Package 'edar'

May 24, 2026

Type Package

Title Convenient Functions for Exploratory Data Analysis

Version 0.0.7

Description
A collection of convenient functions to facilitate common tasks in exploratory data analysis.
Some common tasks include generating summary tables of variables,
displaying tables as a 'flextable' or a 'kable' and visualising variables using 'ggplot2'.
Labels stating the source file with run time can be easily generated
for annotation in tables and plots.

License MIT + file LICENSE

Encoding UTF-8

URL https://github.com/soutomas/edar/

BugReports https://github.com/soutomas/edar/issues

RoxygenNote 7.3.3

Imports dplyr, flextable, ggplot2, ggpubr, grDevices, janitor,
kableExtra, knitr, magrittr, patchwork, purrr, rlang,
rstudioapi, scales, stats, tidyr, tidyselect, xgxr

Suggests gt

Depends R (>= 4.2.0)

NeedsCompilation no

Author Tomas Sou [aut, cre] (ORCID: <https://orcid.org/0000-0002-7570-5545>)

Maintainer Tomas Sou <tomas.sou@carexer.com>

Repository CRAN

Date/Publication 2026-05-24 16:10:02 UTC

Contents

`fc` *Copy files and rename with date*

Description

Copy files to destination and rename with date and a tag as desired.

Usage

```
fc(..., des = "", tag = "", td = TRUE)
```

Arguments

`...`	<chr> A vector of file paths of the source files to copy and rename.
`des`	<chr> Destination folder. "." to rename files at the current location.
`tag`	<chr> Tag to the filename.
`td`	<lgl> TRUE to add today (yymmdd) to the filename.

Value

A logical vector indicating if the operation succeeded for each of the files.

Examples

```
## Not run:
# Copy a file to home directory
tmp = tempdir()
fc("f1.R","f2.R",des=tmp)

## End(Not run)
```

`ft` *flextable wrapper*

Description

Sugar function for default flextable output.

Usage

```
ft(d, fnote = NULL, ttl = NULL, sig = 8, dig = 2, src = 0, omit = "")
```

Arguments

`d`	<dfr> A data frame.
`fnote`	<chr> Footnote.
`ttl`	<chr> Title.
`sig`	<int> Number of significant digits to compute.
`dig`	<int> Number of decimal places to display.
`src`	<int> Either 1 or 2 to add source label over 1 or 2 lines.
`omit`	<chr> Text to omit from the source label.

Value

A flextable object.

Examples

```
mtcars |> head() |> ft()
mtcars |> head() |> ft(src=1)
mtcars |> head() |> ft("Footnote")
mtcars |> head() |> ft("Footnote",src=1)
mtcars |> head() |> ft(sig=2,dig=1)
```

ft_def *flextable defaults*

Description

Sugar function to set flextable defaults. The arguments are passed to `flextable::set_flextable_defaults()`.

Usage

```
ft_def(
  show = FALSE,
  font = "Calibri Light",
  fsize = 10,
  pad = 3,
  na = "",
  nan = "",
  ...
)
```

Arguments

show	<lgl> TRUE to show values after the update.
font	<chr> Font family - for `font.family`.
fsize	<int> Font size (in point) - for `font.size`.
pad	<int> Padding space around text - for `padding`.
na	<chr> A value to display instead of NA - for `na_str`
nan	<chr> A value to display instead of NaN - for `nan_str`
...	Additional arguments to pass to `flextable::set_flextable_defaults()`

Value

A list containing previous default values.

See Also

`flextable::set_flextable_defaults()`.

Examples

```
## Not run:
ft_def()

## End(Not run)
```

geo_cv *Geometric coefficient of variation*

Description

Compute geometric coefficient of variation (GCV)

Usage

```
geo_cv(x)
```

Arguments

x	<num> A vector of values.

Value

Geometric coefficient of variation

Examples

```
geo_cv(rlnorm(10))
```

geo_mean *Geometric mean*

Description

Compute geometric mean.

Usage

```
geo_mean(x)
```

Arguments

x	<num> A vector of values.

Value

Geometric mean.

Examples

```
geo_mean(rlnorm(10))
```

geo_sd *Geometric standard deviation*

Description

Compute geometric standard deviation (GSD)

Usage

```
geo_sd(x)
```

Arguments

x	<num> A vector of values.

Value

Geometric standard deviation

Examples

```
geo_sd(rlnorm(10))
```

ggbox *Box plot wrapper for categorical covariates*

Description

Create box plots for a chosen variable by all discrete covariates in a dataset. Numeric variables will be dropped, except the chosen variable to plot.

Usage

```
ggbox(d, var, cats, alpha = 0.1, show = TRUE, nsub = TRUE, ...)
```

Arguments

d	<dfr> A data frame.
var	<var> A variable to plot as unquoted name.
cats	<var> Optional. Categorical variables to plot as a vector of unquoted names.
alpha	<num> Alpha value for ggplot2::geom_jitter.
show	<lgl> TRUE to show data using ggplot2::geom_jitter.
nsub	<lgl> Show number of observations.
...	Additional arguments for ggplot2::geom_boxplot.

Value

A ggplot object.

Examples

```
d = mtcars |> mutate(across(c(am,carb,cyl,gear,vs),factor))
d |> ggbox(mpg)
d |> ggbox(mpg,alpha=0.5)
d |> ggbox(mpg,show=FALSE)
d |> ggbox(mpg,nsub=FALSE)
d |> ggbox(mpg,c(cyl,vs))
```

---

gghist *Histogram wrapper for continuous covariates*

Description

Create histograms for all numeric variables in a dataset. Non-numeric variables will be dropped.

Usage

```
gghist(d, cols, bins = 30, nsub = TRUE, ...)
```

Arguments

d	<dfr> A data frame.
cols	<var> Optional. Columns to plot as a vector of unquoted names.
bins	<int> Number of bins.
nsub	<lgl> Show number of observations.
...	Additional arguments for ggplot2::geom_histogram.

Value

A ggplot object.

Examples

```
iris |> gghist()
iris |> gghist(c(Sepal.Width,Sepal.Length))
```

ggout *Save ggplot with output path*

Description

Save ggplot with output path

Usage

```
ggout(plt, fpath, lab = "", omit = "", ...)
```

Arguments

plt	A ggplot object.
fpath	<chr> File path to save output and add to label.
lab	<chr> Custom label to use instead of fpath.
omit	<chr> Text to omit from the label.
...	Other arguments to pass to ggplot2::ggsave().

Value

The file path of the output.

See Also

ggplot2::ggsave()

Examples

```
## Not run:
fpath = "../output.png"
iris |> gghist() |> ggout(fpath)

## End(Not run)
```

ggsrc *Add source file label to a ggplot object*

Description

Add a label with the current source file path and run time to a ggplot object.

Usage

```
ggsrc(plt, span = 1, size = 8, col = "grey55", lab = NULL, omit = "")
```

Arguments

plt	A ggplot object.
span	<num> Number of lines: either 1 or 2.
size	<num> Text size.
col	<chr> Colour of the text.
lab	<chr> Custom label to use instead of the default.
omit	<chr> Text to omit from the label.

Value

A ggplot object with the added label.

Examples

```
p = mtcars |> ggxy(mpg,hp)
p |> ggsrc()
p |> ggsrc(lab="My label")
p |> ggsrc(lab="My label",omit="My ")
```

---

ggtpp *Time-profile plot wrapper*

Description

Create plots for time profile data such as PK and PD plots.

Usage

```
ggtpp(
  d,
  x,
  y,
  id,
  ...,
  nsub = TRUE,
  logx = FALSE,
  logy = FALSE,
  alpha_point = 0.2,
  alpha_line = 0.1,
  xlab = NULL,
  ylab = NULL,
  ttl = NULL,
  sttl = NULL,
  cap = NULL
)
```

Arguments

d	<dfr> A data frame.
x, y	<var> Variables for x- and y-axis as unquoted names
id	<var> Variable for grouping ID such as subject ID as unquoted name.
...	Arguments to pass to ggplot2::aes for additional mapping.
nsub	<lgl> TRUE to show number of subjects as per id in caption.
logx, logy	<lgl> TRUE to log x- and y-axis.
alpha_point	<num> Alpha value for ggplot2::geom_point.
alpha_line	<num> Alpha value for ggplot2::geom_line.
xlab, ylab	<chr> Labels for x- and y-axis.
ttl, sttl, cap	<chr> Title. Subtitle. Caption.

Value

A ggplot object.

Examples

```
Theoph |> ggtpp(x=Time,y=conc,id=Subject)
```

---

ggvio *Violin plot wrapper for categorical covariates*

Description

Create violin plots for a chosen variable by all discrete covariates in a dataset. Numeric variables will be dropped, except the chosen variable to plot.

Usage

```
ggvio(d, var, cats, alpha = 0.1, show = TRUE, nsub = TRUE, ...)
```

Arguments

d	<dfr> A data frame.
var	<var> A variable to plot as unquoted name.
cats	<var> Optional. Categorical variables to plot as a vector of unquoted names.
alpha	<num> Alpha value for ggplot2::geom_jitter.
show	<lgl> TRUE to show data using ggplot2::geom_jitter.
nsub	<lgl> Show number of observations.
...	Additional arguments for ggplot2::geom_violin.

Value

A ggplot object.

Examples

```
d = mtcars |> mutate(across(c(am,carb,cyl,gear,vs),factor))
d |> ggvio(mpg)
d |> ggvio(mpg,alpha=0.5)
d |> ggvio(mpg,show=FALSE)
d |> ggvio(mpg,nsub=FALSE)
d |> ggvio(mpg,c(cyl,vs))
```

---

ggxy *XY scatter plot wrapper*

Description

Create basic XY scatter plot for quick data exploration. Default to show Pearson correlation coefficient with p-value using `ggpubr::stat_cor()`. For more complex plots, it is recommended to use ggplot2::ggplot2 directly.

Usage

```
ggxy(
  d,
  x,
  y,
  ...,
  lm = TRUE,
  se = TRUE,
  cor = TRUE,
  pv = 0.001,
  nsub = TRUE,
  legend = TRUE,
  asp = 1
)
```

Arguments

d	<dfr> A data frame.
x, y	<var> Variables for x- and y-axis as unquoted names.
...	Arguments to pass to `ggplot2::aes()` for additional mapping.
lm	<lgl> TRUE to add regression line from linear model.
se	<lgl> TRUE to show standard error with the regression line.
cor	<lgl> TRUE to show Pearson correlation coefficient with p-value.
pv	<dbl> Precision of p-value, e.g., 0.001 to show 3 decimal places, NULL for no rounding.
nsub	<lgl> Show number of observations.
legend	<lgl> TRUE to show legend.
asp	<num> For `aspect.ratio` in `ggplot2::theme()`.

Value

A ggplot object.

See Also

ggpubr::stat_cor()

Examples

```
mtcars |> ggxy(wt,hp)
mtcars |> ggxy(wt,hp,col=factor(gear))
mtcars |> ggxy(wt,hp,col=factor(gear),legend=FALSE)
mtcars |> ggxy(wt,hp,col=factor(gear),pch=factor(am))
mtcars |> ggxy(wt,hp,nsub=FALSE)
mtcars |> ggxy(wt,hp,pv=NULL)
mtcars |> ggxy(wt,hp,lm=FALSE)
mtcars |> ggxy(wt,hp,se=FALSE)
mtcars |> ggxy(wt,hp,cor=FALSE)
```

---

hexn *Generate hex colour codes*

Description

Generate a vector of hex colour codes for the desired number of colours. Colours are generated by evenly splitting hue in the range `[0,360]` in the HCL colour space using grDevices::hcl. The output is meant to follow the default colours used in ggplot2::ggplot2.

Usage

```
hexn(n, show = FALSE)
```

Arguments

n	<int> Number of colours to output.
show	<lgl> TRUE to show the output colours.

Value

A vector of hex colour codes that can be used for plotting.

Examples

```
hexn(6,FALSE)
hexn(4,TRUE)
```

kb *kable wrapper*

Description

Sugar function for default kable output.

Usage

```
kb(d, fnote = NULL, cap = NULL, sig = 8, dig = 2, src = 0, omit = "")
```

Arguments

d	<dfr> A data frame.
fnote	<chr> Footnote.
cap	<chr> Caption.
sig	<int> Number of significant digits to compute.
dig	<int> Number of decimal places to display.
src	<int> Either 1 or 2 to add source label over 1 or 2 lines.
omit	<chr> Text to omit from the source label.

Value

A kable object.

Examples

```
mtcars |> head() |> kb()
mtcars |> head() |> kb(src=1)
mtcars |> head() |> kb("Footnote")
mtcars |> head() |> kb("Footnote",src=1)
mtcars |> head() |> kb(sig=2,dig=1)
```

label_src *Generate source file label*

Description

Generate a label with the current source file path and run time, assuming that the source file is in the current working directory. In interactive sessions, the function is designed to work in a script file in RStudio and uses `rstudioapi` to get the file path. It will return empty if run in the console directly.

Usage

```
label_src(span = 1, omit = "", tz = TRUE, fname = FALSE)
```

Arguments

span	<int> Number of lines: either 1 or 2.
omit	<chr> Text to omit from the label.
tz	<lgl> FALSE to exclude time stamp.
fname	<lgl> TRUE to return the file name only.

Value

A label showing the source file path with a time stamp.

Examples

```
label_src()
label_src(tz=FALSE)
label_src(fname=TRUE)
```

---

label_tz *Generate time stamp label*

Description

Generate a label with a time stamp indicating the run time.

Usage

```
label_tz(omit = "")
```

Arguments

omit	<chr> Text to omit from the label.

Value

A label with time stamp.

Examples

```
label_tz()
```

summ_by *Summarise continuous variables by group*

Description

Summarise all continuous variables by group. Non-numeric variables will be dropped.

Usage

```
summ_by(
  d,
  cols,
  ...,
  pct = c(0.25, 0.75),
  geo = FALSE,
  xname = "",
  view = FALSE
)
```

Arguments

d	<dfr> A data frame.
cols	<var> Optional. Columns to summarise as unquoted names.
...	<var> Optional. Columns to group by as unquoted names.
pct	<num> A vector of two indicating the percentiles to compute.
geo	<lgl> TRUE to add geometric mean and standard devaition.
xname	<chr> Characters to omit in output column names.
view	<lgl> TRUE to print output as flextable.

Value

A data frame of summarised variables.

Examples

```
d = mtcars |> dplyr::mutate(vs=factor(vs), am=factor(am))
d |> summ_by()
d |> summ_by(geo=TRUE)
d |> summ_by(pct=c(0.1,0.9))
d |> summ_by(mpg)
d |> summ_by(mpg,vs)
d |> summ_by(mpg,vs,am)
d |> summ_by(c(mpg,disp))
d |> summ_by(c(mpg,disp),vs)
d |> summ_by(c(mpg,disp),vs,xname="mpg_")
# Grouping without column selection is possible but rarely useful in large dataset
d |> summ_by(,vs)
```

summ_cat *Summarise categorical variables*

Description

Summarise all categorical variables. Numeric variables will be dropped.

Usage

```
summ_cat(d, ..., var, view = FALSE)
```

Arguments

d	<dfr> A data frame.
...	<var> Optional. Columns to summarise.
var	<var/int> (name or index) Optional. A variable to extract as a data frame.
view	<lgl> TRUE to show output as flextable.

Value

A list containing summaries for all categorical variables or a data frame showing the summary of a selected variable.

Examples

```
d = mtcars |> dplyr::mutate(dplyr::across(c(cyl,vs,am,gear,carb),factor))
d |> summ_cat()
d |> summ_cat(cyl,vs)
d |> summ_cat(var=cyl)
d |> summ_cat(var=1)
```

tab2v *Table wrapper for two variables*

Description

Wrapper function to produce summary tables for two variables.

Usage

```
tab2v(d, x, y)
```

Arguments

d	<dfr> A data frame.
x, y	<var> Variables to tabulate.

Value

A tabyl object.

Examples

```
# example code
mtcars |> tab2v(vs,cyl)
mtcars |> tab2v(vs,am)
mtcars |> tab2v(vs,am)
mtcars |> tab2v(vs,gear)
```

Index