Package 'personograph'

August 29, 2016

Title Pictographic Representation of Treatment Effects

Version 0.1.3

Author Joel Kuiper [aut, cre], Iain Marshall [aut]

Maintainer Joel Kuiper <jkuiper@doctorevidence.com>

Description Visualizes treatment effects using person icons, similar to Cates (NNT) charts.

License LGPL (>= 2.0, < 3) | Mozilla Public License

LazyData true

URL https://github.com/joelkuiper/personograph

Depends R (>= 3.1.0), grImport

Imports stats, grDevices, grid

Suggests meta

NeedsCompilation no

Repository CRAN

Date/Publication 2015-11-07 00:08:38

R topics documented:

personograph-package *Generate personograph plots from data*

Description

A personograph (Kuiper-Marshall plot) is a pictographic representation of (relative) harm and benefit from an intervention. It is similar to Visual Rx (Cates Plots). Each icon on the grid is colored to indicate whether that percentage of people is harmed by the intervention, would benefit from the intervention, has good outcome regardless of intervention, or bad outcome regardless of intervention. This terminology is similar to that of Uplift Modelling.

Details

The plot function `personograph` is implemented in such a way that it's easy to just pass a named list of percentages, colors, and an icon. Making it potentially useful for other use cases as well.

The example code will generate the following graph if `higher_is_better=F`:

Example from rMeta

Funding & Acknowledgments: This software was commissioned and sponsored by Doctor Evidence. The Doctor Evidence mission is to improve clinical outcomes by finding and delivering medical evidence to healthcare professionals, medical associations, policy makers and manufacturers through revolutionary solutions that enable anyone to make informed decisions and policies using medical data that is more accessible, relevant and readable.

Source & Issues: Source code and issue tracker can be found on Github.

See Also

`personograph`

uplift

Examples

```
# Example data
data <- read.table(textConnection('
          name ev.trt n.trt ev.ctrl n.ctrl
1     Auckland     36   532      60    538
2        Block      1    69       5     61
3        Doran      4    81      11     63
4        Gamsu     14   131      20    137
5     Morrison      3    67       7     59
6 Papageorgiou      1    71       7     75
7      Tauesch      8    56      10     71
'
), header=TRUE)

sm <- "RR" # The outcome measure (either Relative Risk or Odds Ratio)
if (requireNamespace("meta", quietly = TRUE)) { # use meta if available
    ## Calculate the pooled OR or RR point estimate
    m <- with(data, meta::metabin(ev.trt, n.trt, ev.ctrl, n.ctrl, sm=sm))
    point <- exp(m$TE.random) # meta returns random effects estimate on the log scale
} else {
    # Calculated Random Effects RR, using the meta package
    point <- 0.5710092
}

# Approximate the Control Event Rates using a weighted median
cer <- w.approx.cer(data[["ev.ctrl"]], data[["n.ctrl"]])

# Calculate the Intervention Event Rates (IER) from the CER and point estimate
ier <- calc.ier(cer, point, sm)

# Calcaulte the "uplift" statistics
# Note that this depends on the direction of the outcome effect (higher_is_better)
u <- uplift(ier, cer, higher_is_better=FALSE)
plot(u, fig.title="Example", fig.cap="Example")
```

`calc.ier` *Calculate the IER (Intervention Event Rates)*

Description

Calculate the IER (Intervention Event Rates)

Usage

```
calc.ier(cer, point, sm)
```

Arguments

cer	Absolute risk with control (calculated; from 0 to 1)
point	Relative risk with intervention (direct from meta-analysis)
sm	The outcome measure, RR or OR as string

Value

Absolute risk of intervention as Intervention Event Rates (IER)

See Also

w.approx.cer

---

personograph *Plots a personograph*

Description

Plots a personograph from a named list with percentages (must sum to 1). A personograph is a graphical represenation of relative benefit or harm, using a grid of icons with different colors. Its intended use is similar to that of Cates Plots (Visual Rx, Number Needed to Treat visualization). Although these could be seen as Kuiper-Marshall plots.

Usage

```
personograph(data, fig.title = NULL, fig.cap = NULL, draw.legend = T,
  icon = NULL, icon.dim = NULL, icon.style = 1, n.icons = 100,
  plot.width = 0.75, dimensions = ceiling(sqrt(c(n.icons, n.icons))),
  fudge = 0.0075, legend.show.zeros = TRUE, force.fill = "ignore",
  round.fn = round.standard, colors = as.colors(data))
```

Arguments

data	A list of names to percentages (from 0 to 1)
fig.title	Figure title
fig.cap	Figure caption
draw.legend	Logical if TRUE (default) will draw the legend
icon	A grImport Picture for the icon, overwrites icon.style
icon.dim	The dimensions of icon as a vector c(width, height) as numerical. Calculated from the dimensions if not supplied
icon.style	A numeric from 1-11 indicating which of the included icons to use, they are mostly variations on the theme
n.icons	Number of icons to draw, defaults to 100

plot.width	The percentage of width that the main plotting area should take (with respect to the frame)
dimensions	A vector of c(rows, columns) for the dimensions of the grid
fudge	Fudge factor for the icon size, substracted from the icon.size
legend.show.zeros	Logical if TRUE indicating whether to show zero (0) values in the legend.
force.fill	A character vector of 'ignore' (default), 'most', 'least', or one of the names from data. Defines the behaviour for cases when the rounding doesn't add up to n.icons. 'ignore' simply draws less icons, 'most' adds an icon to the largest group, 'least' to the smallest. If a name from data is supplied it will added to that element
round.fn	Function that is applied to round the percentages from data to n.icons. See also force.fill
colors	A vector of names to colors, must match the names in data. Uses gray.colors style if none supplied

Details

Supplying your own icon: You can supply your own icon by setting icon to a grImport Picture. A Picture can be loaded with grImport::readPicture which requires a grImport XML file. Obtaining this file from a standard SVG or PDF graphics file requires conversion. The easiest way is to convert your original file to PDF and then to PostScript (PS) with the command-line pdf2ps tool, then tracing it with grImport::PostScriptTrace. See the grImport package documentation for more details.

Value

None.

Examples

```
data <- list(first=0.9, second=0.1)
personograph(data)
# With colors
personograph(data, colors=list(first="red", second="blue"))
# With different icon.style
personograph(data, icon.style=4) # numeric from 1-11
# Plot a thousand in a 20x50 grid
personograph(data, n.icons=1000, dimensions=c(20,50), plot.width=0.75)
```

---

uplift *"Uplift" from IER and CER*

Description

Calculates the percentage (from 0 to 1) of people who have an intervention benefit, intervention harm, bad outcome regardless, and good outcome regardless from the Intervention Event Rates (IER) and Control Event Rates (CER). Note that the result depends on the direction of the outcome measure, e.g. `higher_is_better = T` (default) for intervention efficacy, `higher_is_better = F` for adverse events.

Usage

```
uplift(ier, cer, higher_is_better = NULL)
```

Arguments

`ier`	Intervention Event Rates
`cer`	Control Event Rates
`higher_is_better`	logical indicating the direction of the outcome measure, default TRUE

Details

The adopted terminology is similar to that of Uplift modelling https://en.wikipedia.org/wiki/Uplift_modelling

Value

A list of S3 class `personograph.uplift` with the following elements:

- `good outcome` people who have a good outcome regardless of intervention
- `bad outcome` people who have a bad outcome regradless of intervention
- `intervention benefit` people who benefit from intervention
- `intervention harm` people who are harmed by intervention

Can be plotted as a personograph with the S3 generic `plot`.

Examples

```
ier <- 0.06368133
cer <- 0.1115242
u <- uplift(ier, cer, higher_is_better=TRUE)
plot(u)
```

`w.approx.cer` *Calculate the CER (Control Event Rates)*

Description

Calculates the CER from the data, this is a approximation of absolute risk in the control population (from 0 to 1).

Usage

```
w.approx.cer(ev.ctrl, n.ctrl)
```

Arguments

`ev.ctrl`	Vector of event rates in the control group (/arm)
`n.ctrl`	Vector of sample sizes in the control group (/arm)

Details

By default it uses a weighted median of the indivdual control event rates. The weighted median has the benefit of always returning an event rate that actually did occur. However, it is possible that this might return a CER of 0. In this case we fall back to a weighted mean, and throw a warning. If this too returns a CER of 0, it probably means that there was not enough data to estimate the control risk accurately. In this case we recommend you obtain an estimate of the risk in the control group, for example from an observational study or expert opinion.

Value

Approximated Control Event Rates (CER)

Index