

Package ‘pivotr’

April 29, 2020

Title Pivot Tables with the Tidyverse

Version 0.0.0.9000

Author Ismail Muller <muller.ismail@gmail.com>

Maintainer Ismail Muller <muller.ismail@gmail.com>

Description Computes pivot tables using the ‘dplyr’ package as backend.

Imports dplyr, magrittr, purrr, rlang, tidyr, tidyselect, lifecycle

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

URL <https://ismailmuller.github.io/pivotr/>

BugReports <https://github.com/ismailmuller/pivotr/issues>

Suggests testthat (≥ 2.1.0), covr, pkgdown

RdMacros lifecycle

R topics documented:

cube	1
pvt	2
Index	3

cube	<i>Computes grouped aggregates</i>
------	------------------------------------

Description

Calculates aggregates functions for groups of variables and all their combinations **Experimental**

Usage

```
cube(df, groups = NULL, ..., .totals = "Total")
```

Arguments

<code>df</code>	A dataframe
<code>groups</code>	variables names
<code>...</code>	calculations passed to <code>'dplyr::summarize(...)'</code>
<code>.totals</code>	A string of how output totals are labeled in the rows and cols

Examples

```
cube(mtcars, c(cyl, am), Avg = mean(mpg))
```

<code>pvt</code>	<i>Pivot table</i>
------------------	--------------------

Description

Calculates aggregates in pivot tables **Experimental**

Usage

```
pvt(df, rows = NULL, cols = NULL, ..., .totals = "Total", .keep_total = "all")
```

Arguments

<code>df</code>	A dataframe
<code>rows</code>	A vector of variable names that will be on the rows of the pivot table
<code>cols</code>	A vector of variable names that will be on the columns of the pivot table
<code>...</code>	calculations that are passed to <code>'dplyr::summarise()'</code> via the <code>'pivotr::cube()'</code> function
<code>.totals</code>	A string of how output totals are labeled in the rows and cols
<code>.keep_total</code>	one of "all", "overall", "none"

Value

a tibble

Author(s)

Ismail Muller

Examples

```
pvt(mtcars, N = n())
pvt(mtcars, cyl, am, N = n(), Avg = mean(mpg))
pvt(mtcars, cyl, am, Avg = mean(wt), N = n())
pvt(mtcars, NULL, am, Avg = mean(wt))
pvt(mtcars, c(vs, am), cyl, N = n(), .keep_total = "overall")
```

Index

cube, [1](#)

pvt, [2](#)