# Package 'voteR'

July 2, 2018

**Title** Variety of Open Tools for Electoral Research

**Version** 0.0.0.9000

**Description**
This package contains a variety of tools and datasets that can be used for electoral research.

**Depends** R (>= 3.4.1),
ggplot2,
tidyverse,
conflicted

**License** MIT

**Encoding** UTF-8

**URL** schliebs.github.io/voteR

**BugReports** m.schliebs@zeppelin-university.net

**LazyData** true

**Imports** dplyr,
ggplot2,
gtools,
hrbrthemes,
magrittr,
stringr,
tidyr

**RoxygenNote** 6.0.1

**Collate** 'constituencymaps.R'
'gles.R'
'importFrom.R'
'make_dictionary.R'
'partycolors.R'
'partynames.R'
'polling.R'
'plotting.R'
'scrape_wahlrecht.R'
'wahldata.R'

**Suggests** knitr,
rmarkdown

**VignetteBuilder** knitr

# R topics documented:



---

`bundestag_laenderebene` *lala*

---

## Description

Bundestag election results for the regional laender level 1949-2017

## Usage

```
bundestag_laenderebene
```

## Format

A data frame with 236 rows and 13 variables:

**year** election year

**land** regional level

**date** election date

**...** party vote share vector

**wbt** voter turnout

**others** other parties

**level** Level of Election (federal or regional)

## distance_function *Calculate Distances*

### Description

Calculate Distances from different parties/koalitions

### Usage

```
distance_function(data_in = "gles2017_out", who = "schwarzgelb",
  issue = "soz")
```

### Value

The vector including all coalitions.

### Examples

```
distance_function(data_in = "gles2017_out",
                  who = "schwarzgelb",
                  issue = "soz")
```

## gles_recode_partyvar *gles_recode_partyvar*

### Description

Recode a multiparty-variable in GESIS-Datasets such as the GERMAN LONGITUDINAL ELECTION STUDY (GLES)

### Usage

```
gles_recode_partyvar(year = 2017, dataset_input = "gles2017",
  dataset_output = "gles2017_out", varname = "q52", own = NULL,
  varlabel = "soz", key = c("a", "b", "c", "d", "e", "f", "g"),
  partynames = c("cdu", "csu", "spd", "linke", "gruene", "fdp", "afd"),
  NAs = "<0", plot = TRUE)
```

### Arguments

| | |
|---|---|
| year | year the GLES-study is from. Defaults to 2017. |
| dataset_input | Character string of the name of a dataframe containing the raw data. |
| dataset_output | Character string of the name the output data frame (may already exist or not). |
| varname | Character string of the name of the original variable. |
| own | May apply: Different variable name for own position (on left-right scales, e.g.) |
| varlabel | Character string of the to-be-assigned variable label. |
| key | Character vector containing original alphabetic party keys. |
| partynames | Character vector containing shortname party keys. |
| NAs | Numeric vector containing to-be-assigned NAs/Missing values. |
| plot | Logical T/F: Show relative frequency barplots while plotting. |

## Value

A data frame containing output dataframe including newly appended new-variables.

## Examples

```
gles_recode_partyvar(year = 2017,
                     dataset_input = "gles2017",
                     dataset_output = "gles2017_out",
                     varname = "q52",
                     varlabel = "soz",
                     key = c("a","b","c","d","e","f","g"),
                     partynames = c("cdu","csu","spd","linke","gruene","fdp","afd"),
                     NAs = c(-97,-98,-99))
```

---

# intrakoadistanz *Calculate Intra-Coalition Heterogeneity*

---

## Description

Calculate Intra-Coalition Heterogeneity from different parties/koalitions

## Usage

```
intrakoadistanz(who = "schwarzgelb", issue = "lr", input = "gles2017_out",
  year = 2017)
```

## Value

The vector including all coalitions.

## Examples

```
distance_function(input = "gles2017_out",
                  who = "schwarzgelb",
                  issue = "soz")
```

---

# koas *Get coalitions*

---

## Description

Get all available coalitions

## Usage

```
koas(year = 2017)
```

## Value

The vector including all coalitions.

## Examples

```
koas(year = 2017)
```

## koa_members *Get coalition members*

### Description

Get parties that are member of a certrain coalition

### Usage

```
koa_members(koalition)
```

### Arguments

| | |
|---|---|
| coalition | Character string containing the name of the coalition.<br>Options are c("jamaika","schwarzgelb","rotgruen","groko","rotrotgruen","ampel","schwarzgruen"). |

### Value

A vector containing all parties included in the coalition.

### Examples

```
koa_members("schwarzgelb")
```

## koa_positions *Get coalition members*

### Description

Calculate mean koalition issue position and create new variables

### Usage

```
koa_positions(data_in = "gles2017_out", coalition = "schwarzgelb",
  issue = "soz")
```

### Arguments

| | |
|---|---|
| data_in | Character string containing the name the dataset. |
| coalition | Character string containing the name of the coalition. |
| issue | Character string containing the issue. |

### Value

The treated dataset.

### Examples

```
koa_members(data_in = "gles2017_out",
            coalition = "schwarzgelb",
            issue = "soz")
```

## laenderbip *German regional GDP data*

### Description

German regional GDP data

### Usage

```
laenderbip
```

### Format

A data frame with 27 rows and 18 variables:

**year** regional level

**...** name of the bundesland

**d** Germany total sum

## landesregierungen *German regional government cabinets for the whole post-war period*

### Description

German regional government cabinets for the whole post-war period

### Usage

```
landesregierungen
```

### Format

A data frame with 325 rows and 11 variables:

**land** regional level

**cabinet** name of the respective cabinet

**years** years cabinet was in office

**dates** specific dates cabinet was in office

**parties** parties forming the coalition

**primeminister** name of the government leader

**party_x** name of the forming parties

**start** year in which gov't started

**end** year in which it ended

## landtagswahlen — *German "Landtagswahlen" election results for the regional laender level 1946-2018*

### Description

German "Landtagswahlen" election results for the regional laender level 1946-2018

### Usage

```
landtagswahlen
```

### Format

A data frame with 233 rows and 13 variables:

**year** election year

**land** regional level

**date** election date

**...** party vote share vector

**wbt** voter turnout

**others** other parties

**level** Level of Election (federal or regional)

## make_dictionary — *Title*

### Description

Title

### Usage

```
make_dictionary(dataset, format = c("wide", "long"))
```

### Arguments

| | |
|---|---|
| dataset | SPSS dataframe in tbl_df format imported with haven. |
| format | specify whether you want a wide (each variable one row) or wide format (each value one row). |

### Value

a data.fram with meta information which can by querried with R-Studio functionalities.

## parties *Get parties*

### Description

Get main parties for Gles analysis

### Usage

```
parties(year = 2017)
```

### Value

The vector including all parties

### Examples

```
parties(year = 2017)
```

## plot_poll *Plot a multiparty poll*

### Description

...

### Usage

```
plot_poll(vote = c(cdu = 0.33, spd = 0.2, fdp = 0.11, linke = 0.09, gruene =
  0.09, afd = 0.12, sonstige = 0.05), order = "alphabetical",
  sample_confidence_bounds = TRUE, sample_n = 1000, n_draw = 10000,
  show_quantiles = c(0.05, 0.95), round = 1, xlab = "Party",
  ylab = "Voteshare", title = "Title", subtitle = "Subtitle",
  caption = "Caption", theme_ipsum = FALSE, grid = "Y")
```

### Arguments

| | |
|---|---|
| vote | A labeled party vote share vector. |
| order | Method to order parties (Default is "alphabetical"; also takes "descending" and "ascending" as well as manual specification of party vector) |
| sample_confidence_bounds | Logical T/F: add empirical dirichlet quantiles |
| sample_n | The number of observations in the poll sample. |
| n_draw | How many samples to draw from the dirichlet distribution. |
| show_quantiles | Vector of quantiles/confidence boundaries to calculate. |
| round | Round to k decimals after comma |
| xlab | x-label string |

| | |
|---|---|
| ylab | y-label string |
| title | title string |
| subtitle | subtitle string |
| caption | caption string |
| theme_ipsum | Pre-applies nice theme from the hrbrthemes-package.(Attention: possible font-issues when Roboto font is not installed on your computer.) |
| grid | (Applies only if theme_ipsum == T) Add a grid (options: "none","Y") |

## Value

A data frame containing n rows of samples for each party.

## Warning

Do not operate heavy machinery within 8 hours of using this function.

## Examples

```
sample_dirichlet_quantiles(vote = c(cdu = 0.33,....
```

---

# sample_dirichlet *Dirichlet-sample of a multinomial election poll*

---

## Description

Calculate a dirichlet-sample of a multinomial election poll

## Usage

```
sample_dirichlet(vote = c(cdu = 0.5, spd = 0.4, fdp = 0.1), sample_n = 1000,
  n_draw = 10000)
```

## Arguments

| | |
|---|---|
| vote | A labeled party vote share vector. |
| sample_n | The number of observations in the poll sample. |
| n_draw | How many samples to draw from the dirichlet distribution. |

## Value

A data frame containing n rows of samples for each party.

## Warning

Do not operate heavy machinery within 8 hours of using this function.

## Examples

```
sample_dirichlet(vote = c(cdu = 0.33,
                          spd = 0.20,
                          fdp = 0.11,
                          linke = 0.09,
                          gruene = 0.09,
                          afd = 0.12,
                          sonstige = 0.05),
                 sample_n = 1000,
                 n_draw = 10000)
```

---

`sample_dirichlet_quantiles`
*Empirical Dirichlet Quantiles from Multinomial Election Poll*

---

## Description

Calculate empirical quantiles from a sample created by `sample_dirichlet` of a multinomial election poll

## Usage

```
sample_dirichlet_quantiles(vote = c(cdu = 0.5, spd = 0.4, fdp = 0.1),
  sample_n = 1000, n_draw = 10000, show_mean = TRUE,
  show_quantiles = c(0.05, 0.95), round = 2)
```

## Arguments

| | |
|---|---|
| `vote` | A labeled party vote share vector. |
| `sample_n` | The number of observations in the poll sample. |
| `n_draw` | How many samples to draw from the dirichlet distribution. |
| `show_mean` | Logical T/F: Show sample mean. |
| `show_quantiles` | Vector of quantiles/confidence boundaries to calculate. |
| `round` | Logical T/F Round Results to k decimal digits. |

## Value

A data frame containing `n` rows of samples for each party.

## Warning

Do not operate heavy machinery within 8 hours of using this function.

## Examples

```
sample_dirichlet_quantiles(vote = c(cdu = 0.33,
                           spd = 0.20,
                           fdp = 0.11,
                           linke = 0.09,
                           gruene = 0.09,
                           afd = 0.12,
                           sonstige = 0.05),
                    sample_n = 1000,
                    n_draw = 10000,
                    show_mean = TRUE,
                    show_quantiles = c(0.05,0.95))
```

---

`structural_modeldata` *Input Data for structural model*

---

## Description

Input Data for structural model

## Usage

```
structural_modeldata
```

## Format

A data frame with 1864 rows and 25 variables:

**year** election year

**land** regional level

**date** election date

**wbt** voter turnout

**party** party name

**vote** vote share

**partytype** classification in spd/union/small and others

**lag_ltw** one-period lag of landtagswahl result

**lag_btw** one-period lag of bundestagswahl result (of that party in that state)

**date_btw** date of the last bundestagswahl

**cabinet** name of the cabinet

**primeminister_name** name of the incumbent prime minister

**party_x** 1st to 3rd party listing coalition members (1st party is party of PM)

**start** year that the incumbent government was formed

**end** year it was replaced

**primeminister** Logical T/F if party is holding incumbency of PM

**gov** Logical T/F if party is incumbent coalition member

**juniorpartner** T/F if party is junior partner (gov but not PM)

**bip** absolute bip in billion €

**bipchange** change in bip in the last 2 years prior to the election

**firsttime** Logical T/F if party is firsttime contender

**distance_btw_lag** distance to last Bundestagswahl in days

**others** other parties

# Index