Package 'blocs'

April 2, 2023

Type Package

Title Estimate and Visualize Voting Blocs' Partisan Contributions

Version 0.1.1

Maintainer Cole Tanigawa-Lau <coletl@stanford.edu>

Description Functions to combine data on voting blocs' size, turnout, and vote choice to estimate each bloc's vote contributions to the Democratic and Republican parties. The package also includes functions for uncertainty estimation and plotting. Users may define voting blocs along a discrete or continuous variable. The package implements methods described in Grimmer, Marble, and Tanigawa-Lau (2023) <doi:10.31235/osf.io/c9fkg>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

Suggests devtools (>= 2.4.3), questionr (>= 0.7.7), reldist (>= 1.7.0), testthat (>= 3.1.3)

Config/testthat/edition 3

RoxygenNote 7.2.0

Imports collapse (>= 1.7.6), dplyr (>= 1.0.6), ggplot2 (>= 3.2.0), ks (>= 1.13.4), mgcv (>= 1.8.39), rlang (>= 1.0.0), tibble (>= 3.0.0)

Depends R (>= 3.6.0)

NeedsCompilation no

Author Justin Grimmer [aut], Will Marble [aut] (<https://orcid.org/0000-0001-9352-5540>), Cole Tanigawa-Lau [aut, cre] (<https://orcid.org/0000-0002-8656-571X>)

Repository CRAN

Date/Publication 2023-04-02 21:00:02 UTC

R topics documented:

`anes` *Sample of 2020 ANES cumulative data file*

Description

Selected columns from the American National Election Studies' 2020 cumulative data file. The final column is an example of the three-valued variable for voting behavior, to be passed to the 'dv_vote3' argument,

Usage

```
anes
```

Format

A data frame with 68,224 rows and 13 columns:

year election year

respid respondent identifier

weight survey weight

race respondent race

gender respondent gender

educ respondent education level

age respondent age

voted respondent's voter turnout

vote_pres respondent's presidential vote

vote_pres_dem flag indicating Democratic presidential vote choice

vote_pres_rep flag indicating Republican presidential vote choice

vote_pres3 Three-valued voting behavior DV coded as follows: -1 for Democrat vote choice, 0 for third-party vote, 1 for Republican vote choice, and NA for no vote.

Source

https://electionstudies.org/data-center/anes-time-series-cumulative-data-file/

---

check_vbdf *Validator for class vbdf*

Description

Validator for class vbdf

Usage

```
check_vbdf(x, tolerance = sqrt(.Machine$double.eps))
```

Arguments

x	object to check
tolerance	tolerance used when checking range of probability estimates

---

estimate_density *Estimate density*

Description

Run kde for weighted density estimation of a x at n_points evenly spaced points between min and max.

Usage

```
estimate_density(x, min, max, n_points = 100, w = NULL, ...)
```

Arguments

x	numeric vector or matrix
min	numeric vector giving the lower bound of evaluation points for each variable in x
max	numeric vector giving the upper bound of evaluation points for each variable in x
n_points	number of evaluation points (estimates)
w	vector of weights. Default uses uniform weighting.
...	further arguments to pass to kde

new_vbdf *Constructor for class vbdf*

Description

Constructor for class vbdf

Usage

```
new_vbdf(x, bloc_var = character(), var_type = c("discrete", "continuous"))
```

Arguments

x	a data.frame
bloc_var	character vector naming the variables to define voting blocs
var_type	string, the type, discrete or continuous

new_vbsum *Constructor for vbdf summaries*

Description

Constructor for vbdf summaries

Usage

```
new_vbsum(x, bloc_var, var_type, summary_type, resamples)
```

Arguments

x	data.frame of uncertainty summary
bloc_var	string, the name of the variable that defines the voting blocs
var_type	string, the type of variable, discrete or continuous
summary_type	string, the type of variable, discrete or continuous
resamples	numeric, the number of bootstrap resamples

Value

A vbsum object

vbdf *Create a vbdf object*

Description

Create a vbdf object holding bloc-level estimates of composition, turnout, and/or vote choice. This function is mostly for internal use, but you may want it to create a vbdf object from your own voting bloc analysis. A valid vbdf object can be used in [vb_difference] and [vb_plot].

Usage

```
vbdf(
  data,
  bloc_var,
  var_type = c("discrete", "continuous"),
  tolerance = sqrt(.Machine$double.eps)
)
```

Arguments

data	data.frame of voting-bloc results to convert to a vbdf object
bloc_var	string, the name of the variable that defines the voting blocs
var_type	string, the type of variable, discrete or continuous
tolerance	tolerance used when checking range of probability estimates

Value

A vbdf object.

vb_continuous *Continuous voting bloc analysis*

Description

Define voting blocs along a **continuous** variable and estimate their partisan vote contributions.

Usage

```
vb_continuous(
  data,
  data_density = data,
  data_turnout = data,
  data_vote = data,
  indep,
  dv_vote3,
```

```
  dv_turnout,
  weight = NULL,
  min_val = NULL,
  max_val = NULL,
  n_points = 100,
  boot_iters = FALSE,
  verbose = FALSE,
  tolerance = sqrt(.Machine$double.eps),
  ...
)
```

Arguments

data	default data.frame to use as the source for density, turnout, and vote choice data.
data_density	data.frame of blocs' composition/density data. Must include any columns named by indep and weight.
data_turnout	data.frame of blocs' turnout data. Must include any columns named by dv_turnout, indep and weight.
data_vote	data.frame of blocs' vote choice data. Must include any columns named by dv_vote3, indep, and weight.
indep	string, column name of the independent variable defining discrete voting blocs.
dv_vote3	string, column name of the dependent variable in data_vote, coded as follows: -1 for Democrat vote choice, 0 for third-party vote, 1 for Republican vote choice, and NA for no vote.
dv_turnout	string, column name of the dependent variable flagging voter turnout in data_turnout. That column must be coded 0 = no vote, 1 = voted.
weight	optional string naming the column of sample weights.
min_val	numeric vector of the same length as indep, Lower bound for the density estimation of each respective indep. See [estimate_density].
max_val	numeric vector of the same length as indep, Upper bound for the density estimation of each respective indep. See [estimate_density].
n_points	scalar, number of points at which to estimate density. See [estimate_density].
boot_iters	integer, number of bootstrap iterations for uncertainty estimation. The default FALSE is equivalent to 0 and does not estimate uncertainty.
verbose	logical, whether to print iteration number.
tolerance	tolerance used when checking range of probability estimates
...	further arguments to pass to kde for density estimation.

Value

a vbdf data.frame with columns for the resample, bloc variable, and, for each resample-bloc combination, four estimates: probability density, turnout, Republican vote choice conditional on turnout, and net Republican votes.

vb_difference *Calculate differences in bloc contributions*

Description

Use vbdf output to calculate differences in blocs' net Republican vote contributions.

Usage

```
vb_difference(
  vbdf,
  estimates = grep("prob|pr_turnout|pr_votedem|pr_voterep|cond_rep|net_rep",
    names(vbdf), value = TRUE),
  sort_col = "year",
  tolerance = sqrt(.Machine$double.eps)
)
```

Arguments

vbdf	data.frame holding the results of voting bloc analyses.
estimates	character vector naming the column(s) in vbdf with which to compute differences.
sort_col	character vector naming the column(s) in vbdf to use for sorting before calling diff.
tolerance	tolerance used when checking range of probability estimates

Value

A vbdf object, plus two types of columns: for each column named in estimates, a column named diff_* containing the difference in each estimate across sort_col values, comp, which contains a string tag for the rows compared (e.g., 2020-2016),

A vbdf object.

vb_discrete *Discrete voting bloc analysis*

Description

Define voting blocs along a **discrete** variable and estimate their partisan vote contributions.

Usage

```
vb_discrete(
  data,
  data_density = data,
  data_turnout = data,
  data_vote = data,
  indep,
  dv_vote3,
  dv_turnout,
  weight = NULL,
  boot_iters = FALSE,
  verbose = FALSE,
  check_discrete = TRUE
)
```

Arguments

`data`	default data.frame to use as the source for density, turnout, and vote choice data.
`data_density`	data.frame of blocs' composition/density data. Must include any columns named by `indep` and `weight`.
`data_turnout`	data.frame of blocs' turnout data. Must include any columns named by `dv_turnout`, `indep` and `weight`.
`data_vote`	data.frame of blocs' vote choice data. Must include any columns named by `dv_vote3`, `indep`, and `weight`.
`indep`	string, column name of the independent variable defining discrete voting blocs.
`dv_vote3`	string, column name of the dependent variable in `data_vote`, coded as follows: -1 for Democrat vote choice, 0 for third-party vote, 1 for Republican vote choice, and NA for no vote.
`dv_turnout`	string, column name of the dependent variable flagging voter turnout in `data_turnout`. That column must be coded 0 = no vote, 1 = voted.
`weight`	optional string naming the column of sample weights.
`boot_iters`	integer, number of bootstrap iterations for uncertainty estimation. The default `FALSE` is equivalent to 0 and does not estimate uncertainty.
`verbose`	logical, whether to print iteration number.
`check_discrete`	logical, whether to check if `indep` is a discrete variable.

Value

A `vbdf` object.

vb_plot *Plot the summary of a voting bloc analysis*

Description

Plot the summary of a voting bloc analysis

Usage

```
vb_plot(
  data,
  x_col = get_bloc_var(data),
  y_col,
  ymin_col,
  ymax_col,
  discrete = length(unique(data[[x_col]])) < 20
)
```

Arguments

data	a vbsum data.frame, the result of [vb_summary].
x_col	string naming the column that defines voting blocs.
y_col	string naming the column of point estimates.
ymin_col	string naming the column to plot as the lower bound of the confidence interval.
ymax_col	string naming the column to plot as the upper bound of the confidence interval.
discrete	logical indicating whether voting blocs are defined along a discrete (not continuous) variable.

Value

a ggplot object

vb_summary *Summarize uncertainty for a vbdf objects*

Description

Summarize uncertainty for a vbdf objects. Analysis must have run with bootstrap iterations. vb_uncertainty is just an alias for vb_summary.

Usage

```
vb_summary(
  object,
  type = c("discrete", "continuous", "binned"),
  estimates = grep("prob|pr_turnout|pr_votedem|pr_voterep|cond_rep|net_rep",
    names(object), value = TRUE),
  na.rm = FALSE,
  funcs = c("mean", "median", "low", "high"),
  low_ci = 0.025,
  high_ci = 0.975,
  bin_col,
  tolerance = sqrt(.Machine$double.eps)
)

vb_uncertainty(
  object,
  type = c("discrete", "continuous", "binned"),
  estimates = grep("prob|pr_turnout|pr_votedem|pr_voterep|cond_rep|net_rep",
    names(object), value = TRUE),
  na.rm = FALSE,
  funcs = c("mean", "median", "low", "high"),
  low_ci = 0.025,
  high_ci = 0.975,
  bin_col,
  tolerance = sqrt(.Machine$double.eps)
)
```

Arguments

object	a vbdf object, usually the output of [vb_discrete], [vb_continuous], or [vb_difference].
type	a string naming the type of independent variable summary. Use "binned" when using the output of [vb_continuous] plus a binned version of the continuous bloc variable.
estimates	character vector naming columns for which to calculate uncertainty estimates.
na.rm	logical indicating whether to remove NA values in estimates.
funcs	character vector of summary functions to apply to estimates. Alternatively, supply your own list of functions, which should accept a numeric vector input and return a scalar.
low_ci	numeric. If you include the string "low" in funcs, then use this argument to control the lower bound of the confidence interval.
high_ci	numeric. If you include the string "high" in funcs, then use this argument to control the upper bound of the confidence interval.
bin_col	character vector naming the column(s) that define the bins. Used only when type is "binned".
tolerance	tolerance used when checking range of probability estimates

Value

A summary object with additional columns for each combination of `estimates` and `funcs`.

`wtd_table` *Weighted frequency table or proportions*

Description

Weighted frequency table or proportions

Usage

```
wtd_table(
  ...,
  weight = NULL,
  na.rm = FALSE,
  prop = FALSE,
  return_tibble = FALSE,
  normwt = FALSE
)
```

Arguments

`...`	vectors of class factor or character, or a list/data.frame of such vectors.
`weight`	optional vector of weights. The default uses uniform weights of 1.
`na.rm`	logical, whether to remove NA values.
`prop`	logical, whether to return proportions or counts. Default returns counts.
`return_tibble`	logical, whether to return a tibble or named vector.
`normwt`	logical, whether to normalize weights such that they sum to 1.

Value

a vector or tibble of counts or proportions by group

Index