Package 'distrr'

January 3, 2019

Title Estimate and Manage Empirical Distributions

Description Tools to estimate and manage empirical distributions, which should work with survey data. One of the main features is the possibility to create data cubes of estimated statistics, that include all the combinations of the variables of interest (see for example functions dcc5() and dcc6()).

Depends R (>= 3.1.2)

Imports magrittr (>= 1.5), dplyr (>= 0.7.4), rlang, utils, stats, tidyr (>= 0.7.0)

Version 0.0.5

License GPL-2

URL https://gibonet.github.io/distrr, https://github.com/gibonet/distrr

Maintainer Sandro Petrillo Burri <gibo.gaf@gmail.com>

RoxygenNote 6.1.0

Encoding UTF-8

LazyData yes

Collate ``distrr.R'' ``invented_data.R'' ``compat-lazyeval.R'' ``dplyr_new_wrappers.R'' ``gibutils.R'' ``jointfuns.R'' ``Fhat_conditional.R'' ``distrr_funs.R'' ``dcc_new.R'' ``wq_df.R''

NeedsCompilation no

Author Sandro Petrillo Burri [aut, cre]

Repository CRAN

Date/Publication 2019-01-03 10:50:06 UTC

R topics documented:

`combn_char` *Generate all combinations of the elements of a character vector*

Description

Generate all combinations of the elements of a character vector

Usage

```
combn_char(x)
```

Arguments

x	a character vector

Value

a nested list. A list whose elements are lists containing the character vectors with the combinations of their elements.

Examples

```
combn_char(c("gender", "sector"))
combn_char(c("gender", "sector", "education"))
```

`dcc` *Data cube creation (dcc)*

Description

Data cube creation (dcc)

Usage

```
dcc(.data, .variables, .fun = jointfun_, ...)

dcc2(.data, .variables, .fun = jointfun_, order_type = extract_unique2,
  ...)

dcc5(.data, .variables, .fun = jointfun_, .total = "Totale",
  order_type = extract_unique4, .all = TRUE, ...)
```

Arguments

.data	data frame to be processed
.variables	variables to split data frame by, as a character vector (c("var1", "var2")).
.fun	function to apply to each piece (default: jointfun_)
...	additional functions passed to .fun.
order_type	a function like extract_unique or extract_unique2.
.total	character string with the name to give to the subset of data that includes all the observations of a variable (default: "Totale").
.all	logical, indicating if functions' have to be evaluated on the complete dataset.

Value

a data cube, with a column for each cateogrical variable used, and a row for each combination of all the categorical variables' modalities. In addition to all the modalities, each variable will also have a "Total" possibility, which includes all the others. The data cube will contain marginal, conditional and joint empirical distributions...

Examples

```
data("invented_wages")
str(invented_wages)
tmp <- dcc(.data = invented_wages,
           .variables = c("gender", "sector"), .fun = jointfun_)
tmp
str(tmp)
tmp2 <- dcc2(.data = invented_wages,
            .variables = c("gender", "education"),
            .fun = jointfun_,
            order_type = extract_unique2)
tmp2
str(tmp2)

# dcc5 works like dcc2, but has an additional optional argument, .total,
# that can be added to give a name to the groups that include all the
# observations of a variable.
tmp5 <- dcc5(.data = invented_wages,
            .variables = c("gender", "education"),
            .fun = jointfun_,
```

```
            .total = "TOTAL",
            order_type = extract_unique2)
tmp5
```

---

dcc6 *Data cube creation*

Description

Data cube creation

Usage

```
dcc6(.data, .variables, .funs_list = list(n = ~dplyr::n()),
  .total = "Totale", order_type = extract_unique4, .all = TRUE)
```

Arguments

.data	data frame to be processed.
.variables	variables to split data frame by, as a character vector (c("var1", "var2")).
.funs_list	a list of function calls in the form of right-hand formula.
.total	character string with the name to give to the subset of data that includes all the observations of a variable (default: "Totale").
order_type	a function like extract_unique or extract_unique2.
.all	logical, indicating if functions' have to be evaluated on the complete dataset.

Examples

```
dcc6(invented_wages,
     .variables = c("gender", "sector"),
     .funs_list = list(n = ~dplyr::n()),
     .all = TRUE)

dcc6(invented_wages,
     .variables = c("gender", "sector"),
     .funs_list = list(n = ~dplyr::n()),
     .all = FALSE)
```

---

distrr *Estimate and manage empirical distributions*

Description

Tools to estimate and manage empirical distributions, which should work with survey data. One of the main features is the possibility to create data cubes of estimated statistics, that include all the combinations of the variables of interest (see for example functions dcc5() and dcc6()).

extract_unique *Functions to be used in conjunction with 'dcc' family*

Description

Functions to be used in conjunction with 'dcc' family

Usage

```
extract_unique(df)

extract_unique2(df)

extract_unique3(df)

extract_unique4(df)
```

Arguments

df	a data frame

Value

a list whose elements are character vectors of the unique values of each column

Examples

```
data("invented_wages")
tmp <- extract_unique(df = invented_wages[ , c("gender", "sector")])
tmp
str(tmp)
```

Fhat_conditional_ *Weighted empirical cumulative distribution function (ecdf), conditional on one or more variables*

Description

Weighted empirical cumulative distribution function (ecdf), conditional on one or more variables

Usage

```
Fhat_conditional_(.data, .variables, x, weights)
```

Arguments

.data	a data frame
.variables	a character vector with one or more column names
x	character vector of length one, with the name of the numeric column whose conditional ecdf has to be estimated
weights	character vector of length one, indicating the name of the positive numeric column of weights, which will be used in the estimation of the conditional ecdf

Value

a data frame, with the variables used to condition, the x variable, and columns wsum (aggregated sum of weights, based on unique values of x) and Fhat (the estimated conditional Fhat). In addition to data frame, the object will be of classes grouped_df, tbl_df and tbl (from package dplyr)

Examples

```
Fhat_conditional_(mtcars,
                  .variables = c("vs", "am"),
                  x = "mpg",
                  weights = "cyl")
```

Fhat_df_ *Weighted empirical cumulative distribution function (data frame version)*

Description

Weighted empirical cumulative distribution function (data frame version)

Usage

```
Fhat_df_(.data, x, weights)
```

Arguments

.data	a data frame
x	name of the numeric column (as character)
weights	name of the weight column (as character)

Value

a data frame with columns: x, wcum and Fhat

Examples

```
data(invented_wages)
Fhat_df_(invented_wages, "wage", "sample_weights")
```

invented_wages *Invented dataset with wages of men and women.*

Description

This dataset has been completely invented, in order to do some examples with the package.

Usage

```
invented_wages
```

Format

A data frame with 1000 rows and 5 variables:

`gender` gender of the worker (`men` or `women`)

`sector` economic sector where the worker is employed (`secondary` or `tertiary`)

`education` educational level of the worker (`I`, `II` or `III`)

`wage` monthly wage of the worker (in an invented currency)

`sample_weights` sampling weights

Details

Every row of the dataset consists in a fake/invented individual worker. For every individual there is his/her gender, the economic sector in which he/she works, his/her level of education and his/her wage. Furthermore there is a column with the sampling weights.

jointfun_ *A minimal function which counts the number of observations by groups in a data frame*

Description

A minimal function which counts the number of observations by groups in a data frame

Usage

```
jointfun_(.data, .variables, ...)
```

Arguments

`.data`	data frame to be processed
`.variables`	variables to split data frame by, as a character vector (`c("var1", "var2")`).
`...`	additional function calls to be applied on the .data

Value

a data frame, with a column for each cateogrical variable used, and a row for each combination of all the categorical variables' modalities.

Examples

```
data("invented_wages")
tmp <- jointfun_(.data = invented_wages, .variables = c("gender", "sector"))
tmp
str(tmp)
```

`only_joint` *Keeps only joint distribution (removes '.total').*

Description

Removes all the rows where variables have value `.total`.

Usage

```
only_joint(.cube, .total = "Totale", .variables = NULL)
```

Arguments

`.cube`	a datacube with 'Totale' modalities
`.total`	modality to eliminate (filter out) (default: "Totale")
`.variables`	a character vector with the names of the categorical variables

Value

a subset of the data cube with only the combinations of all variables modalities, without the "margins".

Examples

```
data(invented_wages)
str(invented_wages)

vars <- c("gender", "education")
tmp <- dcc2(.data = invented_wages,
            .variables = vars,
            .fun = jointfun_,
            order_type = extract_unique2)
tmp
str(tmp)
only_joint(tmp, .variables = vars)

# Compare dimensions (number of groups)
dim(tmp)
dim(only_joint(tmp, .variables = vars))
```

wq *Empirical weighted quantile*

Description

Empirical weighted quantile

Usage

```
wq(x, weights, probs = c(0.5))
```

Arguments

x	A numeric vector
weights	A vector of (positive) sample weights
probs	a numeric vector with the desired quantile levels (default 0.5, the median)

Value

The weighted quantile (a numeric vector)

References

Ferrez, J., Graf, M. (2007). Enquète suisse sur la structure des salaires. Programmes R pour l'intervalle de confiance de la médiane. (Rapport de méthodes No. 338-0045). Neuchâtel: Office fédéral de statistique.

Examples

```
wq(x = rnorm(100), weights = runif(100))
```

Index