

Package: OpenRepGrid.ic (via r-universe)

October 24, 2024

License MIT + file LICENSE

Title Interpretive Clustering for Repertory Grids

Type Package

LazyLoad yes

Description Shiny UI to identify cliques of related constructs in repertory grid data. See Burr, King, & Heckmann (2020) <[doi:10.1080/14780887.2020.1794088](https://doi.org/10.1080/14780887.2020.1794088)> for a description of the interpretive clustering (IC) method.

Version 0.6.2

Date 2024-02-18

Imports graphics, grDevices, utils, shiny, shinyjs, shinyBS, shinythemes, shinyWidgets, shinydashboard, shinydashboardPlus (>= 2.0.0), shinycssloaders, shinyFeedback, rintrojs, formattable, openxlsx, DT, magrittr, dplyr, stringr, reshape2, scales, splines, igraph, tidyr, withr, RColorBrewer, tidyverse

Suggests knitr, rmarkdown, covr, testthat

Encoding UTF-8

URL <https://github.com/markheckmann/OpenRepGrid.ic>

BugReports <https://github.com/markheckmann/OpenRepGrid.ic/issues>

RoxygenNote 7.3.0

Repository <https://markheckmann.r-universe.dev>

RemoteUrl <https://github.com/markheckmann/openrepgrid.ic>

RemoteRef HEAD

RemoteSha 922aace4b288bb9e1becf8ec47bb464751da4f48

Contents

calculate_similarity	2
check_excel_input	2
check_excel_input_test	3

create_excel_output	3
ic	4
network_graph_images	4
OpenRepGrid.ic	5

Index	8
--------------	----------

calculate_similarity *Calculate similarity matrix*

Description

Calculate similarity matrix

Usage

calculate_similarity(x, min_matches = 6, align_poles = TRUE)

Arguments

x	Grid data.
min_matches	Minimal number of matches to considers constructs as related.
align_poles	Align positive poles on the right and negative poles on the left.

check_excel_input *Check if Excel input file contains valid data*

Description

Check if Excel input file contains valid data

Usage

check_excel_input(x)

Arguments

x	Data from Excel input file.
---	-----------------------------

`check_excel_input_test`*Check if Excel input file contains valid data*

Description

Check if Excel input file contains valid data

Usage

```
check_excel_input_test(x)
```

Arguments

x Data from Excel input file.

`create_excel_output` *Create output Excel file*

Description

Loads the supplied workbook and adds calculations

Usage

```
create_excel_output(file, data = list())
```

Arguments

file Path to workbook.
data Named list of data objects to add to Excel file. The following contents are expected: TODO

Value

Path to saved file.

ic *Launch app in browser*

Description

Launch app in browser

Usage

```
ic(display.mode = "auto", launch.browser = TRUE)
```

Arguments

display.mode auto by default, can also be showcase. See [runApp](#).

launch.browser Boolean, set TRUE (default) to open the app in the browser. See [runApp](#).

Examples

```
if (interactive()) {  
  ic()  
}
```

network_graph_images *Build network graph plots*

Description

Detects maximal cliques and saves images of network graphs into tempfile. Tempfile paths and info on cliques are returned.

Usage

```
network_graph_images(  
  x,  
  min_clique_size = 3,  
  show_edges = TRUE,  
  min_matches = 6,  
  label_wrap_width = 15,  
  label_max_length = -1,  
  indicate_direction = show_edges,  
  colorize_direction = TRUE,  
  colorize_cliques = TRUE,  
  colorize_poles = TRUE,  
  align_poles = TRUE,  
  alpha = 0.1,
```

```

    valence_prefix = FALSE,
    border_default = "#987824",
    fill_default = "#00000008",
    image_border_color = grey(0.6),
    seed = 0
)

```

Arguments

x	A dataframe with a grid.
min_clique_size	Minimal size of cliques to be considered.
show_edges	Whether to show edges in plot.
min_matches	Minimal number of matching scores between constructs to be marked as related.
label_wrap_width	Width of wrapped element label text.
label_max_length	Trim element label at max length characters.
indicate_direction, colorize_direction	Indicate direction of relatedness by edge label +/- and edge color (red, green). Only applied if show_edges = TRUE.
colorize_cliques	Draw cliques in different colors? (default TRUE).
colorize_poles	Colorize positive/negative/neutral poles as red, green, and gray respectively (default TRUE).
align_poles	Align preferred poles on the same side.
alpha	Alpha color value for cliques fillings (default .1).
valence_prefix	Add (+/-) pole prefix to indicate preference. Empty means no preference.
border_default, fill_default	Default border and fill color of polygon encircling clique constructs. Used when colorize_cliques is FALSE. Use NA for no color.
image_border_color	Color of border around generated graph images. If NULL or NA no border is drawn.
seed	Seed number passed to set . seed. Will determine the orientation of the graph.

Description

The **OpenRepGrid.ic** package implements *Interpretive Clustering (IC)* as outlined in Burr, King, and Heckmann (2020). The authors describe a variant of construct clustering which uses a procedure from graph theory called **maximal cliques enumeration**. Given a similarity measure, in our case the number of matching scores between two constructs, a network graph of relatedness between constructs is construed. A clique is a group of constructs which are all mutually related, given some cut-off criterion for relatedness (e.g. 6 matching scores in a grid with 7 elements). While the paper also describes an offline approach to identify the construct cliques, this software automates the process. Under the hood, the package uses the *igraph* package for clique identification.

The package also contains a shiny based UI you can start via the function `ic()`. Visit <http://ic.openrepgrid.org> for an online version. An introduction to the software is also available on [YouTube](#). Below you find an example of how to process a repgrid in an Excel file using code only.

Author(s)

Maintainer: Mark Heckmann <heckmann.mark@gmail.com> ([ORCID](#)) [copyright holder]

Authors:

- Viv Burr
- Nigel King

References

Burr, V. King, N. & Heckmann, M. (2020) The qualitative analysis of repertory grid data: Interpretive Clustering, *Qualitative Research in Psychology*, [doi:10.1080/14780887.2020.1794088](https://doi.org/10.1080/14780887.2020.1794088)

See Also

Useful links:

- <https://github.com/markheckmann/OpenRepGrid.ic>
- Report bugs at <https://github.com/markheckmann/OpenRepGrid.ic/issues>

Examples

```
# The shiny package is just a small UI wrapper around the
# the workhorse core functions. Here is how to call them.

library(tidyverse)
library(openxlsx)
library(igraph)
library(OpenRepGrid.ic)

file <- system.file("extdata", "sylvia.xlsx", package = "OpenRepGrid.ic")
file_out <- str_replace(file, ".xlsx$", " CLIQUES.xlsx") %>% basename

x <- read.xlsx(file)           # read grid
tests <- check_excel_input(x) # check if input format is correct
l <- network_graph_images(x, min_clique_size = 3,
```

```
                show_edges = TRUE,
                min_matches = 6) # produce images
file_tmp <- create_excel_output(file, 1) # create Excel file
# file.show(file_tmp) # not run during tests

# open images saved as temp files (as shown in output Excel file)
file.show(l$img_all_constructs)
file.show(l$img_all_constructs_full_labels)
file.show(l$img_all_constructs_separate_poles)

file.show(l$img_cliques_only)
file.show(l$img_cliques_only_full_labels)
file.show(l$img_cliques_only_separate_poles)

# calculation results used in network_graph_images
# some of them are also contained in Excel file
s <- calculate_similarity(x)
s
```

Index

* **OpenRepGrid.ic**

OpenRepGrid.ic, [5](#)

* **package**

OpenRepGrid.ic, [5](#)

calculate_similarity, [2](#)

check_excel_input, [2](#)

check_excel_input_test, [3](#)

create_excel_output, [3](#)

ic, [4](#)

network_graph_images, [4](#)

OpenRepGrid.ic, [5](#)

OpenRepGrid.ic-package

(OpenRepGrid.ic), [5](#)

runApp, [4](#)