# Package 'SARP.compo'

January 13, 2018

**Type** Package

**Title** Network-based Interpretation of Changes in Compositional Data

**Version** 0.0.3

**Date** 2018-01-09

**Maintainer** Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

**License** Artistic-2.0

**Description** Provides a set of functions to interpret changes in compositional data based on a network representation of all pairwise ratio comparisons: computation of all pairwise ratio, construction of a p-value matrix of all pairwise tests of these ratios between conditions, conversion of this matrix to a network.

**Suggests**

**Depends** igraph,car

**Imports**

**Encoding** UTF-8

**Language** uk

**NeedsCompilation** no

**Author** Emmanuel Curis [aut, cre]

**Repository** CRAN

**Date/Publication** 2018-01-13 00:38:17 UTC

## R topics documented:

BpLi *Circadian Genes Expression in Bipolar Disorder Patients*

## Description

These two datasets give the expression level of main circadian genes in lymphoblastoid cells from bipolar disorder patients, as determined by qRT-PCR. Results are expressed in cycle thresholds (CT) units.

## Usage

```
data(BpLi_J2)
data(BpLi_J4)
```

## Format

Each dataset is a data frame with 78 rows and 26 columns. Each row give the RNA quantification of circadian and control genes for lymphoblastoid cells of a given patient, either with or without lithium in the culture medium.

| | | |
|---|---|---|
| Phenotype | factor | Patient phenotype, either responding (R) or not (NR) to lithium |
| Patient | integer | Patient code |
| Li | factor | Is lithium present (Oui) or not (Non) in the culture medium of the cells |

Other columns are the different circadian genes and reference genes. Three sets of experiments were done, using three different dilutions according to the explored gene: 1/20 (PER3, BHLHE41, NR1D1, DBP), 1/100 (GSK3b, RORA, PER1, PER2, CLOCK, ARNTL, CRY2, BHLHE40) and 1/200 (ARNTL2, TIMELESS, CRY1, CSNK1E). The numerical suffixe after the reference gene name (SDHA or HPRT) gives this dilution level – for instance, SDHA_20 is for the 1/20 dilution level, HRPT_100 for the 1/100 level...

Patients were classified as presenting a good response (R) or a lack of responce (NR) to lithium treatment based on the ALDA scale, see the original publication for details. Lymphoblastoid cells from each patients, obtained from blood samples, were cultivated for 2 (BpLi_J2) or 4 (BpLi_J4) days either with or without LiCl.

## Source

Data courtesy communicated, and allowed to be included in the package, by the authors.

## References

Geoffrey, P. A., Curis E., Courtin, C., Moreira, J., Morvillers, T., Etain, B., Laplanche, J.-L., Bellivier, F. and Marie-Claire, C. (2017). Lithium response in bipolar disorders and core clock genes expression. World J Biol Psychiatry, doi: 10.1080/15622975.2017.1282174.

`calc.rapports` *Compute all pairwise ratios of a set of variables*

## Description

This function computes all pairwise ratios or differences of a set of variables in a given data frame

## Usage

```
calc.rapports( d, noms, log = FALSE, isoler = FALSE )
```

## Arguments

| | |
|---|---|
| `d` | The data frame that contains the variables. Other objects will be coerced as data frames using `as.data.frame` |
| `noms` | A character vector containing the column names of the compositional variables to be used for ratio computations. Names absent from the data frame will be ignored with a warning.<br>Optionnally, an integer vector containing the column numbers can be given instead. They will be converted to column names before further processing. |
| `log` | If `TRUE`, values in the columns are assumed to be log-transformed, and consequently ratios are computed as differences of the columns. The result is in the log scale.<br>If `FALSE`, values are assumed to be raw data and ratios are computed directly. |
| `isoler` | If `TRUE`, the result data frame will not include the original values. |

## Details

Use this function to compute all pairwise ratio of a set of numerical variables. If non-numerical variables are given in the list of variables, they will be ignores with a warning.

Since the ratio of variables i and j is the inverse of the ratio of variables j and i, only one of them is computed. The order is determined by the order of the variables in `noms`. In matrix notations, only the upper right matrix is computed, withour the diagonal.

## Value

These function returns the original data.frame with additional columns corresponding to all pairwise ratios added as the last columns of the data.frame.

These variables have their name constructed as the concatenation of the names of the two variables used, the first one being at the numerator, separated with a dot and with the additional suffix .r (or .r.log is working on difference of logarithms).

Their order is determined by the order given in noms: the first variable of the list, V1, is used to compute ratios with all others (V1/V2, V1/V3 and so on). Then the second one is used for ratios further ones (V2/V3 and so on), and so on until the last one.

## Note

This function is mainly for designing a step-by-step analysis or control purposes. To avoid waste of memory, most of the functions in the package actually compute "on fly" the ratios when constructing the matrix or the data frame of p-values.

## Author(s)

Emmanuel Curis (<emmanuel.curis@parisdescartes.fr>)

## See Also

creer.Mp to create a matrix of p-values for all pairwise tests of ratio changes.

## Examples

```
# load the potery data set
data( poteries )

# Compute all ratios in the potery data set
d.r <- calc.rapports( d = poteries, noms = c( 'Al', 'Fe', 'Mg', 'Ca',
'Na' ) )
names( d.r )
head( d.r )

identical( d.r$Al.Fe.r, d.r$Al / d.r$Fe )
```

---

# creer_graphe *Create a graph using a set of p-values from pairwise tests*

---

## Description

These functions construct an undirected graph, using the igraph package, to represent and investigate the results of all testing of all ratios of components of a compositional vector.

## Usage

```
grf.Mp( Mp, p = 0.05, reference = NULL, groupes = NULL )
```

## Arguments

| | |
|---|---|
| Mp | A square, symmetric data frame containing $p$-values. Element in row $i$ and line $j$ should contain the $p$-value for testing the $\frac{i}{j}$ ratio. The diagonal is ignored. |
| p | The $p$-value cutoff for adding an edge between two nodes. See details. |
| reference | |
| groupes | |

## Details

Consider a compositional vector of $n$ components. These $n$ are seen as the nodes of a graph. Nodes $i$ and $j$ will be connected if and only if the $p$-value for the test of the $\frac{i}{j}$ ratio is higher than the cutoff, p – that is, if the test is **not** significant at the level $p$.

Strongly connected sets of nodes will represent components that share a similar behaviour between the conditions tested, whereas unrelated sets of nodes will have a different behaviour.

## Value

These function returns the created graph. It is an igraph object on which any igraph function can be applied, including plotting, and searching for graph components, cliques or communities.

## Author(s)

Emmanuel Curis (<emmanuel.curis@parisdescartes.fr>)

## See Also

creer.Mp to create a matrix of $p$-values for all possible ratios of a compositional vector.

## Examples

```
# load the potery data set
data( poteries )

# Compute one-way ANOVA p-values for all ratios in this data set
Mp <- creer.Mp( poteries, c( 'Al', 'Na', 'Fe', 'Ca', 'Mg' ),
                f.p = anva1.fpc, v.X = 'Site' )
Mp

# Make a graphe from it and plot it
plot( grf.Mp( Mp ) )
```

---

# creer_matrice *Create p-values matrix from pairwise tests of all possible ratios of a compositional vector*

---

## Description

This function performs hypothesis testing on all possible pairwise ratios or differences of a set of variables in a given data frame, and store their results in a (symmetric) matrix

## Usage

```
creer.Mp( d, noms, f.p, log = FALSE, en.log = !log,
          nom.var = 'R', ... )
```

## Arguments

| | |
|---|---|
| d | The data frame that contains the compositional variables. Other objects will be coerced as data frames using as.data.frame |
| noms | A character vector containing the column names of the compositional variables to be used for ratio computations. Names absent from the data frame will be ignored with a warning.<br>Optionnally, an integer vector containing the column numbers can be given instead. They will be converted to column names before further processing. |
| f.p | An R function that will perform the hypothesis test on a single ratio (or log ratio, depending on log and en.log values).<br>This function should return a single numerical value, typically the p-value from the test.<br>This function must accept at least two named arguments: d that will contain the data frame containing all required variables and variable that will contain the name of the column that contains the (log) ratio in this data frame. All other needed arguments can be passed through ....<br>Such functions are provided for several common situations, see references at the end of this manual page. |
| log | If TRUE, values in the columns are assumed to be log-transformed, and consequently ratios are computed as differences of the columns. The result is in the log scale.<br>If FALSE, values are assumed to be raw data and ratios are computed directly. |
| en.log | If TRUE, the ratio will be log-transformed before applying the hypothesis test computed by f.p. Don't change the default unless you really know what you are doing. |
| nom.var | A length-one character vector giving the name of the variable containing a single ratio (or log-ratio). No sanity check is performed on it: if you experience strange behaviour, check you gave a valid column name, for instance using make.names. |
| ... | additional arguments to f.p, passed unchanged to it. |

## Details

This function constructs a $n \times n$ matrix, where n = length( noms ) (after eventually removing names in noms that do not correspond numeric variables). Term $(i, j)$ in this matrix is the result of the f.p function when applied on the ratio of variables noms[ i ] and noms[ j ] (or on its log, if either (log == TRUE) && (en.log == FALSE) or (log == FALSE) && (en.log == TRUE)).

The f.p function is always called only once, for $i < j$, and the other term is obtained by symmetry.

The diagonal of the matrix is filled with 1 without calling f.p, since corresponding ratios are always identically equal to 1 so nothing useful can be tested on.

## Value

These function returns the matrix obtained as described above, with row an column names set to the names in noms (after conversion into column names and removing all non-numeric variables).

### Note

Since the whole matrix is stored and since it is a dense matrix, memory consumption (and computation time) increases as $n^2$. For compositional data with a large number of components, like in RNA-Seq data, consider instead creating a file.

### Author(s)

Emmanuel Curis (<emmanuel.curis@parisdescartes.fr>)

### See Also

Predefined `f.p` functions: `anva1.fpc` for one-way analysis of variance; `kw.fpc` for the non-parametric equivalent (Kruskal-Wallis test).

`grf.Mp` to create a graphe from the obtained matrix.

### Examples

```
# load the potery data set
data( poteries )

# Compute one-way ANOVA p-values for all ratios in this data set
Mp <- creer.Mp( poteries, c( 'Al', 'Na', 'Fe', 'Ca', 'Mg' ),
                f.p = anva1.fpc, v.X = 'Site' )
Mp

# Make a graphe from it and plot it
plot( grf.Mp( Mp ) )
```

---

## poteries *Composition of Roman poteries*

---

### Description

This data set gives the oxide composition of several potteries found in five different archaelogic sites of the United Kingdom. Composition was obtained using atomic absorption spectrometry.

### Usage

```
data(poteries)
```

### Format

A data frame with 14 columns and 48 rows. Each row gives the composition of a pottery (columns 2 to 10), the archaelogical site where it was found (columns 6 and 7):

| | | |
|---|---|---|
| ID | factor | Pottery sample identifier (see original paper appendix) |
| Al | numeric | Percentage of aluminium oxide |
| Fe | numeric | Percentage of iron oxide |

| | | |
|---|---|---|
| Mg | numeric | Percentage of magnesium oxide |
| Ca | numeric | Percentage of calcium oxide |
| Na | numeric | Percentage of natrium oxide |
| K | numeric | Percentage of kalium oxide |
| Ti | numeric | Percentage of titanim oxide |
| Mn | numeric | Percentage of manganese oxide |
| Ba | numeric | Percentage of baryum oxide |
| Site | factor | Kiln site |
| Pays | factor | Location of the kiln site |
| Couleur | factor | External color of the pottery |
| Date | factor | Approximate date of the pottery |

## Note

The DASL version of the dataset, as presented in the "Pottery stoty", does not include data on the poteries from the Gloucester site, neither the data on K, Ti, Mn and Ba oxides. It neither includes the color and date informations, and codes sites as their first letter only.

The DASL version of the dataset exists in the `car` package, as the `Pottery` dataset (with two locations differently spelled).

## Source

Downloaded from the DASL (Data and Story Library) website, and completed from the original paper of Tubb et al.

## References

A. Tubb, A. J. Parker, and G. Nickless (1980). The analysis of Romano-British pottery by atomic absorption spectrophotometry. *Archaeometry*, 22, 153-171.

Hand, D. J., Daly, F., Lunn, A. D., McConway, K. J., and E., O. (1994) *A Handbook of Small Data Sets*. Chapman and Hall – for the short version of the dataset.

## Examples

```
data( poteries )
# Reconstruct the car version of this dataset
dcar <- poteries[ , c( 'Al', 'Fe', 'Mg', 'Ca', 'Na', 'Site' ) ]
dcar <- droplevels( dcar[ -which( dcar$Site == "College of Art" ), c( 6, 1:5 ) ] )
levels( dcar$Site )[ c( 1, 3, 4 ) ] <- c( "AshleyRails", "Islethorns", "Llanedyrn" )

# Reconstruct the DASL version of this dataset
dasl <- dcar[ , c( 2:6, 1 ) ]
levels( dasl$Site ) <- c( 'A', 'C', 'I', 'L' )
```

tests *Utility functions to obtain p-values from tests on individual ratios*

## Description

These functions can be used in the functions to perform analysis on all pairwise ratios of a compositional dataset

## Usage

```
anva1.fpc( d, variable, v.X, frm = NULL, ... )
rls.fpc( d, variable, v.X, frm = NULL, ... )
kw.fpc( d, variable, v.X, frm = NULL, ... )
anva_SC.fpc( d, variable, frm, SC = 1, type = 1, ... )
```

## Arguments

| | |
|---|---|
| d | The data frame that contains the ratio to test, and all variables of the original data frame that where not used as compositional data. |
| variable | A length-one character vector containing the names of the variable corresponding to the ratio (or log-ratio) to test. |
| v.X | The **name** of the explanatory (independant, predictor) variable. This variable should be a factor for anva1.fpc and kw.fpc and a numeric for rls.fpc. |
| frm | The formula to use.<br>Defaults to as.formula( paste0( variable, "~", v.X ) ) for anva1.fpc and kw.fpc. Providing the formula speeds up the computation, since it avoids repeating the construction step for each ratio.<br>For anva_SC.fpc, giving the formula is mandatory and variable is unused. Beware of the term order to select the right sum of squares to test! |
| SC | For anova.SC, the number of the line to use in the analysis of variance table to get a p-value, see details. |
| type | For anova.SC, the kind of sums of square to be used when constructing the analysis of variance table, see details. |
| ... | additional arguments |

## Details

These functions are only wrapper to some commonly used tests. The correspondance is as follow

| | | |
|---|---|---|
| anva1.fpc | One-way analysis of variance | anova(lm())[ 1, 5 ] |
| rls.fpc | Simple linear regression | anova(lm())[ 1, 5 ] |
| kw.fpc | Kruskal-Wallis test | kruskal.test()$p.value |

anva_SC.fpc is a generic wrapper for lm using any formula. It then extracts the *p*-value of the line given by SC in the analysis of variance table. If type = 1, the table is built using anova and corresponds to type 1 (sequential sum of square). If type = 2 or type = 3, the table is built using car::Anova and corresponds either to type 2 or type 3 sums of squares.

## Value

These function returns the *p*-value from the corresponding test.

## Note

rls.fpc is an exact synonym for anva1.fpc, since the underlying theory is the same. Distinction is made to help users without a formal statistical background to find the right test.

## Author(s)

Emmanuel Curis (<emmanuel.curis@parisdescartes.fr>)

## See Also

kruskal.test, lm, anova, Anova for corresponding tests.

creer.Mp to use these functions to create a matrix of *p*-values.

## Examples

```
# load the potery data set
data( poteries )

# Compute one-way ANOVA p-values for all ratios in this data set
Mp <- creer.Mp( poteries, c( 'Al', 'Na', 'Fe', 'Ca', 'Mg' ),
                f.p = anva1.fpc, v.X = 'Site', frm = R ~ Site )
Mp

# Make a graphe from it and plot it
plot( grf.Mp( Mp ) )
```

# Index