

Package ‘Dominance’

July 2, 2014

Version 1.0.0

Date 2014-04-29

Title ADI (average dominance index), social network graphs with dual directions, and music notation graph

Depends R (>= 2.14.0)

Imports igraph,chron,gdata

Suggests

Description The package can calculate ADI (Average Dominance Index) and can build social network graphs with dual directions, can build a Music Notation Graph

License GPL (>= 2)

URL <http://www.r-project.org>, <http://www.Konstanze-Krueger.de/r>

BugReports see maintainer

Author Konstanze Krueger [aut],Knut Krueger [cre, aut]

Maintainer Knut Krueger <Knut.Krueger@equine-science.de>

NeedsCompilation no

Repository CRAN

Date/Publication 2014-05-07 14:41:31

R topics documented:

Dominance-package	2
ADI	2
data_ADI	4
data_Musicnotation	5
data_Network_1	6
data_Network_2	7

detect_bits	8
Musicnotation	8
search.win.lose	10
Sociogram	11

Index	13
--------------	-----------

Dominance-package *Dominance Calculation and Graphs in Animals*

Description

The package can calculate ADI (Average Dominance Index) and can build social network graphs with dual directions, can build a Music Notation Graph

Details

The following are sources of information on **Dominance** package:

DESCRIPTION file `library(help="Dominance")`

This file package?Dominance

Some help files

[ADI](#)

[Sociogram](#)

[Musicnotation](#)

Author(s)

Knut Krueger

Maintainer: Who to complain to <Knut.Krueger@equine-science.de>

ADI *Average Dominance Index*

Description

computes the Average Dominance Index

Usage

`ADI(data_sheet, bytes, ...)`

Arguments

data_sheet	<p>either a data.frame f.e imported from a data sheet containing "Name","item.number" "action.from. ","action.to","kind.of.action" "name.of.action","action.number","classification","weighting"</p> <p>or only "action.from. ","action.to","kind.of.action"if exists actions and items</p> <p>actions: with "name.of.action","action.number","classification","weighting" items with "Name","item.number"</p>
bytes	a string where each enabled action is set to 1 and each disabled action is set to 0
...	<p>actions with "name.of.action","action.number","classification","weighting"</p> <p>Classification 1 for aggressive behaviour 2 for non aggressive behaviour like retreat</p> <p>Weighting the factor which should be used to calculate the behavior (1 for aggressive -1 for none aggressiv)</p> <p>Setting a aggressiv behaviour to 2 means it is count double</p> <p>vcolors as much colors as items, colors will returned as sorted ADI colors means color 1 = item rank 1, color 2 = item rank 2, and so on</p>

Value

ADI returns a list with
ADI - the Average Dominance index
Colors - the colors supported by vcolors sorted by ADI of the items
ADI_count_matrix - the counts from which the ADI was calculated

Author(s)

Knut Krueger, Prof. Dr. Konstanze Krueger

References

The Construction of Dominance Order: Comparing Performance of Five Methods Using an Individual-Based Model C. K. Hemelrijk, J. Wantia and L. Gygas, Behaviour Vol. 142, No. 8 (Aug., 2005), pp. 1037-1058

<http://www.jstor.org/stable/4536286>

On using the DomWorld model to evaluate dominance ranking methods , de Vries, Han, Behaviour, Volume 146, Number 6, 2009 , pp. 843-869(27)

<http://dx.doi.org/10.1163/156853909X412241>

Examples

```
## you can either use:
data_sheet=data.frame ("action.from"=c(1,4,2,3,4,3,4,3,4,3,4,3,4),
                        "action.to"=c(4,1,1,4,3,4,3,4,3,4,3,4,3),
                        "kind.of.action"= c(4,1,1,4,3,4,3,4,3,4,3,4,3),stringsAsFactors=FALSE)
items= data.frame ("Name"=c("item1","item2","item3","item4","item5","item6") ,
                  "item.number"=c(1:6),stringsAsFactors=FALSE)
actions=data.frame("name.of.action"= c("leading","following","approach","bite","threat to bite",
                                       "kick","threat to kick", "chase","retreat"),
                  "action.number"=c(1:9),
                  "classification"=c(1,2,1,1,1,1,1,1,2) ,
                  "weighting"=c(1,-1,1,1,1,1,1,1,-1),stringsAsFactors=FALSE)

## all encounters without leading and following
bytes= "001111111"
ADI(data_sheet,items=items,actions=actions,bytes)
## or you can use a complete f.e Excel sheet
## you can save this data as basic excel sheet to work with
data(data_ADI)
bytes= "001111111"
ADI(data_ADI,bytes)
```

 data_ADI

Demodata for ADI

Description

Demodata to calculate an ADI

Usage

```
data(data_ADI)
```

Format

A data frame with 17 observations on the following 10 variables.

Name a character vector
 item.number a numeric vector
 action.from. a numeric vector
 action.to a numeric vector
 kind.of.action a numeric vector
 observation.number a numeric vector
 name.of.action a character vector
 action.number a numeric vector
 classification a numeric vector
 weighting a numeric vector

Examples

```
data(data_ADI)
```

`data_Musicnotation` *Demodata for Musicnotation*

Description

Demodata to show an Musicnotation Graph

Usage

```
data(data_Musicnotation)
```

Format

A data frame with 15 observations on the following 11 variables.

`action.from` a numeric vector

`action.to` a numeric vector

`kind.of.action` a numeric vector

`Time` a character vector

`Name` a character vector

`item.number` a numeric vector

`dominance.order` a numeric vector

`name.of.action` a character vector

`action.number` a numeric vector

`classification` a numeric vector

`weighting` a numeric vector

Examples

```
data(data_Musicnotation)
```

`data_Network_1`*Demodata for Social network Graph*

Description

A dataset to show a bigger sociogramm

Usage

```
data(data_Network_1)
```

Format

A data frame with 800 observations on the following 16 variables.

Name a numeric vector

Beschreibung a character vector

item.number a numeric vector

dominance.order a character vector

age a character vector

sex a character vector

action.from. a numeric vector

action.to a numeric vector

kind.of.action a numeric vector

time a POSIXct

test.2.kind.of.action a numeric vector

test.3.kind.of.action a numeric vector

name.of.action a character vector

action.number a numeric vector

classification a numeric vector

weighting a numeric vector

Examples

```
data(data_Network_1)
```

`data_Network_2`*Demodata for Social network Graph*

Description

A dataset to show a bigger sociogramm

Usage

```
data(data_Network_2)
```

Format

A data frame with 800 observations on the following 16 variables.

`Name` a numeric vector

`Beschreibung` a character vector

`item.number` a numeric vector

`dominance.order` a character vector

`age` a character vector

`sex` a character vector

`action.from.` a numeric vector

`action.to` a numeric vector

`kind.of.action` a numeric vector

`time` a POSIXct

`test.2.kind.of.action` a numeric vector

`test.3.kind.of.action` a numeric vector

`name.of.action` a character vector

`action.number` a numeric vector

`classification` a numeric vector

`weighting` a numeric vector

Examples

```
data(data_Network_2)
```

detect_bits *for internal use only*

Description

for internal use only

Usage

```
detect_bits(bits, set = TRUE)
```

Arguments

bits

set if set = TRUE returns all true bits if set = false returns all false bits

Author(s)

Knut krueger

Examples

```
print('for internal use only')
```

Musicnotation *Music Notation Graph*

Description

computes Music Notation graphs

Usage

```
Musicnotation(data_sheet, ...)
```


Arguments

`data_sheet` either a data.frame f.e imported from a data sheet containing
 "Name","item.number"
 "action.from.,"action.to","kind.of.action"
 "name.of.action","action.number","classification","weighting"

or only "action.from.,"action.to","kind.of.action"if exists actions and items

actions: with "name.of.action","action.number","classification","weighting"
 items with "Name","item.number"

... colors: a factor of colors as much as actions
 lwd: line width if lwd_arrows is not used also for line width arrows
 show_items: items to be shown -
 angel_arrows: The angel aof the arrow head default 20
 length_arrows: the lenght of the arrow default 0.05
 lwd_arrows: the line width of the arrows default 1
 actions_colors: a vector of colors for actions f.e to show one special action
 starting_time:
 ending_time: builds the graph with data bewteen starting and ending time
 sort_dominace: sort the graph by ADI if sort_dominance is present
 user_colors: a vector of colors as much as items to show differetn colors for
 items
 color_bits: a vector of colors as much as items 1 shows the horse colored 0 in
 black (defined with actions_colors)

Value

returns the ADI

Author(s)

Knut Krueger

References

Chase, I. D. (2006). Music notation: a new method for visualizing social interaction in animals and humans. *Front Zool*, 3, 18.

<http://dx.doi.org/10.1186%2F1742-9994-3-18>

Examples

```
## you can eihter use:
dataM=data.frame ("action.from"=c(1,2,3,4,5,1,1,1,1,1,1,1,1,3,4),
                  "action.to"= c(2,3,4,5,6,2,3,4,5,6,3,4,3,4,3),
                  "kind.of.action"= c(4,1,1,4,3,4,3,4,3,4,3,4,3,4,3),
                  "Time"=c("03:15:00","03:17:30","03:20:00","03:20:30","03:21:00","03:21:30",
                           "03:22:00","03:22:30","03:23:00","03:23:30","03:25:00","03:25:30",
```

```

"03:26:00","03:26:30","03:27:00"),stringsAsFactors=FALSE)

items= data.frame ("Name"=c("item1","item2","item3","item4","item5","item6") ,
                  "item.number"=c(1:6),stringsAsFactors=FALSE)
actions=data.frame("name.of.action"= c("leading","following","approach","bite","threat to bite",
                                       "kick","threat to kick", "chase","retreat"),
                  "action.number"=c(1:9),
                  "classification"=c(1,2,1,1,1,1,1,1,2) ,
                  "weighting"=c(1,-1,1,1,1,1,1,1,-1),stringsAsFactors=FALSE)
## all encounters without leading and following
bytes= "001111111"
## set colors for special encounters
color= c("green","green","red","red","red","red","red","red")

Musicnotation(data_sheet=dataM,actions=actions,items=items,sort_dominance=TRUE)
## or you can use a complete f.e Excel sheet
## you can save this data as basic excel sheet to work with
data(data_Musicnotation)
Musicnotation(data_sheet=data_Musicnotation,sort_dominance=TRUE)

```

search.win.lose

for internal use only

Description

counts the wins and loses

Usage

```
search.win.lose(data_sheet, ...)
```

Arguments

data_sheet

...

Author(s)

Knut Krueger

Examples

```
print('for internal use only')
```

Sociogram

*Social Network Graphs***Description**

computes social network graphs with igraph

Usage

```
Sociogram(data_sheet, bits, ...)
```

Arguments

<code>data_sheet</code>	either a data.frame f.e imported from a data sheet containing "Name","item.number" "action.from.,"action.to","kind.of.action" "name.of.action","action.number","classification","weighting" or only "action.from.,"action.to","kind.of.action"if exists actions and items data_sheet: with "action.from.,"action.to","kind.of.action" items with "Name","item.number"
<code>bits</code>	as sting where each enabled action is set to 1 and each disabled action is set to 0
<code>...</code>	Postscript: FALSE (default) or path to PS output file soziogram_layout: layout.auto,layout.random,layout.circle,layout.sphere,layout.fruchterman.reingold,lay curved: how much the lines between the nodes ar curved 0 ist stright, default is 0.2 scal_value: the multiplicator of the nodes, default: 1/3 linesize_add: value to add to the linesize (helpful for a graph with thin lines. default 0 log: log2 size of linewidth, default= false canvas.width: default 1000 canvas.height: default 800 tkplot: interactiv tkplot, default = true

Value

<code>sociogram</code>	the igraph object
<code>counts_circles</code>	the count of cirles
<code>count_interactions</code>	the count of interactions
<code>line_size</code>	the used linesize
<code>counts_circles</code>	vector of min and max line size

the last for are helpful to change circle size and linewidth

Author(s)

Knut Krueger

Examples

```

## you can either use:
data_sheet=data.frame ("action.from"=c(1,4,2,3,4,3,4,3,4,3,4,3,4,3,4),
                        "action.to"=c(4,1,1,4,3,4,3,4,3,4,3,4,3,4,3),
                        "kind.of.action"= c(4,1,1,4,3,4,3,4,3,4,3,4,3,4,3),stringsAsFactors=FALSE)
items= data.frame ("Name"=c("item1","item2","item3","item4","item5","item6") ,
                  "item.number"=c(1:6),stringsAsFactors=FALSE)
actions=data.frame("name.of.action"= c("leading","following","approach","bite","threat to bite",
                                       "kick","threat to kick", "chase","retreat"),
                  "action.number"=c(1:9),
                  "classification"=c(1,2,1,1,1,1,1,1,2) ,
                  "weighting"=c(1,-1,1,1,1,1,1,1,-1),stringsAsFactors=FALSE)
## all encounters without leading and following
bytes= "001111111"
Sociogram(data_sheet,items=items,actions=actions,bytes)
## mor you can use a complete f.e Excel sheet
## you can save this data as basic excel sheet to work with
data(data_Network_1)
## set 1 for action you want to show
bytes= "00111111111000000000"
Sociogram(data_Network_1,bytes)

```

Index

- *Topic **ADI**
 - Dominance-package, [2](#)
 - *Topic **Music Notation Graph**
 - Dominance-package, [2](#)
 - *Topic **Social Network Graph**
 - Dominance-package, [2](#)
 - *Topic **\textasciitildekw1**
 - ADI, [2](#)
 - Musicnotation, [8](#)
 - Sociogram, [11](#)
 - *Topic **\textasciitildekw2**
 - ADI, [2](#)
 - Musicnotation, [8](#)
 - Sociogram, [11](#)
 - *Topic **datasets**
 - data_ADI, [4](#)
 - data_Musicnotation, [5](#)
 - data_Network_1, [6](#)
 - data_Network_2, [7](#)
- ADI, [2](#), [2](#)
- data_ADI, [4](#)
- data_Musicnotation, [5](#)
- data_Network_1, [6](#)
- data_Network_2, [7](#)
- detect_bits, [8](#)
- Dominance (Dominance-package), [2](#)
- Dominance-package, [2](#)
- Musicnotation, [2](#), [8](#)
- search.win.lose, [10](#)
- Sociogram, [2](#), [11](#)