

Package: partycoloR (via r-universe)

May 17, 2026

Type Package

Title Extract Party Colors and Logos from Wikipedia

Version 0.3.0

Description Extract political party colors and logos from English Wikipedia party pages. Provides functions to scrape party infoboxes for color codes (HEX or HTML color names) and logo images. Includes integration with the Party Facts database for easy party lookups. Designed for political scientists and party researchers working with electoral and party data. For Party Facts, see Döring and Regel (2019) <[doi:10.1177/1354068818820671](https://doi.org/10.1177/1354068818820671)> and Bederke, Döring, and Regel (2023) <[doi:10.7910/DVN/TJINLQ](https://doi.org/10.7910/DVN/TJINLQ)>.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports httr (>= 1.4.0), rvest (>= 1.0.0), xml2 (>= 1.3.0), purrr (>= 0.3.0), stringr (>= 1.4.0), tibble (>= 3.0.0), rlang (>= 0.4.0), curl (>= 4.0)

Suggests testthat (>= 3.0.0), knitr, rmarkdown, ggplot2, scales, dplyr, tidyr

VignetteBuilder knitr

URL <https://lwarode.github.io/partycoloR/>,
<https://github.com/lwarode/partycoloR>

BugReports <https://github.com/lwarode/partycoloR/issues>

Config/testthat/edition 3

Config/pak/sysreqs libicu-dev libxml2-dev libssl-dev

Repository <https://lwarode.r-universe.dev>

Date/Publication 2026-02-16 11:24:04 UTC

RemoteUrl <https://github.com/lwarode/partycolor>

RemoteRef HEAD

RemoteSha bc1567899c762ac4d31f67349c4e22e3f4da0c39

Contents

clear_partycolor_cache	2
download_party_logo	3
get_party_color	3
get_party_color_by_name	5
get_party_info	6
get_party_info_by_name	7
get_party_logo	8
get_party_logo_by_name	9
get_partyfacts_wikipedia	10
lookup_party_url	11
party_data	12
wikipedia_party_color	13

Index **14**

clear_partycolor_cache
Clear the partycoloR cache

Description

Clears cached data (Partyfacts dataset) from the current R session.

Usage

```
clear_partycolor_cache()
```

Value

Invisible NULL.

Examples

```
clear_partycolor_cache()
```

download_party_logo *Download party logo image*

Description

Downloads a party logo image from a URL and saves it to a file.

Usage

```
download_party_logo(logo_url, destfile, overwrite = FALSE)
```

Arguments

logo_url	A character string with the logo URL (from [get_party_logo()]).
destfile	Path where the image should be saved.
overwrite	Logical. If 'TRUE', overwrite existing files.

Value

Invisible 'TRUE' if successful, 'FALSE' otherwise.

Examples

```
if (curl::has_internet()) {  
  logo_url <- get_party_logo(  
    "https://en.wikipedia.org/wiki/Democratic_Party_(United_States)"  
  )  
  tmp_file <- tempfile(fileext = ".png")  
  download_party_logo(logo_url, tmp_file)  
  unlink(tmp_file)  
}
```

get_party_color *Extract party color from Wikipedia*

Description

Extracts the primary party color (or all colors) from a political party's English Wikipedia page. The function scrapes the party infobox for color information.

Usage

```
get_party_color(url, all_colors = FALSE, normalize = TRUE, use_cache = TRUE)
```

Arguments

<code>url</code>	A character vector of Wikipedia URLs for political party pages.
<code>all_colors</code>	Logical. If <code>'FALSE'</code> (default), returns only the first/primary color. If <code>'TRUE'</code> , returns all colors as a list.
<code>normalize</code>	Logical. If <code>'TRUE'</code> (default), attempts to normalize color values to uppercase hex codes. Named colors (e.g., "red") are converted to hex codes.
<code>use_cache</code>	Logical. If <code>'TRUE'</code> (default), looks up data in the bundled <code>'party_data'</code> dataset first before scraping Wikipedia. If <code>'FALSE'</code> , always scrapes live from Wikipedia. Using the cache is much faster and reduces load on Wikipedia servers.

Details

By default, this function uses the bundled `'party_data'` dataset which contains pre-scraped data for major political parties. This provides instant lookups without network requests. If a party is not in the bundled data, the function automatically falls back to live Wikipedia scraping. Set `'use_cache = FALSE'` to always scrape fresh data from Wikipedia.

The function works by scraping the Wikipedia infobox (vcard table) for spans with background-color style attributes. This depends on Wikipedia's current HTML structure and may occasionally fail if the page structure changes.

For use with `'dplyr::mutate()'`, this function is vectorized over the `'url'` argument. Each URL is processed independently.

Value

If `'all_colors = FALSE'`, a character vector of hex color codes (or NA for failed extractions). If `'all_colors = TRUE'`, a list of character vectors containing all colors for each URL.

Examples

```
if (curl::has_internet()) {
  # Fast lookup using bundled data (default)
  get_party_color("https://en.wikipedia.org/wiki/Democratic_Party_(United_States)")

  # Force live scraping for most recent data
  get_party_color("https://en.wikipedia.org/wiki/Democratic_Party_(United_States)",
                 use_cache = FALSE)

  # Multiple parties
  urls <- c(
    "https://en.wikipedia.org/wiki/Democratic_Party_(United_States)",
    "https://en.wikipedia.org/wiki/Republican_Party_(United_States)"
  )
  get_party_color(urls)

  # Get all colors (some parties have multiple)
  get_party_color(urls, all_colors = TRUE)
}
```

`get_party_color_by_name`*Get party color by name*

Description

A convenience function that combines party lookup and color extraction. Searches for a party by name, finds its Wikipedia URL, and extracts the party color.

Usage

```
get_party_color_by_name(  
  party_name,  
  country = NULL,  
  all_colors = FALSE,  
  data = NULL  
)
```

Arguments

<code>party_name</code>	A character string with the party name to search for.
<code>country</code>	Optional. ISO 3-letter country code to filter results.
<code>all_colors</code>	Logical. If 'TRUE', returns all colors. Default is 'FALSE'.
<code>data</code>	Optional. A Partyfacts dataset.

Value

If exactly one party is found, returns the color(s). If multiple parties match, returns a tibble with party info and colors. Returns NA if no party is found.

Examples

```
if (curl::has_internet()) {  
  # Get color for German SPD  
  get_party_color_by_name("SPD", country = "DEU")  
  
  # Search more broadly  
  get_party_color_by_name("Labour", country = "GBR")  
}
```

get_party_info	<i>Extract party information from Wikipedia</i>
----------------	---

Description

Extracts both the party color(s) and logo URL from a political party's English Wikipedia page in a single call.

Usage

```
get_party_info(url, all_colors = FALSE, use_cache = TRUE)
```

Arguments

url	A character vector of Wikipedia URLs for political party pages.
all_colors	Logical. If 'FALSE' (default), returns only the primary color in the 'color' column. If 'TRUE', adds an 'all_colors' list-column.
use_cache	Logical. If 'TRUE' (default), looks up data in the bundled 'party_data' dataset first before scraping Wikipedia. If 'FALSE', always scrapes live from Wikipedia. Using the cache is much faster and reduces load on Wikipedia servers.

Details

By default, this function uses the bundled 'party_data' dataset which contains pre-scraped data for major political parties. This provides instant lookups without network requests. If a party is not in the bundled data, the function automatically falls back to live Wikipedia scraping. Set 'use_cache = FALSE' to always scrape fresh data from Wikipedia.

This function combines [get_party_color()] and [get_party_logo()] into a single call, which is more efficient when you need both pieces of information as it only fetches each Wikipedia page once.

Value

A tibble with columns: 'url', 'color' (primary color as hex), 'logo_url', and optionally 'all_colors' (list of all colors).

Examples

```
if (curl::has_internet()) {
  # Fast lookup using bundled data (default)
  urls <- c(
    "https://en.wikipedia.org/wiki/Democratic_Party_(United_States)",
    "https://en.wikipedia.org/wiki/Republican_Party_(United_States)",
    "https://en.wikipedia.org/wiki/Social_Democratic_Party_of_Germany"
  )
  get_party_info(urls)

  # Force live scraping for most recent data
```

```
get_party_info(urls, use_cache = FALSE)

# Include all colors
get_party_info(urls, all_colors = TRUE)
}
```

get_party_info_by_name

Get party info by name

Description

A convenience function that combines party lookup with color and logo extraction. Searches for a party by name, finds its Wikipedia URL, and extracts both color and logo.

Usage

```
get_party_info_by_name(
  party_name,
  country = NULL,
  all_colors = FALSE,
  data = NULL
)
```

Arguments

party_name	A character string with the party name to search for.
country	Optional. ISO 3-letter country code to filter results.
all_colors	Logical. If 'TRUE', includes all colors as a list-column. Default is 'FALSE'.
data	Optional. A Partyfacts dataset.

Value

A tibble with party info, color, and logo_url columns. Returns a tibble with zero rows if no party is found.

Examples

```
if (curl::has_internet()) {
  # Get info for German SPD
  get_party_info_by_name("SPD", country = "DEU")

  # Search more broadly with all colors
  get_party_info_by_name("Labour", country = "GBR", all_colors = TRUE)
}
```

get_party_logo	<i>Extract party logo URL from Wikipedia</i>
----------------	--

Description

Extracts the party logo image URL from a political party's English Wikipedia page. The function scrapes the party infobox for the logo image.

Usage

```
get_party_logo(url, use_cache = TRUE)
```

Arguments

url	A character vector of Wikipedia URLs for political party pages.
use_cache	Logical. If 'TRUE' (default), looks up data in the bundled 'party_data' dataset first before scraping Wikipedia. If 'FALSE', always scrapes live from Wikipedia. Using the cache is much faster and reduces load on Wikipedia servers.

Details

By default, this function uses the bundled 'party_data' dataset which contains pre-scraped logo URLs for major political parties. This provides instant lookups without network requests. If a party is not in the bundled data, the function automatically falls back to live Wikipedia scraping. Set 'use_cache = FALSE' to always scrape fresh data from Wikipedia.

The function looks for logo images in the Wikipedia infobox. The returned URL is typically a Wikimedia Commons thumbnail URL. Note that some party pages may not have logos, or the logo may be in a non-standard location.

The returned URLs point to image files hosted on Wikimedia servers. These can be used directly in R graphics or downloaded for further processing.

For use with 'dplyr::mutate()', this function is vectorized over the 'url' argument.

Value

A character vector of logo image URLs (or NA for failed extractions or pages without logos).

Examples

```
if (curl::has_internet()) {  
  # Fast lookup using bundled data (default)  
  get_party_logo("https://en.wikipedia.org/wiki/Democratic_Party_(United_States)")  
  
  # Force live scraping for most recent data  
  get_party_logo("https://en.wikipedia.org/wiki/Democratic_Party_(United_States)",  
                use_cache = FALSE)  
  
  # Multiple parties
```

```
urls <- c(
  "https://en.wikipedia.org/wiki/Democratic_Party_(United_States)",
  "https://en.wikipedia.org/wiki/Republican_Party_(United_States)"
)
get_party_logo(urls)
}
```

get_party_logo_by_name

Get party logo by name

Description

A convenience function that combines party lookup and logo extraction. Searches for a party by name, finds its Wikipedia URL, and extracts the party logo URL.

Usage

```
get_party_logo_by_name(party_name, country = NULL, data = NULL)
```

Arguments

party_name	A character string with the party name to search for.
country	Optional. ISO 3-letter country code to filter results.
data	Optional. A Partyfacts dataset.

Value

If exactly one party is found, returns the logo URL. If multiple parties match, returns a tibble with party info and logo URLs. Returns NA if no party is found.

Examples

```
if (curl::has_internet()) {
  # Get logo for German SPD
  get_party_logo_by_name("SPD", country = "DEU")

  # Search more broadly
  get_party_logo_by_name("Labour", country = "GBR")
}
```

get_partyfacts_wikipedia

Download Partyfacts Wikipedia data

Description

Downloads the current Partyfacts Wikipedia dataset containing party names, countries, and Wikipedia URLs for thousands of political parties worldwide.

Usage

```
get_partyfacts_wikipedia(cache = TRUE)
```

Arguments

cache Logical. If 'TRUE' (default), caches the downloaded data for the current R session to avoid repeated downloads.

Details

The data comes from the Partyfacts project (<<https://partyfacts.herokuapp.com/>>), which links party datasets and provides Wikipedia URLs for political parties. The data is downloaded from the Partyfacts GitHub repository.

Value

A tibble with columns: country, partyfacts_id, url, name_short, name, name_native, year_founded, year_dissolved.

See Also

[lookup_party_url()] for searching parties in the dataset.

Examples

```
if (curl::has_internet()) {  
  # Download the dataset  
  pf_data <- get_partyfacts_wikipedia()  
  
  # View parties from Germany  
  pf_data[pf_data$country == "DEU", ]  
}
```

lookup_party_url	<i>Look up party Wikipedia URL</i>
------------------	------------------------------------

Description

Search the Partyfacts Wikipedia dataset for a party by name and/or country.

Usage

```
lookup_party_url(party_name, country = NULL, data = NULL, exact = FALSE)
```

Arguments

party_name	A character string to search for in party names. The search is case-insensitive and matches partial names.
country	Optional. ISO 3-letter country code (e.g., "DEU", "USA", "GBR") to filter results.
data	Optional. A Partyfacts dataset (from [get_partyfacts_wikipedia()]). If not provided, downloads the data automatically.
exact	Logical. If 'TRUE', requires exact match on party name. Default is 'FALSE' (partial matching).

Value

A tibble with matching parties and their Wikipedia URLs.

Examples

```
if (curl::has_internet()) {  
  # Search for parties with "Democratic" in the name  
  lookup_party_url("Democratic")  
  
  # Search within a specific country  
  lookup_party_url("SPD", country = "DEU")  
  
  # Exact match  
  lookup_party_url("CDU", country = "DEU", exact = TRUE)  
}
```

party_data

Bundled party colors and logos dataset

Description

A dataset containing pre-scraped party colors and logos from Wikipedia for political parties in the Party Facts database. This dataset is bundled with the package to provide fast lookups without requiring live Wikipedia scraping.

Usage

party_data

Format

A tibble with 5 columns:

url Wikipedia URL of the party page (character)

color Primary party color as normalized HEX code, e.g., "#0015BC" (character)

all_colors List column containing all party colors as character vector (list)

logo_url URL to the party logo image on Wikimedia Commons (character)

last_updated Date when the data was scraped from Wikipedia (Date)

Details

This dataset contains pre-scraped color and logo information for political parties from the Party Facts Wikipedia dataset, covering parties worldwide that have English Wikipedia pages with available color data.

The data is used by default in `[get_party_color()]`, `[get_party_logo()]`, and `[get_party_info()]` to provide instant lookups. If a party is not found in this dataset, the functions automatically fall back to live Wikipedia scraping.

Color codes are normalized to uppercase 6-digit HEX format (e.g., "#FF0000"). Logo URLs point to images hosted on Wikimedia Commons.

To regenerate or update this dataset, see the script at `'data-raw/update.R'` in the package source.

Source

Data scraped from English Wikipedia party pages. Party list from Party Facts database: Döring and Regel (2019) <doi:10.1177/1354068818820671> and Bederke, Döring, and Regel (2023) <doi:10.7910/DVN/TJINLQ>.

See Also

`[get_party_color()]`, `[get_party_logo()]`, `[get_party_info()]`

Examples

```
# View structure of bundled data
str(party_data)

# Number of parties in bundled dataset
nrow(party_data)

# Count of parties by availability of data
sum(!is.na(party_data$color)) # Parties with colors
sum(!is.na(party_data$logo_url)) # Parties with logos

# Example: Find a specific party
## Not run:
party_data[grepl("Democratic.*United States", party_data$url), ]

## End(Not run)
```

wikipedia_party_color *Extract party colors from Wikipedia (legacy function)*

Description

```
‘r lifecycle::badge("deprecated")‘
```

This function is deprecated. Please use [get_party_color()] instead.

Usage

```
wikipedia_party_color(party_url_list)
```

Arguments

party_url_list A list of Wikipedia political party URLs

Value

A data frame / tibble containing extracted color codes from Wikipedia

Examples

```
if (curl::has_internet()) {
  party_list <- c(
    "https://en.wikipedia.org/wiki/Democratic_Party_(United_States)",
    "https://en.wikipedia.org/wiki/Republican_Party_(United_States)"
  )
  wikipedia_party_color(party_list)
}
```

Index

* datasets

- party_data, [12](#)

- clear_partycolor_cache, [2](#)

- download_party_logo, [3](#)

- get_party_color, [3](#)
- get_party_color_by_name, [5](#)
- get_party_info, [6](#)
- get_party_info_by_name, [7](#)
- get_party_logo, [8](#)
- get_party_logo_by_name, [9](#)
- get_partyfacts_wikipedia, [10](#)

- lookup_party_url, [11](#)

- party_data, [12](#)

- wikipedia_party_color, [13](#)