

# Package: drhutools (via r-universe)

December 6, 2024

**Type** Package

**Title** Political Science Academic Research Gears

**Version** 1.0.0

**Maintainer** Yue Hu <yuehu@tsinghua.edu.cn>

**Description** Using these tools to simplify the research process of political science and other social sciences. The current version can create folder system for academic project in political science, calculate psychological trait scores, visualize experimental and spatial data, and set up color-blind palette, functions used in academic research of political psychology or political science in general.

**URL** <https://www.drhuyue.site/software/drhutools/>

**BugReports** <https://github.com/sammo3182/drhutools/issues>

**License** MIT + file LICENSE

**Encoding** UTF-8

**Depends** R(>= 3.6.0)

**Imports** dplyr, ggplot2, purrr, htmltools, sf, htmlwidgets, jsonlite, leaflet, sp, stats, methods, ganimate, magick, mapview, webshot, animation, png, grDevices

**LazyData** true

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Suggests** knitr, remotes, rmarkdown

**VignetteBuilder** knitr

**Config/pak/sysreqs** libfontconfig1-dev libfreetype6-dev libgdal-dev gdal-bin libgeos-dev make libmagick++-dev gsfonts texlive libicu-dev libpng-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev zlib1g-dev

**Repository** <https://sammo3182.r-universe.dev>

**RemoteUrl** <https://github.com/sammo3182/drhutools>

**RemoteRef** HEAD

**RemoteSha** 9e86b2dfe507ef82d1f73ac31a247bb4a1b81fa3

## Contents

cdplot . . . . .	2
folderSystem . . . . .	3
gb_cols . . . . .	3
gb_pal . . . . .	4
goodmap . . . . .	5
leafletcn.map.names . . . . .	7
scale_color_gb . . . . .	7
scale_fill_gb . . . . .	8
toy_poly . . . . .	9
traits . . . . .	9

**Index** **11**

---

cdplot	<i>Visualizing the experimental outcome with cumulative distribution functions.</i>
--------	---

---

## Description

Packed ggplot2 function to compare the empirical cumulative distribution functions (ECDF) between the treatment and control groups in an experiment or quasi-experiment.

## Arguments

data	A data.frame including two columns, one recording the outcome and the other recording the assignment. The assignment column must be named as group.
ks_test	A logical option to indicate whether to show the Kolmogorov-Smirnov test result in the bottom-right corner. The default value is FALSE.
point_size	An integer to indicate the size of the points at the largest difference. The default value is 3.
point_color	An character or function to indicate the color of the points at the largest difference. The default value is drhutools::gb_cols("red").
link_color	An character or function to indicate the color of the link at the largest difference. The default value is drhutools::gb_cols("red").

## Value

A list of ggplot2 objects comparing the ECDFs between the control and treatment groups and identifying at most three largest differences.

**Examples**

```
data("PlantGrowth")

plot_plant <- cdplot(PlantGrowth, ks_test = TRUE)
plot_plant
```

---

folderSystem	<i>Folder hierarchy creator for academic research</i>
--------------	---

---

**Description**

Folder hierarchy creator for academic research

**Usage**

```
folderSystem()
```

**Details**

The function constructs a standardized folder hierarchy encompassing codes/, data/, output/, paper/, and document/. These directories are designated for storing programming scripts, data sets, processed outputs, manuscript drafts (including related images and submission documents), and assorted materials, respectively. This organization facilitates easy retrieval of project components, ensuring that members of Dr. Hu's Amazing Team can efficiently navigate and manage shared resources. This structure is advocated for collaborative projects to maintain uniformity across team operations.

**Value**

Local folders under the project's working directory.

---

gb_cols	<i>Function to extract colors as hex codes.</i>
---------	---

---

**Description**

Function to extract colors as hex codes.

**Usage**

```
gb_cols(...)
```

**Arguments**

... A character vector indicating the names of colors. Options includes "gold", "black", "orange", "red", "light green", "green", "light blue", "blue", "light grey", "grey", and "dark grey".

**Value**

A character of hex codes.

---

 gb\_pal

---

*Function to interpolate a gold-black color palette.*


---

**Description**

Function to interpolate a gold-black color palette.

**Usage**

```
gb_pal(palette = "main", reverse = FALSE, ...)
```

**Arguments**

palette A character vector indicating the name of palette in gb\_palettes. Available palettes:

- main: Gold and black colors.
- tricol: Gold, black, and dark grey to create a gradual effect.
- digitMixed: Five-pack colors specified for digital publications.
- printMixed: Five-pack colors specified for printed publications.
- full: A palette including all the colors gb\_cols can call.

reverse A logic vector indicating whether the palette should be reversed; the default is FALSE.

... Additional arguments to pass to colorRampPalette()

**Value**

A function that takes an integer argument (the required number of colors) and returns a character vector of colors interpolating the given sequence.

---

goodmap	<i>The goodmap function is designed to create interactive PNG Map or GIF Map from a provided data file. It supports two types of maps: point and polygon. The function can visualize data by either plotting points based on geographical coordinates or highlighting regions polygon based on their administrative boundaries (province or city level). Additionally, the function can generate animated that showcase the change of data.</i>
---------	---

---

### Description

If the map type is point, the color and size of the points will be determined by the `value_set` column in the data file, which means the different value of each point. If the map type is polygon, the color of the polygons will be determined by the average value of the `value_set` column for each city or province in the data file.

### Usage

```
goodmap(
  data_file,
  type = "point",
  level = NULL,
  animate = FALSE,
  animate_var = NULL,
  map_center = c(35.8617, 104.1954),
  zoom_level = 4,
  color_type = "numeric",
  custom_colors = NULL,
  point_radius = 5,
  legend_opacity = 0.7,
  legend_name = NULL,
  width = 800,
  height = 900
)
```

### Arguments

<code>data_file</code>	Dataframe. When generate point map, <code>data_file</code> should include required columns such as <code>g_lat</code> and <code>g_lon</code> . When generate polygon map, <code>data_file</code> should include required columns such as <code>prov</code> or <code>city</code> . The <code>prov</code> columns must be complete, official names rather than any shortened form or abbreviation. If there is only incomplete names or geocodes in your <code>data_file</code> , we recommend you to use function <code>regioncode</code> as a one-step solution to these conversion from incomplete names. Ensure the file is formatted correctly with appropriate column headers.
<code>type</code>	A string specifying the type of map to generate. Options are <code>point</code> for point maps using <code>g_lat</code> and <code>g_lon</code> , or <code>polygon</code> for maps with administrative boundaries.

level	A string specifying the level of administrative boundaries for polygon maps. Acceptable values are <code>province</code> or <code>city</code> . This parameter is required if <code>type</code> is <code>polygon</code> .
animate	A logical value indicating whether to generate an animation from the maps. The default is <code>FALSE</code> . If <code>animate</code> is <code>FALSE</code> , the whole data will be generated as a PNG file. If <code>animate</code> is <code>TRUE</code> , an animation will be generated from all panel data, the <code>animate_var</code> must be assigned.
animate_var	A string specifying the variable to animate over. The default is <code>NULL</code> , the whole data will be generated as a PNG file. If an animation is needed, <code>data_file</code> should include required the <code>animate_set</code> column. <code>animate_set</code> includes a series of unique values that represents year or month, or other categorical variable. The frame number of the animation depends on the <code>animate_set</code> .
map_center	A numeric vector of length 2 specifying the latitude and longitude for the center of the map view. Default is <code>c(35.8617, 104.1954)</code> , which is approximately the center of China.
zoom_level	A numeric value specifying the zoom level for the map. Default is 3.
color_type	If the data is discrete, such as types or categories, choose <code>factor</code> . If the data is continuous, such as temperature or pressure, choose <code>numeric</code> . Default is <code>numeric</code> .
custom_colors	A vector of colors for customizing the color gradient. Default is <code>NULL</code> , which uses the predefined color palette.
point_radius	A numeric value specifying the radius for point markers on point maps. Default is 5.
legend_opacity	A numeric value specifying the opacity of the legend. Default is 0.7.
legend_name	The name of the legend. Default is <code>Value</code> .
width	A numeric value specifying the width of the map images. Default is 800.
height	A numeric value specifying the height of the map images. Default is 900.

### Value

Image in the viewer.

### Examples

```
goodmap(
  toy_poly,
  type = "polygon",
  level = "province"
)
```

---

leafletcn.map.names     *Leaflet Map Names Dataset*

---

**Description**

A internal dataset providing city names and corresponding file names for mapping with Leaflet.

**Usage**

```
leafletcn.map.names
```

**Format**

A data frame with city names and file names.

**Value**

A data.frame object

---

scale\_color\_gb     *Fill scale constructors for gold & black colors*

---

**Description**

Fill scale constructors for gold & black colors

**Usage**

```
scale_color_gb(palette = "main", discrete = TRUE, reverse = FALSE, ...)
```

**Arguments**

palette	A character vector indicating the name of palette in gb_palettes. Available palettes: <ul style="list-style-type: none"> <li>• main: Gold and black colors.</li> <li>• tricol: Gold, black, and dark grey to create a gradual effect.</li> <li>• digitMixed: Five-pack colors specified for digital publications.</li> <li>• printMixed: Five-pack colors specified for printed publications.</li> <li>• full: A palette including all the colors gb_cols can call.</li> </ul>
discrete	A logic vector indicating whether color aesthetic is discrete or not; the default is "main".
reverse	A logic vector indicating whether the palette should be reversed.
...	Additional arguments passed to discrete_scale() or scale_color_gradientn(), used respectively when discrete is TRUE or FALSE

**Value**

No return value, called for modifying the appearance of the plot.

---

scale\_fill\_gb

*Color scale constructors for gold & black colors*

---

**Description**

Color scale constructors for gold & black colors

**Usage**

```
scale_fill_gb(palette = "main", discrete = TRUE, reverse = FALSE, ...)
```

**Arguments**

palette	A character vector indicating the name of palette in gb_palettes. Available palettes: <ul style="list-style-type: none"> <li>• main: Gold and black colors.</li> <li>• tricol: Gold, black, and dark grey to create a gradual effect.</li> <li>• digitMixed: Five-pack colors specified for digital publications.</li> <li>• printMixed: Five-pack colors specified for printed publications.</li> <li>• full: A palette including all the colors gb_cols can call.</li> </ul>
discrete	A logic vector indicating whether color aesthetic is discrete or not; the default is "main".
reverse	A logic vector indicating whether the palette should be reversed.
...	Additional arguments passed to discrete_scale() or scale_color_gradientn(), used respectively when discrete is TRUE or FALSE

**Value**

No return value, called for modifying the appearance of the plot.



---

toy_poly	<i>Toy Dataset for goodmap</i>
----------	--------------------------------

---

**Description**

A sample dataset designed to illustrate the functionality of the goodmap function.

**Usage**

```
toy_poly
```

**Format**

A geographic data frame with 6 rows and 5 variables:

**id** Optional data identifier.

**city** Prefectural names, required for city-level map plotting.

**prov** Provincial names, required for provincial-level map plotting.

**animate\_set** A numeric vector used to group data for animated plotting.

**value\_set** Values used to fill each polygon in the map.

**Value**

A data.frame object

---

traits	<i>Calculate psychological traits based on</i>
--------	--

---

**Description**

Calculate psychological traits based on

**Usage**

```
traits(survey)
```

**Arguments**

survey            Psychological survey data as a data.frame.

**Details**

The current version can calculate the TOSCA-3SC scores and Grit-O score.

**Value**

A data frame with trait scores for each observations

**References**

Duckworth, Angela L., Christopher Peterson, Michael D. Matthews, and Dennis R. Kelly. 2007. "Grit: Perseverance and Passion for Long-Term Goals." *Journal of Personality and Social Psychology* 92(6): 1087–1101. doi:10.1037/0022-3514.92.6.1087.

Tangney, June P. 1990. "Assessing Individual Differences in Proneness to Shame and Guilt: Development of the Self-Conscious Affect and Attribution Inventory." *Journal of Personality and Social Psychology* 59(1): 102–11. doi:10.1037/0022-3514.59.1.102.

**Examples**

```
column_names <- c("Q3|R3", "Q3|R4", "Q4|R3", "Q4|R4", "Q5|R5", "Q5|R6", "Q6|R3", "Q6|R4", "Q7|R3",  
"Q7|R4", "Q8|R5", "Q8|R6", "Q9|R5", "Q9|R6", "Q10|R5", "Q10|R6", "Q11|R5", "Q11|R6", "Q12|R3",  
"Q12|R4", "Q13|R3", "Q13|R4", "Q14|1", "Q15|1", "Q16|1", "Q17|1", "Q18|1", "Q19|1", "Q20|1",  
"Q21|1", "Q22|1", "Q23|1", "Q24|1", "Q25|1")  
  
toy_data <- data.frame(matrix(sample(1:5, 10 * length(column_names), replace = TRUE),  
ncol = length(column_names)))  
  
names(toy_data) <- column_names  
  
traits(toy_data)
```

# Index

## \* datasets

leafletcn.map.names, 7

toy\_poly, 9

cdplot, 2

folderSystem, 3

gb\_cols, 3

gb\_pal, 4

goodmap, 5

leafletcn.map.names, 7

scale\_color\_gb, 7

scale\_fill\_gb, 8

toy\_poly, 9

traits, 9