

# Package ‘pollstR’

March 4, 2016

**Title** Client for the HuffPost Pollster API

**Description** Client for the HuffPost Pollster API, which provides access to U.S. opinion polls since 2004.

**Version** 1.2.2

**Date** 2016-03-04

**Depends** R (>= 3.0.3)

**Imports** httr, plyr, jsonlite

**Suggests** knitr, dplyr, ggplot2, rmarkdown, testthat

**URL** <https://github.com/rOpenGov/pollstR>

**BugReports** <https://github.com/rOpenGov/pollstR/issues>

**VignetteBuilder** knitr

**License** GPL-2

**LazyLoad** yes

**LazyData** true

**Collate** 'pollstr-package.R' 'chart.R' 'charts.R' 'polls.R'

**RoxygenNote** 5.0.1

**NeedsCompilation** no

**Author** Jeffrey B. Arnold [aut, cre],  
Thomas J. Leeper [aut],  
Drew Linzer [aut]

**Maintainer** Jeffrey B. Arnold <[jeffrey.arnold@gmail.com](mailto:jeffrey.arnold@gmail.com)>

**Repository** CRAN

**Date/Publication** 2016-03-04 08:43:23

## R topics documented:

pollstR . . . . .	2
pollstr_chart . . . . .	2
pollstr_charts . . . . .	3
pollstr_polls . . . . .	4

**Index****6**


---

pollstR *R client for the Huffpost Pollster API*

---

**Description**

This package provides an R interface to the Huffington Post Pollster API. Pollster provides programmatic access to opinion polls collected by the Huffington Post.

**Details**

See <http://elections.huffingtonpost.com/pollster/api> for more details on the API.

---

pollstr\_chart *Return a single chart*

---

**Description**

Return a single chart. This includes both current and historical estimates by date.

**Usage**

```
pollstr_chart(slug, convert = TRUE)
```

**Arguments**

slug            The slug-name of the chart to be returned.  
 convert        Rearrange the data returned by the API into easier to use data frames.

**Value**

If `convert=TRUE`, then a "pollstr\_chart" object with elements

- title Title of the chart.
- slug Slug (URL-friendly title) of the chart.
- topic Topic of the chart.
- short\_title Short title of the chart.
- poll\_count Number of polls in the chart.
- last\_updated Time the chart was last updated.
- url URL of the chart.
- estimates A data frame with an observation for each choice and the current estimates.
- estimates\_by\_date A data frame with an observation for each choice at each date, with estimates.

Otherwise, a "list" in the original structure of the json returned by the API.

**References**

<http://elections.huffingtonpost.com/pollster/api>

**Examples**

```
## Not run:
chart1 <- pollstr_chart('2012-general-election-romney-vs-obama')

## End(Not run)
```

---

pollstr_charts	<i>Get list of available charts</i>
----------------	-------------------------------------

---

**Description**

Get list of available charts

**Usage**

```
pollstr_charts(topic = NULL, state = NULL, showall = NULL,
  convert = TRUE)
```

**Arguments**

topic	Only include charts related to a specific topic. See <a href="http://elections.huffingtonpost.com/pollster/api">http://elections.huffingtonpost.com/pollster/api</a> for examples.
state	Only include charts from a single state. Use 2-letter state abbreviations. "US" will return all national charts.
showall	logical Include charts for races that were once possible but didn't happen (e.g. Gingrich vs. Obama 2012)
convert	Rearrange the data returned by the API into easier to use data frames.

**Value**

If convert=TRUE, a "pollstr\_charts" object with elements

charts Data frame with data on charts.

estimates Data frame with current estimates from each chart. The column slug matches this data frame to charts

Otherwise, a "list" in the original structure of the json returned by the API.

**References**

<http://elections.huffingtonpost.com/pollster/api>

**Examples**

```
## Not run:
# Get charts related to Washington
wa <- pollstr_charts(state='WA')
# Get national charts
us_charts <- pollstr_charts(state='US')
# Get charts in the topic '2016-president'
gov <- pollstr_charts(topic='2016-president')
# Get all charts
allcharts <- pollstr_charts()

## End(Not run)
```

---

pollstr_polls	<i>Get a list of polls</i>
---------------	----------------------------

---

**Description**

Get a list of polls

**Usage**

```
pollstr_polls(page = 1, chart = NULL, state = NULL, topic = NULL,
  before = NULL, after = NULL, sort = FALSE, showall = NULL,
  max_pages = 1, convert = TRUE)
```

**Arguments**

page	Return page number
chart	List polls related to the specified chart. Chart names are the slug returned by pollstr_charts.
state	Only include charts from a single state. Use 2-letter pstate abbreviations. "US" will return all national charts.
topic	Only include charts related to a specific topic. See the <a href="http://elections.huffingtonpost.com/pollster/api">http://elections.huffingtonpost.com/pollster/api</a> for examples.
before	Only list polls that ended on or bfore the specified date.
after	Only list polls that ended on or bfore the specified date.
sort	If TRUE, then sort polls by the last updated time.
showall	Include polls for races that were once possible but didn't happen (e.g. Gingrich vs. Obama 2012)
max_pages	Maximum number of pages to get.
convert	Rearrange the data returned by the API into easier to use data frames.

**Value**

If `convert=TRUE`, a "pollstr\_polls" object with elements

`polls` A data.frame with entries for each poll.

`questions` A data.frame with entries for each question asked in the polls.

`survey_houses` A data.frame with the survey houses of the polls. There can be multiple survey houses for a poll.

`sponsors` A data.frame with the sponsors of the polls. Not all polls have sponsors.

Otherwise, a "list" in the original structure of the json returned by the API.

**References**

<http://elections.huffingtonpost.com/pollster/api>

**Examples**

```
## Not run:  
# Get polls related to a chart pulled programmatically with  
# pollstr_charts()  
all_charts <- pollstr_charts()  
pollstr_polls(chart=all_charts$slug[1])  
# Lookup polls related to a specific topic  
pollstr_polls(topic='2016-president')  
  
## End(Not run)
```

# Index

[pollstR](#), [2](#)  
[pollstR-package \(pollstR\)](#), [2](#)  
[pollstr\\_chart](#), [2](#)  
[pollstr\\_charts](#), [3](#)  
[pollstr\\_polls](#), [4](#)