

# Package ‘voteogram’

July 22, 2025

**Type** Package

**Title** United States House and Senate Voting Cartogram Generators

**Version** 0.3.2

**Date** 2023-03-08

**Maintainer** Bob Rudis <bob@rud.is>

**Description** 'ProPublica' <<https://projects.propublica.org/represent/>> makes United States Congress member votes available and has developed their own unique cartogram to visually represent this data. Tools are provided to retrieve voting data, prepare voting data for plotting with 'ggplot2', create vote cartograms and theme them.

**URL** <https://github.com/hrbrmstr/voteogram>

**BugReports** <https://github.com/hrbrmstr/voteogram/issues>

**Encoding** UTF-8

**License** AGPL

**Suggests** testthat, covr, knitr, rmarkdown

**Depends** R (>= 3.2.0)

**Imports** ggplot2 (>= 3.4.1), dplyr, scales, jsonlite, tibble, utils

**RoxygenNote** 7.2.1

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Bob Rudis [aut, cre] (ORCID: <<https://orcid.org/0000-0001-5670-2640>>),  
ProPublica [cph] (Cartogram SVGs used to make the plot data.frames),  
GovTrack [cph] (Cartogram SVG used to make the plot data.frame)

**Repository** CRAN

**Date/Publication** 2023-03-08 12:20:02 UTC

## Contents

fortify.pprc	2
house_carto	2

print.pprc . . . . .	3
roll_call . . . . .	4
senate_carto . . . . .	5
theme_voteogram . . . . .	6
voteogram . . . . .	6
voteogram-colors . . . . .	7
voteogram-exports . . . . .	7

<b>Index</b>	<b>8</b>
--------------	----------

---

fortify.pprc	<i>Fortify a <code>roll_call()</code> (pprc) object</i>
--------------	---

---

### Description

Fortify a `roll_call()` (pprc) object

### Usage

```
## S3 method for class 'pprc'
fortify(model, data, ...)
```

### Arguments

model	a <code>roll_call()</code> (pprc) object
data	unused
...	unused

---

house_carto	<i>Produce a ProPublica- or GovTrack-style House roll call vote cartogram</i>
-------------	---

---

### Description

Produce a ProPublica- or GovTrack-style House roll call vote cartogram

### Usage

```
house_carto(
  vote_tally,
  style = c("pp", "gt", "propublica", "govtrack"),
  pp_square = FALSE
)
```

**Arguments**

vote_tally	either a pprc object (the result of a call to <code>roll_call()</code> ) or a data.frame of vote tallies for the house. It expects 3 columns. <code>state_abbrev</code> : the 2-letter U.S. state abbreviation; <code>district</code> : either 1 or 2 to distinguish between each representative; <code>party</code> : R, D or ID; <code>position</code> : yes, no, present, none for how the representative voted.
style	either ProPublica-ish (pp or propublica) or GovTrack-ish (gt or govtrack)
pp_square	if TRUE then no "state borders" will be drawn, but distinct Representative squares. If FALSE then the cartogram will be very close to the ProPublica cartograms.

**Value**

a ggplot2 object that you can further customize with scales, labels, etc.

**Note**

No "theming" is applied to the returned ggplot2 object. You can use `theme_voteogram()` if you need a base theme. Also, GovTrack-style cartograms will have `coord_equal()` applied by default.

**Examples**

```
## Not run:
# what you'd normally do
rep <- roll_call("house", 115, 1, 256)

## End(Not run)

# using a saved object
rep <- readRDS(system.file("extdata", "rep.rds", package = "voteogram"))

house_carto(rep, pp_square = TRUE)
```

---

print.pprc

*Better default print function for `roll_call()` (pprc) objects*


---

**Description**

Better default print function for `roll_call()` (pprc) objects

**Usage**

```
## S3 method for class 'pprc'
print(x, ...)
```

**Arguments**

x	a <code>roll_call()</code> (pprc) object
...	ignored

---

roll_call	<i>Get Voting Record for House or Senate By Number, Session &amp; Roll Call Number</i>
-----------	--

---

### Description

Get Voting Record for House or Senate By Number, Session & Roll Call Number

### Usage

```
roll_call(critter = c("house", "senate"), number, session = c(1L, 2L), rcall)
```

### Arguments

critter	one of 'house' or 'senate'
number	valid congress number. ProPublica seems to have data going back to the 101st Congress, so valid values are '101'-present Congress number ('115' as of the creation date of the package).
session	a valid session numbner (i.e. '1' or '2' and valid for current year)
rcall	roll call vote number

### Value

a 'list', one component of which is a 'votes' 'data.frame'

### Note

Try to cache this data if at all possible. ProPublica is a non-profit organization and this data comes from their Amazon S3 buckets. Every access in a given month ticks down the "free" counter.

### Examples

```
## Not run:  
# these make API calls so they aren't run in the examples  
rep <- roll_call("house", 115, 1, 256)  
sen <- roll_call("senate", 115, 1, 110)  
  
## End(Not run)
```

---

senate_carto	<i>Produce a Senate cartogram</i>
--------------	-----------------------------------

---

## Description

Produce a Senate cartogram

## Usage

```
senate_carto(vote_tally)
```

## Arguments

`vote_tally` either a pprc object (the result of a call to `roll_call()`) or a data.frame of vote tallies for the senate. It expects 3 columns. `state_abbrev` : the 2-letter U.S. state abbreviation; `district` : either 1 or 2 to distinguish between each senator; `party` : R, D or ID; `position` : yes, no, present, none for how the senator voted.

## Value

a ggplot2 object that you can further customize with scales, labels, etc.

## Note

No "theming" is applied to the returned ggplot2 object. You can use `theme_voteogram()` if you need a base theme.

## Examples

```
## Not run:  
# what you'd normally do  
sen <- roll_call("senate", 115, 1, 110)  
  
## End(Not run)  
  
# Using a saved object  
sen <- readRDS(system.file("extdata", "sen.rds", package = "voteogram"))  
  
senate_carto(sen)
```

---

theme_voteogram	<i>voteogram ggplot2 theme</i>
-----------------	--------------------------------

---

### Description

Provides a very basic theme with no background, grid, axis text or axis ticks and an easy way to turn the legend on or off.

### Usage

```
theme_voteogram(legend = TRUE)
```

### Arguments

legend            if 'FALSE' no legend is shown

### Examples

```
## Not run:  
# what you'd normally do  
sen <- roll_call("senate", 115, 1, 110)  
  
## End(Not run)  
  
# using a saved object  
sen <- readRDS(system.file("extdata", "sen.rds", package="voteogram"))  
  
senate_carto(sen) +  
  theme_voteogram()
```

---

voteogram	<i>U.S. House and Senate Voting Cartogram Generators</i>
-----------	--

---

### Description

ProPublica' <<https://projects.propublica.org/represent/>> makes United States Congress member votes available and has developed their own unique cartogram to visually represent this data. Tools are provided to retrieve voting data, prepare voting data for plotting with 'ggplot2', create vote cartograms and theme them.

### Author(s)

Bob Rudis (bob@rud.is)

---

voteogram-colors      *voteogram manual scale colors*

---

**Description**

Manual color scales fill and color scale values (in the event you need to use them) outside the main plotting functions.

**Usage**

vote\_carto\_fill

vote\_carto\_color

**Format**

An object of class character of length 13.

An object of class character of length 13.

---

voteogram-exports      *voteogram exported operators*

---

**Description**

The following functions are imported and then re-exported from the voteogram package to enable use of the magrittr pipe operator with no additional library calls

# Index

## \* datasets

- voteogram-colors, [7](#)
- %>%(voteogram-exports), [7](#)
  
- fortify.pprc, [2](#)
  
- house\_carto, [2](#)
  
- print.pprc, [3](#)
  
- roll\_call, [4](#)
- roll\_call(), [2](#), [3](#), [5](#)
  
- senate\_carto, [5](#)
  
- theme\_voteogram, [6](#)
- theme\_voteogram(), [3](#), [5](#)
  
- vote\_carto\_color (voteogram-colors), [7](#)
- vote\_carto\_fill (voteogram-colors), [7](#)
- voteogram, [6](#)
- voteogram-colors, [7](#)
- voteogram-exports, [7](#)