Package 'TheseusPlot'

August 28, 2025

Type Package

Title Visualizing Decomposition of Differences in Rate Metrics

Version 0.1.1

Description Provides tools for decomposing differences in rate metrics between two groups into contributions from individual subgroups and visualizing them as a ``Theseus Plot''. Inspired by the story of the Ship of Theseus, the method replaces subgroup data from one group with that of another step by step, recalculating the overall metric at each stage to quantify subgroup contributions. A Theseus Plot combines the stepwise progression of a waterfall plot with the comparative bars of a bar chart, offering an intuitive way to understand subgroup-level effects.

License MIT + file LICENSE

URL https://github.com/hoxo-m/TheseusPlot

BugReports https://github.com/hoxo-m/TheseusPlot/issues

Depends R (>= 4.1.0)

Imports dplyr, ggplot2, forcats, memoise, R6, rlang, stats, stringr, tibble, tidyr, waterfalls

Suggests nycflights13

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Koji Makiyama [aut, cre, cph],
Shinichi Takayanagi [med],
Daisuke Ichikawa [exp],
LY Corporation Analytics Solution Enhancement Team [spn]

Maintainer Koji Makiyama <hoxo.smile@gmail.com>

Repository CRAN

Date/Publication 2025-08-28 07:50:06 UTC

Contents

`continuous_config` *Continuous Variable Configuration for Theseus Plot*

Description

The `continuous_config()` function creates a configuration object for handling continuous variables in Theseus plots. It controls how continuous data is binned into discrete categories for contribution calculations and visualization.

Usage

```
continuous_config(
  n = 10L,
  pretty = TRUE,
  split = c("count", "width", "rate"),
  breaks = NULL
)
```

Arguments

`n`	integer. Number of bins to create for a continuous variable.
`pretty`	logical. If TRUE, use pretty breaks for bin edges.
`split`	string. Method for binning continuous variables. Options are:
	`"count"` divide the variable into bins with roughly equal number of observations.
	`"width"` divide the range of the variable into equal-width bins.
	`"rate"` divide based on differences in outcome rates between bins.
`breaks`	numeric vector specifying custom break points.

Value

A list containing binning parameters (`n`, `pretty`, `split`, `breaks`) to be used in plotting or contribution calculations for continuous variables.

Examples

```
library(TheseusPlot)
continuous_config(n = 5, pretty = FALSE, split = "rate")
```

`create_ship` *Creates a Ship Object for Generating Theseus Plots*

Description

Creates a ship object, which serves as a container for data and methods to generate Theseus plots for decomposing differences in rate metrics.

Usage

```
create_ship(
  data1,
  data2,
  y = "y",
  labels = c("Original", "Refitted"),
  ylab = NULL,
  digits = 3L,
  text_size = 1
)
```

Arguments

`data1`	data frame representing the first group (e.g., the baseline or "original" data).
`data2`	data frame representing the second group (e.g., the comparison or "refitted" data).
`y`	column name specifying the outcome variable used to compute the rate metric (default is `"y"`). Typically, this is a binary indicator (e.g., 0/1) that is aggregated to form rates.
`labels`	character vector of length 2 giving the labels for the two groups. The first corresponds to `data1`, the second to `data2`. Default is `c("Original", "Refitted")`.
`ylab`	string specifying the y-axis label for plots. If `NULL` (default), no label is displayed.
`digits`	integer indicating the number of decimal places to use for displaying numeric values (default is 3).
`text_size`	numeric value specifying the relative size of text elements in plots (default is 1.0).

Value

A ShipOfTheseus object, which can be used with `plot()` to create Theseus plots.

Examples

```
library(dplyr)
library(TheseusPlot)

data <- nycflights13::flights |>
  filter(!is.na(arr_delay)) |>
  mutate(on_time = arr_delay <= 0)

data1 <- data |> filter(month == 1)
data2 <- data |> filter(month == 2)

create_ship(data1, data2, y = on_time)
```

`ShipOfTheseus` *An R6 Class for Generating Theseus Plot*

Description

The 'ShipOfTheseus' class decomposes the difference in outcome rates between two datasets and visualizes the results as a Theseus Plot. It provides methods to compute contributions of individual attributes, summarize results in tables, and generate waterfall-style plots for intuitive interpretation.

Methods

Public methods:

- `ShipOfTheseus$new()`
- `ShipOfTheseus$table()`
- `ShipOfTheseus$plot()`
- `ShipOfTheseus$plot_flip()`
- `ShipOfTheseus$clone()`

Method `new()`**:** The constructor of the ShipOfTheseus class.

Usage:

```
ShipOfTheseus$new(data1, data2, outcome, labels, ylab, digits, text_size)
```

Arguments:

`data1` data frame representing the first group (e.g., the baseline or "original" data).

`data2` data frame representing the second group (e.g., the comparison or "refitted" data).

`outcome` string specifying the outcome variable used to compute the rate metric (default is "y"). Typically, this is a binary indicator (e.g., 0/1) that is aggregated to form rates.

`labels` character vector of length 2 giving the labels for the two groups. The first corresponds to 'data1', the second to 'data2'. Default is c("Original", "Refitted").

`ylab` string specifying the y-axis label for plots. If NULL (default), no label is displayed.

`digits` integer indicating the number of decimal places to use for displaying numeric values (default is 3).

`text_size` numeric value specifying the relative size of text elements in plots (default is 1).

Returns: A ShipOfTheseus object, which can be used with `plot()` to create Theseus plots.

Method `table()`**:** Generate a contribution table for a given column.

Usage:

```
ShipOfTheseus$table(column_name, n = Inf, continuous = continuous_config())
```

Arguments:

`column_name` string. The name of the column to analyze.

`n` integer. Maximum number of top contributing attributes to display. If the number of attributes exceeds 'n', the remaining are aggregated.

`continuous` list. A configuration list for handling continuous variables (e.g., specifying number of bins or custom breaks).

Returns: A tibble summarizing each attribute's contribution to the difference between the two groups, including counts, total outcomes, and rates for each subgroup.

Method `plot()`**:** Generate a Theseus plot for a specified column

Usage:

```
ShipOfTheseus$plot(
  column_name,
  n = 10L,
  main_item = NULL,
  bar_max_value = NULL,
  levels = NULL,
  continuous = continuous_config()
)
```

Arguments:

`column_name` The name of the column to visualize.

`n` integer. Maximum number of top contributing attributes to display. Remaining attributes are aggregated if necessary.

`main_item` string. The attribute used as the reference for scaling the bar heights.

`bar_max_value` numeric. Maximum value for scaling the contribution bars.

`levels` character vector specifying the display order of attributes.

`continuous` list. Configuration for handling continuous variables (e.g., number of bins or custom breaks).

Returns: A ggplot object representing the Theseus Plot for the specified column.

Method `plot_flip()`**:** Generate a Theseus plot for a specified column

Usage:

```
ShipOfTheseus$plot_flip(
  column_name,
  n = 10L,
  main_item = NULL,
  bar_max_value = NULL,
  levels = NULL,
  continuous = continuous_config()
)
```

Arguments:

`column_name` The name of the column to visualize.

`n` integer. Maximum number of top contributing attributes to display. Remaining attributes are aggregated if necessary.

`main_item` string. The attribute used as the reference for scaling the bar heights.

`bar_max_value` numeric. Maximum value for scaling the contribution bars.

`levels` character vector specifying the display order of attributes.

`continuous` list. Configuration for handling continuous variables (e.g., number of bins or custom breaks).

Returns: A ggplot object representing the Theseus Plot for the specified column.

Method `clone()`**:** The objects of this class are cloneable with this method.

Usage:

```
ShipOfTheseus$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

Index