Package ‘timelineS’

October 14, 2022

Type    Package
Title   Timeline and Time Duration-Related Tools
Version 0.1.1
Date    2016-08-21
Author  Dahee Lee [aut, cre], Dustin Tingley [aut]
Maintainer Dahee Lee <dhlee99@gmail.com>
Description An easy tool for plotting annotated timelines, grouped timelines, and exploratory graphics (boxplot/histogram/density plot/scatter plot/line plot). Filter, summarize date data by duration and convert to calendar units.

URL    https://github.com/daheelee/timelineS
BugReports https://github.com/daheelee/timelineS/issues
License GPL-2
LazyData TRUE
Imports base, graphics, magrittr, dplyr, lubridate, ggplot2, stats
Depends R (>= 3.1.0)
RoxygenNote 5.0.1
NeedsCompilation no
Repository CRAN
Date/Publication 2016-08-22 14:13:31

R topics documented:

durCalc .................................................. 2
durPlot .................................................. 3
durSummary ............................................. 5
life_country ........................................... 5
life_exp .................................................. 6
mj_life ................................................... 6
timelineG ............................................... 6
timelineS .............................................. 7

Index 10
**durCalc**

**Filter Dataset by Date Duration**

**Description**

Calculates the duration between two dates, use it as a filter to select rows that satisfy the length criteria. Returns the dataset with additional columns regarding the length of durations in different units.

**Usage**

```
durCalc(df = NULL, start, end, timeunit = "day", filterlength = NA, filterlonger = TRUE, year = 365.25, month = 30.42)
```

**Arguments**

- `df` Data frame containing start and end dates.
- `start` Column in df for start dates or a date to use as start date.
- `end` Column in df for end dates or a date to use as as end date.
- `timeunit` Unit of time to be used in plots. "day(s)", "week(s)", "month(s)", "quarter(s)", "semiannual", "halfyear", or "year(s)".
- `filterlength` A time length to use as filter.
- `filterlonger` If TRUE, the function gives rows with longer durations than specified in filter-length. If FALSE, gives rows with shorter durations.
- `year` Number of days to use as a year. Default is 365.25.
- `month` Number of days to use as a month. Default is 30.42.

**Details**

Additional columns returned with the filtered rows are: length of duration in days, in specified time unit, and in calendar units, and how much longer/shorter the durations are compared to filter length in calendar units.

If no `filterlength` is provided, then returns all rows with length of duration in days and calendar units.

You can use dates for `start` and `end` and provide no `df` to get the length of duration between the dates in calendar units. See example.

**Value**

A subset of original data frame with additional columns in specified time units and calendar units.

**Author(s)**

Dahee Lee
durPlot

Graphs and Summary for Date Durations

Description

Plots boxplot, histogram, density plot, scatter plot, line plot and prints summary statistics for date
duration data.

Usage

durPlot(df, start, end, group = NA, timeunit = "days", plot_type = "all",
        facet = FALSE, facet.nrow = NULL, theme = NULL, other = NULL,
        fill_color = "black", line_color = "black", groupcolor = TRUE,
        point_size = 2, alpha = NA, binwidth = 0.5, show_legend = TRUE,
        titleFALSE, title_boxplot = "Boxplot", title_histogram = "Histogram",
        title_density = "Density Plot", title_scatter = "Scatter Plot",
        title_line = "Line Plot")

Arguments

df Data frame containing start dates, end dates and groups.
start Column in df for start dates.
end Column in df for end dates.
group Column in df for groups. Default is NA.
timeunit Unit of time to be used in plots. "day(s)"", "week(s)"", "month(s)"", "quarter(s)",
            "semiannual", "halfyear", or "year(s)".
plot_type One of "all", "boxplot", "histogram", "density", "scatter", "line". Default is "all".
facet If TRUE, wraps plots in group facets
facet.nrow Number of rows for facet wrap
theme Add theme elements if needed.
other  
  Add other elements if needed.

fill_color  
  Fill color

line_color  
  Line color

groupcolor  
  If FALSE, fill_color and line_color used for all groups. Default is TRUE.

point_size  
  Point size for scatterplot

alpha  
  Color transparency [0,1]

binwidth  
  Binwidth for histogram; default 0.5.

show_legend  
  Default is TRUE

title  
  If TRUE, puts main titles for each plot

title_boxplot  
  Title for boxplot title

title_histogram  
  Title for histogram

title_density  
  Title for density plot

title_scatter  
  Title for scatter plot

title_line  
  Title for line plot

Details

  The function also returns summary statistics for the specified date duration.

Author(s)

  Dahee Lee

See Also

  timelineS, timelineG, durSummary, durCalc

Examples

  durPlot(life_exp, start = "Birth", end = "Death", group = "Country",
        timeunit = "years", facet = TRUE, binwidth = 3, alpha = 0.7, title = TRUE)

  durPlot(life_exp, start = "Birth", end = "Death", group = "Country",
        timeunit = "years", alpha = 0.5, title = TRUE)
Description

Returns summary statistics for date duration data (for each group if group is provided)

Usage

durSummary(df, start, end, group = NA, timeunit = "days")

Arguments

df       Data frame containing start and end dates.
start    Column in df for start dates.
end      Column in df for end dates.
group    Column in df for groups. Default NA.
timeunit Unit of time to be used in plots. "day(s)"", "week(s)"", "month(s)"", "quarter(s)", "semiannual", "halfyear", or "year(s)".

Details

1 year = 365.25 days, 1 month = 30.42 days, 1 year = 52.14 weeks

Author(s)

Dahee Lee

See Also

durPlot, durCalc

Examples

durSummary(life_exp, start = "Birth", end = "Death", group = "Country", timeunit = "years")

life_country

Dates of birth and death, gender and phases

Format

dataframe of name(character), country(character), gender(character), phase(character) and dates(date)
Description

Dates of birth and death, country, gender and names

Format

dataframe of name(character), country(character), gender(character), and dates(date)

mj_life

Data for timelineS function example in timelineS package

Description

Events and dates of Michael Jackson’s life

Format

dataframe of events(character) and dates(date)

timelineG

Faceted Timelines for Grouped Data

Description

Plots faceted timelines for grouped data.

Usage

timelineG(df, start, end, names, phase = NA, group1 = NA, group2 = NA, width = 2, color = "grey", theme = NULL, other = NULL)

Arguments

df Data frame containing start dates, end dates, groups, phases, and names for each timeline.
start Column in df for start dates.
end Column in df for end dates.
names Column in df for names of each timeline
phase Column in df for phases.
group1  Column in df for groups to be used as the rows of the tabular display. Default is NA.
group2  Column in df for groups to be used as the columns of the tabular display. Default is NA.
width  Width of each timeline. Default is 2.
color  Color of timelines, only used when phase is not provided.
theme  Add theme elements if needed.
other  Add other elements if needed.

Author(s)
Dahee Lee

See Also
timelineS

Examples

### Plot timelines row-grouped by "Country"
timelineG(df = life_country, start = "Start", end = "End", names = "Name", phase = "Phase", group1 = "Country")

### Plot timelines row-grouped by "Country" and column-grouped by "Gender"
timelineG(df = life_country, start = "Start", end = "End", names = "Name", phase = "Phase", group1 = "Country", group2 = "Gender")

### Plot timelines, no group
timelineG(df = life_country, start = "Start", end = "End", names = "Name",color = "grey")

---

**timelineS**  
*Timeline with Event Labels*

Description

Plots a horizontal timeline with event descriptions at corresponding dates.

Usage

timelineS(df, main = NA, xlab = NA, buffer.days = 600, line.width = 5, line.color = "gray44", scale = "year", scale.format = "%Y", scale.font = 2, scale.orient = 1, scale.above = FALSE, scale.cex = 1, scale.tickwidth = 2, labels = paste(df[[1]], df[[2]]), label.direction = "downup", label.length = c(0.5,0.5,0.8,0.8), label.position = c(1,3), label.color = "gray44", label.cex = 0.8, label.font = 1, label.angle = 0, pch = 20, point.cex = 1, point.color = "gray44")
Arguments

- **df**: Data frame for events and dates. First column for event names and second column for dates in Date class.
- **main**: Title of the plot.
- **xlab**: X axis label.
- **buffer.days**: Additional days to add before and after the event dates on the timeline. Default is 600 days.
- **line.width**: Timeline width; default 5
- **line.color**: Timeline color.
- **scale**: Scale on timeline. One of "year", "quarter", "month", "week" or "day". See `seq.Date`.
- **scale.format**: Scale format; default "%Y".
- **scale.font**: Integer specifying font of scale. Default is 2. (1:plain, 2:bold, 3:italic, 4:bold italic, 5:symbol).
- **scale.orient**: Orientation of scale; default 1(upright)
- **scale.above**: If TRUE, the scale shows above the line.
- **scale.cex**: Scale font size relative to cex.
- **scale.tickwidth**: Width of scale tick; default 2.
- **labels**: Event labels. Events and corresponding dates as default.
- **label.direction**: Direction of labels from timeline. "downup", "updown", "up", or "down", default is "downup". See details.
- **label.length**: Distance of event label from the timeline. Could be a single value or a vector of lengths. Default is c(0.5, 0.5, 0.8, 0.8). See details.
- **label.position**: Integer specifying label positions; default c(1,3). See details.
- **label.color**: Label color(s).
- **label.cex**: Font size(s) of event labels; default 0.8.
- **label.font**: Integer specifying label font; default 1.
- **label.angle**: Angle of text in the label.
- **pch**: End point symbol(s).
- **point.cex**: End points size(s).
- **point.color**: End points color(s).

Details

**label.direction** indicates the direction of event labels from timeline. "downup" and "updown" plots alternating labels; "up" puts all the labels above and "down" below the timeline.

**label.length** could be a single number or a numeric vector. For label directions "downup" and "updown", use between 0 and 0.9, and for "up" and "down", use between 0 and 1.6. For example, `label.length = 0.5` produces all the labels at equal lengths, and `label.length = c(0.5, 0.5, 0.8, 0.8)` repeats the sequence of lengths.

The positions for **label.position** are 1: below 2: left 3: above 4: right.
### Default down-up labels
```r
timelineS(mj_life, main = "Life of Michael Jackson")
```

### Labels above timeline and other change in aesthetics
```r
timelineS(mj_life, main = "Life of Michael Jackson", label.direction = "up", label.length = c(0.2,0.8,0.4,1.2), label.position = 3, line.color = "blue", label.color = "blue", point.color = "blue", pch = "+")
```
Index

* durCalc
  life_exp, 6
* durPlot,
  life_exp, 6
* durSummary,
  life_exp, 6
* timelineGroup
  life_country, 5

axis.Date, 9
durCalc, 2, 4, 5, 9
durPlot, 3, 3, 5, 9
durSummary, 3, 4, 5

life_country, 5
life_exp, 6
mj_life, 6
seq.Date, 8
timelineG, 4, 6, 9
timelineS, 4, 7, 7