

Package: pivotea (via r-universe)

August 30, 2024

Title Create Pivot Table Easily

Version 1.0.2

Description Pivot easily by specifying rows, columns, values and split.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

URL <https://github.com/matutosi/pivotea>,
<https://matutosi.github.io/pivotea/>

LazyData true

Imports dplyr, purrr, rlang, tidyr

Suggests ggplot2, knitr, rmarkdown, spelling, testthat (>= 3.0.0),
tibble

Config/testthat/edition 3

Language en-US

VignetteBuilder knitr

Depends R (>= 2.10)

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

RemoteUrl <https://github.com/matutosi/pivotea>

RemoteRef HEAD

RemoteSha 0d3b372861431a053a5ec0f879d091e699f7ac7d

Contents

add_group_sub	2
extract_col	3
has_col	4
hogwarts	4

na2empty	5
omit_na_cols	6
omit_na_rows	7
pivot	8
replace_col	9
split_force	9
validate_col	10

Index	12
--------------	-----------

add_group_sub	<i>Add sub index within group</i>
---------------	-----------------------------------

Description

Add sub index within group

Usage

```
add_group_sub(df, group, sep = "_", tmp_col = "tmp_col")
```

Arguments

df	A dataframe.
group	A string or string vector. When vector, the first string will be used for adding sub index.
sep	A string for separator.
tmp_col	A string of colnames for temporary use.

Value

A dataframe.

Examples

```
library(dplyr)
add_group_sub(mtcars, c("am", "gear"))
add_group_sub(mtcars, c("cyl", "am"))
```

extract_col	<i>Helper for na_col_omit()</i>
-------------	---------------------------------

Description

Helper for na_col_omit()

Usage

```
extract_col(col, df)
```

Arguments

col	A string or string vector.
df	A dataframe.

Value

A vector.

Examples

```
library(tidyr)
library(dplyr)
library(purrr)
library(ggplot2)
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "teacher", "room"),
        split = c("house", "grade"))
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "room", "house", "grade"),
        split = c("teacher"))
starwars |>
  pivot(row = "homeworld", col = "species", value = "name", split = "sex")
msleep |>
  pivot(row = "vore", col = "conservation", value = "name") |>
  na2empty() |>
  print(n = Inf)
tibble::as_tibble(Titanic) |>
  pivot(row = "Age", col = c("Sex", "Survived"),
        value = "n", split = "Class")
diamonds |>
  pivot(row = "cut", col = "color", value = "price", split = "clarity")
```

has_col *Detect if df has col*

Description

Detect if df has col

Usage

```
has_col(df, col)
```

Arguments

df A dataframe.
col A string or string vector.

Value

A dataframe.

Examples

```
colnames(mtcars)  
has_col(mtcars, c("mpg", "cyl"))  
has_col(mtcars, c("mpg", "foo"))
```

hogwarts *Timetable in Hogwarts School of Witchcraft and Wizardry.*

Description

Timetable in Hogwarts School of Witchcraft and Wizardry.

Usage

```
hogwarts
```

Format

A data frame with 548 rows and 7 variable:

grade Grades in school.

house Houses. G: Gryffindor, S: Slytherin, R: Ravenclaw, and H: Hufflepuff.

wday Abbreviations of day of the week.

hour Hours.

teacher Teachers.

subject Subjects.

room

Examples

```
data(hogwarts)
hogwarts
```

```
na2empty           replace NA character into ""
```

Description

replace NA character into ""

Usage

```
na2empty(df)
```

Arguments

df A dataframe.

Value

A dataframe.

Examples

```
library(tidyr)
library(dplyr)
library(purrr)
library(ggplot2)
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "teacher", "room"),
        split = c("house", "grade"))
hogwarts |>
  pivot(row = "hour", col = "wday",
```

```

      value = c("subject", "room", "house", "grade"),
      split = c("teacher"))
starwars |>
  pivot(row = "homeworld", col = "species", value = "name", split = "sex")
msleep |>
  pivot(row = "vore", col = "conservation", value = "name") |>
  na2empty() |>
  print(n = Inf)
tibble::as_tibble(Titanic) |>
  pivot(row = "Age", col = c("Sex", "Survived"),
        value = "n", split = "Class")
diamonds |>
  pivot(row = "cut", col = "color", value = "price", split = "clarity")

```

omit_na_cols

Remove all NA cols

Description

Remove all NA cols

Usage

```
omit_na_cols(df)
```

Arguments

df A dataframe.

Value

A dataframe.

Examples

```

library(tidyr)
library(dplyr)
library(purrr)
library(ggplot2)
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "teacher", "room"),
        split = c("house", "grade"))
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "room", "house", "grade"),
        split = c("teacher"))
starwars |>
  pivot(row = "homeworld", col = "species", value = "name", split = "sex")

```

```
msleep |>
  pivot(row = "vore", col = "conservation", value = "name") |>
  na2empty() |>
  print(n = Inf)
tibble::as_tibble(Titanic) |>
  pivot(row = "Age", col = c("Sex", "Survived"),
        value = "n", split = "Class")
diamonds |>
  pivot(row = "cut", col = "color", value = "price", split = "clarity")
```

omit_na_rows

Remove all NA rows

Description

Remove all NA rows

Usage

```
omit_na_rows(df)
```

Arguments

df A dataframe.

Value

A dataframe.

Examples

```
library(tidyr)
library(dplyr)
library(purrr)
library(ggplot2)
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "teacher", "room"),
        split = c("house", "grade"))
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "room", "house", "grade"),
        split = c("teacher"))
starwars |>
  pivot(row = "homeworld", col = "species", value = "name", split = "sex")
msleep |>
  pivot(row = "vore", col = "conservation", value = "name") |>
  na2empty() |>
  print(n = Inf)
```

```
tibble::as_tibble(Titanic) |>
  pivot(row = "Age", col = c("Sex", "Survived"),
        value = "n", split = "Class")
diamonds |>
  pivot(row = "cut", col = "color", value = "price", split = "clarity")
```

pivot

Pivot easily by specifying rows, columns, values and split.

Description

Pivot easily by specifying rows, columns, values and split.

Usage

```
pivot(df, row, col, value, split = NULL, sep = "_", rm_empty_df = TRUE)
```

Arguments

df	A dataframe.
row, value	A string or string vector.
col	A string or string vector.
split	A string or string vector.
sep	A string for separator.
rm_empty_df	A logical for removing empty df.

Value

A dataframe.

Examples

```
library(tidyr)
library(dplyr)
library(purrr)
library(ggplot2)
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "teacher", "room"),
        split = c("house", "grade"))
hogwarts |>
  pivot(row = "hour", col = "wday",
        value = c("subject", "room", "house", "grade"),
        split = c("teacher"))
starwars |>
  pivot(row = "homeworld", col = "species", value = "name", split = "sex")
msleep |>
```



```
  pivot(row = "vore", col = "conservation", value = "name") |>
  na2empty() |>
  print(n = Inf)
tibble::as_tibble(Titanic) |>
  pivot(row = "Age", col = c("Sex", "Survived"),
        value = "n", split = "Class")
diamonds |>
  pivot(row = "cut", col = "color", value = "price", split = "clarity")
```

replace_col	<i>Replace a col with a data.frame.</i>
-------------	---

Description

Replace a col with a data.frame.

Usage

```
replace_col(df, replace)
```

Arguments

df, replace A dataframe.

Value

A dataframe.

Examples

```
(state <- tibble::tibble(state = state.name, area = state.area))
(abb <- tibble::tibble(state = state.name, abb = state.abb))
replace_col(state, abb)
```

split_force	<i>Split by force with "" when split is NULL</i>
-------------	--

Description

Split by force with "" when split is NULL

Usage

```
split_force(df, split)
```

Arguments

df A dataframe.
split A string or string vector.

Value

A dataframe.

Examples

```
split_force(mtcars, split = NULL)  
split_force(mtcars, split = c("cyl"))
```

validate_col

Validate col

Description

Validate col

Usage

```
validate_col(df, col)
```

Arguments

df A dataframe.
col A string or string vector.

Value

A dataframe.

Examples

```
library(tidyr)  
library(dplyr)  
library(purrr)  
library(ggplot2)  
hogwarts |>  
  pivot(row = "hour", col = "wday",  
        value = c("subject", "teacher", "room"),  
        split = c("house", "grade"))  
hogwarts |>  
  pivot(row = "hour", col = "wday",  
        value = c("subject", "room", "house", "grade"),  
        split = c("teacher"))
```

```
starwars |>
  pivot(row = "homeworld", col = "species", value = "name", split = "sex")
msleep |>
  pivot(row = "vore", col = "conservation", value = "name") |>
  na2empty() |>
  print(n = Inf)
tibble::as_tibble(Titanic) |>
  pivot(row = "Age", col = c("Sex", "Survived"),
        value = "n", split = "Class")
diamonds |>
  pivot(row = "cut", col = "color", value = "price", split = "clarity")
```

Index

* datasets

hogwarts, 4

add_group_sub, 2

extract_col, 3

has_col, 4

hogwarts, 4

na2empty, 5

omit_na_cols, 6

omit_na_rows, 7

pivot, 8

replace_col, 9

split_force, 9

validate_col, 10