Package ‘ibawds’

January 4, 2022

Type Package
Title Functions and Datasets for the Data Science Course at IBAW
Version 0.3.2
Author Stefan Lanz
Maintainer Stefan Lanz <slanz1137@gmail.com>
Description A collection of useful functions and datasets for the Data Science Course at IBAW in Lucerne.
License MIT + file LICENSE
URL https://stibu81.github.io/ibawds/
BugReports https://github.com/stibu81/ibawds/issues
Encoding UTF-8
LazyData true
Language en-GB
RoxygenNote 7.1.2
Depends R (>= 3.6.0), dslabs
Imports stats, grDevices, methods, rlang, ggplot2, scales, dplyr, stringr, magrittr, Ecdat, kableExtra
Suggests tidyverse, rmarkdown, knitr, reshape2, lubridate, ggrepel, writexl, cowplot, DT, gutenbergr, tidytext, rvest, Lahman, pdftools, HistData, titanic, BiocManager, waldo, usethis, vdiffr, testthat (>= 3.0.0), covr
Config/testthat/edition 3
NeedsCompilation no
Repository CRAN
Date/Publication 2022-01-04 21:00:08 UTC
R topics documented:

- bills .......................................................... 2
- define_latex_stats ........................................... 3
- distribution_plot ........................................... 3
- get_cran_history ............................................ 5
- grading_tables ............................................. 6
- install_ibawds .............................................. 7
- mtcars2 ....................................................... 7
- n_available_packages ....................................... 8
- rand_with_cor ............................................... 8
- rescale ....................................................... 9
- seatbelts ..................................................... 10
- set_slide_options .......................................... 11

Index .......................................................... 13

---

**bills** Summarised Data on Restaurant Bills

**Description**

Summary of data on restaurant bills from the dataset `reshape2::tips`. Labels are in German.

**Usage**

`bills`

**Format**

A data frame with 8 rows and 4 variables:

- **sex**  sex of the bill payer
- **time**  time of day
- **smoker**  whether there were smokers in the party
- **mean_bill**  mean of all the bills in dollars
**define_latex_stats**

**Define LaTeX commands for statistical symbols**

**Description**

Add the definitions for various useful LaTeX equation symbols for statistics to an RMarkdown document.

**Usage**

```r
define_latex_stats()
```

**Details**

Run this function from within a code chunk in a RMarkdown document with options `results = "asis"` and `echo = FALSE` (see "Examples"). It only works for pdf output.

It defines the following macros: \( \E \), \( \P \), \( \Var \), \( \Cov \), \( \Cor \), \( \SD \), \( \SE \), \( \Xb \), \( \Yb \).

**Value**

The function returns `NULL` invisibly. The command definitions are output as a side effect.

**Examples**

```r
## Not run:
# add this code chunk to a RMarkdown document
```r
```r
```r
```
```r
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
`---`
Usage

distribution_plot(
  fun,
  range,
  ..., 
  points = NULL,
  var = "x",
  title = "Verteilungsfunktion"
)

density_plot(
  fun, 
  range,
  ..., 
  from = NULL,
  to = NULL,
  points = NULL,
  var = "x",
  title = "Dichte"
)

Arguments

fun    a density or distribution function that takes quantiles as its first argument.
range  numeric vector of length two giving the range of quantiles to be plotted.
...    further arguments that are passed to fun().
points numeric vector giving quantiles where the function should be marked with a red dot.
var    character giving the name of the quantile variable. This is only used to label the axes.
title  character giving the title of the plot
from, to numeric values giving start and end of a range where the area under the density will be shaded. If only one of the two values is given, the shading will start a negative infinity or go until positive infinity, respectively.

Value

ggplot object

Examples

# plot density of the normal distribution
density_plot(dnorm, c(-5, 7),
  mean = 1, sd = 2,
  to = 3)

# plot distribution function of the Poisson distribution
get_cran_history

distribution_plot(ppois, c(0, 12),
    lambda = 4,
    points = c(2, 6, 10),
    var = "y")

---

get_cran_history  History of the Number of Available CRAN Packages

Description

Get a data frame containing the number of packages available for historic dates back to 21 June 2001.

Usage

get_cran_history()

Details

Data on the number of packages on CRAN between 2001-06-21 and 2014-04-13 is obtained from Ecdat::CRANpackages. This data was collected by John Fox and Spencer Graves. Intervals between data points are irregularly spaced.

Newer data was obtained using the function n_available_packages() which extracts the information from CRAN snapshots on MRAN. One data point per quarter is available starting on 2014-10-01.

Value

a tibble with columns date and n_packages

Examples

library(ggplot2)
cran_history <- get_cran_history()
ggplot(cran_history, aes(x = date, y = n_packages)) +
    geom_point()
**grading_tables**  
*Tables Used for Grading the Papers*

**Description**

These functions create two tables that can be used for the grading of the student’s papers.

**Usage**

```r
create_minreq_table(repro, n_tab, n_plot_kinds, n_plots, n_stat)
create_grading_table(p_text, p_tab, p_plot, p_code, p_stat)
```

**Arguments**

- `repro`: logical, is the paper reproducible?
- `n_tab`: integer, number of tables
- `n_plot_kinds`: integer, number of different kinds of plots
- `n_plots`: integer, number of plots
- `n_stat`: integer, number of statistical computations
- `p_text`: numeric between 0 and 5, points given for the text
- `p_tab`: numeric between 0 and 5, points given for the tables
- `p_plot`: numeric between 0 and 5, points given for the plots
- `p_code`: numeric between 0 and 5, points given for the code
- `p_stat`: numeric between 0 and 5, points given for the statistic computations

**Details**

The tables are created using `knitr::kable()` and `kableExtra` is used for additional styling. `create_minreq_table()` creates a table that checks that the minimal requirements are satisfied:

- the paper must be reproducible
- there must be at least one table and two kinds of plots
- there must be at least 5 plots and tables
- there must be at least two statistical computations

The table lists for each of those requirement whether it is satisfied or not.

`create_grading_table()` creates a table that gives grades in percent for each of five categories:

- Text
- Tables
- Plots
- Code
- Statistical computations

In each category, up to five points may be awarded. The last row of the table gives the percentage over all categories.
install_ibawds

**Value**

both functions return an object of class kableExtra.

---

**install_ibawds**

*Install the R-Packages Required for the Course*

---

**Description**

A number of R-packages are used in the courses and the video lectures. They are also dependencies of this package. Use `install_ibawds()` to install the packages that are not yet installed.

**Usage**

`install_ibawds()`

**Details**

This function checks whether all the packages that `ibawds` depends on, imports or suggests are installed. In interactive sessions, it either informs the user that all packages are installed or asks to install missing packages. The function relies on `rlang::check_installed()`.

**Value**

nothing or NULL invisibly

---

**mtcars2**

*Dataset mtcars without row names*

---

**Description**

In the `mtcars` dataset, the names of the car models are stored as row names. However, when working with `ggplot2` and other packages from the `tidyverse`, it is convenient to have all data in columns. `mtcars2` is a variant of `mtcars` that contains car models in a column instead of storing them as row names.

**Usage**

`mtcars2`

**Format**

A data frame with 32 rows and 12 variables. The format is identical to `mtcars` and details can be found in its documentation. The only difference is that the car model names are stored in the column `model` instead of row names.
### n_available_packages  
*Number of Available R Packages and R Versions from MRAN*

**Description**

MRAN has an archive of Snapshots of CRAN dating back to September 17 2014. These functions return the number of available packages and the available R version according to the snapshot of https://cran.r-project.org on MRAN.

**Usage**

```r
n_available_packages(date = Sys.Date())
available_r_version(date = Sys.Date())
```

**Arguments**

- `date`: the date of the snapshot to be used. It can be a `Date` object or a character in the format `%Y-%m-%d`.

**Details**

MRAN has data starting from September 17 2014. Data for a few selected dates before September 17 2014 can be obtained from the dataset `Ecdat::CRANpackages`. A more complete dataset ranging from 2001 until today can be obtained with `get_cran_history()`.

Note that for some dates there is no snapshot on MRAN. The function will return an error in those cases.

**Value**

- the number of available packages as an integer or the R version number as a character

**See Also**

- `get_cran_history()`

### rand_with_cor  
*Create a Random Vector With Fixed Correlation With Another Vector*

**Description**

`rand_with_cor()` creates a vector of random number that has correlation `rho` with a given vector `y`. Also mean and standard deviation of the random vector can be fixed by the user. By default, they will be equal to the mean and standard deviation of `y`, respectively.
Usage

\texttt{rand\_with\_cor(y, rho, mu = mean(y), sigma = sd(y))}

Arguments

\textit{y} \quad \text{a numeric vector}
\textit{rho} \quad \text{numeric value between -1 and 1 giving the desired correlation.}
\textit{mu} \quad \text{numeric value giving the desired mean}
\textit{sigma} \quad \text{numeric value giving the desired standard deviation}

Value

a vector of the same length as \(y\) that has correlation \(\rho\) with \(y\).

Source

This solution is based on an answer by \texttt{whuber} on Cross Validated.

Examples

\begin{verbatim}
x <- runif(1000, 5, 8)

# create a random vector with positive correlation
y1 <- rand_with_cor(x, 0.8)
all.equal(cor(x, y1), 0.8)

# create a random vector with negative correlation
# and fixed mean and standard deviation
y2 <- rand_with_cor(x, -0.3, 2, 3)
all.equal(cor(x, y2), -0.3)
all.equal(mean(y2), 2)
all.equal(sd(y2), 3)
\end{verbatim}

\textbf{rescale}

\textit{Rescale Mean And/Or Standard Deviation of a Vector}

Description

Rescale Mean And/Or Standard Deviation of a Vector

Usage

\texttt{rescale(x, mu = mean(x), sigma = sd(x))}
Arguments

- `x` numeric vector
- `mu` numeric value giving the desired mean
- `sigma` numeric value giving the desired standard deviation

Details

By default, mean and standard deviation are not changed, i.e., `rescale(x)` is identical to `x`. Only if a value is specified for `mu` and/or `sigma` the mean and/or the standard deviation are rescaled.

Value

A numeric vector with the same length as `x` with mean `mu` and standard deviation `sigma`.

Examples

```r
x <- runif(1000, 5, 8)

# calling rescale without specifying mu and sigma doesn't change anything
all.equal(x, rescale(x))

# change the mean without changing the standard deviation
x1 <- rescale(x, mu = 3)
all.equal(mean(x1), 3)
all.equal(sd(x1), sd(x))

# rescale mean and standard deviation
x2 <- rescale(x, mu = 3, sigma = 2)
all.equal(mean(x2), 3)
all.equal(sd(x2), 2)
```

seatbelts

*Road Casualties in Great Britain 1969-84*

Description

Extract of the data in the `Seatbelts` dataset as a data frame. The original dataset is a multiple time series (class `mts`). Labels are in German.

Usage

`seatbelts`
set_slide_options

Description

Set options for ggplot plots and tibble outputs for IBAW slides.

Usage

```r
set_slide_options(
  ggplot_text_size = 22,
  ggplot_margin_pt = rep(10, 4),
  tibble_print_max = 12,
  tibble_print_min = 8
)
```

Arguments

- `ggplot_text_size` 
  Text size to be used in ggplot2 plots. This applies to all texts in the plots.
- `ggplot_margin_pt` 
  numeric vector of length 4 giving the sizes of the top, right, bottom, and left margins in points.
- `tibble_print_max` 
  Maximum number of rows printed for a tibble. Set to Inf to always print all rows.
- `tibble_print_min` 
  Number of rows to be printed if a tibble has more than `tibble_print_max` rows.

Details

The function uses `ggplot2::theme_update()` to modify the default theme for ggplot and `options()` to set base R options that influence the printing of tibbles.

Note that if you make changes to these options in a R Markdown file, you may have to delete the `knitr` cache in order for the changes to apply.
Value

A named list (invisibly) with two elements containing the old values of the options for the ggplot theme and the base R options, respectively. These can be used to reset the ggplot theme and the base R options to their previous values.
Index

* datasets
  bills, 2
  mtcars2, 7
  seatbelts, 10

available_r_version
  (n_available_packages), 8

bills, 2

create_grading_table (grading_tables), 6
create_minreq_table (grading_tables), 6

define_latex_stats, 3
density_plot (distribution_plot), 3
distribution_plot, 3

Ecdat::CRANpackages, 5, 8

get_cran_history, 5
get_cran_history(), 8
ggplot2::theme_update(), 11
grading_tables, 6

install_ibawds, 7

kableExtra, 6
knitr::kable(), 6

mtcars, 7
mtcars2, 7

n_available_packages, 8
n_available_packages(), 5

options(), 11

rand_with_cor, 8
rescale, 9
reshape2::tips, 2
rlang::check_installed(), 7

Seatbelts, 10
seatbelts, 10
set_slide_options, 11