data.cube

March 23, 2018

---

as.data.cube.data.frame
Transform a data.frame *into a* data.cube.

Description

as.data.cube transforms a data.frame into a data.cube by interpreting its rows as observations and its columns either as categorical descriptions of the observations (elements within data.cube dimensions) or as numerical descriptions of the observations (values taken by the data.cube variables).

Usage

```
## S3 method for class 'data.frame'
as.data.cube(df, dim, var)
```

Arguments

df	A data.frame to be transformed into a data.cube.
dim	A list of unquoted expressions giving the columns of df to be used as dimensions for the data.cube. If names are provided, they are interpreted as names for the corresponding dimensions.
var	A list of unquoted expressions giving the columns of df to be used as variables for the data.cube. If names are provided, they are interpreted as names for the corresponding variables.

Value

A data.cube resulting from the transformation of df.

as.data.frame.data.cube *Transform a* data.cube *into a* data.frame.

Description

as.data.frame transforms a data.cube into a data.frame by presenting observations in rows, and dimensions and variables in columns.

Usage

```
## S3 method for class 'data.cube'
as.data.frame(dc, complete = FALSE)
```

Arguments

dc	A data.cube to be transformed into a data.frame.
complete	A logical indicating if observations for which variables all equal zero should be presented in the output data.frame. If not, the output data.frame hence provides a sparse representation of the data.

Value

A data.frame resulting from the transformation of dc.

data.cube *Easy Processing of Multidimensional Data.*

Description

data.cube is an R package for the exploration of multidimensional datasets and for the detection of statistical outliers within. It is mainly a tool for data exploration, allowing to have a first glance at it and to formulate research hypotheses to be later tested.

The package defines a new data structure called data.cube that can be fed with a classical data.frame encoding a list of numeric variables described according to several categorical dimensions. For example, in the case of Twitter data, it can be the number of tweets (numeric variable) that have been published by a given user (first dimension) about a given topic (second dimension) at a given date (third dimension). The input data.frame hence takes the form of a list of quadruplets (user, topic, date, number of tweets).

Statistical outliers can then be identified among the observations by first selecting some dimensions of interest, that is by subsetting or by aggregating the input dimensions. If needed, observations can also be normalised according to the marginal values along the selected dimensions, thus comparing the observed value to an expected value obtained by the uniform redistribution of the selected marginal values. Different statistical tests can then be chosen to measure the deviation between the observed and the expected values. The package finally allows to retrieve a list of positive outliers, that is observations that are significantly higher than expected.

Note that the current implementation is optimised for sparse data.

This package has been developed by researchers of the Complex Networks team, within the Computer Science Laboratory of Paris 6 (LIP6), for the ODYCCEUS project, founded by the European Commission FETPROACT 2016-2017 program under grant 732942.

Links:

- Complex Networks team: `http://www.complexnetworks.fr/`
- LIP6: `https://www.lip6.fr/`
- ODYCCEUS project: `https://www.odycceus.eu/`

Contact:

- Robin Lamarche-Perrin: `<Robin.Lamarche-Perrin@lip6.fr>`
  See also my webpage: `https://www-complexnetworks.lip6.fr/~lamarche/`

List of main collaborators:

- Robin Lamarche-Perrin (CNRS, ISC-PIF, LIP6)
- Audrey Wilmet (UPMC, LIP6)
- Léonard Panichi (UPMC, LIP6)

`property.data.cube` *Get basic properties.*

Description

A collection of functions to get basic properties of a `data.cube`.

Usage

```
## S3 method for class 'data.cube'
dim.names(dc)

## S3 method for class 'data.cube'
elm.names(dc, ...)

## S3 method for class 'data.cube'
var.names(dc)

## S3 method for class 'data.cube'
dim.nb(dc)

## S3 method for class 'data.cube'
elm.nb(dc, ...)

## S3 method for class 'data.cube'
var.nb(dc)
```

Arguments

`dc`	A `data.cube`.
`...`	Optional. Names of the dimensions of interest.

Value

A character vector giving the names of the dimensions of `dc`.

A named list of character vectors giving the names of the elements in the dimensions of `dc`.

A character vector giving the names of the variables of `dc`.

An integer giving the number of dimensions of `dc`.

A named list of integers giving the number of elements in the dimensions of `dc`.

An integer giving the number of variables of `dc`.

`select.dim.data.cube` *Select dimensions of interest.*

Description

`select.dim` selects a subset of dimensions within the `data.cube` by aggregating together the elements of each unselected dimension. This consists in replacing the datacube by one of its marginal dataplane.

Usage

```
## S3 method for class 'data.cube'
select.dim(dc, ...)
```

Arguments

dc A datacube.

... A list of unquoted expressions giving the dimensions to select.

Value

The resulting datacube.

select.elm.data.cube *Select or remove elements within a given dimension.*

Description

select.elm and remove.elm respectively select and remove a subset of elements within a particular dimensions. This consists in suppressing all observations that corresponds to the unselected or removed elements.

Usage

```
## S3 method for class 'data.cube'
select.elm(dc, dim, elm = NULL, top.nb = NULL,
  bot.nb = NULL, var = NULL, suppress = FALSE)

## S3 method for class 'data.cube'
remove.elm(dc, dim, elm = NULL, top = NULL,
  bot = NULL, var = NULL, suppress = FALSE)
```

Arguments

dc A datacube.

dim An unquoted expression giving the dimension whose elements are selected or removed.

elm Optional. A character vector giving the names of the elements to select or to remove.

top.nb bot.nb Optional. An integer giving the number of elements to select or to remove among the ones with the highest or the lowest values of the specified variable (see var).

var Optional. Can be specified if either top.nb or bot.nb is specified. An unquoted expression giving the variable to consider when selecting or removing the top or the bottom elements. If not specified, the datacube's first variable is taken. Can also be model, ratio, or deviation if a marginal model has been computed.

suppress A logical indicating if the corresponding observations should be fully suppressed (recomputing the datacube's marginals) or simply hidden (no longer shown when processing the datacube).

Value

The resulting datacube.

Index