.
- **Social participation**: Social participation (factor vector): No, Yes.
- **Political participation**: Political participation (factor vector): No, Yes.
- **cweight**: cweight (numeric vector): Cases weight.

**References**


**Examples**

```r
data("ess")
head(ess,10)
```

---

**events**

*Data: Attributes of the dice events.*

**Description**

Data frame with the attributes of the events of dice.

**Usage**

```r
data("events")
```

**Format**

A data frame with 10 observations on the following 4 variables:

- **name**: a factor vector with 10 levels
- **label**: a factor vector with 10 levels
- **frequency**: a numeric vector
- **type**: a factor vector with 2 levels

**Source**

```r
data(dice); coin.dice<-coin(dice); asNodes(coin.dice)
```

**References**

See dice.
expectedList

Examples

data(events)

Expected list.

Description

Converts a coin object to a links data frame with coincidences and expected values.

Usage

expectedList(data, names = NULL, min = 1, confidence=FALSE)

Arguments

data is a coin object. See coin

names a character vector.

min minimum value of the statistic to include the edge in the list.

confidence add the confidence interval if TRUE.

Value

A links data frame with coincidences and expected values.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women", "Man; Man", "Undet.; Women; Man"))
data <- dichotomize(frame, "A", sep = "; ")[2:4]
C <- coin(data) # coincidence matrix
expectedList(C) # edge data frame
families  
*Data: Italian families in the Renaissance.*

**Description**

Data frame with the characteristics of powerful families of Renaissance Italy.

**Usage**

```r
data("families")
```

**Format**

A data frame with 16 families (rows) and 6 characteristics.

- `name`  Family's name
- `f.Marriages`  number of marriage links
- `f.Business`  number of business links
- `wealth`  wealth's index
- `priorates`  number of priorates on control
- `seat`  At least priorate

**Source**


**Examples**

```r
data("families")
head(families)
```

finches  
*Data: Finches' attributes in Galapagos islands.*

**Description**

Data frame with events as result.

**Usage**

```r
data("finches")
```
Format

A data frame with 13 observations (pinches) and 4 variables (name and characteristics):

- name: Genus and species of the finche
- frequency: number of islands where the finche can be found
- type: Genus of the finche
- species: name of the file containing the picture of the finche

References


Examples

```r
data(finches)
head(finches, 10)
```

```r
class(finches)
```

fromIgraph

Produce interactive networks from igraph objects.

Description

fromIgraph produce an interactive network from an igraph object.

Usage

```r
fromIgraph(G, ...)
```

Arguments

- `G`: an igraph object.
- `...`: Any `netCoin` argument.

Value

This function returns a `netCoin` object. If the 'dir' attribute is specified, the function creates a folder in the computer with an HTML document named index.html which contains the produced graph. This file can be directly opened with your browser and sent to a web server to work properly.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/
Data: Finches’ presence in Galapagos Islands.

Description

Data frame with absence(0) presence(1) of finches in the Galagos Islands.

Usage

data("Galapagos")

Format

A data frame with 17 localizations (islands) and 13 variables (Genus and species of the finches):

- Geospiza magnirostris
- Geospiza fortis
- Geospiza fuliginosa
- Geospiza difficilis
- Geospiza scandens
- Geospiza conirostris
- Camarhynchus psitticula
- Camarhynchus pauper
- Camarhynchus parvulus
- Platyspiza crassirostris
- Cactospiza pallida
- Cactospiza heliobates
- Certhidea olivacea

References


Examples

data(Galapagos)
head(Galapagos, 10)
Description

produces a netCoin object from a set of glm regressions.

Usage

```r
glmCoin(formulas, data, weights=NULL, pmax=.05,
         twotail=FALSE, showArrows=TRUE,
         frequency = FALSE, percentage = TRUE,
         color="variable", lwidth="z.value",
         circle= NA, language=c("en","es","ca"),
         igraph=FALSE, ...)
```

Arguments

- `formulas` A set of formulas separated, followed by the family and a return. For example:
  ```r
  model <- "counts ~ outcome + treatment, poisson counts ~ outcome, poisson"
  ```
- `data` Data frame containing the variables in the model.
- `weights` Optional vector of weights to be used in the fitting process.
- `pmax` Selection of links with Pr(>|z|) less than p (one-tail by default).
- `twotail` Logical value indicating if twotail test must be applied. Default=FALSE.
- `showArrows` a logical value true if the directional arrows are to be shown. Default = FALSE.
- `frequency` a logical value true if frequencies are to be shown. Default=FALSE.
- `percentage` a logical value true if percentages are to be shown. Default=TRUE.
- `color` Nodes’ attribute to be used for expressing color ("variable" by default).
- `lwidth` Nodes’ attribute to be used for width of arrows ("z.value" by default).
- `circle` Degree of rotation in case of fixed circled dependent variables.
- `language` Language of the graph controls.
- `igraph` Produces an igraph object instead of a netCoin object if TRUE.
- `...` Any netCoin argument.

Value

This function creates a netCoin object (or igraph) and, if stated, a folder in the computer with an HTML document named index.html which contains the produced graph. This file can be directly opened with your browser and sent to a web server to work properly.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/
## incTime

### Time incidences.

Convert a data frame with two numbers (normally a beginning year and end year) into an incidences matrix whose rows are the intermediate numbers, and whose columns are the content of the names column.

#### Usage

```r
incTime(data, name = "name", beginning = "birth", end = "death")
```

#### Arguments

- `data`: a data frame a name and two numbers.
- `name`: Column with the names (default= "name").
- `beginning`: Column with the beginning number to include (default= "birth").
- `end`: Column with the end number to include (default= "death").

#### Value

A data frame in which the two first columns are source and target. The rest of the columns are sim.=(1+threshold-real difference) and dist.=(difference between numbers)

#### Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See [https://sociocav.usal.es/blog/modesto-escobar/](https://sociocav.usal.es/blog/modesto-escobar/)

#### Examples

```r
# From sociologists data
data("sociologists")
head(incTime(sociologists))[,1:5]
```
**layoutCircle**  
*Produce a circle layout of any number of nodes.*

**Description**

`layoutCircle` produces a circle layout of any number of nodes.

**Usage**

```r
layoutCircle(N, nodes, deg=0, name=NULL)
```

**Arguments**

- **N**: a data frame of nodes.
- **nodes**: a vector specifying nodes.
- **deg**: degrees to rotate.
- **name**: name of column with node names.

**Value**

This function returns the input data frame of nodes with the resulting layout applied.

**Author(s)**

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

---

**layoutGrid**  
*Produce a layout of any number of nodes.*

**Description**

`layoutGrid` produces a grid layout of any number of nodes.

**Usage**

```r
layoutGrid(N, string, name=NULL)
```

**Arguments**

- **N**: a data frame of nodes.
- **string**: a character vector specifying grouped nodes.
- **name**: name of column with node names.
Value

This function returns the input data frame of nodes with the resulting layout applied.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Description

Data: Links between Italian families in the Renaissance.

Usage

data("links")

Format

A data frame with 36 links (rows) amongst 16 Italian families in the Renaissance.

Albizzi
Acciaiuoli
Barbadori
Bischeri
Castellani
Guadagni
Lamberteschi
Medici
Pazzi
Peruzzi
Ridolfi
Salviati
Strozzi
Tornabuoni
Ginori
Pucci

link  Type of link: marriage or business
**Source**


**Examples**

```r
data("links")
head(links)
```

**Description**

Display the lower part of a matrix with a specified number of decimals.

**Usage**

```r
lower(matrix, decimals = 3)
```

**Arguments**

- `matrix`: a symmetric similarity/distance matrix
- `decimals`: number of decimals to be displayed

**Value**

A data frame of characters.

**Author(s)**

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

**Examples**

```r
# From a random incidence matrix I(25X4)
I <- matrix(rbinom(100, 1, .5), nrow = 25, ncol = 4,
dimnames = list(NULL, c("A", "B", "C", "D")))
lower(sim(I, "Jaccard"), 2)
```
mobileEdges  

*Mobile Edges.*

**Description**

Convert a data frame with one number (normally a year) into an edge list form with those whose numbers (years) have a difference lower or equal to a quantity.

**Usage**

```r
mobileEdges(data, name = 1, number = 2, difference=0)
```

**Arguments**

- `data`: a data frame with a name and a number (year).
- `name`: Column with the names (default= first column).
- `number`: Column with the number (year) to compare (default= second column).
- `difference`: Minimum difference between numbers of every two pair of names to create the edge or link (default=15).

**Value**

A data frame in which the two first columns are source and target. The rest of the columns are `sim.=(1+threshold-real difference)` and `dist.=(difference between numbers)`

**Author(s)**

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See [https://sociocav.usal.es/blog/modesto-escobar/](https://sociocav.usal.es/blog/modesto-escobar/)

**Examples**

```r
# From a random incidence matrix I(25X4)
data("sociologists")
mobileEdges(sociologists)
```
multigraphCreate  Produce interactive multi graphs.

Description

multigraphCreate produce an interactive multi graph.

Usage

multigraphCreate(..., mode = c("default","parallel","frame"),
    frame = 0, speed = 50, dir = "MultiGraph", show = TRUE)

Arguments

  ...          coin graphs (netCoin, barCoin, timeCoin) objects or html "directories".
  mode         a string specifying the displaying mode. The "default" displays graphs one by
                one, "parallel" splits screen and "frame" allows dynamic graphs in time.
  frame        number of frame to start a dynamic network.
  speed        a percentage for frame speed in dynamic networks.
  dir          a "character" string representing the directory where the graph will be saved.
  show         a logical value true if the graph is to be shown. Default = TRUE.

Value

The function creates a folder in your computer with an HTML document named index.html which
contains the graph. This file can be directly opened with your browser.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See
https://sociocav.usal.es/blog/modesto-escobar/

Examples

## Not run:
# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women",
                          "Man; Man", "Undet.; Women; Man"))
C <- coin(data) # coincidence matrix
N <- asNodes(C) # node data frame
E <- edgeList(C,c("frequency","expected","haberman")) # edge data frame
bC <- barCoin(data,dichotomies="_all") # barCoin object
cC <- barCoin(data,dichotomies="_all",expected=TRUE) # barCoin object
nC <- netCoin(N,E) # netCoin object
multigraphCreate("Bar graph" = bC,
                  "Conditional bar graph" = cC,
"Net graph"=nC,
dir="./example") # See ./example/index.html file

## End(Not run)

```r
netCoin
```

## Description

`netCoin` produces a netCoin object of coincidences. Its input has to be two data.frames: one of attributes of events or nodes, and the other of attributes of the edges or links.

## Usage

```r
netCoin(nodes, links = NULL, tree = NULL, name = NULL, label = NULL,
        labelSize = NULL, size = NULL, color = NULL, shape = NULL,
        legend = NULL, ntext = NULL, info = NULL,
        orderA = NULL, orderD = NULL, group = NULL, community = NULL,
        lwidth = NULL, lweight = NULL, lcolor = NULL, ltext = NULL,
        nodeFilter = NULL, linkFilter = NULL, degreeFilter = NULL,
        nodeBipolar = FALSE, linkBipolar = FALSE, defaultColor = "#1f77b4",
        distance = 10, repulsion = 25, zoom = 1,
        scenarios = NULL, main = NULL, help = NULL, helpOn = FALSE, cex = 1,
        background = NULL, layout = NULL, limits = NULL, controls = 1:5,
        mode = c("network","heatmap"),
        showCoordinates = FALSE, showArrows = FALSE, showLegend = TRUE,
        showAxes = FALSE, axesLabels = NULL, language = c("en","es","ca"),
        image = NULL, imageNames = NULL, dir = NULL)
```

## Arguments

- **nodes**: a data frame with at least one vector of names.
- **links**: a data frame with at least two vectors with source and target, including names of nodes.
- **tree**: a data frame with two vectors: source and target, describing relationships between nodes.
- **name**: name of the vector with names in the nodes data frame. By default, if language="en", name is "name".
- **label**: name of the vector with labels in the nodes data frame.
- **group**: name of the vector with groups in the nodes data frame.
- **community**: algorithm to make communities: `edge_betweenness("ed")`, `fast_greedy("fa")`, `label_prop("la")`, `leiden_eigen("le")`, `louvain("lo")`, `optimal("op")`, `spinglass("sp")`, `walktrap("wa")`
- **labelSize**: name of the vector with label size in the nodes data frame.
size
name of the vector with size in the nodes data frame.

color
name of the vector with color variable in the nodes data frame.

shape
name of the vector with shape variable in the nodes data frame.

legend
name of the vector with the variable to represent as a legend in the nodes data frame.

ntext
name of the vector with html text in the nodes data frame.

info
name of the vector with information to display in a panel in the nodes data frame.

orderA
name of the vector with node ascending order in the nodes data frame.

orderD
name of the vector with node descending order in the nodes data frame.

lwidth
name of the vector with width variable in the links data frame.

lweight
name of the vector with weight variable in the links data frame.

lcolor
name of the vector with color variable in the links data frame.

ltext
name of the vector with labels in the links data frame.

nodeFilter
condition for filtering nodes.

linkFilter
condition for filtering links.

degreeFilter
numeric vector to filter the resulting network by degree.

defaultColor
a character vector giving a valid html color.

linkBipolar
a logical value that polarizes negative and positive link values in the graphical representation. Default = FALSE.

repulsion
a percentage for repulsion between nodes.

distance
a percentage for distance of links.

zoom
a number between 0.1 and 10 to start displaying zoom.

scenarios
a note showing number of scenarios.

main
upper title of the graph.

note
lower title of the graph.

help
text of the graph.

helpOn
Should the help be shown at the beginning?

background
background color or image of the graph.

layout
a matrix with two columns or an algorithm to elaborate the coordinates: davidson.harel drl("da"), circle("ci"), fruchterman.reingold("fr"), gem("ge"), grid("gr"), kamada.kawai("ka"), lgl("lg"), mds("md"), random("ra"), reingold.tilford("re"), star("sta"), sugiyama("sug")

limits
vector indicating size references to display layout, must be a numeric vector of length 4: x1, y1, x2, y2.

cex
number indicating the amount by which plotting text should be scaled relative to the default. Default = 1.
controls  a numeric vector indicating which controls will be shown. 1 = sidebar, 2 = selection buttons, 3 = tables, 4 = sliders & buttons, 5 = export buttons. NULL hide all controls, negative values deny each control and 0 deny all.

mode  a character vector indicating the graph mode allowed: network, heatmap or both (both by default).

showCoordinates  a logical value true if the coordinates are to be shown in tables. Default = FALSE.

showArrows  a logical value true if the directional arrows are to be shown. Default = FALSE.

showLegend  a logical value true if the legend is to be shown. Default = TRUE.

showAxes  a logical value true if the axes are to be shown. Default = FALSE.

axesLabels  a character vector giving the axes names.

language  a character vector (es=spanish; en=english; ca=catalan).

image  name of the vector with image files in the nodes data frame.

imageName  name of the vector with names for image files in the nodes data frame.

dir  a "character" string representing the directory where the web files will be saved.

Value

This function returns a netCoin object. If the 'dir' attribute is specified, the function creates a folder in the computer with an HTML document named index.html which contains the produced graph. This file can be directly opened with your browser and sent to a web server to work properly.

Note

nodes and links arguments can be substituted by a netCoin object to add or change options to it.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women", "Man; Man", "Undet.; Women; Man"))
data <- dichotomize(frame, "A", sep = " "; ")[2:4]
C <- coin(data) # coincidence matrix
N <- asNodes(C) # node data frame
E <- edgeList(C) # edge data frame
netCoin(N, E) # netCoin object
Description

`netCorr` produces a network object of correlations. Its input has to be at least one set of quantitative variables.

Usage

```r
netCorr(variables, weight=NULL, pairwise=FALSE,
minimum=-Inf, maximum=Inf, sort=FALSE, decreasing=TRUE,
frequency=FALSE, means=TRUE,
method="pearson", criteria="p", Bonferroni=FALSE,
minL=0, maxL=Inf,
sortL=NULL, decreasingL=TRUE,
igraph=FALSE, ...)
```

Arguments

- **variables**: a data frame with at least two quantitative variables.
- **weight**: a vector of weights. Optimal for data.framed tables
- **pairwise**: Pairwise mode of handling missing values if TRUE. Listwise by default.
- **minimum**: minimum frequency to be considered
- **maximum**: maximum frequency to be considered
- **sort**: sort the correlation matrix according to the frequency of the events
- **decreasing**: decreasing or increasing sort of the matrix
- **frequency**: a logical value true if frequencies are to be shown. Default=FALSE.
- **means**: a logical value true if means are to be shown. Default=TRUE.
- **method**: a vector of statistics of similarity. Pearson correlation by default. spearman and kendall are also possible
- **criteria**: statistic to be use for selection criteria.
- **Bonferroni**: Bonferroni criterium of the signification test.
- **minL**: minimum value of the statistic to include the edge in the list.
- **maxL**: maximum value of the statistic to include the edge in the list.
- **sortL**: sort the list according to the values of a statistic. See below
- **decreasingL**: order in a decreasing way.
- **igraph**: Produces an igraph object instead of a netCoin object if TRUE
- **...**: Any `netCoin` argument.
**Value**

The function creates a netCoin object and eventually a folder in the computer with an HTML document named index.html which contains the produced graph. This file can be directly opened with your browser and sent to a web server to work properly.

**Author(s)**

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See [https://sociocav.usal.es/blog/modesto-escobar/](https://sociocav.usal.es/blog/modesto-escobar/)

**Examples**

```r
# A character column (with separator)
data(iris)
netCorr(iris[,1:4],ltext="value",
     main="Correlations between measurements of Iris Species",
```

---

**Description**

`pathCoin` produces a netCoin object from a lavaan object, i.e., parameters of structural equation model.

**Usage**

```r
pathCoin(model, estimates=c("b","se","z","pvalue","beta"),
         fitMeasures=c("chisq", "cfi", "rmsea"), ...)
```

**Arguments**

- `model` a lavaan object.
- `estimates` A vector with at least one element amongst "b", "se", "z", "pvalue", "beta".
- `fitMeasures` Default values: "chisq", "df", "pvalue", "cfi", "rmsea"
- `...` Any netCoin argument.

**Value**

The function creates a netCoin object and eventually a folder in the computer with an HTML document named index.html which contains the produced graph. This file can be directly opened with your browser and sent to a web server to work properly.

**Author(s)**

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See [https://sociocav.usal.es/blog/modesto-escobar/](https://sociocav.usal.es/blog/modesto-escobar/)
Examples

# Classic Wheaton et al. model
library(lavaan)
lower <- '
11.834
6.947 9.364
6.819 5.091 12.532
4.783 5.028 7.495 9.986
wheaton.cov <- getCov(lower,
   names = c("anomia67", "powerless67","anomia71", "powerless71",
   "education", "sei"))
wheaton.model <- '
# latent variables
ses =~ education + sei
alien67 =~ anomia67 + powerless67
alien71 =~ anomia71 + powerless71
# regressions
alien71 ~ alien67 + ses
alien67 ~ ses
# correlated residuals
anomia67 ~~ anomia71
powerless67 ~~ powerless71
'
fit <- sem(wheaton.model, sample.cov = wheaton.cov, sample.nobs = 932)
pathCoin(fit)

propCoin

Express Coin Entries as Fraction of Marginal Table

Description

This is like 'prop.table' for 'coin' objects.

Usage

propCoin(x, margin= 0, decimals=1)

Arguments

x 'coin' object.
margin index, or vector of indices to generate margin for.
decimals integer indicating the number of decimal places to be used.

Value

Table like 'x' expressed relative to 'margin'.
Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

```r
## Random incidence matrix: 25 scenarios, 4 events.
I <- matrix(rbinom(100, 1, .5), nrow = 25, ncol = 4,
            dimnames = list(NULL, c("A", "B", "C", "D")))
C <- coin(I, sort = TRUE)
propCoin(C, 1)
```

Description

savePajek produces a .net (.paj) file from a netCoin object.

Usage

```r
savePajek(net, file="file.net", arcs=NULL, edges=NULL,
          partitions=NULL, vectors=NULL)
```

Arguments

- `net`: a netCoin object.
- `file`: The name of the file without extension. It will be .net or .paj according to the data. The default is file.net or file.paj
- `arcs`: Names of netCoin$links to be included and considered as arcs in the Pajek file.
- `edges`: Names of netCoin$links to be included and considered as edges in the Pajek file.
- `partitions`: Names of netCoin$nodes to be included and considered as partitions in the Pajek file.
- `vectors`: Names of netCoin$nodes to be included and considered as vectors in the Pajek file.

Value

The function creates a file with vertices and arcs or edges of a netCoin object. Vectors and partitions can be also included.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/
shinyCoin

Examples

## Not run:
# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women",
                         "Man; Man", "Undet.; Women; Man"))
data <- dichotomize(frame, "A", add=FALSE, sep = "; ")
graph <- allNet(data) # graph from an incidence matrix

savePajek(graph,"graph",edges="Haberman") # save graph.net file

## End(Not run)

shinyCoin

Include netCoin Plots in Shiny.

Description

Load a netCoin plot to display in shiny.

Usage

shinyCoin(x)

Arguments

x      is a netCoin, barCoin or timeCoin object.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

## Not run:
# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women",
                         "Man; Man", "Undet.; Women; Man"))
data <- dichotomize(frame, "A", sep="; ")[2:4]
C <- coin(data) # coincidence matrix
N <- asNodes(C) # node data frame
E <- edgeList(C) # edge data frame
net <- netCoin(N, E) # netCoin object
shinyCoin(net)

## End(Not run)
**Description**

It calculates a similarity/distance matrix from either an incidence data frame/matrix or a coin object.

**Usage**

```r
sim(input, procedures="Jaccard", level=.95, distance=FALSE,
minimum=1, maximum=Inf, sort=FALSE, decreasing=FALSE,
weight = NULL, pairwise = FALSE)
```

**Arguments**

- `input`: a binary data frame or a coin object, let’s say an R list composed by a number of scenarios ($n$) and a coincidence matrix with frequencies ($f$).
- `procedures`: a vector of statistics of similarity. See details below.
- `level`: confidence level
- `distance`: convert the similarity matrix into a distance matrix
- `minimum`: minimum frequency to obtain a similarity/distance measure.
- `maximum`: maximum frequency to obtain a similarity/distance measure.
- `sort`: sort the list according to the values of a statistic. See details below
- `decreasing`: order in a decreasing way.
- `weight`: a vector of weights. Optimal for data.framed tables
- `pairwise`: Pairwise mode of handling missing values if TRUE. Listwise by default.

**Details**

Possible measures in procedures are:

- Frequencies (f), Relative frequencies (x), Conditional frequencies (i), Coincidence degree (cc), Probable degree (cp),
- Expected (e), Confidence interval (con)
- Matching (m), Rogers & Tanimoto (t), Gower (g), Sneath (s), Anderberg (and),
- Jaccard (j), Dice (d), antiDice (a), Ochiai (o), Kulczynski (k),
- Hamann (ham), Yule (y), Pearson (p), odds ratio (od), Russell (r),
- Haberman (h), Z value of Haberman (z),
- Hypergeometric p greater value (hyp).

**Value**

A similarity/distance matrix.
sociologists

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

# From a random incidence matrix I(25X4)
I<-matrix(rbinom(100,1,.5),nrow=25,ncol=4, 
  dimnames=list(NULL,c("A","B","C","D")))
sim(I)
#Same results
C<-coin(I)
sim(C)

sociologists  Data: Sociologists born in the 19th century.

Description

Data frame with names, birth and death year data, birth country and movement.

Usage

data("sociologists")

Format

A data frame with 33 observations and the following 4 variables (events) to study coincidences in time:

name : name and last name of the sociologist
birth : birth year
death : death year
birthcountry : birth country
movements : movement or school of thought

Source

Own elaboration from manuals of sociology.

References

See events.
Examples

```r
data(sociologists)
head(sociologists, 10)
tail(sociologists, 10)
```

**surCoin**

*Networked coincidences from a data frame.*

**Description**

`surCoin` produces a network object of coincidences from a data frame converting variables into dichotomies.

**Usage**

```r
surCoin(data, variables=names(data), commonlabel=NULL,
  dichotomies=NULL, valueDicho=1, metric=NULL, exogenous=NULL,
  weight=NULL, subsample=FALSE, pairwise=FALSE,
  minimum=1, maximum=nrow(data), sort=FALSE, decreasing=TRUE,
  frequency=FALSE, percentages=TRUE,
  procedures="Haberman", criteria="Z", Bonferroni=FALSE,
  support=-Inf, minL=-Inf, maxL=Inf,
  directed=FALSE, diagonal=FALSE, sortL=NULL, decreasingL=TRUE,
  igraph=FALSE, coin=FALSE, dir=NULL, ...)
```

**Arguments**

- **data**
  a data frame
- **variables**
  a vector of variables included in the previous data frame
- **commonlabel**
  a vector of variables whose names are to be included in nodes labels
- **dichotomies**
  a vector of dichotomous variables to appear as just one category
- **valueDicho**
  value to be selected for dichotomous variables. Default is 1
- **metric**
  a vector of metrics
- **exogenous**
  a vector of variables whose relations amongst them are of no interest. None by default
- **weight**
  a vector of weights. Optimal for data.framed tables
- **subsample**
  restrict the analysis to scenarios with at least one event
- **pairwise**
  Pairwise mode of handling missing values if TRUE. Listwise by default.
- **minimum**
  minimum frequency to be considered
- **maximum**
  maximum frequency to be considered
- **sort**
  sort the coincidence matrix according to frequency of events
- **decreasing**
  decreasing or increasing sort of the matrix
- **frequency**
  a logical value true if frequencies are to be shown. Default=FALSE.
percentages  a logical value true if percentages are to be shown. Default=TRUE.
procedures  a vector of statistics of similarity. See below.
criteria  statistic to be use for selection criteria.
Bonferroni  Bonferroni criterium of the signification test.
support  minimum value of the frequency of the coincidence to be edged
minL  minimum value of the statistic to include the edge in the list.
maxL  maximum value of the statistic to include the edge in the list.
directed  includes same edges only once.
diagonal  includes auto-links
sortL  sort the list according to the values of a statistic. See below
decreasingL  order in a decreasing way.
igraph  Produces an igraph object instead of a netCoin object if TRUE.
coin  Only return the coincidences matrix if TRUE
dir  a "character" string representing the directory where the web files will be saved.
...  Any netCoin argument.

Details

Possible measures in procedures are

- Frequencies (f), Relative frequencies (x), Conditional frequencies (i), Coincidence degree (cc),
  Probable degree (cp),
- Expected (e), Confidence interval (con)
- Matching (m), Rogers & Tanimoto (t), Gower (g), Sneath (s), Anderberg (and),
- Jaccard (j), Dice (d), antiDice (a), Ochiai (o), Kulczynski (k),
- Hamann (ham), Yule (y), Pearson (p), odds ratio (od), Rusell (r),
- Haberman (h), Z value of Haberman (z),
- Hypergeometric p greater value (hyp).
- Convert a matrix into an edge list (shape).

Value

This function creates a netCoin object (or igraph) and, if stated, a folder in the computer with an
HTML document named index.html which contains the produced graph. This file can be directly
opened with your browser and sent to a web server to work properly.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See
https://sociocav.usal.es/blog/modesto-escobar/
Examples

# A data frame with two variables Gender and Opinion
frame<-data.frame(Gender=c(rep("Man",3),rep("Woman",3)),
                   Opinion=c("Yes","Yes","No","No","No","Yes"))
surCoin(frame,commonlabel="") # network object

Description

timeCoin produces a timeCoin object.

Usage

timeCoin(nodes, name = "name", start = "start", end = "end",
         group = NULL, text = NULL, main = NULL, note = NULL, cex = 1,
         language = c("en","es","ca"), dir = NULL)

Arguments

nodes      a data frame with at least two vectors of names and incidences.
name       name of the vector with names in the nodes data frame.
start      name of the vector with starts in the nodes data frame.
end        name of the vector with ends in the nodes data frame.
group      name of the vector with groups in the nodes data frame.
text       name of the vector with html text in the nodes data frame.
main       upper title of the graph.
ote        lower title of the graph.
cex         number indicating the amount by which plotting text should be scaled relative to the default. Default = 1.
language    a character vector (es=spanish; en=english; ca=catalan).
dir         a "character" string representing the directory where the web files will be saved.

Value

Object of class timeCoin.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/
Examples

# Database of 19th century sociologists
data(sociologists)
timeCoin(sociologists,"name","birth","death","birthcountry")

toIgraph

igraph object.

Description

igraph object from a netCoin object.

Usage

toIgraph(net)

Arguments

net is a netCoin object. See netCoin

Value

An igraph object.

Author(s)

Modesto Escobar, Department of Sociology and Communication, University of Salamanca. See https://sociocav.usal.es/blog/modesto-escobar/

Examples

# A character column (with separator)
frame <- data.frame(A = c("Man; Women", "Women; Women", "Man; Man", "Undet.; Women; Man"))
data <- dichotomize(frame, "A", sep="; ")[2:4]
C <- coin(data) # coincidence matrix
N <- asNodes(C) # node data frame
E <- edgeList(C) # edge data frame
net <- netCoin(N, E) # netCoin object
toIgraph(net) # conversion into a igraph object
Index

*Topic datasets
  dice, 11
  ess, 15
  events, 16
  families, 18
  finches, 18
  Galapagos, 20
  links, 24
  sociologists, 37

allNet, 4
asNodes, 4, 6
barCoin, 7

coin, 3, 9, 17
coocur, 10

dice, 11, 16
dichotomize, 12
distant, 13

directList, 3, 4, 14
ess, 15
events, 11, 16, 37
expectedList, 17

families, 18
finches, 18
fromIgraph, 19

Galapagos, 20
glmCoin, 21

incTime, 22

layoutCircle, 23
layoutGrid, 23
links, 24
lower, 25

mobileEdges, 26

multigraphCreate, 27

netCoin, 4, 5, 19, 21, 28, 31, 32, 39, 41
netCoin-package, 2
netCorr, 31

pathCoin, 32
propCoin, 33

savePajek, 34
shinyCoin, 35
sim, 3, 13, 36
sociologists, 37
surCoin, 38

timCoin, 40
toIgraph, 41