Package 'animbook'

July 22, 2025

Title Visualizing Changes in Performance Measures and Demographic Affiliations using Animation

Version 1.0.0

Description Create an interactive visualization to be used for communication purposes. Providing the function for preparing, plotting, and animating the data.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

URL https://github.com/KrisanatA/animbook

BugReports https://github.com/KrisanatA/animbook/issues

Depends R (>= 2.10)

LazyData true

Suggests knitr, rmarkdown

VignetteBuilder knitr

Imports dplyr, gganimate, ggplot2, plotly, purrr, RColorBrewer, rlang, stats, tibble, tidyr, tidyselect

NeedsCompilation no

Author Krisanat Anukarnsakulchularp [aut, cre, cph] (ORCID: <https://orcid.org/0009-0008-5638-7124>), Dianne Cook [aut] (ORCID: <https://orcid.org/0000-0002-3813-7155>)

Maintainer Krisanat Anukarnsakulchularp <krisanat.anu@gmail.com>

Repository CRAN

Date/Publication 2023-12-05 17:50:07 UTC

Contents

`aeles` *Australian election study data*

Description

The aes dataset contains the answers to the surveys that were done in 2019 for the election. This dataset only includes the id, year, party, and gender from the full survey data. The year column comes from the transformations of two different questions to see whether the voter voted for the same party in 2016 and 2019 or not, and if not, who did they vote for before?

Usage

```
aeles
```

Format

A data frame with 1,468 rows and 4 variables

id The id of the respondent

year Year

party Party that the respondent votes for in the House of Representatives

gender Gender of the respondent

Source

This dataset is from the following; Australian Election Study https://dataverse.ada.edu.au/file.xhtml?fileId=18013&version=3.0

anim_animate *Modified the ggplot object*

Description

This function will modify the ggplot object before the user can pass it to the rendering of choice.

Usage

```
anim_animate(plot)
```

Arguments

plot	ggplot object

Value

A gganimate object if the rendering is gganimate or a ggplot object if the rendering is plotly.

Examples

```
animbook <- anim_prep(data = osiris, id = ID, values = sales, time = year, group = japan)

plot <- wallaby_plot(animbook)

animate <- anim_animate(plot)

plotly::ggplotly(animate)
```

anim_prep *Transformed numerical into categorized data*

Description

This function transformed the numerical data into the categorized format by grouping data and scaling values.

Usage

```
anim_prep(
  data,
  id = NULL,
  values = NULL,
  time = NULL,
  group = NULL,
```

```
  ncat = 5L,
  breaks = NULL,
  label = NULL,
  group_scaling = NULL,
  scaling = "rank"
)
```

Arguments

data	A data frame contained the numerical values.
id	The column name that represents the identifiers variable.
values	The column name contains the numeric values.
time	The column name that represents the time variable.
group	The column name that represents the distinguished group between the values.
ncat	The number of categories to be created for scaling values.
breaks	A vector of breaks for creating bins.
label	A vector of labels to represent the qtile.
group_scaling	The column name that will be used for grouping the variable before scaling.
scaling	The scaling method to be used; "rank" or "absolute".

Details

The function takes the input data and performs several operations to transformed it into categorized format. It is done by grouping data, scales values, and assigned the qtile.

Value

A categorized data.

Examples

```
# rank scaling
anim_prep(data = osiris, id = ID, values = sales, time = year)

# group_rank scaling
anim_prep(data = osiris, id = ID, values = sales, time = year,
group_scaling = country)

# absolute scaling
anim_prep(data = osiris, id = ID, values = sales, time = year,
scaling = "absolute")

# group_absolute scaling
anim_prep(data = osiris, id = ID, values = sales, time = year,
group_scaling = country, scaling = "absolute")
```

`anim_prep_cat` *Transformed category data into a categorized format*

Description

This function transformed the category data in the categorized format by ordering the values.

Usage

```
anim_prep_cat(
  data,
  id = NULL,
  values = NULL,
  time = NULL,
  group = NULL,
  order = NULL,
  label = NULL
)
```

Arguments

`data`	A data frame contained the category values.
`id`	The column name that represents the identifiers variable.
`values`	The column name contains the category values.
`time`	The column name that represents the time variable.
`group`	The column name that represents the distinguished group between the values.
`order`	A vector of order for sorting the category values.
`label`	A vector of labels to represent the qtile.

Details

The function takes the input data, ordering the values, and assigning the variable names.

Value

A categorized data.

Examples

```
anim_prep_cat(data = aeles, id = id, values = party, time = year)
```

cat_change *Simulated data with some change (category)*

Description

This data has changed from category A to E between two-time points.

Usage

```
cat_change
```

Format

A data frame with 400 rows and 4 variables

id The id of the organisation

time time

gp Either X or Y

qnt Quantile group for the two times

Examples

```
d <- anim_prep_cat(cat_change, id = id, values = qnt,
time = time, group = gp)

d_p <- wallaby_plot(d, height = 1)

d_p_anim <- anim_animate(d_p)
```

dbl_change *Simulated data with some change (numerical)*

Description

This data contained the numerical values for each observation.

Usage

```
dbl_change
```

Format

A data frame with 400 rows and 4 variables

id The id of the organisation

time time

gp Either X or Y

values Numerical values represent sales

Examples

```
d <- anim_prep(dbl_change, id = id, values = values,
time = time, group = gp)

d_p <- wallaby_plot(d, height = 1)

d_p_anim <- anim_animate(d_p)
```

`funnel_web_plot` *Turn the data into a facetted plot*

Description

This function takes in the data which has been prepared by either `anim_prep()` or `anim_prep_cat()` and return the ggplot object. The user can still modify the plot the same as normal using the ggplot2 function.

Usage

```
funnel_web_plot(data, group_palette = NULL, ...)
```

Arguments

`data`	The animbook object returned from the prep function.
`group_palette`	The vector of the palette used by the function to supply the color to each group.
`...`	Additional arguments for customization. See details for more information.

Details

This function takes prepared data and generates a ggplot object. The funnel web plot is the plot that shows the line facetted plot showing the pattern between time period. The line jitter can be controlled using additional arguments such as height and width to control the appearance. For the shading area, the alpha argument can be used.

Value

Return a ggplot object.

Examples

```
animbook <- anim_prep(data = osiris, id = ID, values = sales, time = year, group = japan)

funnel_web_plot(animbook)
```

kangaroo_plot *Turn the data into a ggplot object for the animate function*

Description

This function takes in the data which has been prepared by the anim_prep() or anim_prep_cat() and return the ggplot object. The user can still modify the plot as usual.

Usage

```
kangaroo_plot(
  data,
  group_palette = NULL,
  shade_palette = NULL,
  rendering = "gganimate",
  time_dependent = FALSE,
  ...
)
```

Arguments

data	The categorized data.
group_palette	The vector of the palette used by the function to supply the color of each group.
shade_palette	The vector of the palette used by the function to supply the color to the shaded area.
rendering	The choice of method used to create and display the plot, either gganimate or plotly.
time_dependent	Logical. Should the visualization be time-dependent? The default is FALSE.
...	Additional arguments for customization. See details for more information.

Details

This function takes categorized data and generates a ggplot object. The kangaroo plot is the plot that shows the movement between groups over time. The point position and point size in the shaded area can be controlled using additional arguments such as height, width, and size. For the shading area, the alpha argument can be used.

Value

Return a ggplot object.

Examples

```
example <- anim_prep(data = dbl_change, id = id, values = values,
time = time, group = gp)

kangaroo_plot(example)
```

osiris *Osiris firm sales data*

Description

The Osiris dataset contains information on listed and major unlisted/delisted companies across the world from 2006 to 2018. This dataset only includes the year, ID, country, sales, and japan variables from the full Osiris dataset to give the user an example of the dataset format.

Usage

```
osiris
```

Format

A data frame with 10,270 rows and 5 variables

year Year

ID BvD(Bureau van Dijk) ID

country Address of incorp. - Country

sales Sales

japan Whether the firm is from Japan or not

Source

This dataset is from the following; Bureau van Dijk https://www.bvdinfo.com/en-gb/our-products/data/international/osiris.

wallaby_plot *Turn the data into a subset plot for animate function*

Description

This function takes in the data which has been prepared by the anim_prep() or anim_prep_cat() and return the ggplot object. The user can still modify the plot as usual.

Usage

```
wallaby_plot(
  data,
  group_palette = NULL,
  shade_palette = NULL,
  rendering = "gganimate",
  time_dependent = FALSE,
  subset = "top",
```

```
  relation = "one_many",
  total_point = NULL,
  x_lab = NULL,
  ...
)
```

Arguments

data	The categorized data.
group_palette	The vector of the palette used by the function to supply the color of each group.
shade_palette	The vector of the palette used by the function to supply the color of each shaded area.
rendering	The choice of method used to create and display the plot, either gganimate or plotly.
time_dependent	Logical. Should the visualization be time-dependent? The default is FALSE.
subset	A character string specifying the variable used for subsetting the data. The "top" and "bottom" strings can also be used in this argument.
relation	The choice of relationship for the values to display on the plot, either "one_many" or "many_one."
total_point	The number of points the users want for the wallaby plot. The default is NULL, where the number of the point is equal to the original number of points.
x_lab	The label for the x-axis.
...	Additional arguments for customization. See details for more information.

Details

This function takes categorized data and generates a ggplot object. The wallaby plot is the plot that shows the movement of the subset data between the start and end of the observable period. The point position and point size in the shaded area can be controlled using additional arguments such as height, width, and size. For the shading area, the alpha argument can be used.

Value

Return a ggplot object.

Examples

```
animbook <- anim_prep(data = osiris, id = ID, values = sales, time = year, group = japan)

wallaby_plot(animbook)
```

Index