Package: datacult (via r-universe)

June 9, 2026

Title Exploratory Data Analysis for Public Policy Applied to Culture

Version 0.1.0

Description Implementation of frequency tables and bar charts for qualitative variables and checkbox fields. This package implements tables and charts used in reports at Funarte (National Arts Foundation) and OBEC (Culture and Creative Economy Observatory) in Brazil, and its main purpose is to simplify the use of R for people with a background in the humanities and arts. Examples and details can be viewed in this presentation from 2026: <https://formacao2026.netlify.app/assets/modulo_3/modulo3#/title-slide>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Imports dplyr, forcats, ggplot2, janitor, rlang, scales

NeedsCompilation no

Author Gilberto Sassi [aut, cre]

Maintainer Gilberto Sassi <sassi.pereira.gilberto@gmail.com>

Depends R (>= 4.1.0)

Config/pak/sysreqs libicu-dev

Repository https://gilberto-sassi.r-universe.dev

Date/Publication 2026-01-27 08:40:15 UTC

RemoteUrl https://github.com/cran/datacult

RemoteRef HEAD

RemoteSha 802d2640c067a6da5501c41c89b3962332d11969

Contents

`bar_chart` *Bar chart with horizontal bar and data label.*

Description

Axis-x has the frequency and data label has the percent value.

Usage

```
bar_chart(
  data,
  variable,
  axis_title = "",
  fill = "blue",
  axis_title_axis_size = 22,
  axis_text_axis_size = 22,
  data_label_size = 15,
  sort = TRUE
)
```

Arguments

`data`	a dataframe object
`variable`	variable name
`axis_title`	axis y label. Defaults to "".
`fill`	bar color. Character scale with hexidecimal color ou named color. Defaults to "blue".
`axis_title_axis_size`	integer scalar. Size of axis labels. Defaults to 22.
`axis_text_axis_size`	integer scalar. Size of values annotated in axis. Defaults to 22.
`data_label_size`	integer scalar. Size of data labels. Defaults to 15
`sort`	logical scalar. If TRUE, the are ordered using frequency. Defaults to TRUE.

Value

a ggplot2 object

Examples

```
df <- data.frame(var = sample(c("A", "B"),
 size = 100, replace = TRUE))
bar_chart(df, var)
```

`bar_chart_checkbox` *Bar chart with frequency and data label with percent.*

Description

This function builds a bar chart with ggplot2 for a field with checkbox (user can select 2 or more options). Each option is one separated column.

Usage

```
bar_chart_checkbox(
  data,
  columns,
  sucess = "checked",
  labels = NULL,
  axis_title = "",
  fill = "blue",
  axis_title_axis_size = 22,
  axis_text_axis_size = 22,
  data_label_size = 15,
  sort = TRUE
)
```

Arguments

`data`	dataframe object
`columns`	character vector. Columns to count.
`sucess`	character scalar. Category indicating the sucess. Defaults to "checked".
`labels`	character vector.Label of each category. Defaults to NULL.
`axis_title`	axis y label. Defaults to "".
`fill`	bar color. Character scale with hexidecimal color ou named color. Defaults to "blue".
`axis_title_axis_size`	integer scalar. Size of axis labels. Defaults to 22.
`axis_text_axis_size`	integer scalar. Size of values annotated in axis. Defaults to 22.
`data_label_size`	integer scalar. Size of data labels. Defaults to 15
`sort`	logical scalar. If TRUE, the are ordered using frequency. Defaults to TRUE.

Value

a ggplot2 object

Examples

```
df <- data.frame(
 x1 = c("checked", "checked", "unchecked"),
 x2 = c("checked", "unchecked", "checked")
)
bar_chart_checkbox(df, c("x1", "x2"))
```

`n_missing` *Number of missing values in a vector.*

Description

Number of missing values in a vector.

Usage

```
n_missing(x)
```

Arguments

`x`	an atomic vector

Value

returns a integer scalar

Examples

```
n_missing(c(1, NA, 3))
```

`n_no_missing` *Nnumber of non missing values in a vector.*

Description

Nnumber of non missing values in a vector.

Usage

```
n_no_missing(x)
```

Arguments

x	an atomic vector

Value

scalar integer

Examples

```
n_no_missing(c(1, NA, 3))
```

---

tab_freq_checkbox *Frequency for Checkbox Fields.*

Description

Each category (or option) is a variable in the dataframe, and we count the number of `sucess` in each column. The column `percent` is the ratio between this count by the observation number.

Usage

```
tab_freq_checkbox(
  data,
  columns,
  sucess = "checked",
  labels = NULL,
  variable_name = NULL
)
```

Arguments

data	dataframe object
columns	character vector. Columns to count.
sucess	character scalar. Category indicating the sucess. Defaults to "checked".
labels	character vector.Label of each category. Defaults to NULL.
variable_name	character scalar. Name of field in the form. Defaults to NULL.

Details

Missing values are ignored.

Value

return a dataframe with `n` and `percent`

Examples

```
data <- data.frame(
  x1 = c("checked", "checked", "unchecked"),
  x2 = c("checked", "unchecked", "checked")
)
tab_freq_checkbox(data, c("x1", "x2"))
```

`tab_freq_cont` *Generate a frequency table to a continuous variable.*

Description

Bin the continuous variable, and count the occurrence of each interval.

Usage

```
tab_freq_cont(
  data,
  variable,
  breaks = NULL,
  labels = NULL,
  include_lowest = TRUE,
  right = FALSE
)
```

Arguments

`data`	dataframe object
`variable`	variable name as character
`breaks`	either a integer number or vector of number. Defaults to `NULL`
`labels`	label for each interval. Defaults to interval names using brackes and parentheses. Defaults to `NULL`.
`include_lowest`	logical value. If `TRUE`, the lower limit of interval belongs to the bin. Defaults to `TRUE`.
`right`	logical value. If `TRUE`, the upper limit of interval belongs to the bin. Defaults to `FALSE`.

Details

if `breaks = NULL`, then `ceiling(1 + log2(n))`, where `n` is the sample size.

Value

a dataframe with distribution frequency

Examples

```
tab_freq_cont(iris, "Sepal.Width")
```

Index