Package 'lapop'

May 8, 2026

Type Package

Title Processing, Visualizing, and Labeling Americas Barometer Data

Version 2.1.5

Author Robert Vidigal [cre, aut] (ORCID:
<https://orcid.org/0000-0001-5713-119X>),
Luke Plutowski [ctb, aut] (ORCID:
<https://orcid.org/0000-0001-8240-6722>)

Maintainer Robert Vidigal <robert.vidigal@vanderbilt.edu>

Description
Labeling, weighting, and plotting data following custom style guidelines for use in reports, presentations, and social media posts. The Center for Global Democracy (formerly the Latin American Public Opinion Project) at Vanderbilt University is a leader in public survey research, best known for the Americas Barometer project. The publicly available data can be downloaded from: <https://www.vanderbilt.edu/lapop/data-access.php>.

URL https://lapop-central.github.io/lapop/

Depends R (>= 4.1.0)

Imports ggplot2, ggtext, ggrepel, showtext, grid, gridtext, gridExtra, sf, sysfonts, systemfonts, svglite, dplyr, srvyr, survey, haven, stats, purrr, tibble, marginaleffects, stringr, zoo

VignetteBuilder knitr

Suggests readstata13, rio, rprojroot, knitr, rmarkdown, tidyr, ggpattern, testthat (>= 3.0.0)

Language en-US

Encoding UTF-8

License MIT + file LICENSE

RoxygenNote 7.3.3

Config/testthat/edition 3

Collate 'bra23.R' 'cm23.R' 'globals.R' 'lapop-deprecated.R' 'lapop_fonts.R' 'lapop_cc.R' 'lapop_cccomb.R' 'lapop_ccm.R'

'lapop_coef.R' 'lapop_dumb.R' 'lapop_fonts_design.R'
'lapop_hist.R' 'lapop_map.R' 'lapop_mline.R' 'lapop_mover.R'
'lapop_save.R' 'lapop_stack.R' 'lapop_ts.R' 'lpr_cc.R'
'lpr_ccm.R' 'lpr_ci.R' 'lpr_coef.R' 'lpr_data.R' 'lpr_dumb.R'
'lpr_extract_notes.R' 'lpr_extract_ros.R' 'lpr_hist.R'
'lpr_mline.R' 'lpr_mover.R' 'lpr_resc.R' 'lpr_set_attr.R'
'lpr_set_ros.R' 'lpr_stack.R' 'lpr_ts.R' 'world.R' 'ym23.R'
'zzz.R'

NeedsCompilation no

Repository CRAN

Date/Publication 2026-04-29 18:30:02 UTC

Contents

`bra23` *bra23: Single-country Single-year Dataset*

Description

A dataset containing the AmericasBarometer Brazil 2023 survey round.

Usage

```
bra23
```

Format

A data frame

ing4 Support for Democracy

b12 Trust in Armed Forces

b13 Trust in the National Legislature

b21 Trust in Political Parties

b31 Trust in the Supreme Court of Justice

fs2 Food security

idio2 Personal economic situation

wealth Wealth

wave Survey round year for regional or multi-country data

pais Country of survey

year Survey round year for single-country data

upm Primary Sampling Unit

strata Stratification

wt Country-specific post stratification weight

Source

LAPOP AmericasBarometer (https://www.vanderbilt.edu/lapop/)

cm23 *cm23: Single-country Multi-year Dataset*

Description

A dataset containing the AmericasBarometer Brazil Country Merge up to 2023.

Usage

```
cm23
```

Format

A data frame

ing4 Support for Democracy

b13 Trust in the National Legislature

b21 Trust in Political Parties

b31 Trust in the Supreme Court of Justice

wave Survey round year for regional or multi-country data

pais Country of survey

year Survey round year for single-country data

upm Primary Sampling Unit

strata Stratification

weight1500 Cross-country and cross-time weight

Source

LAPOP AmericasBarometer (https://www.vanderbilt.edu/lapop/)

lapop_cc *LAPOP Cross-Country Bar Graphs*

Description

This function creates bar graphs for comparing values across countries using LAPOP formatting.

Usage

```
lapop_cc(
  data,
  outcome_var = data$prop,
  lower_bound = data$lb,
  vallabel = data$vallabel,
  upper_bound = data$ub,
  label_var = data$proplabel,
  ymin = 0,
  ymax = 100,
  lang = "en",
  highlight = "",
  main_title = "",
  source_info = "LAPOP",
  subtitle = "",
  sort = "",
  color_scheme = "#784885",
  label_size = 5,
  max_countries = 30,
  label_angle = 0
)
```

Arguments

data	Data Frame. Dataset to be used for analysis. The data frame should have columns titled vallabel (values of x-axis variable (e.g. pais); character vector), prop (outcome variable; numeric), proplabel (text of outcome variable; character), lb (lower bound of estimate; numeric), and ub (upper bound of estimate; numeric). Default: None (must be supplied).
vallabel, outcome_var, label_var, lower_bound, upper_bound	Character, numeric, character, numeric, numeric. Each component of the plot data can be manually specified in case the default columns in the data frame should not be used (if, for example, the values for a given variable were altered and stored in a new column). x
ymin, ymax	Numeric. Minimum and maximum values for y-axis. Default: 0 to 100.
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".
highlight	Character. Country of interest. Will highlight (make darker) that country's bar. Input must match entry in "vallabel" exactly. Default: None.
main_title	Character. Title of graph. Default: None.
source_info	Character. Information on dataset used (country, years, version, etc.), which is added to the bottom-left corner of the graph. Default: LAPOP ("Source: LAPOP Lab" will be printed).
subtitle	Character. Describes the values/data shown in the graph, e.g., "percentage of Mexicans who say...)". Default: None.

sort	Character. Method of sorting bars. Options: "hi-lo" (highest to lowest y value), "lo-hi" (lowest to highest), "alpha" (alphabetical by vallabel/x-axis label). Default: Order of data frame.
color_scheme	Character. Color of bars. Takes hex number, beginning with "#". Default: #784885.
label_size	Numeric. Size of text for data labels (percentages above bars). Default: 5.
max_countries	Numeric. Threshold for automatic x-axis label rotation. When the number of unique country labels exceeds this value, labels will be rotated for better readability. Default: 20.
label_angle	Numeric. Angle (in degrees) to rotate x-axis labels when max_countries is exceeded. Default: 0.

Value

Returns an object of class ggplot, a ggplot figure showing average values of some variables across multiple countries.

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()

df <- data.frame(
vallabel = c("PE", "CO", "BR", "PN", "GT", "DO", "MX", "BO", "EC"),
prop      = c(36.1, 19.3, 16.6, 13.3, 13.0, 11.1,  9.5,  9.0,  8.1),
proplabel = c("36%" ,"19%" ,"17%" ,"13%" ,"13%" ,"11%" ,"10%", "9%", "8%"),
lb        = c(34.9, 18.1, 15.4, 12.1, 11.8,  9.9,  8.3,  7.8,  6.9),
ub        = c(37.3, 20.5, 17.8, 14.5, 14.2, 12.3, 10.7, 10.2,  9.3)
)
lapop_cc(df,
         main_title = "Normalization of Intimate Partner Violence in LAC Countries",
         subtitle = "% who say domestic violence is private matter",
         source_info = "LAPOP Lab, AmericasBarometer 2021",
         highlight = "PE",
         ymax = 50)
```

---

lapop_cccomb *LAPOP Bar Graphs*

Description

This function shows a bar graph for categorical variables using LAPOP formatting.

Usage

```
lapop_cccomb(
  cc1,
  cc2,
  subtitle1 = "",
  subtitle2 = "",
  main_title = "",
  source_info = "",
  lang = "en",
  color_scheme = "#784885",
  file_name = "",
  width_px = 895,
  height_px = 600
)
```

Arguments

cc1, cc2	lapop_cc (ggplot) object. Graphic for left and right panes, respectively.
subtitle1, subtitle2	Character. Describes the values/data shown in the graph, e.g., "Percent who agree that...". Default: None.
main_title	Character. Title of graph. Default: None.
source_info	Character. Information on dataset used (country, years, version, etc.), which is added to the end of "Source: LAPOP Lab" in the bottom-left corner of the graph. Default: None (only "Source: LAPOP Lab" will be printed).
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".
color_scheme	Character. Color of bars. Takes hex numbers, beginning with "#". Default: "#008381".
file_name	Character. If desired, supply file path + name to save graph.
width_px, height_px	Numeric. Width and height of saved graph in pixels. Default: 895, 600.

Value

Returns an object of class ggplot, a ggplot bar graph.

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()
df1 <- data.frame(vallabel = c("Crime victim", "Non-victim"),
prop = c(36.1, 19.3),
```

```
proplabel = c("36%" ,"19%"),
lb = c(34.9, 18.1),
ub = c(37.3, 20.5))

df2 <- data.frame(vallabel = c("Crime victim", "Non-victim"),
prop = c(45, 15),
proplabel = c("45%" ,"15%"),
lb = c(43, 13),
ub = c(47, 16))

ccx <- lapop_cc(df1)
ccy <- lapop_cc(df2)

lapop_cccomb(ccx, ccy,
subtitle1 = "% who support democracy",
subtitle2 = "% who are satisfied with democracy",
main_title = "Crime victims are more supportive of and satisfied with democracy",
source_info = ", AmericasBarometer 2023")
```

---

lapop_ccm *LAPOP Cross-Country Bar Graphs*

Description

This function creates bar graphs for comparing values across countries using LAPOP formatting.

Usage

```
lapop_ccm(
  data,
  pais = data$pais,
  outcome_var = data$prop,
  lower_bound = data$lb,
  upper_bound = data$ub,
  label_var = data$proplabel,
  var = data$var,
  ymin = 0,
  ymax = 100,
  lang = "en",
  main_title = "",
  source_info = "",
  subtitle = "",
  sort = "",
  y_label = "",
  x_label = "",
  highlight = "",
  color_scheme = c("#784885", "#008381", "#C74E49"),
```

```
  label_size = 4,
  text_position = 0.7
)
```

Arguments

`data`	Data Frame. Dataset to be used for analysis. The data frame should have columns titled pais (values of x-axis variable (usually pais); character vector), prop (outcome variable; numeric), proplabel (text of outcome variable; character), lb (lower bound of estimate; numeric), ub (upper bound of estimate; numeric), and var (labels of secondary variables; character). Default: None (must be supplied).
`pais`, `outcome_var`, `label_var`, `lower_bound`, `upper_bound`, `var`	Character, numeric, character, numeric, numeric, character. Each component of the plot data can be manually specified in case the default columns in the data frame should not be used (if, for example, the values for a given variable were altered and stored in a new column).
`ymin`, `ymax`	Numeric. Minimum and maximum values for y-axis. Default: 0 to 100.
`lang`	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".
`main_title`	Character. Title of graph. Default: None.
`source_info`	Character. Information on dataset used (country, years, version, etc.), which is added to the end of "Source: " in the bottom-left corner of the graph. Default: None (only "Source: " will be printed).
`subtitle`	Character. Describes the values/data shown in the graph, e.g., "percentage of Mexicans who say...)". Default: None.
`sort`	Character. Method of sorting bars. Options: "var1" (highest to lowest on variable 1), "var2" (highest to lowest on variable 2), "var3" (highest to lowest on variable 3), "alpha" (alphabetical along x-axis/pais). Default: Order of data frame.
`y_label`	Character. Y-axis label.
`x_label`	Character. X-axis label.
`highlight`	Character. Country of interest. Will highlight (make darker) that country's bar. Input must match entry in "vallabel" exactly. Default: None.
`color_scheme`	Character. Color of bars. Takes hex number, beginning with "#". Default: "#784885", "#008381", "#C74E49".
`label_size`	Numeric. Size of text for data labels (percentages above bars). Default: 4.
`text_position`	Numeric. Amount that text above error bars should be offset (to avoid overlap). Default: 0.7

Value

Returns an object of class `ggplot`, a ggplot figure showing average values of some variables across multiple countries.

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()

df <- data.frame(pais = c(rep("HT", 2), rep("PE", 2), rep("HN", 2), rep("CO", 2),
             rep("UY", 2), rep("CR", 2), rep("EC", 2), rep("CL", 2),
              rep("BR", 2), rep("BO", 2), rep("JA", 2), rep("PN", 2)),
              var = rep(c("countfair1", "countfair3"), 3),
              prop = c(30, 38, 40, 49, 57, 33, 80, 54, 30, 43, 61, 42,
                       38, 54, 74, 61, 50, 34, 48, 34, 72, 41, 58, 57),
              proplabel = c("30%", "38%", "40%", "49%", "57%", "33%",
                            "80%", "54%", "30%", "43%", "61%", "42%",
                            "38%", "54%", "74%", "61%", "50%", "34%",
                            "48%", "34%", "72%", "41%", "58%", "57%"),
              lb = c(27, 35, 37, 46, 54, 30, 77, 51, 27, 40, 58, 39,
                     35, 51, 71, 58, 47, 31, 45, 31, 69, 38, 55, 54),
              ub = c(33, 41, 43, 52, 60, 36, 83, 57, 33, 46, 64, 45,
                     41, 57, 77, 64, 53, 37, 51, 37, 75, 44, 61, 60))

lapop_ccm(df, sort = "var", source_info = ", AmericasBarometer")
```

---

lapop_coef *LAPOP Regression Graphs*

Description

This function creates plots of regression coefficients and predicted probabilities using LAPOP formatting.

Usage

```
lapop_coef(
  data,
  coef_var = data$coef,
  label_var = data$proplabel,
  varlabel_var = data$varlabel,
  lb = data$lb,
  ub = data$ub,
  pval_var = data$pvalue,
  lang = "en",
  main_title = "",
  subtitle = "",
  source_info = "",
  ymin = NULL,
```

```
  ymax = NULL,
  pred_prob = FALSE,
  color_scheme = "#784885",
  subtitle_h_just = 0
)
```

Arguments

data	Data Frame. Dataset to be used for analysis. The data frame should have columns titled coef (regression coefficients/predicted probabilities; numeric), proplabel (text of outcome variable; character), varlabel (names of variables to be plotted; character), lb (lower bound of coefficient estimate; numeric), ub (upper bound of estimate; numeric), and pvalue (p value of coefficient estimate; numeric). Default: None (must be supplied).
coef_var, label_var, varlabel_var, lb, ub, pval_var	Numeric, character, character, numeric, numeric, numeric. Each component of the data to be plotted can be manually specified in case the default columns in the data frame should not be used (if, for example, the values for a given variable were altered and stored in a new column).
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".
main_title	Character. Title of graph. Default: None.
subtitle	Character. Describes the values/data shown in the graph, e.g., "Regression coefficients". Default: None.
source_info	Character. Information on dataset used (country, years, version, etc.), which is added to the end of "Source: " in the bottom-left corner of the graph. Default: None (only "Source: " will be printed).
ymin, ymax	Numeric. Minimum and maximum values for y-axis. Default: dynamic.
pred_prob	Logical. Is the graph showing predicted probabilities (instead of regression coefficients)? Will only change text in the legend, not the data. Default: FALSE.
color_scheme	Character. Color of bars. Takes hex number, beginning with "#". Default: "#784885" (purple).
subtitle_h_just	Numeric. Move the subtitle/legend text left (negative numbers) or right (positive numbers). Ranges from -100 to 100. Default: 0.

Value

Returns an object of class ggplot, a ggplot figure showing coefficients or predicted probabilities from a multivariate regression.

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()
df <- data.frame(
  varlabel = c("Intimate\nPartner", "wealth", "Education", "Age", "Male"),
  coef = c(0.02, -0.07, -0.24, 0.01, 0.11),
  lb = c(-0.002, -0.110, -0.295, -0.060, 0.085),
  ub = c(0.049, -0.031, -0.187, 0.080, 0.135),
  pvalue = c(0.075, 0.000, 0.000, 0.784, 0.000),
  proplabel = c("0.02", "-0.07", "-0.24", "0.01", "0.11")
)

lapop_coef(df,
           main_title = "Demographic and Socioeconomic Predictors of Normalizing IPV",
           pred_prob = TRUE,
           source_info = ", AmericasBarometer 2021",
           ymin = -0.3,
           ymax = 0.2)
```

---

lapop_dumb *LAPOP Dummbell Graphs*

Description

This function creates "dumbbell" graphs, which show averages for a variable across countries over two time periods, using LAPOP formatting.

Usage

```
lapop_dumb(
  data,
  ymin = 0,
  ymax = 100,
  lang = "en",
  main_title = "",
  source_info = "",
  subtitle = "",
  sort = "wave2",
  order = "hi-lo",
  color_scheme = c("#008381", "#A43D6A"),
  subtitle_h_just = 40,
  subtitle_v_just = -18,
  text_nudge = 6,
  drop_singles = FALSE
)
```

Arguments

data	Data Frame. Dataset to be used for analysis. The data frame should have columns titled pais (country name; character), wave1 (name of first wave/year (all rows are the same); character), prop1 (outcome variable values for the first wave; numeric), proplabel1 (text of outcome variable for first wave; character), wave2 (name of second wave/year (all rows are the same); character), prop2 (outcome variable values for the second wave; numeric), proplabel2 (text of outcome variable for second wave; character). Default: None (must be supplied).
ymin, ymax	Numeric. Minimum and maximum values for y-axis. Defaults: 0 and 100.
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".
main_title	Character. Title of graph. Default: None.
source_info	Character. Information on dataset used (country, years, version, etc.), which is added to the end of "Source: " in the bottom-left corner of the graph. Default: LAPOP (only "Source: "LAPOP Lab will be printed).
subtitle	Character. Describes the values/data shown in the graph, e.g., "Percent who agree that...". Default: None.
sort	Character. The metric by which the data are sorted. Options: "wave1" (outcome variable in first wave), "wave2" (outcome variable in wave 2), "diff" (difference between the two waves), "alpha" (alphabetical by country name). Default: "wave2".
order	Whether data should be sorted from low to high or high to low on the sort metric. Options: "hi-lo" and "lo-hi". Default: "hi-lo".
color_scheme	Character. Color of data points. Must supply two values. Takes hex numbers, beginning with "#". Default: "#482677", "#3CBC70".
subtitle_h_just, subtitle_v_just	Numeric. Move the subtitle/legend text left/down (negative numbers) or right/up (positive numbers). Ranges from -100 to 100. Defaults: 40, -18.
text_nudge	Numeric. Move text of data further or closer to data point. Default: 6.
drop_singles	Logical. Should rows with only one dot be removed? Default: FALSE.

Value

Returns an object of class ggplot, a ggplot figure showing average values of some variable in two time periods across multiple countries (a dumbbell plot).

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()
df <- data.frame(pais = c("Haiti", "Peru", "Honduras", "Colombia", "Ecuador",
```

```
                          "Panama", "Bolivia", "Argentina", "Paraguay",
                          "Dom. Rep.", "Brazil", "Jamaica", "Nicaragua",
                          "Guyana", "Costa Rica", "Mexico", "Guatemala",
                          "Chile", "Uruguay", "El Salvador"),
                 wave1 = rep("2018/19", 20),
                 prop1 = c(NA, 30, 58, 40, 49, 57, 33, 68, 38, 46, 30,
                           31, 70, NA, 43, 25, 38, 31, 34, 41),
                 proplabel1 = c(NA, "30%", "58%", "40%", "49%", "57%", "33%",
                                "68%", "38%", "46%", "30%", "31%", "70%", NA,
                                "43%", "25%", "38%", "31%", "34%", "41%"),
                 wave2 = rep("2021", 20),
                 prop2 = c(86, 73, 69, 67, 67, 65, 65, 65, 63, 62, 62,
                           57, 56, 56, 55, 55, 54, 51, 46, 42),
                 proplabel2 = c("86%", "73%", "69%", "67%", "67%", "65%", "65%",
                                "65%", "63%", "62%", "62%", "57%", "56%", "56%",
                                "55%", "55%", "54%", "51%", "46%", "42%"))

lapop_dumb(df,
main_title = paste0("Personal economic conditions worsened across LAC"),
         subtitle = "% personal economic situation worsened",
         source_info = "Source: LAPOP Lab, AmericasBarometer 2018/19-2021")
```

`lapop_fonts` *LAPOP Fonts*

Description

This function loads fonts needed for LAPOP graph formatting. No arguments needed; just run lapop_fonts() at the beginning of your session.

Usage

```
lapop_fonts()
```

Value

No return value, called for side effects

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()
```

lapop_fonts_design *LAPOP Fonts (design)*

Description

This function loads fonts needed for LAPOP graph formatting. In contrast to lapop_fonts(), this renders text as text instead of polygons, which allows post-hoc editing.

Usage

```
lapop_fonts_design()
```

Value

No return value, called for side effects

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu>

lapop_hist *LAPOP Bar Graphs*

Description

This function shows a bar graph for categorical variables using LAPOP formatting.

Usage

```
lapop_hist(
  data,
  outcome_var = data$prop,
  label_var = data$proplabel,
  cat_var = data$cat,
  ymin = 0,
  ymax = 100,
  lang = "en",
  main_title = "",
  subtitle = "",
  source_info = "LAPOP",
  order = FALSE,
  color_scheme = "#008381"
)
```

Arguments

data	Data Frame. Dataset to be used for analysis. The data frame should have columns titled cat (labels of each category in variable; character), prop (outcome variable value; numeric), and proplabel (text of outcome variable value; character). Default: None (must be provided).
cat_var, outcome_var, label_var	Character, numeric, character. Each component of the data to be plotted can be manually specified in case the default columns in the data frame should not be used (if, for example, the values for a given variable were altered and stored in a new column).
ymin, ymax	Numeric. Minimum and maximum values for y-axis. Defaults: 0, 100.
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".
main_title	Character. Title of graph. Default: None.
subtitle	Character. Describes the values/data shown in the graph, e.g., "Percent who agree that...". Default: None.
source_info	Character. Information on dataset used (country, years, version, etc.), which is added to the bottom-left corner of the graph. Default: LAPOP ("Source: LAPOP Lab" will be printed).
order	Logical. Should bars be ordered from most frequent response to least? Default: FALSE.
color_scheme	Character. Color of bars. Takes hex numbers, beginning with "#". Default: "#008381".

Value

Returns an object of class ggplot, a ggplot bar graph.

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()

df <- data.frame(
cat = c("Far Left", 1, 2, 3, 4, "Center", 6, 7, 8, 9, "Far Right"),
prop = c(4, 3, 5, 12, 17, 23, 15, 11, 5, 4, 1),
proplabel = c("4%", "3%", "5%", "12%", "17%", "23%", "15%", "11%", "5%", "4%", "1%")
)
lapop_hist(df,
           main_title = "Centrists are a plurality among Peruvians",
           subtitle = "Distribution of ideological preferences",
           source_info = "Source: LAPOP Lab, AmericasBarometer Peru 2019",
           ymax = 27)
```

lapop_map *LAPOP World and Americas Map Graph*

Description

The ‘lapop_map()‘ generates a stylized choropleth map using ISO2 country codes from both continuous and factor variables. It is designed to map cross-country results from ‘lpr_cc()‘ and supports either a full world map (‘survey = "CSES"‘) # or an Americas-only map (‘survey = "AmericasBarometer"‘).

Usage

```
lapop_map(
  data,
  outcome = "value",
  pais_lab = "pais_lab",
  survey = c("CSES", "AmericasBarometer"),
  zoom = 1,
  main_title = "",
  subtitle = "",
  palette = c("#F2A344", "#D97A1E", "#BF5A00", "#8A3900", "#4A1E00"),
  source_info = "LAPOP",
  lang = "en",
  selected_countries = NULL
)
```

Arguments

data	A data frame containing ISO2 country codes and a value to map.
outcome	String. Column name containing the numeric or categorical variable to visualize.
pais_lab	String. Column name containing ISO2 country codes (e.g., ‘"US"‘, ‘"BR"‘).
survey	Either ‘"CSES"‘ (full world map) or ‘"AmericasBarometer"‘ (Americas only).
zoom	Numeric (0–1). Controls how tightly the map zooms when ‘survey = "AmericasBarometer"‘. Default is ‘1‘.
main_title	Character. Title of graph. Default: None.
subtitle	Character. Describes the values/data shown in the graph, e.g., "percentage of Mexicans who say...)". Default: None.
palette	Vector of up to 5 colors for continuous and factor variables.
source_info	Character. Information on dataset used (country, years, version, etc.), which is added to the bottom-left corner of the graph. Default: LAPOP ("Source: LAPOP Lab" will be printed).
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".

selected_countries
: Character or NULL. ISO2 code of the currently selected country (e.g. from ‘input$pais‘ in Shiny). When not ‘NULL‘, countries with no data are rendered with diagonal stripes instead of solid gray. Default: ‘NULL‘ (solid ‘"#dddddf"‘).

Value

A ‘ggplot2‘ choropleth map object.

Author(s)

Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
# Standalone — solid gray for no-data countries
lapop_fonts()
 data_cont <- data.frame(
  vallabel = c("US", "AR", "VE", "CH", "EC", "BO"),
  prop = c(37, 52, 80, 17, 69, 94)
)
lapop_map(data_cont, pais_lab = "vallabel", outcome = "prop", zoom = 0.9,
       survey = "AmericasBarometer", main_title = "Latin America and Caribbean Countries",
          subtitle = "% of respondents")


if (interactive()) {
  lapop_map(data_cont, pais_lab = "vallabel", outcome = "prop", zoom = 0.9,
            survey = "AmericasBarometer", selected_countries = "BR")
}
```

---

lapop_mline *LAPOP Multi-line Time-Series Graphs*

Description

This function creates a time series graph utilizing multiple lines representing values of an outcome variable for different values of a secondary variable – for example, support for democracy over time by country. This function is designed to be used for AmericasBarometer data. The maximum number of lines is four. Unlike the lapop_ts() single-line time series graph, this function will not print confidence lines nor will it show text values for each year (just the final/most recent year).

Usage

```
lapop_mline(
  data,
  varlabel = data$varlabel,
  wave_var = as.character(data$wave),
  outcome_var = data$prop,
```

```
  label_var = data$proplabel,
  point_var = data$prop,
  ymin = 0,
  ymax = 100,
  main_title = "",
  source_info = "",
  subtitle = "",
  lang = "en",
  legend_h_just = 40,
  legend_v_just = -20,
  subtitle_h_just = 0,
 color_scheme = c("#784885", "#008381", "#c74e49", "#2d708e", "#a43d6a", "#202020"),
  percentages = TRUE,
  all_labels = FALSE,
  ci = FALSE,
  legendnrow = 1
)
```

Arguments

`data`	Data Frame. Dataset to be used for analysis. The data frame should have columns titled varlabel (values of secondary variable which will be used to make each line; character), wave (survey wave/year; character), prop (outcome variable; numeric), proplabel (text of outcome variable; character). Default: None (must be supplied).
`varlabel`, `wave_var`, `outcome_var`, `label_var`, `point_var`	Character, character, numeric, character, numeric. Each component of the data to be plotted can be manually specified in case the default columns in the data frame should not be used (if, for example, the values for a given variable were altered and stored in a new column).
`ymin`, `ymax`	Numeric. Minimum and maximum values for y-axis. Default: 0, 100.
`main_title`	Character. Title of graph. Default: None.
`source_info`	Character. Information on dataset used (country, years, version, etc.), which is added to the end of "Source: " in the bottom-left corner of the graph. Default: None (only "Source: " will be printed).
`subtitle`	Character. Describes the values/data shown in the graph, e.g., "Percent of Mexicans who agree...". Default: None.
`lang`	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. #' Takes either "en" (English) or "es" (Spanish). Default: "en".
`legend_h_just`, `legend_v_just`	Numeric. Changes location of legend. From 0 to 100. (secondary variable labels). Defaults: 40, -20.
`subtitle_h_just`	Numeric. Moves subtitle left to right. From 0 to 1. (secondary variable labels). Defaults: 0 (left justify).

color_scheme	Character. Color of lines and dots. Takes hex number, beginning with "#". Must specify four values, even if four are not used. Default: c("#784885", "#008381", "#c74e49", "#2d708e", "#a43d6a", "#202020").
percentages	Logical. Is the outcome variable a percentage? Set to FALSE if you are using means of the raw values, so that the y-axis adjusts accordingly. Default: TRUE.
all_labels	Logical. If TRUE, show text above all points, instead of only those in the most recent wave. Default: FALSE.
ci	Logical. Add "tie fighter" confidence intervals. Only recommended when each line represents a different variable.
legendnrow	Numeric. How many rows for legend labels. Default: 1.

Value

Returns an object of class ggplot, a ggplot line graph showing values of a variable over time.

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
df <- data.frame(varlabel = c(rep("Honduras", 9), rep("El Salvador", 9),
                              rep("Mexico", 9), rep("Guatemala", 9)),
                 wave = rep(c("2004", "2006", "2008", "2010", "2012",
                              "2014", "2016/17", "2018/19", "2021"), 4),
                 prop = c(19, 24, 21, 15, 11, 32, 41, 38, 54,
                          29, 29, 25, 24, 24, 28, 36, 26, 32,
                          14, 16, 14, 16, 9, 14, 18, 19, 26,
                          21, 15, 18, 20, 14, 18, 17, 25, 36),
                 proplabel = c("19%", "24%", "21%", "15%", "11%", "32%",
                               "41%", "38%", "54%",
                               "29%", "29%", "25%", "24%", "24%", "28%",
                               "36%", "26%", "32%",
                               "14%", "16%", "14%", "16%", "9%", "14%",
                               "18%", "19%", "26%",
                               "21%", "15%", "18%", "20%", "14%", "18%",
                               "17%", "25%", "36%"))
require(lapop); lapop_fonts()
lapop_mline(df,
    main_title = "Intentions to emigrate in Guatemala, Honduras and Mexico reached their highs",
      subtitle = "% who intend to migrate in:",
      source_info = ", AmericasBarometer 2004-2021")
```

lapop_mover *LAPOP Multiple-Over/Breakdown Graphs*

Description

This function shows the values of an outcome variable for subgroups of a secondary variable, using LAPOP formatting.

Usage

```
lapop_mover(
  data,
  lang = "en",
  main_title = "",
  subtitle = "",
  qword = NULL,
  source_info = "LAPOP",
  rev_values = FALSE,
  rev_variables = FALSE,
  subtitle_h_just = 0,
  ymin = 0,
  ymax = 100,
  x_lab_angle = 90,
  color_scheme = c("#784885", "#008381", "#c74e49", "#2d708e", "#a43d6a")
)
```

Arguments

data	Data Frame. Dataset to be used for analysis. The data frame should have columns titled varlabel (name(s)/label(s) of secondary variable(s); character), vallabel (names/labels of values for secondary variable; character), prop (outcome variable value; numeric), proplabel (text of outcome variable value; character), lb (lower bound of estimate; numeric), and ub (upper bound of estimate; numeric). Default: None (must be provided).
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".
main_title	Character. Title of graph. Default: None.
subtitle	Character. Describes the values/data shown in the graph, e.g., "Percent who agree that...". Default: None.
qword	Character. Describes the question wording shown in the graph, e.g., "Do you agree that...". Default: NULL.
source_info	Character. Information on dataset used (country, years, version, etc.), which is added to the bottom-left corner of the graph. Default: LAPOP ("Source: LAPOP Lab" will be printed).

rev_values	Logical. Should the order of the values for each variable be reversed? Default: FALSE.
rev_variables	Logical. Should the order of the variables be reversed? Default: FALSE.
subtitle_h_just	Numeric. Move the subtitle/legend text left (negative numbers) or right (positive numbers). Ranges from -100 to 100. Default: 0.
ymin, ymax	Numeric. Minimum and maximum values for y-axis. Defaults: 0 and 100.
x_lab_angle	Numeric. Angle/orientation of the value labels. Default: 90.
color_scheme	Character. Color of data points and text for each secondary variable. Allows up to 6 values. Takes hex numbers, beginning with "#". Default: c("#784885", "#008381", "#c74e49", "#2d708e", "#a43d6a") (purple, teal, green, olive, sap green, pea soup).

Value

Returns an object of class ggplot, a ggplot figure showing average values of some variable broken down by one or more secondary variables (commonly, demographic variables).

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
df <- data.frame(varlabel = c(rep("Gender", 2), rep("Age", 6),
                              rep("Education", 4), rep("Wealth", 5)),
                 vallabel = c("Women", "Men", "18-25", "26-35", "36-45",
                              "46-55", "56-65", "66+", "  None", "Primary",
                              "Secondary", "Post-Sec.", "Low", "2",
                              "3", "4", "High"),
                 prop = c(20, 22, 21, 24, 22, 21, 17, 15, 20, 18, 21, 25, 21,
                          21, 21, 21, 22),
                 proplabel = c("20%", "22%", "21%", "24%", "22%", "21%",
                               "17%", "15%", "20%", "18%", "21%", "25%",
                               "21%", "21%", "21%", "21%", "22%"),
                 lb = c(19, 21, 20, 23, 21, 20, 15, 13, 16, 17, 20, 24, 20,
                        20, 20, 20, 21),
                 ub = c(21, 23, 22, 25, 23, 22, 19, 17, 24, 19, 22, 26, 22,
                        22, 22, 22, 23))
require(lapop); lapop_fonts
lapop_mover(df,
            main_title = paste0("More educated, men, and younger individuals",
                                " in the LAC region are the\nmost likely",
                                  " to be crime victims"),
            subtitle = "% victim of a crime", qword = "",
            source_info = "Source: LAPOP Lab, AmericasBarometer",
            ymin = 0,
            ymax = 40)
```

lapop_save *LAPOP Save*

Description

This function creates exports graphs created using the LAPOP templates.

Usage

```
lapop_save(
  figure,
  filename,
  format = "svg",
  logo = FALSE,
  width_px = 895,
  height_px = 500
)
```

Arguments

figure	Ggplot object.
filename	File path + name to be saved + .filetype.
format	Character. Options: "png", "svg". Default = "svg".
logo	Logical. Should logo be added to plot? Default: FALSE.
width_px	Numeric. Width in pixels. Default: 750.
height_px	Numeric. Height in pixels.

Value

Saves a file (in either .svg or .png format) to provided directory.

Examples

```
df <- data.frame(
cat = c("Far Left", 1, 2, 3, 4, "Center", 6, 7, 8, 9, "Far Right"),
prop = c(4, 3, 5, 12, 17, 23, 15, 11, 5, 4, 1),
proplabel = c("4%", "3%", "5%", "12%", "17%", "23%", "15%", "11%", "5%", "4%", "1%")
)
myfigure <- lapop_hist(df,
          main_title = "Centrists are a plurality among Peruvians",
          subtitle = "Distribution of ideological preferences",
          source_info = "Peru, 2019",
          ymax = 27
)

f <- file.path(tempdir(), "fig1.svg")
lapop_save(myfigure, f, format = "svg", width_px = 800)
```

lapop_stack *LAPOP Stacked Bar Graphs*

Description

This function shows a stacked bar graph using LAPOP formatting.

Usage

```
lapop_stack(
  data,
  outcome_var = data$prop,
  prop_labels = data$proplabel,
  var_labels = data$varlabel,
  value_labels = data$vallabel,
  xvar = NULL,
  lang = "en",
  main_title = "",
  subtitle = "",
  source_info = "LAPOP",
  rev_values = FALSE,
  rev_variables = FALSE,
  hide_small_values = TRUE,
  order_bars = FALSE,
  subtitle_h_just = 0,
  fixed_aspect_ratio = TRUE,
  legendnrow = 1,
 color_scheme = c("#2D708E", "#008381", "#C74E49", "#784885", "#a43d6a", "#202020")
)
```

Arguments

data	Data Frame. Dataset to be used for analysis. The data frame should have columns titled varlabel (name(s)/label(s) of variable(s) of interest; character), vallabel (names/labels of values for each variable; character), prop (outcome variable value; numeric), and proplabel (text of outcome variable value; character). Default: None (must be provided).
outcome_var, prop_labels, var_labels, value_labels	Numeric, character, character, character. Each component of the data to be plotted can be manually specified in case the default columns in the data frame should not be used (if, for example, the values for a given variable were altered and stored in a new column).
xvar	Character. Column name to group the plots by. This should match a column name in the dataset. Default: NULL (no grouping).
lang	Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. Takes either "en" (English) or "es" (Spanish). Default: "en".

`main_title` Character. Title of graph. Default: None.

`subtitle` Character. Describes the values/data shown in the graph, e.g., "Percent who support...". Default: None.

`source_info` Character. Information on dataset used (country, years, version, etc.), which is added to the end of "Source: " in the bottom-left corner of the graph. Default: LAPOP ("Source: LAPOP Lab" will be printed).

`rev_values` Logical. Should the order of the values for each variable be reversed? Default: FALSE.

`rev_variables` Logical. Should the order of the variables be reversed? Default: FALSE.

`hide_small_values` Logical. Should labels for categories with less than 5 percent be hidden? Default: TRUE.

`order_bars` Logical. Should categories be placed in descending order for each bar? Default: FALSE. showing the distributions of multiple categorical variables.

`subtitle_h_just` Numeric. Move the subtitle/legend text left (negative numbers) or right (positive numbers). Ranges from -100 to 100. Default: 0.

`fixed_aspect_ratio` Logical. Should the aspect ratio be set to a specific value (0.35)? This prevents bars from stretching vertically to fit the plot area. Set to false when you have a large number of bars (> 10). Default: TRUE.

`legendnrow` Numeric. How many rows for legend labels. Default: 1.

`color_scheme` Character. Color of data bars for each value. Allows up to 6 values. Takes hex numbers, beginning with "#". Default: c("#2D708E", "#008381", "#C74E49", "#784885", "#a43d6a","#202020") (navy blue, turquoise, teal, green, sap green, pea soup).

Value

Returns an object of class `ggplot`, a ggplot stacked bar graph

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
df <- data.frame(varlabel = c(rep("Politicians can\nidentify voters", 5),
                              rep("Wealthy can\nbuy results", 5),
                              rep("Votes are\ncounted correctly", 5)),
                 vallabel = rep(c("Always", "Often", "Sometimes",
                                  "Never", "Other"), 3),
                 prop = c(36, 10, 19, 25, 10, 46, 10, 23, 11, 10, 35,
                          10, 32, 13, 10),
                 proplabel = c("36%", "10%", "19%", "25%", "10%", "46%",
                               "10%", "23%", "11%", "10%", "35%", "10%",
                               "32%", "13%", "10%"))
```

```
require(lapop); lapop_fonts()
lapop_stack(df,
      main_title = "Trust in key features of the electoral process is low in Latin America",
        subtitle = "% believing it happens:",
        source_info = "Source: LAPOP Lab, AmericasBarometer 2019")
```

---

lapop_ts *LAPOP Time-Series Graphs*

Description

This function creates time series graphs using LAPOP formatting. If there are waves missing at the beginning or end of the series, the function will omit those waves from the graph (i.e., the x-axis will range from the earliest wave for which data is supplied to the latest). If there are waves missing in the middle of the series, those waves will be displayed on the x-axis, but no data will be shown.

Usage

```
lapop_ts(
  data,
  outcome_var = data$prop,
  lower_bound = data$lb,
  upper_bound = data$ub,
  wave_var = as.character(data$wave),
  label_var = data$proplabel,
  point_var = data$prop,
  ymin = 0,
  ymax = 100,
  main_title = "",
  source_info = "LAPOP",
  subtitle = "",
  lang = "en",
  color_scheme = "#A43D6A",
  percentages = TRUE,
  label_vjust = -2.1,
  max_years = 15,
  label_angle = 0,
  ci_type = "linerange"
)
```

Arguments

data	Data Frame. Dataset to be used for analysis. The data frame should have columns titled wave (survey wave/year; character vector), prop (outcome variable; numeric), proplabel (text of outcome variable; character); lb (lower bound of estimate; numeric), and ub (upper bound of estimate; numeric). Default: None (must be supplied).

`wave_var`, `outcome_var`, `label_var`, `lower_bound`, `upper_bound`, `point_var`
: Character, numeric, character, numeric, numeric, character. Each component of the data to be plotted can be manually specified in case the default columns in the data frame should not be used (if, for example, the values for a given variable were altered and stored in a new column).

`ymin`, `ymax`
: Numeric. Minimum and maximum values for y-axis. Default: 0, 100.

`main_title`
: Character. Title of graph. Default: None.

`source_info`
: Character. Information on dataset used (country, years, version, etc.), which is added to the bottom-left corner of the graph. Default: LAPOP ("Source: LAPOP Lab" will be printed).

`subtitle`
: Character. Describes the values/data shown in the graph, e.g., "Percent of Mexicans who agree...". Default: None.

`lang`
: Character. Changes default subtitle text and source info to either Spanish or English. Will not translate input text, such as main title or variable labels. #' Takes either "en" (English) or "es" (Spanish). Default: "en".

`color_scheme`
: Character. Color of lines and dots. Takes hex number, beginning with "#". Default: "#A43D6A" (red).

`percentages`
: Logical. Is the outcome variable a percentage? Set to FALSE if you are using means of the raw values, so that the y-axis adjusts accordingly. Default: TRUE.

`label_vjust`
: Numeric. Customize vertical space between points and their labels. Default: -2.1.

`max_years`
: Numeric. Threshold for automatic x-axis label rotation. When the number of unique country labels exceeds this value, labels will be smaller and if necessary rotated for better readability. Default: 15 years.

`label_angle`
: Numeric. Angle (in degrees) to rotate x-axis labels when max_years is exceeded. Default: 0.

`ci_type`
: Character. Controls how confidence intervals are displayed in the time-series plot. This parameter only affects how the confidence interval is visualized; the point estimate and line plot remain unchanged. Options:

  - `"linerange"` (default): Draws upper and lower bounds as dashed lines.
  - `"errorbar"`: Displays confidence intervals using vertical error bars centered on the point estimate.
  - `"ribbon"`: Shows a shaded confidence band between the lower and upper bounds.
  - `"none"`: Suppresses confidence interval display.

Details

The input data must have a specific format to produce a graph. It must include columns for the survey wave (wave), the outcome variable (prop), the lower bound of the estimate (lb), the upper bound of the estimate (ub), and a string for the outcome variable label (proplabel).

Value

Returns an object of class `ggplot`, a ggplot line graph showing values of a variable over time.

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); lapop_fonts()

df <- data.frame(wave = c("2008", "2010", "2016/17", "2018/19", "2021"),
prop = c(23.2, 14.4, 35.8, 36.6, 40),
proplabel = c("23.2%", "14.4%", "35.8%", "36.6%", "40.0%"),
lb = c(20.2, 11.9, 33.3, 33.1, 38),
ub = c(26.2, 16.9, 38.3, 40.1, 42)
)

lapop_ts(df,
main_title = "Ecuadorians are becoming more interested in politics",
subtitle = "% politically interested",
source_info = "Source: LAPOP Lab, AmericasBarometer Ecuador 2006-2021",
ymin = 0,
ymax = 55)
```

---

lpr_cc *LAPOP Cross-Country Bar Graph Pre-Processing*

Description

This function creates dataframes which can then be input in lapop_cc for comparing values across countries with a bar graph using LAPOP formatting.

Usage

```
lpr_cc(
  data,
  outcome,
  xvar = "pais_lab",
  rec = list(c(1, 1)),
  rec2 = list(c(1, 1)),
  rec3 = list(c(1, 1)),
  rec4 = list(c(1, 1)),
  ci_level = 0.95,
  mean = FALSE,
  filesave = "",
  cfmt = "",
  sort = "y",
  order = "hi-lo",
  ttest = FALSE,
  keep_nr = FALSE
)
```

Arguments

data	A survey object. The data that should be analyzed.
outcome	Outcome variable(s) of interest to be plotted across countries. It can handle a single variable across countries, or multiple variables instead of multiple countries. See examples below.
xvar	Grouping variable. Default: pais_lab. It can handle other variables grouping like year/wave.
rec	Numeric. The minimum and maximum values of the outcome variable that should be included in the numerator of the percentage. For example, if the variable is on a 1-7 scale and rec is c(5, 7), the function will show the percentage who chose an answer of 5, 6, 7 out of all valid answers. Default: c(1, 1).
rec2	Numeric. Same as rec(). Default: c(1, 1).
rec3	Numeric. Same as rec(). Default: c(1, 1).
rec4	Numeric. Same as rec(). Default: c(1, 1).
ci_level	Numeric. Confidence interval level for estimates. Default: 0.95
mean	Logical. If TRUE, will produce the mean of the variable rather than rescaling to percentage. Default: FALSE.
filesave	Character. Path and file name to save the dataframe as csv.
cfmt	changes the format of the numbers displayed above the bars. Uses sprintf string formatting syntax. Default is whole numbers for percentages and tenths place for means.
sort	Character. On what value the bars are sorted: the x or the y. Options are "y" (default; for the value of the outcome variable), "xv" (for the underlying values of the x variable), "xl" (for the labels of the x variable, i.e., alphabetical).
order	Character. How the bars should be sorted. Options are "hi-lo" (default) or "lo-hi".
ttest	Logical. If TRUE, will conduct pairwise t-tests for difference of means between all individual x levels and save them in attr(x, "t_test_results"). Default: FALSE.
keep_nr	Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99) and use them in the denominator when calculating percentages. The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_cc

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(ym23); data(bra23)

# Set Survey Context on a small cross-country subset
ym23_small <- subset(ym23, pais %in% c(1, 15, 17))
ym23lpr <- lpr_data(ym23_small)

# Single variable in Multiple Countries
lpr_cc(data = ym23lpr,
      outcome = "ing4",
      rec = c(5, 7),
      xvar = "pais")

# Multiple variables in Single Country

bra23lpr <- lpr_data(bra23, wt = TRUE)
lpr_cc(data = bra23lpr,
outcome = c("b12", "b13"),
rec = c(5, 7))
```

---

lpr_ccm *LAPOP Grouped Bar Graph Pre-Processing*

Description

This function creates dataframes which can then be input in lapop_ccm for comparing values for multiple variables across countries with a bar graph using LAPOP formatting.

Usage

```
lpr_ccm(
  data,
  outcome_vars,
  xvar = "pais_lab",
  rec1 = c(1, 1),
  rec2 = c(1, 1),
  rec3 = c(1, 1),
  ci_level = 0.95,
  mean = FALSE,
  filesave = "",
  cfmt = "",
  sort = "y",
  order = "hi-lo",
  ttest = FALSE,
  keep_nr = FALSE
)
```

Arguments

data	A survey object. The data that should be analyzed.
outcome_vars	Character vector. Outcome variable(s) of interest to be plotted across country (or other x variable). Max of 3 (three) variables.
xvar	Character string. Outcome variables are broken down by this variable. You can set xvar to "wave" or "year" for cross-time comparisons. Default: pais_lab.
rec1, rec2, rec3	Numeric. The minimum and maximum values of the outcome variable that should be included in the numerator of the percentage. For example, if the variable is on a 1-7 scale and rec1 is c(5, 7), the function will show the percentage who chose an answer of 5, 6, 7 out of all valid answers. Can also supply one value only, to produce the percentage that chose that value out of all other values. Default: c(1, 1).
ci_level	Numeric. Confidence interval level for estimates. Default: 0.95
mean	Logical. If TRUE, will produce the mean of the variable rather than rescaling to percentage. Default: FALSE.
filesave	Character. Path and file name to save the dataframe as csv.
cfmt	Character. Changes the format of the numbers displayed above the bars. Uses sprintf string formatting syntax. Default is whole numbers for percentages and tenths place for means.
sort	Character. On what value the bars are sorted. Options are "y" (default; for the value of the first outcome variable), "xv" (for the underlying values of the x variable), "xl" (for the labels of the x variable, i.e., alphabetical).
order	Character. How the bars should be sorted. Options are "hi-lo" (default) or "lo-hi".
ttest	Logical. If TRUE, will conduct pairwise t-tests for difference of means between all outcomes vs. all x-vars and save them in attr(x, "t_test_results"). Default: FALSE.
keep_nr	Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99) and use them in the denominator when calculating percentages. The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_ccm()

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(ym23)

# Set Survey Context on a small cross-country subset
```

```
ym23_small <- subset(ym23, pais %in% c(1, 15, 17))
ym23lpr <- lpr_data(ym23_small)

# Multiple outcomes over countries
lpr_ccm(ym23lpr,
outcome_vars = c("b12", "b18"),
rec1 = c(1, 3),
rec2 = c(5, 7))

# Multiple outcomes over years

lpr_ccm(ym23lpr,
outcome_vars = c("b12", "b18"),
xvar = "wave",
rec1 = c(1, 3),
rec2 = c(5, 7),
ttest = TRUE)
```

`lpr_ci` *Compute Design-Based Proportion and Confidence Interval*

Description

Computes a weighted proportion (mean of a binary outcome) and its confidence interval using complex survey design features. When stratification and PSU variables are supplied, the function uses Taylor linearized variance estimation via the **survey** package.

Usage

```
lpr_ci(
  data,
  outcome,
  weight = "weight1500",
  strata = NULL,
  psu = NULL,
  conf.level = 0.95,
  na.rm = TRUE
)
```

Arguments

`data`	A data frame containing the outcome and survey design variables.
`outcome`	Character string. Name of a binary variable coded 0/1.
`weight`	Character string. Name of the sampling weight variable. Default is ‘"weight1500"‘.
`strata`	Character string. Name of the stratification variable. Default is ‘NULL‘. If provided together with ‘psu‘, a complex survey design is used.

psu	Character string. Name of the primary sampling unit (cluster) variable. Default is ‘NULL‘.
conf.level	Numeric. Confidence level for the interval. Default is ‘0.95‘.
na.rm	Logical. If ‘TRUE‘, rows with missing values in the required variables are removed prior to estimation.

Details

If both ‘strata‘ and ‘psu‘ are provided, a full complex survey design is declared. If they are ‘NULL‘, the function computes a weighted estimate assuming simple random sampling (SRS) with weights.

Lonely PSUs are handled using ‘survey.lonely.psu = "adjust"‘.

Variance estimation is performed using Taylor linearization as implemented in svymean. When both ‘strata‘ and ‘psu‘ are supplied, clustering and stratification are incorporated in the variance estimator.

If ‘strata‘ and ‘psu‘ are not provided, the function assumes a weighted simple random sample and estimates variance accordingly.

Value

A data frame with:

prop Estimated proportion (mean of binary outcome).

lb Lower bound of the confidence interval.

ub Upper bound of the confidence interval.

se Standard error of the estimate.

Examples

```
# Design-based estimate
data(cm23)
lpr_ci(data = cm23,
         outcome = "b13",
         weight = "weight1500",
         strata = "strata",
         psu = "upm")

# Weighted SRS estimate
data(bra23)
lpr_ci(data = bra23,
         outcome = "b13",
         weight = "wt")
```

lpr_coef *LAPOP Regression Coefficients Graph Pre-Processing*

Description

This function creates a data frame which can then be input in lapop_coef() for plotting regression coefficients graph using LAPOP formatting.

Usage

```
lpr_coef(
  outcome = NULL,
  xvar = NULL,
  interact = NULL,
  model = "linear",
  data = NULL,
  estimate = c("coef"),
  vlabs = NULL,
  omit = NULL,
  filesave = NULL,
  replace = FALSE,
  level = 95
)
```

Arguments

outcome	Dependent variable for the svyglm regression model. (e.g., "outcome_name"). Only one variable allowed.
xvar	Vector of independent variables for the svyglm regression model (e.g., "xvar1+xvar2+xvar3" and so on). Multiple variables are allowed.
interact	Interaction terms (e.g., "xvar1‘*‘xvar2 + xvar3‘:‘xvar4"). Supports ‘:‘ and ‘*‘ operators for interacting variables. Optional, default is NULL.
model	Model family object for glm. Default is gaussian regression (i.e., "linear"). For a logit model, use model="binomial"
data	Survey design data from lpr_data() output.
estimate	Character. Graph either the coefficients (i.e., ‘coef‘) or the change in probabilities (i.e., ‘contrast‘). Default is "coef."
vlabs	Character. Rename variable labels to be displayed in the graph produced by lapop_coef(). For instance, vlabs=c("old_varname" = "new_varname").
omit	Character. Do not display coefficients for these independent variables. Default is to display all variables included in the model. To omit any variables you need to include the raw "varname" in the omit argument.
filesave	Character. Path and file name with csv extension to save the dataframe output.
replace	Logical. Replace the dataset output if it already exists. Default is FALSE.
level	Numeric. Set confidence level in numeric values; default is 95 percent.

Value

Returns a data frame, with data formatted for visualization by lapop_coef

Author(s)

Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(bra23)

# Set Survey Context
bra23lpr <- lpr_data(bra23, wt = TRUE)

# Example 1: Linear model
lpr_coef(data = bra23lpr,
 outcome = "ing4",
 xvar = "wealth+idio2",
 model = "linear",
 estimate = "coef")

# Example 2: Logit model with contrasts
lpr_coef(data = bra23lpr,
 outcome = "fs2",
 xvar = "wealth+idio2",
 model = "binomial",
 estimate = "contrast")

# Example 3: Interactive linear model
lpr_coef(data = bra23lpr,
 outcome = "ing4",
 xvar = "wealth+idio2",
 interact = "wealth*idio2",
 model = "linear",
 estimate = "coef")

# Example 4: Interactive logit model
lpr_coef(data = bra23lpr,
         outcome = "fs2",
         xvar = "wealth+idio2",
         interact = "wealth*idio2",
         model = "binomial",
         estimate = "contrast")
```

---

lpr_data *LAPOP Data Processing*

Description

This function takes LAPOP datasets and adds survey features such as sampling design effects, outputting a svy_tbl object that can then be analyzed using lpr_ wrangling commands.

Usage

```
lpr_data(data_path, wt = FALSE)
```

Arguments

data_path	The path for a AmericasBarometer data or a an existing dataframe.
wt	Logical. If TRUE, use 'wt' (weights only for single-country single-year data) instead of 'weight1500' (the default weights for multiple-country and multiple-year data). Default: FALSE.

Value

Returns a svy_tbl object

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
data(bra23)
data(cm23)

bra23w <- lpr_data(bra23, wt = TRUE)
cm23w <- lpr_data(cm23)
```

---

lpr_dumb *LAPOP Dumbbell Graphs*

Description

This function creates dataframes which can then be input in lapop_dumb for comparing means of a variable across countries and two waves using LAPOP formatting.

Usage

```
lpr_dumb(
  data,
  outcome,
  xvar = "pais",
  over,
  rec = c(1, 1),
  ci_level = 0.95,
  mean = FALSE,
  filesave = "",
  cfmt = "",
  sort = "prop2",
  order = "hi-lo",
  ttest = FALSE,
  keep_nr = FALSE
)
```

Arguments

data	A survey object. The data that should be analyzed.
outcome	Outcome variable(s) of interest to be plotted across countries and waves, supplied as a character string or vector of strings.
xvar	Character. The grouping variable to be plotted along the x-axis (technically, the vertical axis for lapop_dumb). Usually country (pais). Default: "pais".
over	Numeric. A vector of values for "wave" that specify which two waves should be included in the plot.
rec	Numeric. The minimum and maximum values of the outcome variable that should be included in the numerator of the percentage. For example, if the variable is on a 1-7 scale and rec is c(5, 7), the function will show the percentage who chose an answer of 5, 6, 7 out of all valid answers. Can also supply one value only, to produce the percentage that chose that value out of all other values. Default: c(1, 1).
ci_level	Numeric. Confidence interval level for estimates. Default: 0.95
mean	Logical. If TRUE, will produce the mean of the variable rather than recoding to percentage. Default: FALSE.
filesave	Character. Path and file name to save the dataframe as csv.
cfmt	Character. Changes the format of the numbers displayed above the bars. Uses sprintf string formatting syntax. Default is whole numbers for percentages and tenths place for means.
sort	Character. On what value the bars are sorted. Options are "prop1" (for the value of the outcome variable in wave 1), "prop2" (default; for the value of the outcome variable in wave 2), "xv" (for the underlying values of the x variable), "xl" (for the labels of the x variable, i.e., alphabetical), and "diff" (for the difference between the outcome between the two waves).
order	Character. How the bars should be sorted. Options are "hi-lo" (default) or "lo-hi".

ttest	Logical. If TRUE, will conduct pairwise t-tests for difference of means between all pais-wave combinations and save them in attr(x, "t_test_results"). Default: FALSE.
keep_nr	Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99) and use them in the denominator when calculating percentages. The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_dumb()

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(cm23)

# Set Survey Context
cm23lpr <- lpr_data(cm23)

# Single outcome over years
lpr_dumb(cm23lpr,
outcome = "ing4",
rec = c(5, 7),
over = c("2018/19", "2023"),
sort = "diff")

# Multiple outcomes over years
lpr_dumb(cm23lpr,
outcome=c("b13","b21", "b31"),
rec=c(5,7),
over=c("2018/19", "2023"))
```

---

lpr_extract_notes *Extract Notes from AmericasBarometer Attributes*

Description

Extracts notes stored in a dataset's attributes and organizes them into a tidy data frame. This function is particularly useful for processing Stata datasets imported into R that contain variable notes in their attributes.

Usage

```
lpr_extract_notes(data)
```

Arguments

data	A dataset (data frame) containing "expansion.fields" in its attributes.

Details

This function processes the attributes of a dataset to extract notes that are typically stored in a specific format. It skips any notes associated with "_dta" (dataset-level notes) and only returns variable-specific notes. The function expects the notes to be organized as a list where each element contains exactly three components: variable name, note ID, and note value.

Value

A data frame with three columns:

variable_name Name of the variable the note belongs to

note_id Identifier for the note

note_value The actual note text

Examples

```
require(lapop); data(bra23)

# Extract the notes
notesBRA23 <- lpr_extract_notes(bra23)
tail(notesBRA23[notesBRA23$variable_name=="ing4",]) # for ing4 variable
```

---

lpr_extract_ros *Extract Response Option (RO) values and texts for all variables into a tidy table.*

Description

Works with: (a) dataset-level dictionaries (e.g., attr(data, "label.table") is a list keyed by "<VAR>_<lang>"), or (b) per-variable attributes (e.g., attr(data[[VAR]], "levels") or factor levels).

Usage

```
lpr_extract_ros(
  data,
  lang_id = "en",
  include_special = FALSE,
  restrict_to_present = TRUE,
  one_row_per_var = FALSE,
  pair_sep = " | ",
  attr_name = "label.table"
)
```

Arguments

data	A data.frame read with readstata13/haven/etc.
lang_id	Language code used in label table names ("en", "es", "pt"). If ‘NULL‘ or ‘""‘, auto-detect per variable (dataset-level only). Ignored for per-variable ‘levels‘.
include_special	Logical; if FALSE, drop codes >= 1000 when codes are numeric. Default FALSE.
restrict_to_present	Logical; if TRUE, keep only codes that appear in the data. Default TRUE.
one_row_per_var	Logical; if TRUE, return one row per variable_name with concatenated ROs. Default FALSE.
pair_sep	String used to separate each "(value) answer_text" pair when collapsing. Default " | ".
attr_name	Name of the attribute that stores RO info. Default "label.table".

Value

If ‘one_row_per_var = FALSE‘: tibble with columns ‘variable_name‘, ‘value‘, ‘answer_text‘. If ‘one_row_per_var = TRUE‘: tibble with columns ‘variable_name‘, ‘answer_text‘ (collapsed pairs).

Author(s)

Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
toy <- data.frame(
  ing4 = c(1L, 2L, 1L),
  b12 = c(1L, 2L, NA_integer_)
)
attr(toy, "label.table") <- list(
  ing4_pt = c("Apoia muito" = 1L, "Apoia" = 2L, "NS/NR" = 1000L),
  b12_pt = c("Muito" = 1L, "Algo" = 2L, "NS/NR" = 1000L)
)

lpr_extract_ros(toy, lang_id = "pt")
lpr_extract_ros(toy, lang_id = "pt", one_row_per_var = TRUE)
```

---

lpr_hist *LAPOP Bar/Histogram Graphs*

Description

This function creates dataframes which can then be input in lapop_hist for showing a bar graph using LAPOP formatting.

Usage

```
lpr_hist(
  data,
  outcome,
  filesave = "",
  cfmt = "",
  sort = "xv",
  order = "lo-hi",
  keep_nr = FALSE
)
```

Arguments

data	A survey object. The data that should be analyzed.
outcome	Character. Outcome variable of interest.
filesave	Character. Path and file name to save the dataframe as csv.
cfmt	Character. Changes the format of the numbers displayed above the bars. Uses sprintf string formatting syntax. Default is whole numbers.
sort	Character. On what value the bars are sorted. Options are "y" (for the value of the outcome variable), "xv" (default; for the underlying values of the x variable), "xl" (for the labels of the x variable, i.e., alphabetical).
order	Character. How the bars should be sorted. Options are "hi-lo" or "lo-hi" (default).
keep_nr	Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99). The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_hist

Author(s)

Shashwat Dhar <shashwat.dhar@vanderbilt.edu> & Luke Plutowski, <luke.plutowski@vanderbilt.edu>

Examples

```
require(lapop); data(bra23)

# Set Survey Context: single country and year (requires wt = T)
bra23lpr <- lpr_data(bra23, wt = TRUE)

lpr_hist(bra23lpr,
outcome = "ing4",
sort = "xv",
order = "hi-lo")
```

lpr_mline *LAPOP Multi-Line Time Series Graph Pre-Processing*

Description

This function creates a dataframe which can then be input in lapop_mline for to show a time series plot with multiple lines. If one "outcome" variable and an ‘xvar‘ variable is supplied, the function produces the values of a single outcome variable, broken down by a secondary variable, across time. If multiple outcome variables (up to four) are supplied, it will show means/percentages of those variables across time (essentially, it allows you to do lpr_ts for multiple variables).

Usage

```
lpr_mline(
  data,
  outcome,
  rec = c(1, 1),
  rec2 = c(1, 1),
  rec3 = c(1, 1),
  rec4 = c(1, 1),
  xvar,
  use_wave = FALSE,
  use_cat = FALSE,
  ci_level = 0.95,
  mean = FALSE,
  filesave = "",
  cfmt = "",
  ttest = FALSE,
  keep_nr = FALSE
)
```

Arguments

data	A survey object. The data that should be analyzed.
outcome	Character vector. Outcome variable(s) of interest to be plotted across time. If only one value is provided, the graph will show the outcome variable, over time, broken down by a secondary variable (x-var). If more than one value is supplied, the graph will show each outcome variable across time (no secondary variable).
rec, rec2, rec3, rec4	Numeric. The minimum and maximum values of the outcome variable that should be included in the numerator of the percentage. For example, if the variable is on a 1-7 scale and rec is c(5, 7), the function will show the percentage who chose an answer of 5, 6, 7 out of all valid answers. Can also supply one value only, to produce the percentage that chose that value out of all other values. Default: c(1, 1).

xvar	Character. Variable on which to break down the outcome variable. In other words, the line graph will produce multiple lines for each value of xvar (technically, it is the z-variable, not the x variable, which is year/wave). Ignored if multiple outcome variables are supplied.
use_wave	Logical. If TRUE, will use "wave" for the x-axis; otherwise, will use "year". Default: FALSE.
use_cat	Logical. If TRUE, will show the percentages of category values of a single variable; otherwise will show percentages of the range of values from rec(). Default FALSE.
ci_level	Numeric. Confidence interval level for estimates. Default: 0.95
mean	Logical. If TRUE, will produce the mean of the variable rather than rescaling to percentage. Default: FALSE.
filesave	Character. Path and file name to save the dataframe as csv.
cfmt	Character. changes the format of the numbers displayed above the bars. Uses sprintf string formatting syntax. Default is whole numbers for percentages and tenths place for means.
ttest	Logical. If TRUE, will conduct pairwise t-tests for difference of means between all individual x levels and save them in attr(x, "t_test_results"). Default: FALSE.
keep_nr	Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99) and use them in the denominator when calculating percentages. The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_mline

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(ym23)

# Set Survey Context
ym23lpr <- lpr_data(ym23)

# Single Variable by Country and Year
lpr_mline(ym23lpr,
outcome = "ing4",
rec = c(5, 7),
xvar = "pais",
use_wave = TRUE)

# Multiple Variables
lpr_mline(ym23lpr,
outcome = c("b12", "b18"),
```

```
rec = c(5, 7),
rec2 = c(1, 2),
rec3 = c(5, 7),
use_wave = TRUE)

# Binary Single Variable by Category
lpr_mline(ym23lpr,
outcome = "pn4",
use_cat = TRUE,
use_wave = TRUE)

# Recode Categorical Variable (max 4-categories)
lpr_mline(ym23lpr,
outcome = "ing4",
rec = c(5, 7),
use_cat = TRUE,
use_wave = TRUE)
```

`lpr_mover` *LAPOP "Multiple-Over" Breakdown Graphs*

Description

This function creates a dataframe which can then be input in lapop_mover() for comparing means across values of secondary variable(s) using LAPOP formatting.

Usage

```
lpr_mover(
  data,
  outcome,
  grouping_vars,
  rec = list(c(1, 1)),
  rec2 = c(1, 1),
  rec3 = c(1, 1),
  rec4 = c(1, 1),
  ci_level = 0.95,
  mean = FALSE,
  filesave = "",
  cfmt = "",
  ttest = FALSE,
  keep_nr = FALSE
)
```

Arguments

`data`	A survey object. The data that should be analyzed.

outcome	Character. Outcome variable(s) of interest to be plotted across secondary variable(s).
grouping_vars	A character vector specifying one or more grouping variables. For each variable, the function calculates the average of the outcome variable, broken down by the distinct values within the grouping variable(s).
rec	Numeric. The minimum and maximum values of the frst outcome variable that should be included in the numerator of the percentage. For example, if the variable is on a 1-7 scale and rec is c(5, 7), the function will show the percentage who chose an answer of 5, 6, 7 out of all valid answers. Can also supply one value only, to produce the percentage that chose that value out of all other values. Default: c(1, 1).
rec2	Numeric. Similar to 'rec' for the second outcome. Default: c(1, 1).
rec3	Numeric. Similar to 'rec' for the third outcome. Default: c(1, 1).
rec4	Numeric. Similar to 'rec' for the fourth outcome. Default: c(1, 1).
ci_level	Numeric. Confidence interval level for estimates. Default: 0.95
mean	Logical. If TRUE, will produce the mean of the variable rather than recoding to percentage. Default: FALSE.
filesave	Character. Path and file name to save the dataframe as csv.
cfmt	Changes the format of the numbers displayed above the bars. Uses sprintf string formatting syntax. Default is whole numbers for percentages and tenths place for means.
ttest	Logical. If TRUE, will conduct pairwise t-tests for difference of means between all individual year-xvar levels and save them in attr(x, "t_test_results"). Default: FALSE.
keep_nr	Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99) and use them in the denominator when calculating percentages. The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_mover

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> && Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(ym23)

# Set SUrvey Context
ym23lpr<-lpr_data(ym23)

# Single DV
lpr_mover(data = ym23lpr,
```

```
 outcome = "ing4",
 grouping_vars = c("q1tc_r", "edre"),
 rec = c(5, 7), ttest = FALSE)

# Multiple DV
lpr_mover(data = ym23lpr,
outcome = c("ing4", "pn4"),
grouping_vars = c("q1tc_r", "edre"),
rec = c(5, 7), rec2 = c(1, 2),
ttest = FALSE)

# Single DV X Single IV
lpr_mover(data = ym23lpr,
outcome="ing4",
grouping_vars="pn4",
rec=c(5,7),
ttest = FALSE)

# Multiple DV X Single IV
lpr_mover(data = ym23lpr,
outcome=c("ing4", "pn4"),
grouping_vars="edre",
rec=c(5,7), rec2=c(1,2),
ttest = FALSE)

# Multiple DV X Multiple IV
lpr_mover(data = ym23lpr,
outcome=c("ing4", "pn4"),
grouping_vars=c("edre", "q1tc_r"),
rec=c(5,7), rec2=c(1,2),
ttest = FALSE)
```

---

lpr_resc *LAPOP Rescale*

Description

This function allows users to rescale and reorder variables. It is designed for variables of class "labelled" but the rescaling will work for numeric and factor variables too.

Usage

```
lpr_resc(
  var,
  min = 0L,
  max = 1L,
  reverse = FALSE,
  only_reverse = FALSE,
```

```
  only_flip = FALSE,
  map = FALSE,
  new_varlabel = NULL,
  new_vallabels = NULL
)
```

Arguments

var	Vector (class "labelled" or "haven_labelled"). The original variable to rescale.
min	Integer. Minimum value for the new rescaled variables; default is 0.
max	Integer. Maximum value for the new rescaled variables; default is 1.
reverse	Logical. Reverse code the variable before rescaling. Default: FALSE.
only_reverse	Logical. Reverse code the variable, but do not rescale. Default: FALSE.
only_flip	Logical. Flip the variable coding. Unlike "only_reverse", this will exactly preserve the values of the old variable. For example, for a variable with codes 1, 2, 3, 5, 10, only_flip will code the values 10, 5, 3, 2, 1 (instead of 10, 9, 8, 6, 1). Generally, reverse should be preferred to preserve the underlying scale. Not compatible with rescale. Default: FALSE.
map	Logical. If TRUE, will print a cross-tab showing the old variable and the new, recoded variable. Used to verify the new variable is coded correctly. Default: FALSE.
new_varlabel	Character. Variable label for the new variable. Default: old variable's label.
new_vallabels	Character vector. Supply custom names for value labels. Default: value labels of old variable.

Value

The input variable rescaled

Author(s)

Luke Plutowski, <luke.plutowski@vanderbilt.edu> & Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(ym23)

# Regular data.frame
ym23$pn4r <- lpr_resc(ym23$pn4,
reverse = TRUE,
map = TRUE)

# LPR data.frame
ym23lpr<-lpr_data(ym23)
ym23lpr$variables$pn4r <- lpr_resc(ym23lpr$variables$pn4,
reverse = TRUE,
map = TRUE)
```

lpr_set_attr *Set Variable Attributes from AmericasBarometer Notes (with propagation)*

Description

Applies notes stored in a data frame as attributes to variables in ‘data‘. If a variable has expanded children (e.g., vb20_1, vb20_2), the attribute is propagated to all of them by default.

Usage

```
lpr_set_attr(
  data,
  notes,
  noteid = character(),
  attribute_name = character(),
  verbose = FALSE,
  propagate = TRUE,
  overwrite = TRUE
)
```

Arguments

data	data.frame with variables to annotate
notes	data.frame with columns variable_name, note_id, note_value
noteid	character scalar; which note_id to use (e.g., "qtext_en")
attribute_name	character scalar; attribute name to set (e.g., "qwording_en")
verbose	logical; if TRUE it prints all variables notes available but not found in data
propagate	logical; if TRUE, also set on <varname>_* children. Useful for nominal variables or multiple response options variables. Default TRUE.
overwrite	logical; if FALSE, do not overwrite existing attribute on a variable. Default TRUE.

Value

The input data frame with attributes applied (i.e., question wording)

`lpr_set_ros` *Set Response Option (ROS) labels for variables in AmericasBarometer datasets*

Description

This function extracts formatted response option labels for AmericasBarometer variables, using label tables stored as attributes. The labels are formatted with their corresponding numeric codes and can be pulled in multiple languages.

Usage

```
lpr_set_ros(data, lang_id = "en", attribute_name = "roslabel")
```

Arguments

`data`	A data frame loaded using readstata13 containing label.table attributes.
`lang_id`	Language identifier for the labels ("en" for English, "es" for Spanish, "pt" for Portuguese). Default is "en" (English).
`attribute_name`	The name of the attribute where the formatted response options string will be stored. Default is "roslabel".

Details

The function looks for label tables stored as attributes of the data frame, with names following the pattern "VARNAME_lang_id" (e.g., "ing4_en" for English labels of variable ing4). Each label table should be a named numeric vector where names are the response labels and values are the corresponding codes.

Special codes (values >= 1000) are excluded from the response options string.

Value

The input data frame with response option labels added to variables as attributes

Examples

```
require(lapop); data(bra23)

# Apply the function
bra23 <- lpr_set_ros(bra23) # Default English
bra23 <- lpr_set_ros(bra23, lang_id = "es", attribute_name = "respuestas") # Spanish
bra23 <- lpr_set_ros(bra23, lang_id = "pt", attribute_name = "ROsLabels_pt") # Portuguese

# View the resulting attribute
attr(bra23$ing4, "roslabel")
attr(bra23$ing4, "respuestas")
attr(bra23$ing4, "ROsLabels_pt")
```

lpr_stack *LAPOP Stacked Bar Graph Pre-Processing*

Description

This function creates dataframes which can then be input in lapop_stack() for plotting variables categories with a stacked bar graph using LAPOP formatting.

Usage

```
lpr_stack(
  data,
  outcome,
  xvar = NULL,
  sort = "y",
  order = "hi-lo",
  filesave = "",
  keep_nr = FALSE
)
```

Arguments

data	The data that should be analyzed. It requires a survey object from lpr_data() function.
outcome	Vector of variables be plotted.
xvar	Character. Outcome variable will be broken down by this variable. Default is NULL
sort	Character. On what value the bars are sorted: the x or the y. Options are "y" (default; for the value of the outcome variable), "xv" (for the underlying values of the x variable), "xl" (for the labels of the x variable, i.e., alphabetical).
order	Character. How the bars should be sorted. Options are "hi-lo" (default) or "lo-hi".
filesave	Character. Path and file name to save the dataframe as csv.
keep_nr	Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99) and use them in the denominator when calculating percentages. The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_stack

Author(s)

Robert Vidigal, <robert.vidigal@vanderbilt.edu>

Examples

```
require(lapop); data(ym23)

# Set Survey Context
ym23lpr<-lpr_data(ym23)

# Multiple outcomes stacked
lpr_stack(data = ym23lpr,
outcome = c("b12", "b18"))

# Single outcome over years
lpr_stack(data = ym23lpr,
outcome = "q14f",
xvar="year")
```

---

lpr_ts *LAPOP Time-Series Line Graph Pre-Processing*

Description

This function creates dataframes which can then be input in lapop_ts for comparing values across time with a line graph using LAPOP formatting.

Usage

```
lpr_ts(
  data,
  outcome,
  rec = c(1, 1),
  use_wave = FALSE,
  ci_level = 0.95,
  mean = FALSE,
  filesave = "",
  cfmt = "",
  ttest = FALSE,
  keep_nr = FALSE
)
```

Arguments

data	A survey object. The data that should be analyzed.
outcome	Character. Outcome variable of interest to be plotted across time.
rec	Numeric. The minimum and maximum values of the outcome variable that should be included in the numerator of the percentage. For example, if the

variable is on a 1-7 scale and rec is c(5, 7), the function will show the percentage who chose an answer of 5, 6, 7 out of all valid answers. Can also supply one value only, to produce the percentage that chose that value out of all other values. Default: c(1, 1).

`use_wave` Logical. If TRUE, will use "wave" for the x-axis; otherwise, will use "year". Default: FALSE.

`ci_level` Numeric. Confidence interval level for estimates. Default: 0.95

`mean` Logical. If TRUE, will produce the mean of the variable rather than rescaling to percentage. Default: FALSE.

`filesave` Character. Path and file name to save the dataframe as csv.

`cfmt` Character. changes the format of the numbers displayed above the bars. Uses sprintf string formatting syntax. Default is whole numbers for percentages and tenths place for means.

`ttest` Logical. If TRUE, will conduct pairwise t-tests for difference of means between all individual x levels and save them in attr(x, "t_test_results"). Default: FALSE.

`keep_nr` Logical. If TRUE, will convert "don't know" (missing code .a) and "no response" (missing code .b) into valid data (value = 99) and use them in the denominator when calculating percentages. The default is to examine valid responses only. Default: FALSE.

Value

Returns a data frame, with data formatted for visualization by lapop_ts()

Author(s)

Berta Diaz, <berta.diaz.martinez@vanderbilt.edu> & Luke Plutowski, <luke.plutowski@vanderbilt.edu>

Examples

```
require(lapop); data(ym23)

# Set Survey Context
ym23lpr<-lpr_data(ym23)

# Run lpr_ts
lpr_ts(ym23lpr,
outcome = "ing4",
use_wave = TRUE,
mean = TRUE,
ttest = TRUE)
```

world *bra23: Single-country Single-year Dataset*

Description

A dataset containing the World Map geometry for lapop_map() function.

Usage

```
world
```

Format

A data frame

pais Country name

pais_lab Country ISO2 code

geometry Polygon with geometry

Source

LAPOP AmericasBarometer (https://www.vanderbilt.edu/lapop/)

ym23 *ym23: Multi-country Single-year Dataset*

Description

A dataset containing the AmericasBarometer Year Merge of 2023.

Usage

```
ym23
```

Format

A data frame

b12 Trust in Armed Forces

b18 Trust in National Police

ing4 Support for Democracy

pn4 Satisfaction with Democracy

vb21n Influence Political Change

q14f Migration Intentions

edre Education

wealth Wealth

q1tc_r Gender

upm Primary Sampling Unit

strata Stratification

wave Survey round year for regional or multi-country data

pais Country of survey

year Survey round year for single-country data

weight1500 Cross-country and cross-time weight

Source

LAPOP AmericasBarometer (https://www.vanderbilt.edu/lapop/) # Ensure this line has a valid source.

Index