Package 'glancedata'

November 23, 2019

Type Package

Title Generate tables and plots to get summaries of data

Version 1.0.1

Date 2019-11-22

Description Generate data frames for all the variables with some sum-
maries and also some plots for numerical variables.
Several functions from the 'tidyverse' and 'GGally' packages are used.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.0.0

Imports GGally, dplyr, forcats, ggplot2, openxlsx, purrr, tibble,
tidyr, gridExtra

Suggests testthat (>= 2.1.0), covr, spelling, knitr, rmarkdown, skimr

Depends R (>= 3.5)

Language en-US

VignetteBuilder knitr

NeedsCompilation no

Author Guillermo Basulto-Elias [aut, cre]

Maintainer Guillermo Basulto-Elias <guillermobasulto@gmail.com>

Repository CRAN

Date/Publication 2019-11-22 23:10:05 UTC

R topics documented:

---

glance_data   *Glance Data*

Description

Provides a summary of data with the the following columns:

name Name of the column.

type Type of the column, equal to "numerical", "logical", "factor", "categorical", or "NA only".

distinct_values Count of distinct values. It ignores NA values. Thus, if a columns only has NAs, then the value of this field will be zero.

minimum Minimum of numerical columns excluding NA values.

median Median of numerical columns excluding NA values.

maximum Maximum of numerical columns excluding NA values.

mean Mean of numerical variables. It ignores NAs.

sd Standard deviation of numerical variables. It ignores NAs.

na_proportion Proportion of NAs.

count Tally of values if the column has 5 values at most. This value (5) can be modified with the parameter limit2tally.

sample_values Sample of (different) values in each column.

Usage

```
glance_data(x, limit2tally = 20)
```

Arguments

x	A dataframe with named columns.
limit2tally	One of the summaries is a tally of the distinct values on each column. If there are too many different values in a column, this summary would be meaningless. This limit2tally is the limit of distinct values to tally. If there are more than that it returns "Too many unique values".

Value

A tibble.

Author(s)

Guillermo Basulto-Elias

Examples

```
glance_data(iris)
```

glance_data_in_workbook

Export glance data to Excel Workbook

Description

It is similar to [glance_data](#) but instead of creating a single dataframe, it creates a list of seven dataframes:

all Same output as [glance_data](#).

summary A tally of columns by type. A dataframe with two columns: column type and count.

all_nas Summary of columns with only NAs.

single_value Summary of columns with a single value besides NAs. It might be numeric or character.

binary Summary of columns with two values besides NAs. It might be numerical or categorical.

numerical Summary of all numerical columns when there are more than two possible values.

categorical Summary of all categorical columns when there are more than two possible values.

Usage

```
glance_data_in_workbook(dataframe, filename = NULL, limit2tally = 20)
```

Arguments

dataframe	Dataframe to be summarized
filename	File name of the Excel file, e.g., "mydataglance.xlsx". By default, no file name is provided and, therefore, no Excel is created.
limit2tally	One of the summaries is a tally of the distinct values on each column. If there are too many different values in a column, this summary would be meaningless. This limit2tally is the limit of distinct values to tally. If there are more than that it returns "Too many unique values".

Details

If a XLSX file name is provided, it will create a file with seven sheets, corresponding to the seven dataframes above.

Finally, the last elements of the list (and the last five sheets if the filename is provided), are disjoint. That is, if a the summary of a column is included in "binary", it will not be included in "numerical".

Value

A list

Author(s)

Guillermo Basulto-Elias

Examples

```
## Create a list of dataframes. If you provide the `filename`
## parameter to be equal to, say, "myglance.xlsx", then it will
## create an Excel workbook and place the content of each
## dataframe in a separate sheet.
glance_data_in_workbook(iris)
```

---

plot_discrete_vars   *Plot Discrete Variables*

Description

Creates a grid of bar plots for variables with at most nvalues (15 is the default) different values.

Usage

```
plot_discrete_vars(x, nvalues = 15, sort_by_frequency = FALSE)
```

Arguments

x   Data frame, tibble or matrix with data.

nvalues   Integer. It creates a bar plot for each variable with at most this amount of differ-
                   ent values. The default value is 15.

sort_by_frequency

                   Logical. It casts every variable to factor and then reorders the levels by fre-
                   quency before generating the plot.

Value

A GGplot object.

Author(s)

Guillermo Basulto-Elias

Examples

```
library(glancedata)

plot_discrete_vars(mtcars)
plot_discrete_vars(mtcars, sort_by_frequency = TRUE)
plot_discrete_vars(iris)

## The following two examples give a warning because they don't
## have any variable with fewer than 15 different values.
plot_discrete_vars(cars)
plot_discrete_vars(state.x77)
```

---

plot_numerical_vars *Plot Continuous Variables*

Description

Graphical summaries of numerical variables using functions from ggplot2 and GGally.

Usage

```
plot_numerical_vars(x, plot_type)
```

Arguments

x   Data frame which may include continuous and discrete variables. Non-continuous
            variables are ignored.

plot_type   Plot type. Current options:

            "pairwise" Calls ggpairs to get plots of pairwise differences. Avoid it if there
                are too many numerical variables.
            "density" Calls geom_density and geom_rug.
            "histogram" Calls geom_histogram and geom_rug.
            "violin" Calls geom_violin and geom_jitter.
            "boxplot" Calls geom_boxplot.
            "qqplot" Calls stat_qq_line and stat_qq.

Value

A ggplot object.

Author(s)

Guillermo Basulto-Elias

Examples

```
library(glancedata)

plot_numerical_vars(iris, "pairwise")
plot_numerical_vars(iris, "density")
plot_numerical_vars(iris, "boxplot")
plot_numerical_vars(iris, "violin")
plot_numerical_vars(iris, "histogram")
plot_numerical_vars(iris, "qqplot")
```

Index