Package: TeachingLab (via r-universe)

November 20, 2024

Title Teaching Lab Custom R Development Package

Version 0.0.1

Description Provides templates, palettes, and useful data analysis for Teaching Lab purposes.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

URL https://github.com/duncangates/teachinglab-duncan

BugReports https://github.com/duncangates/teachinglab-duncan/issues

Imports assertthat, bookdown, dplyr, fontawesome, forcats, ggplot2, ggtext, googlesheets4, glue, grid, gt, htmltools, knitr, lubridate, magrittr, purrr, qualtRics, readr, rlang, rmarkdown, scales, stats, stringr, tibble, tidyr, tidyselect, tidytext, treemapify, waffle

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Depends R (>= 3.10)

Suggests testthat (>= 3.0.0), janitor

Config/testthat/edition 3

Config/pak/sysreqs make libarchive-dev libicu-dev libjpeg-dev libpng-dev libxml2-dev libssl-dev libnode-dev libx11-dev zlib1g-dev

Repository https://teachinglab.r-universe.dev

RemoteUrl https://github.com/duncangates/TeachingLab

RemoteRef HEAD

RemoteSha 458aa7263b4570769a47775c5aedd3e463774117

Contents

`agree_strongly_agree` *Agree/Strongly agree*

Description

Gets the percent that agree and strongly agree

Usage

```
agree_strongly_agree(data, question)
```

Arguments

`data`	the data
`question`	a string - the question to get the percentage for

Value

a string

agree_strongly_agree2 *Agree/Strongly agree*

Description

Gets the percent that agree and strongly agree with different variable names

Usage

```
agree_strongly_agree2(data, name)
```

Arguments

data	the data
name	a string - the question to get the percentage for

Value

a string

calc_nps *Calculate nps score*

Description

Calculate nps score

Usage

```
calc_nps(x)
```

Arguments

x	A vector of nps scores

Value

Returns the nps score With formula % promoters - % detractors where promoters are 9 or 10 ratings and detractors are 0 to 6

check_email_approved *Verify email from vector*

Description

Verifies that an email is in a list

Usage

```
check_email_approved(email, approved_emails_list)
```

Arguments

email	the email provided
approved_emails_list	the email to check the list for

Value

TRUE or FALSE

check_email_domain *Verify email domain*

Description

Verifies that an email is of a specified domain

Usage

```
check_email_domain(email, domain)
```

Arguments

email	the email provided
domain	the domain to check that it is from

Value

TRUE or FALSE

coalesce_by_column *Coalesce everything*

Description

Takes all columns and splices them into dots for combination

Usage

```
coalesce_by_column(df)
```

Arguments

df	the dataframe

Value

the dataframe coalesced

Examples

```
df <- data.frame(
  a = c(1, 2, 3),
  b = c(NA, NA, NA),
  id = c("xxx", "xxx", "xxx")
)
df %>%
  dplyr::group_by(id) %>%
  dplyr::summarise_all(TeachingLab::coalesce_by_column)
```

colorize *Set Color in rmd with HTML or LaTex*

Description

Creates code to generate a color in x text with color, color.

Usage

```
colorize(x, color)
```

Arguments

x	the text to be colored
color	the color of the text

Value

Returns a string with code to render the correctly colored text

conditionally *Conditionally Perform Function*

Description

Wraps a function and conditionally performs it given certain arguments

Usage

```
conditionally(fun)
```

Arguments

fun	the function to wrap

Examples

```
library(dplyr)
cond_filter <- conditionally(filter)
cond_select <- conditionally(select)
```

conditional_slice_sample *Conditionally slice*

Description

Conditionally reduces the data

Usage

```
conditional_slice_sample(data, max)
```

Arguments

data	the data
max	a max length to slice the data for

Value

a string

course_feedback_graph *Course survey feedback graph dependent on race and content area*

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
course_feedback_graph(race_filter = "All", content_area_filter = "All")
```

Arguments

race_filter	the filter to apply to the race column, one of "All", "White", "Black or African American", "Asian", "Hispanic/Latino", "More than one race", or "Other"
content_area_filter	the content area to filter for, one otf "All", "ELA", "Math", or "Other"

Value

prints a ggplot object

course_quotes *Dashboard End of Course Quotes*

Description

Creates a gt table for qualitative responses of the end of course survey

Usage

```
course_quotes(
  data = TeachingLab::get_course_survey(),
  size = 10,
  n = 3,
  all = F,
  save = NULL,
  include_columns = NULL
)
```

Arguments

data	the data to be input
size	the number of rows of quotes to return
n	The number of words to highlight
all	F by default, whether or not to return ALL data
save	NULL by default, a save path to make inside current package directory
include_columns	A vector of additional columns to include in table

Value

Returns a gt, unless save in which case it will return a saved file with gtsave

---

dashboard_ipg_plot *IPG Forms Grapher*

Description

Creates a graph specifically for IPG Forms data

Usage

```
dashboard_ipg_plot(
  data,
  name,
  wrap = 60,
  sizing = 1,
  split = F,
  numeric = F,
  factor_level = NULL
)
```

Arguments

data	the data
name	the column name for the data frame to focus on
wrap	passes to str_wrap for the title
sizing	the base text size multiplier
split	if it is sequenced by commas split it
numeric	if it is numeric reorder the factors
factor_level	option for factor levels

Value

a ggplot object

dashboard_ipg_plot_ts *IPG Forms Grapher*

Description

Creates a graph specifically for IPG Forms data

Usage

```
dashboard_ipg_plot_ts(
  data,
  name,
  wrap = 60,
  sizing = 1,
  split = F,
  numeric = F,
  factor_level = NULL
)
```

Arguments

data	the data
name	the column name for the data frame to focus on
wrap	passes to str_wrap for the title
sizing	the base text size multiplier
split	if it is sequenced by commas split it
numeric	if it is numeric reorder the factors
factor_level	option for factor levels

Value

a ggplot object

drop1 *HTML/CSS Button Content Expander*

Description

Creates a button that will expand or hide content

Usage

```
drop1(before = T, options, envir, name)
```

Arguments

before	button default, collapsed/not collapsed
options	unclear
envir	also unclear
name	chunk name

Value

html wrapper

---

end_coaching_feedback_graph

End of Coaching feedback graph dependent on race and content area

---

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
end_coaching_feedback_graph(race_filter = "All", content_area_filter = "All")
```

Arguments

race_filter	the filter to apply to the race column, one of "All", "White", "Black or African American", "Asian", "Hispanic/Latino", "More than one race", or "Other"
content_area_filter	the content area to filter for, one otf "All", "ELA", "Math", or "Other"

Value

prints a ggplot object

`fake_bar_graph_create` *FAKE Knowledge Assessment Graph Summary*

Description

Creates a bar graph specifically for Knowledge Assessments for SXSW Report

Usage

```
fake_bar_graph_create(
  title,
  fake_data_fun = "p_and_n",
  custom_n_range = 0,
  custom_n = NULL,
  custom_x_axis_labels = NULL,
  custom_p_1 = NULL,
  custom_p_2 = NULL,
  randomness = 0,
  multiple_labels = NULL,
  know_graph = F
)
```

Arguments

`title`	the title for the knowledge assessment to make plot for
`fake_data_fun`	p_and_n, p_and_n_split
`custom_n_range`	custom n range the number to add/subtract for random number generation
`custom_n`	custom n, one number
`custom_x_axis_labels`	custom x-axis labels
`custom_p_1`	custom percentage 1, vector with 1: initial percentage, and 2: amount to set maximum increase
`custom_p_2`	custom percentage 2, vector with 1: initial percentage, and 2: amount to set maximum increase
`randomness`	multiplier for amount of randomness in custom percentages, default 0
`multiple_labels`	a special label maker to add group labels in bar charts
`know_graph`	if it is a knowledge assessments graph then add % correct to title

Value

a ggplot object

`fake_line_graph_create` *FAKE Knowledge Assessment Line Graph Summary*

Description

Creates a line graph specifically for Knowledge assessments for SXSW Report

Usage

```
fake_line_graph_create(
  title,
  fake_data_fun = "time_data",
  labels = c("Test 1", "Test 2"),
  x_axis = "yearly",
  y_axis_label = NULL,
  lines = 2,
  custom_n = NULL,
  custom_n_range = 0
)
```

Arguments

`title`	the title for the plot
`fake_data_fun`	time_data
`labels`	The labels for the lines
`x_axis`	the labels for the x-axis: yearly or pre-pl-post-pl
`y_axis_label`	A custom y axis label
`lines`	number of lines to create, either 1 or 2
`custom_n`	custom n, one number
`custom_n_range`	custom n range the number to add/subtract for random number generation

Value

a ggplot object

file.path2 *File path*

Description

Gives the file path without double slash bug

Usage

```
file.path2(..., fsep = .Platform$file.sep)
```

Arguments

...	The file path
fsep	the file separation

Value

fp a file path

find_highlight *Word highlighting*

Description

Finds most common words in string

Usage

```
find_highlight(string, n = 3, print = F)
```

Arguments

string	the string to evaluate
n	the number of words to find
print	whether or not to print the highlighted words

Value

a vector of strings

first_up *First Letter Uppercase*

Description

First Letter Uppercase

Usage

```
first_up(x)
```

Arguments

x	string

Value

string

get_course_survey *End of Course Dashboard Data*

Description

Gets dashboard data by reading it in from data folder

Usage

```
get_course_survey(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, optional to update end of course data or not
year	"21_22" or "22_23"

Value

Returns a tibble

get_current_partner_sites *Ongoing list of partner sites*

Description

Gets data from Google Sheet of partner sites

Usage

```
get_current_partner_sites(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, whether or not to pull the updated version
year	the year to get the site for

Value

Returns a tibble of sites

get_diagnostic_survey *Diagnostic Survey Update*

Description

Get the diagnostic survey

Usage

```
get_diagnostic_survey(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, optional updating
year	"21_22" or "22_23"

Value

A tibble

get_end_coaching *Ongoing Coaching Feedback Survey Data*

Description

Gets data from the Ongoing Coaching Feedback Survey

Usage

```
get_end_coaching(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, whether or not to pull the updated version
year	"21_22" or "22_23"

Value

Returns a tibble

get_followup_educator *Follow Up Educator Survey Data*

Description

Gets data from the Follow Up Educator Survey

Usage

```
get_followup_educator(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, whether or not to pull the updated version
year	"21_22" or "22_23"

Value

Returns a tibble

get_ipg_forms *IPG Data*

Description

Gets data from IPG forms

Usage

```
get_ipg_forms(update = FALSE, year = "22_23")
```

Arguments

update	FALSE
year	"21_22" or "22_23"

Value

Returns a tibble

get_knowledge_assessments *Knowledge Assessments Update*

Description

Get the knowledge assessments survey

Usage

```
get_knowledge_assessments(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, optional updating
year	"21_22" or "22_23"

Value

A tibble

get_lesson_analysis *Lesson Plan Analysis Data*

Description

Gets data from Lesson Plan Analysis forms

Usage

```
get_lesson_analysis(update = FALSE)
```

Arguments

update	FALSE

Value

Returns a tibble

get_ongoing_coaching *Coaching Participant Feedback Data*

Description

Gets data from Coaching Participant Feedback

Usage

```
get_ongoing_coaching(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, whether or not to pull the updated version
year	"21_22" or "22_23"

Value

Returns a tibble

get_season *Get Season*

Description

Takes a date and finds the season

Usage

```
get_season(date)
```

Arguments

date	the date

Value

the season

Examples

```
get_season(as.POSIXct("2016-01-01 12:00:00"))
```

get_session_survey *End of Session Dashboard Data*

Description

Gets dashboard data by reading it in from data folder

Usage

```
get_session_survey(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, whether or not to update
year	"21_22" or "22_23"

Value

Returns a tibble

get_student_scores_mississippi
Student Scores Mississippi

Description

Get student scores for mississippi data

Usage

```
get_student_scores_mississippi(update = FALSE)
```

Arguments

update	FALSE, optional updating

Value

A tibble

get_student_scores_mississippi2
Student Scores Mississippi Round 2

Description

Get student scores for follow up mississippi data

Usage

```
get_student_scores_mississippi2(update = FALSE)
```

Arguments

update	FALSE, optional updating

Value

A tibble

get_student_survey *Student Survey Data*

Description

Gets data from Student Survey

Usage

```
get_student_survey(update = FALSE, year = "22_23")
```

Arguments

update	FALSE whether or not to update the data
year	"21_22" or "22_23"

Value

Returns a tibble

get_student_work *Student Work Data*

Description

Gets metadata about student work files

Usage

```
get_student_work(update = FALSE, year = "22_23")
```

Arguments

update	FALSE, whether or not to pull the updated version
year	"21_22" or "22_23"

Value

Returns a tibble

get_student_work_grades *Ongoing list of partner sites*

Description

Gets data from Google Sheet of partner sites

Usage

```
get_student_work_grades(year = "22_23")
```

Arguments

year	the year to get the site for

Value

Returns a tibble of sites

grade_ipg *Grade IPG Data*

Description

function for grading different parts of the ipg forms

Usage

```
grade_ipg(x, type = "character")
```

Arguments

x	the data
type	character or numeric

Value

a percentage of correct either by checking 3 or 4 or yes

gtable_extract_grob *gtable_extract_grob*

Description

Helper function to extract a grob from gtables by name.

Usage

```
gtable_extract_grob(g, pattern = "guide-box")
```

Arguments

g	the gtable to extract the grob
pattern	grob name or pattern to match

Value

g, a grob matching the specified pattern

gtable_remove_grob *gtable_remove_grob*

Description

Helper function to remove grobs by name, from gtables

Usage

```
gtable_remove_grob(g, pattern = "guide-box")
```

Arguments

g	gtable with the grob removed
pattern	grob name or pattern to match

Value

g, with pattern removed.

gt_arrow *Arrow maker for gt table*

Description

Makes an html column for a gt table

Usage

```
gt_arrow(data, colors = c("#800000", "#98AFC7"), column_one, column_two)
```

Arguments

data	the gt table to make arrows for
colors	the color style to apply
column_one	The first column to compare
column_two	The second column to compare

Value

a colored gt table

Examples

```
gt_arrow(data = mtcars, colors = c("red", "blue"), column_one = cyl, column_two = disp)
```

gt_know_assess *Knowledge Assessment Graphs*

Description

Creates a graph specifically for Knowledge Assessments in mid year reports

Usage

```
gt_know_assess(data, know_assess)
```

Arguments

data	the data
know_assess	the knowledge assessment to make a table for

Value

a gt table

gt_percent_n *GT or ggplot maker*

Description

makes a gt table with percent and n colored

Usage

```
gt_percent_n(
  df,
  column,
  custom_title,
  no_title = T,
  base_font = 10,
  heading_font = 14,
  custom_column_name = "",
  viz_type = "gt"
)
```

Arguments

df	the data frame
column	the column to get count and percent from
custom_title	the title for the table
no_title	make the table have no title
base_font	overall table font size
heading_font	title font size
custom_column_name	a custom name for the column
viz_type	gt by default, also has ggplot options like pie chart, waffle, or treemap

Value

a gt table

gt_theme_tl *Create Teaching Lab theme to a gt table*

Description

Create Teaching Lab theme to a gt table

Usage

```
gt_theme_tl(
  data,
  all_caps = F,
  align = "center",
  base_font = 16,
  heading_font = 20,
  ...
)
```

Arguments

data	An existing gt table object
all_caps	Whether or not to capitalize titles
align	Align options are "left", "center", "right" or NULL
base_font	the font size
heading_font	the title font size
...	Optional additional arguments to gt::table_options()

Value

Creates a gt theme as a pipeable function

Examples

```
mtcars %>% utils::head() %>% gt::gt() %>% TeachingLab::gt_theme_tl()
```

gt_tl_color *General Table Maker*

Description

Makes a gt table with teaching lab color style

Usage

```
gt_tl_color(gt_table, color, column, scale = 1)
```

Arguments

gt_table	the gt table to color
color	the color style to apply
column	The column to apply the color to
scale	the scale to apply to

Value

a colored gt table

---

highlight_fun *Word highlighting*

Description

Provides html formatted highlighting

Usage

```
highlight_fun(data, highlight = TeachingLab::find_highlight(data))
```

Arguments

data	the data to highlight
highlight	the words to highlight

Value

a vector of strings

---

html_dependency_bootstrap *create an html dependency for bootstrap (function copied from rmarkdown)*

Description

create an html dependency for bootstrap (function copied from rmarkdown)

Usage

```
html_dependency_bootstrap(theme = "bootstrap")
```

Arguments

theme	"bootstrap" is the only option as of yet

html_dependency_jquery_stickytableheaders *create an html dependency for jquery-stickytableheaders*

Description

create an html dependency for jquery-stickytableheaders

Usage

```
html_dependency_jquery_stickytableheaders()
```

html_dependency_magnific_popup *create an html dependency for Magnific popup*

Description

create an html dependency for Magnific popup

Usage

```
html_dependency_magnific_popup()
```

html_template *Shared HTML template function*

Description

Shared HTML template function

Usage

```
html_template(
  template_name,
  template_path,
  template_dependencies,
  pandoc_args,
  ...
)
```

Arguments

template_name	template file name
template_path	template location
template_dependencies	js and css dependencies
pandoc_args	pandoc additional args
...	additional arguments to pass to pkgdown args

---

html_wrap *HTML Text Wrapping*

Description

Takes a string and inserts at the requested intervals

Usage

```
html_wrap(string, n = 40)
```

Arguments

string	the string
n	the width of the string before a tag

Value

the same string with inserted at the requested interval

Examples

```
html_wrap("a random string that has about 40 characters in it")
```

---

id_maker *ID Maker*

Description

takes initials and id column and makes an ID

Usage

```
id_maker(initials, birthday)
```

Arguments

initials	the initials
birthday	the birthday

ipg_plot *IPG Forms Grapher*

Description

Creates a graph specifically for IPG Forms data

Usage

```
ipg_plot(
  data,
  name,
  save_name,
  height = 5,
  width = 8.5,
  wrap = 60,
  sizing = 1,
  dpi = 300,
  split = F,
  numeric = F,
  factor_level = NULL,
  save = FALSE
)
```

Arguments

data	the data
name	the column name for the data frame to focus on
save_name	the name to save
height	height
width	width
wrap	passes to str_wrap for the title
sizing	the base text size multiplier
dpi	the dpi to save with
split	if it is sequenced by commas split it
numeric	if it is numeric reorder the factors
factor_level	option for factor levels
save	FALSE

Value

a ggplot

knowledge_assess_detailed_score
Knowledge Assessments Question Scoring 2022-2023

Description

function to grade and output each question per respondent in a knowledge assessments survey

Usage

```
knowledge_assess_detailed_score(survey_id, survey_name)
```

Arguments

survey_id	Qualtrics survey id
survey_name	Name of survey - should match Qualtrics name

Value

a dataframe of columns: prepost, site, know_assess, date, question, score, question2, answer, max_score

knowledge_assess_select_score
Knowledge Assessments Scoring 2022-2023

Description

function to grade and output specific set of data points for knowledge assessments

Usage

```
knowledge_assess_select_score(survey_id, survey_name)
```

Arguments

survey_id	Qualtrics survey id
survey_name	Name of survey - should match Qualtrics name

Value

a dataframe of columns: id, percent, prepost, site, know_assess, date

know_assess_summary *Knowledge Assessment Graph Summary*

Description

Creates a graph specifically for Knowledge Assessments in mid year reports

Usage

```
know_assess_summary(data, know_assess, summary_path = "report_summary_images")
```

Arguments

data	the data
know_assess	the knowledge assessment to make plot for
summary_path	optional path to save plot to a file, if NULL does not save anywhere

Value

a ggplot

mutate_cond *Conditionally mutate*

Description

Conditionally changes the data

Usage

```
mutate_cond(.data, condition, ..., envir = parent.frame())
```

Arguments

.data	the data
condition	a specific condition to test for
...	additional arguments for mutate
envir	parent.frame()

Value

a string

neg_cond_filter *Negative Conditional Filter*

Description

Conditionally filters value given that it is not the first parameter, for use in shiny apps

Usage

```
neg_cond_filter(data, if_not_this, filter_this, dat_filter)
```

Arguments

data	the dataframe to apply filter
if_not_this	If value is not this
filter_this	Filter for this
dat_filter	Data column object to filter

Value

filtered dataframe

notin *Find elements x not in a vector y*

Description

Find elements x not in a vector y

Usage

```
x %!in% y
```

Arguments

x	A vector of what shouldn't exist
y	A vector to check against

Value

Returns elements not in vector

ongoing_coaching_feedback_graph *Ongoing Coaching feedback graph dependent on race and content area*

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
ongoing_coaching_feedback_graph(
  race_filter = "All",
  content_area_filter = "All"
)
```

Arguments

race_filter	the filter to apply to the race column, one of "All", "White", "Black or African American", "Asian", "Hispanic/Latino", "More than one race", or "Other"
content_area_filter	the content area to filter for, one otf "All", "ELA", "Math", or "Other"

Value

prints a ggplot object

partner_file_remove *Remove Image Files Render R Markdown*

Description

remove files from images/report_images and images/report_summary_images and render specified rmd

Usage

```
partner_file_remove(partner, content_area = NULL, input, output_dir, ...)
```

Arguments

partner	The partner to render a report for
content_area	content area to filter by default to NULL and ignored
input	the rmd to use to parametrically generate reports
output_dir	the output directory for the files
...	arguments passed to `rmarkdown::render`

password_generator *Password Generator*

Description

Creates a password of n length

Usage

```
password_generator(length = 8)
```

Arguments

length	the length of the password

Value

a string

percent_agree *Percent Agree/Strongly agree*

Description

Calculates percent that are 4, 5, agree, or strongly agree

Usage

```
percent_agree(agree_col)
```

Arguments

agree_col	the vector or column to summarise

Value

integer

p_and_n *Fake p1, p2, n1, n2 data*

Description

Creates fake data for knowledge assessments in SXSW report

Usage

```
p_and_n(
  p1_range = c(40:80),
  p2_range = c(70:100),
  n1_range = stats::rnorm(1, mean = 80, sd = 10),
  n2_range = stats::rnorm(1, mean = 50, sd = 10)
)
```

Arguments

p1_range	the range of values to sample for p1
p2_range	the range of values to sample for p2
n1_range	the range of values to sample for n1
n2_range	the range of values to sample for n2

Value

a randomised dataset

p_and_n_split *Fake p1_1, p1_2, p2_1, p2_2, n1, n2 data*

Description

Creates fake data for knowledge assessments in SXSW report

Usage

```
p_and_n_split(
  p1_1_range = c(30:70),
  p1_2_range = c(70:100),
  p2_1_range = c(80:95),
  p2_2_range = c(85:100),
  n1_range = stats::rnorm(1, mean = 80, sd = 10),
  n2_range = stats::rnorm(1, mean = 50, sd = 10)
)
```

Arguments

p1_1_range	the lower range of values to sample for p1
p1_2_range	the upper range of values to sample for p1
p2_1_range	the lower range of values to sample for p2
p2_2_range	the upper range of values to sample for p2
n1_range	the range of values to sample for n1
n2_range	the range of values to sample for n2

Value

a randomised dataset

---

quote_viz *Quote Visualization*

Description

takes a dataframe and makes a gt table or ggplot that shows a quote

Usage

```
quote_viz(
  data,
  text_col = colnames(data)[1],
  viz_type = "gt",
  custom_highlight = NULL,
  n = 3,
  print = T,
  width = 60,
  title = NULL,
  suppress_warnings = T,
  align = "center",
  save = T,
  ...
)
```

Arguments

data	the dataframe
text_col	columns to create table for
viz_type	ggplot or gt visualization
custom_highlight	a vector, optional custom highlighting
n	integer, number of words to auto-highlight

print	T, whether or not to print the highlighted words to console
width	The width of the table generated
title	the title of the ggplot or gt
suppress_warnings	T/F suppression of warnings
align	the table alignment: "left", "center", "right"
save	if TRUE actually make the gt
...	Arguments passed onto the gt table

Value

a ggplot/gt that visualizes text

Examples

```
## Not run:
df <- TeachingLab::survey_monkey
colnames(df)[1] <- "What learning are you excited to try?"
quote_viz(
  data = df,
  text_col = "What learning are you excited to try?",
  viz_type = "gt",
  title = "Responses from Survey Monkey"
)

## End(Not run)
```

---

rand_vect *Random Vector*

Description

Random Vector

Usage

```
rand_vect(N, M, sd = 1, pos.only = TRUE)
```

Arguments

N	number of scalars to create
M	number to sum to
sd	Standard Deviation
pos.only	T

relabel_qualtrics_df *Qualtrics Data Frame Relabel*

Description

Switches the labels and column names of a data frame, or just changes the column names to the labels

Usage

```
relabel_qualtrics_df(df, switch = TRUE)
```

Arguments

df	the relevant dataframe
switch	T/F whether or not to switch the names

Value

a data.frame

Examples

```
## Not run:
qualtrics_survey |>
  relabel_qualtrics_df()

## End(Not run)
```

round2 *Round*

Description

round that actually round up 0.5 as it should be

rounding function as an alternative to round due to occasional decimal errors and problems with rounding from 0.5

Usage

```
round2(x, n)
```

Arguments

x	the vector to round
n	the number of digits to round

Value

the vector provided rounded

`round_even` *Round to nearest even number*

Description

Takes a whole number and "rounds" it to the nearest even number.

Usage

```
round_even(x)
```

Arguments

`x`	the number to round

Value

an integer

Examples

```
round_even(17)
```

`runif_round` *Generate one random number from a min and max*

Description

Generate one random number from a min and max

Usage

```
runif_round(min, max)
```

Arguments

`min`	the minimum number allowed
`max`	the maximum number allowed

Value

a single integer

scale_colour_tl *Discrete color & fill scales based on the Teaching Lab palette*

Description

See tl_palette().

Usage

```
scale_colour_tl(n, color = "blue", ...)

scale_color_tl(n, color = "blue", ...)

scale_fill_tl(
  n,
  color = c("blue", "orange", "purple", "green", "teal", "tl_colors"),
  ...
)
```

Arguments

n	the number of colors
color	the color from tl_palette
...	Arguments passed on to ggplot2::discrete_scale

aesthetics The names of the aesthetics that this scale works with.

scale_name The name of the scale that should be used for error messages associated with this scale.

palette A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take (e.g., scales::hue_pal()).

name The name of the scale. Used as the axis or legend title. If waiver(), the default, the name of the scale is taken from the first mapping used for that aesthetic. If NULL, the legend title will be omitted.

breaks One of:

- NULL for no breaks
- waiver() for the default breaks (the scale limits)
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output. Also accepts rlang lambda function notation.

labels One of:

- NULL for no labels
- waiver() for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)

- An expression vector (must be the same length as breaks). See ?plotmath for details.
- A function that takes the breaks as input and returns labels as output. Also accepts rlang lambda function notation.

`limits` One of:

- `NULL` to use the default scale values
- A character vector that defines possible values of the scale and their order
- A function that accepts the existing (automatic) values and returns new ones. Also accepts rlang lambda function notation.

`na.translate` Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

`na.value` If `na.translate = TRUE`, what aesthetic value should the missing values be displayed as? Does not apply to position scales where `NA` is always placed at the far right.

`drop` Should unused factor levels be omitted from the scale? The default, `TRUE`, uses the levels that appear in the data; `FALSE` uses all the levels in the factor.

`guide` A function used to create a guide or its name. See `guides()` for more information.

`super` The super class to use for the constructed scale

`scale_dollar_labels` *Five thirty-eight style formatter for currency*

Description

Five thirty-eight style formatter for currency

Usage

```
scale_dollar_labels(labels)
```

Arguments

`labels`	vector of labels

Value

formatted percent labels

scale_percent_labels *Five thirty-eight style formatter for percentages*

Description

Five thirty-eight style formatter for percentages

Usage

```
scale_percent_labels(labels)
```

Arguments

labels	vector of labels

Value

formatted percent labels

scale_ratio_labels *Five thirty-eight style formatter for Ratios*

Description

Five thirty-eight style formatter for Ratios

Usage

```
scale_ratio_labels(labels)
```

Arguments

labels	vector of labels

Value

formatted ratio labels

`score_compare_plot` *Graph maker for score comparisons*

Description

Makes a ggplot comparison table

Usage

```
score_compare_plot(
  data,
  question,
  order,
  prepost,
  score,
  split_variable,
  title
)
```

Arguments

`data`	the data to use
`question`	the question column to use for comparison
`order`	the order to keep the questions in
`prepost`	The column to compare before and after
`score`	the scores to compare
`split_variable`	the text detected in the variable to create segments with (from prepost)
`title`	the title of the gt

Value

a colored gt table

Examples

```
## Not run:
score_compare_plot(data, question, prepost, score)

## End(Not run)
```

score_knowledge_question
Knowledge Assessments Scoring

Description

Score a knowledge assessments question by selecting a grouping and choosing the percent that is correct

Usage

```
score_knowledge_question(data, question, correct)
```

Arguments

data	the data to evaluate
question	a vector of questions
correct	a vector of correct answers

Value

a dataframe of format question1, question2, question3, with percents as answers

score_one_question_mindsets
Mindsets scoring

Description

Calculate percentage correct for mindsets & expectations for just pre or post

Usage

```
score_one_question_mindsets(
  data,
  question,
  coding,
  na_remove = F,
  likert = c(5, 6)
)
```

Arguments

data	the dataframe to be analyzed
question	the initial column to be selected
coding	the coding to check for
na_remove	whether or not to drop NAs at the start of the evaluation
likert	whether or not the scale is likert with 5 points or 6

Value

Returns a dataframe with the percent, correct, number of non-na responses, the question itself, and the percent that sustained/improved

---

score_question *Calculate percentage of a question (column) in data (data) that is in the right answer (coding)*

Description

Calculate percentage of a question (column) in data (data) that is in the right answer (coding)

Usage

```
score_question(data, question, coding, grouping = NULL)
```

Arguments

data	the dataframe to be analyzed
question	the column to be selected
coding	the coding to check for
grouping	NULL a vector of variables to group_by

Value

Returns a dataframe with the percent, correct, number of non-na responses, and question itself

score_question_improved *Scores pre and post and percent improved/sustained*

Description

Calculate percentage of a pre question (column) and post question (column) in data (data) that is in the right answer (coding).

Usage

```
score_question_improved(
  data,
  question_pre,
  question_post,
  coding,
  middle_value
)
```

Arguments

data	the dataframe to be analyzed
question_pre	the initial column to be selected
question_post	the comparison column to be selected
coding	the coding to check for
middle_value	the middle value to check for when calculating scores

Value

Returns a dataframe with the percent, correct, number of non-na responses, the question itself, and the percent that sustained/improved

score_question_mindsets *Mindsets scoring*

Description

Calculate percentage correct for mindsets & expectations

Usage

```
score_question_mindsets(
  data,
  question_pre,
  question_post,
  coding,
  na_remove = F,
  likert = c(5, 6)
)
```

Arguments

data	the dataframe to be analyzed
question_pre	the initial column to be selected
question_post	the comparison column to be selected
coding	the coding to check for
na_remove	whether or not to drop NAs at the start of the evaluation
likert	whether or not the scale is likert with 5 points or 6

Value

Returns a dataframe with the percent, correct, number of non-na responses, the question itself, and the percent that sustained/improved

---

score_question_number *A title*

Description

Calculate percentage of a question (column) in data (data) that is on the positive or numeric side (coding) with a grading twist, where three is worth 1 and 1/2 or 4/5 is worth 2

Usage

```
score_question_number(
  data,
  question_pre,
  question_post,
  coding,
  likert = c(5, 6)
)
```

Arguments

data	the dataframe to be analyzed
question_pre	the "pre-tl pl" column to be selected
question_post	the "post-tl pl"column to be selected
coding	the coding to check for
likert	whether the likert scale has 5 or 6 points

Value

Returns a dataframe with the percent, correct, number of non-na responses, and question itself

---

selection_sum *Summarise a selection*

Description

finds the average sum of all selected columns

Usage

```
selection_sum(data, select, group = NULL)
```

Arguments

data	the data
select	the data to select
group	the data to group by

Value

a tibble of the selected data's percent correct

select_useful_text *Useful text selector*

Description

Gets all text over a specified nps and tibble

Usage

```
select_useful_text(data, text_col, name, n = 15)
```

Arguments

data	the data to filter the text for
text_col	the text column to select
name	the new name for the text column
n	15 observations to return by default

Value

returns a single tibble column

session_agree_plot *Dashboard Agree Type % Plot*

Description

Creates a plot for session survey data that shows % in each Likert category

Usage

```
session_agree_plot(data)
```

Arguments

data	the data to be input

Value

Returns a ggplot

session_agree_plot_ts *Dashboard Time Series Plot*

Description

Creates a plot for session survey data with an adjustable time series component

Usage

```
session_agree_plot_ts(data, scale = "1 month")
```

Arguments

data	the data to be input
scale	the date scale to use for the plot

Value

Returns a ggplot

session_feedback_graph *End of session feedback graph dependent on race and content area*

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
session_feedback_graph(race_filter = "All Races", content_area_filter = "All")
```

Arguments

race_filter	the filter to apply to the race column, one of "All", "White", "Black or African American", "Asian", "Hispanic/Latino", "More than one race", or "Other"
content_area_filter	the content area to filter for, one otf "All", "ELA", "Math", or "Other"

Value

prints a ggplot object

session_quotes *Dashboard End of Session Quotes*

Description

Creates a gt table for qualitative responses to the end of session survey

Usage

```
session_quotes(
  data = TeachingLab::get_session_survey(),
  size = 10,
  n = 3,
  all = F,
  save = NULL,
  include_columns = NULL
)
```

Arguments

data	the data to be input
size	the number of rows of quotes to return
n	The number of words to highlight
all	Whether or not to return ALL data
save	NULL by default, a save path to make inside current package directory
include_columns	A vector of additional columns to include in table

Value

Returns a gt, unless save in which case it will return a saved file with gtsave

single_sort_view *Unique, sort, view*

Description

Gets all unique elements of a vector in a sorted view

Usage

```
single_sort_view(vector)
```

Arguments

vector	the vector to view

Value

a string

Examples

```
# example code
 c("stuff", "and", "more") |> single_sort_view()
```

---

stop_words *Various lexicons for English stop words*

Description

English stop words from three lexicons, as a data frame. The snowball and SMART sets are pulled from the tm package. Note that words with non-ASCII characters have been removed.

Format

A data frame with 1149 rows and 2 variables:

word An English word

lexicon The source of the stop word. Either "onix", "SMART", or "snowball"

Source

- http://www.lextek.com/manuals/onix/stopwords1.html
- https://www.jmlr.org/papers/volume5/lewis04a/lewis04a.pdf
- http://snowball.tartarus.org/algorithms/english/stop.txt

---

string_replace *String Replace*

Description

Detects a specific string pattern and replaces entire string with it

Usage

```
string_replace(string, string_detect, string_replace, print = FALSE)
```

Arguments

string	the string to look within
string_detect	the string to detect
string_replace	the string to replace detected string with
print	FALSE whether or not to print the string

Value

a string

---

student_bar_chart *Student percent agree/strongly agree*

Description

Automatically dodged bar chart for student data

Usage

```
student_bar_chart(
  data,
  col_select,
  agree_select,
  string_remove,
  title,
  legend_position = c(0.8, 0.25),
  race_filter = "All"
)
```

Arguments

data	the data for the plotter to use, should include all columns of interest
col_select	the columns to select with `tidyselect::contains`
agree_select	the type of agree/strongly agree to select, for example also often/always
string_remove	NULL by default, provides an optional string removal
title	the title for the plot
legend_position	c(0.8, 0.25) by default, adjustable
race_filter	a parameter to filter for a specific race from the survey

Value

a ggplot object

survey_monkey *A survey monkey text selection*

Description

Survey monkey reviews from the course survey in response to the question What is the learning from this course that you are most excited about trying out?, as a data frame.

Format

A data frame with 87 rows and 1 variable

Source

https://www.surveymonkey.com/r/TLendofcourse

table_maker *General Table Maker*

Description

Makes a gt table for mindsets question set

Usage

```
table_maker(
  data,
  column_names,
  title,
  spanner,
  n1,
  n2,
  rows_positive,
  rows_negative,
  improve_col,
  bottom_row
)
```

Arguments

data	the dataframe to be analyzed
column_names	the names of the columns
title	The tables title
spanner	The gt spanner

n1	the n for fall
n2	the n for spring
rows_positive	the positive rows
rows_negative	the negative_rows
improve_col	name for the improve/sustain column
bottom_row	the bottom row which has "n = " instead of data

Value

Returns an unsaved gt table

---

temp_save *Temporary Image Save and Return*

Description

Creates a temporary save location for an image and return it with `knitr::include_graphics`

Usage

```
temp_save(img)
```

Arguments

img	the object to save as an image

Value

a png image

---

theme_final_report *Teaching Lab Report Theme*

Description

A theme for Teaching Lab RMD Reports

Usage

```
theme_final_report()
```

Value

a ggplot object

theme_irp *My ggplot2 theme heavy credits for influencing the theme function go to @hrbrmstr (Bob Rudis)*

Description

It requires installing Calibri fonts unless you change the font parameters

Usage

```
theme_irp(
  base_family = "Roboto Condensed",
  base_size = 11,
  strip_text_family = "Futura Medium",
  strip_text_size = 12,
  plot_title_family = "Futura Medium",
  plot_title_size = 18,
  plot_title_margin = 10,
  subtitle_family = "Roboto Condensed",
  subtitle_size = 12,
  subtitle_margin = 15,
  caption_family = "Roboto Condensed",
  caption_size = 9,
  caption_margin = 10,
  axis_title_family = "Roboto Condensed",
  axis_title_size = 9,
  axis_title_just = "mm",
  dark = FALSE,
  grid = TRUE,
  axis = FALSE,
  ticks = FALSE
)
```

Arguments

base_family	base font family
base_size	base font size
strip_text_family	facet label font family
strip_text_size	facet label text size
plot_title_family	plot tilte family
plot_title_size	plot title font size

plot_title_margin
: plot title margin

subtitle_family
: plot subtitle family

subtitle_size
: plot subtitle size

subtitle_margin
: plot subtitle margin

caption_family
: plot caption family

caption_size
: plot caption size

caption_margin
: plot caption margin

axis_title_family
: axis title font family

axis_title_size
: axis title font size

axis_title_just
: axis title font justification blmcrt

dark
: axis TRUE, FALSE,

grid
: panel grid (TRUE, FALSE, or a combination of X, x, Y, y)

axis
: axis TRUE, FALSE

ticks
: ticks axis TRUE, FALSE

Details

https://www.google.com/fonts

---

theme_tl *Teaching Lab Custom Ggplot2 Theme*

Description

It requires installing Roboto, Calibri fonts unless you change the font parameters

https://www.google.com/fonts

Usage

```
theme_tl(
  base_family = "Calibri",
  base_size = 14,
  strip_text_family = base_family,
  strip_text_size = 15,
  plot_title_family = "Calibri",
  plot_title_size = 20,
  plot_title_margin = 10,
  subtitle_family = "Roboto",
```

```
  subtitle_size = 15,
  subtitle_margin = 15,
  caption_family = "Roboto",
  caption_size = 11,
  caption_margin = 10,
  axis_title_family = "Calibri",
  axis_title_size = 12,
  axis_title_just = "mm",
  axis_text_size = 10.5,
  dark = FALSE,
  grid = TRUE,
  axis = FALSE,
  ticks = FALSE,
  markdown = FALSE,
  legend = F
)
```

Arguments

base_family	base font family
base_size	base font size
strip_text_family	facet label font family
strip_text_size	facet label text size
plot_title_family	plot title family
plot_title_size	plot title font size
plot_title_margin	plot title margin
subtitle_family	plot subtitle family
subtitle_size	plot subtitle size
subtitle_margin	plot subtitle margin
caption_family	plot caption family
caption_size	plot caption size
caption_margin	plot caption margin
axis_title_family	axis title font family
axis_title_size	axis title font size
axis_title_just	axis title font justification blmcrt

axis_text_size	axis text size
dark	dark mode TRUE, FALSE
grid	panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
axis	axis TRUE, FALSE
ticks	ticks TRUE, FALSE
markdown	enabled ggtext markdown styling TRUE, FALSE
legend	default no legend with F

TLDefault *Convert to an HTML document*

Description

Format for converting from R Markdown to an HTML document.

Usage

```
TLDefault(
  fig_width = 8,
  fig_height = 5,
  fig_caption = TRUE,
  highlight = "kate",
  lightbox = FALSE,
  thumbnails = FALSE,
  gallery = FALSE,
  toc_depth = 2,
  embed_fonts = TRUE,
  fontawesome = TRUE,
  use_bookdown = FALSE,
  pandoc_args = NULL,
  md_extensions = NULL,
  mathjax = "rmdformats",
  ...
)
```

Arguments

fig_width	Default width (in inches) for figures
fig_height	Default width (in inches) for figures
fig_caption	TRUE to render figures with captions
highlight	Syntax highlighting style. Supported styles include "default", "tango", "pygments", "kate", "monochrome", "espresso", "zenburn", "haddock", and "textmate". Pass NULL to prevent syntax highlighting.
lightbox	if TRUE, add lightbox effect to content images

thumbnails	if TRUE display content images as thumbnails
gallery	if TRUE and lightbox is TRUE, add a gallery navigation between images in lightbox display
toc_depth	adjust table of contents depth
embed_fonts	if TRUE, use local files for fonts used in the template. This leads to bigger files but ensures that these fonts are available. If FALSE they are downloaded from Google Web Fonts.
fontawesome	if TRUE use fontawesome dependencies
use_bookdown	if TRUE, uses html_document2 instead of html_document, thus providing numbered sections and cross references
pandoc_args	arguments passed to the pandoc_args argument of rmarkdown html_document
md_extensions	arguments passed to the md_extensions argument of rmarkdown html_document
mathjax	set to NULL to disable Mathjax insertion
...	Additional function arguments passed to R Markdown html_document

Details

CSS adapted from the readtheorg theme of the org-html-themes project : https://github.com/fniessen/org-html-themes, which is itself inspired by the Read the docs theme : https://readthedocs.org/.

Value

R Markdown output format to pass to render

---

tl_likert *TL Themed Stacked Bar Chart*

Description

Automatically scaled stacked bar chart with TL theming

Usage

```
tl_likert(data, title = "% Selected", string_remove = NULL, string_wrap = 80)
```

Arguments

data	the data for the plotter to use, should include all columns of interest
title	the title for the plot
string_remove	NULL by default, provides an optional string removal
string_wrap	80 by default, the amount by which to wrap y axis text

Value

a ggplot object

tl_pal *A muted, qualitative color palette*

Description

A muted, qualitative color palette

Usage

```
tl_pal()
```

Examples

```
## Not run:
library(scales)
scales::show_col(tl_pal_blue()(9))

## End(Not run)
```

tl_palette *Teaching Lab Color Palette Maker*

Description

Teaching Lab Color Palette Maker

Usage

```
tl_palette(
  color = c("blue", "orange", "purple", "green", "teal", "tl_colors"),
  theme = c("dark"),
  n = 6
)
```

Arguments

color	the color palette to generate
theme	if theme is light or dark
n	number of colors to generate

Value

color ramp palette function

tl_pal_blue *TL Default Blue Palette*

Description

TL Default Blue Palette

Usage

```
tl_pal_blue
```

Format

An object of class `character` of length 8.

tl_score *Grade Data*

Description

function for grading in general

Usage

```
tl_score(data, answer)
```

Arguments

`data`	the data
`answer`	the answer

Value

a percentage of correct

tl_score_count *Grade Data*

Description

function for generally getting the count of a data that is correct

Usage

```
tl_score_count(data, answer)
```

Arguments

data	the data
answer	the answer

Value

a count of correct

tl_score_numeric *IPG Scoring Numeric*

Description

function for grading in general

Usage

```
tl_score_numeric(x)
```

Arguments

x	the numbers to apply to

Value

a rounded average

tl_score_percent *Get the percent of* data *in* answer

Description

function for general grading of percentages

Usage

```
tl_score_percent(data, answer)
```

Arguments

data	the data
answer	the answer

Value

a vector percent of correct

tl_select_percent *Get the percent of a column that equals specific values*

Description

Automatically scaled stacked bar chart with TL theming

Usage

```
tl_select_percent(data, percent_equal)
```

Arguments

data	the data for the plotter to use, should include all columns of interest
percent_equal	string inputs to find the percent of the column that equals those values

Value

a percentage as a string

`tl_summary_table` *Teaching Lab Summary Tables*

Description

Creates tables for responses to the Educator Diagnostic Survey

Usage

```
tl_summary_table(
  data,
  save = F,
  summarise = T,
  grouping = NULL,
  n_size = NULL,
  n_size_single = NULL,
  explain = F,
  prepost = T,
  admin = F,
  rename = F
)
```

Arguments

`data`	the data to be input
`save`	F by default, a save path inside current working directory
`summarise`	TRUE by default will summarise all data selected
`grouping`	includes "equitable", "high_expectations", and "crse"
`n_size`	NULL an optional n_size to include
`n_size_single`	NULL an optional single value for n_size
`explain`	add explanation for different percentage levels to the table
`prepost`	F by default for making a prepost dataframe
`admin`	F by default for prepost admin dataframe
`rename`	F by default for making prepost dataframe

Value

Returns a gt table, unless save in which case it will return a saved file with gtsave

Index