Package 'RankingProject'

January 29, 2020

Type Package

Title The Ranking Project: Visualizations for Comparing Populations

Version 0.2.0

Date 2020-01-28

Description Functions to generate plots and tables for comparing independently-sampled populations. Companion package to ``A Primer on Visualizations for Comparing Populations, Including the Issue of Overlapping Confidence Intervals'' by Wright, Klein, and Wieczorek (2019) <DOI:10.1080/00031305.2017.1392359> and ``A Joint Confidence Region for an Overall Ranking of Populations'' by Klein, Wright, and Wieczorek (2020, in press).

License GPL-2

LazyData TRUE

Depends R (>= 2.10)

Suggests knitr (>= 1.15.1), rmarkdown (>= 0.9.6), tikzDevice (>= 0.10-1)

VignetteBuilder knitr

URL http://github.com/civilstat/RankingProject

BugReports http://github.com/civilstat/RankingProject/issues

RoxygenNote 7.0.2

NeedsCompilation no

Author Jerzy Wieczorek [cre, aut],
Joel Beard [ctb],
Andy Liaw [ctb],
Robert Gentleman [ctb],
Martin Maechler [ctb]

Maintainer Jerzy Wieczorek <jerzywieczorek@gmail.com>

Repository CRAN

Date/Publication 2020-01-29 19:50:05 UTC

R topics documented:

RankingProject	*The Ranking Project: Visualizations for Comparing Populations*

Description

Functions to generate plots and tables for comparing independently-sampled populations. Companion package to "A Primer on Visualizations for Comparing Populations, Including the Issue of Overlapping Confidence Intervals" (Wright, Klein, and Wieczorek, 2017, *The American Statistician*, in press). See the Intro vignette (html) for an overview and examples: vignette("intro",package = "RankingProject"). See the Primer vignette (pdf) for code which replicates the main figures from the article: vignette("primer",package = "RankingProject").

Details

The "comparison" plots are based on figures and S code from Almond et al. (2000). The present package does not contain a direct modification of their S code, but draws inspiration from it. Their script was originally hosted at Statlib at http://stat.cmu.edu/S/comprB and may still be found at Statlib mirrors such as http://ftp.uni-bayreuth.de/math/statlib/S/comprB.

The code for the "columns" plots is directly based on R's stats::heatmap() function, with minor modifications to remove dendograms and allow the heatmap to be placed inside a larger layout().

References

Almond, R.G., Lewis, C., Tukey, J.W., and Yan, D. (2000). "Displays for Comparing a Given State to Many Others," *The American Statistician*, vol. 54, no. 2, 89-93.

Wright, T., Klein, M., and Wieczorek, J. (2017). "A Primer on Visualizations for Comparing Populations, Including the Issue of Overlapping Confidence Intervals," *The American Statistician*, in press.

RankPlot *Figure containing a plot of ranking data.*

Description

RankPlot creates a figure with a plot of ranking data, from among several options for showing uncertainty in the ranked estimates. This function is meant for use within RankPlotWithTable, which draws a ranking table aligned with this plot of the data in one combined figure.

Usage

```
RankPlot(
  est,
  se,
  names,
  refName = NULL,
  confLevel = 0.9,
  plotType = c("individual", "difference", "comparison", "columns"),
  tiers = 1,
  GH = FALSE,
  Bonferroni = ifelse(plotType == "individual", "none", "demi"),
  tikzText = FALSE,
  cex = 1,
  tickWidth = NULL,
  rangeFactor = 1.2,
  textPad = 0,
  legendX = "topleft",
  legendY = NULL,
  legendText = NULL,
  lwdReg = 1,
  lwdBold = 3,
  thetaLine = 1,
  xlim = NULL
)
```

Arguments

est, se	Vectors containing the point estimate and its standard error for each area.
names	Vector containing the name of each area. Abbreviations may be preferable to full names (e.g. "CO" instead of "Colorado") since these names will be displayed directly on the plot.
refName	String containing the name of the reference area; must be one of the values in names. Required for plotType = c("difference","comparison"). Optional for plotType = "individual" (where it only determines the row above/below which the names are plotted to the right/left of the intervals; if unspecified, defaults to median rank); or for plotType = "columns" (where it selects one column to be highlighted by vertical lines, if specified).

confLevel	Number between 0 and 1: confidence level for individual (uncorrected) hypothesis tests and/or confidence intervals. E.g. with `plotType = "individual"`, `confLevel = 0.9` will plot individual 90% confidence intervals. If using `GH = TRUE` and/or `Bonferroni != "none"`, the Goldstein-Healy and/or Bonferroni corrections will be applied to the `confLevel` baseline.
plotType	Which type of ranking plot to use. See vignettes for examples and details. • `"individual"` is used for usual individual confidence intervals, with or without Goldstein-Healy adjustment and/or (demi or full) Bonferroni corrections. • `"difference"` shows confidence intervals for the differences between the reference area `refName` and all other areas. • `"comparison"` also compares the reference area `refName` to all others, but using the "comparison intervals" of Almond et al. (2000). • `"columns"` plots a grid of shaded columns, where each column uses shading to report demi-Bonferroni-corrected significance tests for comparing the reference area (labeled at the bottom of the column) with all other areas.
tiers	Numeric, either 1 for usual confidence intervals, or 2 for two-tiered intervals. 2 can only be used with `plotType = "individual"`, when either `GH = TRUE` or `Bonferroni != "none"` or both. In that case, the "inner tiers" run between each interval's cross-bars, and the "outer tiers" run past the cross-bars all the way to the ends of each interval. One of the tiers will show uncorrected `confLevel*100%` confidence intervals, and the other tier will show the Golstein-Healy and/or Bonferroni adjusted intervals. A legend will show which tier is which; usually Goldstein-Healy alone gives shorter intervals (inner tier), but Bonferroni corrections make them into longer intervals (outer tier).
GH	Logical, for whether or not to plot adjusted confidence intervals at an "average" `confLevel*100%` confidence level as in Goldstein and Healy (1995). Can only be used with `plotType = "individual"`.
Bonferroni	Whether and how to correct for multiple comparisons by a Bonferroni correction to the confidence level of the tests or intervals. `"none"` performs no correction; `"demi"` corrects for comparing one reference area to all `n-1` other areas; and `"full"` corrects for comparing all possible `choose(n-1,2)` pairs of areas. If `GH = TRUE`, the Goldstein-Healy adjustment is performed first, and any Bonferroni correction is applied afterwards. Settings `"none"` and `"full"` can only be used with `plotType = "individual"`; all other plot types use the setting `"demi"`.
tikzText	Logical, for whether or not to format text for tikz plotting.
cex	**C**haracter **ex**pansion factor for the points use to plot each area's point estimate, and for the text used to plot each area's name next to its interval.
tickWidth	Numeric height of the cross-bars on interval endpoints (or inner tiers, if `tiers = 2`). The function tries to leave a reasonable amount of space between intervals plotted in different rows, but sometimes it may help to adjust `tickWidth` manually.
rangeFactor	Numeric multiple by which to expand the range of the data when setting the x-axis limits. The function tries to leave sufficient room for plotting margins

	of error and names next to each area, but sometimes it may help to adjust `rangeFactor` manually.
textPad	Numeric amount by which to shift the text of `names` past the interval endpoints when plotting. Positive values shift outwards (towards the edges of the plot); negative values shift inwards.
legendX, legendY	
	The x and y co-ordinates used to position the legend; see [legend](#) for details on specifying x by keyword.
legendText	String, or string vector, with legend text. By default, each plot type adds informative legend text, but the user may override. To remove legends entirely, set `legendText=NA`.
lwdReg	Positive number for the line width of regular lines. Used for all intervals when `plotType = "individual"`, or for intervals not significantly different from the reference area when `plotType = c("difference","comparison")`.
lwdBold	Positive number for the line width of bold lines. Used for intervals significantly different from the reference area when `plotType = c("difference","comparison")`.
thetaLine	Number for how many lines below bottom axis to display "theta" or other default x-axis labels (which depend on `plotType`).
xlim	Vector of 2 numbers for x-axis limits. If `NULL`, will be automatically set using range of data expanded by `rangeFactor`.

Details

Users may wish to modify this code and write their own plot function, which can be swapped into `figureFunction` within [RankPlotWithTable](#). Be aware that RankPlotWithTable uses [layout](#) to arrange the table and plot side-by-side, so `layout` cannot be used within a new `figureFunction`.

See Goldstein and Healy (1995) for details on the "average" confidence level procedure used when `GH = TRUE`. See Almond et al. (2000) for details on the "comparison intervals" procedure.

References

Almond, R.G., Lewis, C., Tukey, J.W., and Yan, D. (2000). "Displays for Comparing a Given State to Many Others," *The American Statistician*, vol. 54, no. 2, 89-93.

Goldstein, H. and Healy, M.J.R. (1995). "The Graphical Presentation of a Collection of Means," *JRSS A*, vol. 158, no. 1, 175-177.

See Also

[RankPlotWithTable](#) and [RankTable](#).

Examples

```
# Plot of 90% confidence intervals for differences
# between each state and Colorado, with demi-Bonferroni correction,
# for US states' mean travel times to work, from the 2011 ACS
data(TravelTime2011)
with(TravelTime2011,
```

```
    RankPlot(est = Estimate.2dec, se = SE.2dec,
             names = Abbreviation, refName = "CO",
             confLevel = 0.90, cex = 0.6,
             plotType = "difference"))
```

RankPlotWithTable   *Figure containing aligned table and plot of ranking data.*

Description

RankPlotWithTable aligns a table of ranking data with a plot of the data, in one combined fig-
ure. See RankTable and RankPlot for details about the default table and plot functions, including
arguments that can be passed to those functions.

Usage

```
RankPlotWithTable(
  tableParList,
  plotParList,
  tableFunction = RankTable,
  plotFunction = RankPlot,
  tableWidthProp = 3/8,
  tikzText = FALSE,
  annotRefName = NULL,
  annotRefRank = NULL,
  annotX = 0
)
```

Arguments

tableParList	A required named list of arguments that will be passed to tableFunction using do.call(). The default tableFunction is RankTable, which requires at least these four arguments: ranks, names, est, se.
plotParList	A required named list of arguments that will be passed to plotFunction using do.call(). The default plotFunction is RankPlot, which requires at least these three arguments: est, se, names.
tableFunction	The function to use for plotting a table of the data on the left-hand side of the layout. Default is RankTable.
plotFunction	The function to use for plotting a figure of the data on the right-hand side of the layout. Default is RankPlot.
tableWidthProp	A number between 0 and 1, for what proportion of the layout's width should be used to plot the table. The remaining proportion 1-tableWidthProp is used to plot the figure.
tikzText	Logical, formats text for tikz plotting if TRUE.

annotRefName, annotRefRank

> Optional rank and name of the reference area, for adding an extra annotation below the figure created by `plotFunction`. Currently centered at 0 on x-axis, so only useful when `plotType = "difference"`. If provided, the list must contain two required named elements (`refFullName` and `refRank`, the reference area's name and rank)

annotX   A number, showing where on the x-axis to center the annotation if `annotRefName` and `annotRefRank` are not `NULL`.

Details

Users may write their own table and plot functions to swap into `tableFunction` and `plotFunction`. Be aware that RankPlotWithTable uses [layout](#) to arrange the table and plot side-by-side, so `layout` cannot be used within either `tableFunction` or `plotFunction`. This can also cause trouble for using the `lattice` package within `plotFunction`.

See Also

[RankPlot](#) and [RankTable](#).

Examples

```
# Table with plot of individual 90% confidence intervals
# for US states' mean travel times to work, from the 2011 ACS
data(TravelTime2011)
tableParList <- with(TravelTime2011,
  list(ranks = Rank, names = State,
       est = Estimate.2dec, se = SE.2dec,
       placeType = "State"))
plotParList <- with(TravelTime2011,
  list(est = Estimate.2dec, se = SE.2dec,
       names = Abbreviation,
       confLevel = .90, plotType = "individual", cex = 0.6))
RankPlotWithTable(tableParList = tableParList,
  plotParList = plotParList)

# Illustrating the use of annotRefName and annotRefRank:
# Table with plot of 90% confidence intervals for differences
# between each state and Colorado, with demi-Bonferroni correction
plotParList$plotType <- "difference"
plotParList$refName <- "CO"
RankPlotWithTable(tableParList = tableParList,
  plotParList = plotParList, annotRefName = "Colorado",
  annotRefRank = TravelTime2011$Rank[which(TravelTime2011$Abbreviation == "CO")])
```

RankTable   *Figure containing a table of ranking data.*

Description

`RankTable` creates a figure with a table of ranking data. This may not look very good plotted on its own. Rather, it is meant for use within `RankPlotWithTable`, which draws this table aligned with a plot of the data in one combined figure.

Usage

```
RankTable(
  ranks,
  names,
  est,
  se,
  placeType = "State",
  col1 = 0.15,
  col2 = 0.6,
  col3 = 0.85,
  col4 = 1,
  textPos = 2,
  titleCex = 0.9,
  titleLift = 1.5,
  contentCex = 0.7,
  columnsPlotRefLine = NULL,
  tikzText = FALSE
)
```

Arguments

`ranks`	Vector containing the rank of each area.
`names`	Vector containing the name of each area.
`est, se`	Vectors containing the point estimate and its standard error for each area. See vignettes for examples of using `formatC` to turn the numeric estimates or SEs into strings, for printing with a consistent number of decimal places.
`placeType`	String, naming the type of places or units being ranked.
`col1, col2, col3, col4`	
	Numeric values between 0 and 1, showing where each column's right-hand-side endpoint is along the table's width. In other words, `colJ` should be the fraction of the table's total width at which the Jth column should end, if using default of right-aligned columns (unless `textPos != 2`). Use `col4 = 1` unless you want the table to be narrower than the space available, or unless you switch to centered or left-aligned columns.
`textPos`	Passed to `pos` argument of `text`. Default of 2 ensures each column of text is right-justified.

titleCex	**C**haracter **ex**pansion factor for column titles.
titleLift	Numeric value for how many row-heights to raise column titles above top row of column contents.
contentCex	**C**haracter **ex**pansion factor for column contents (all column text except the titles).
columnsPlotRefLine	
	Optional numeric value. If not NULL, how many row-heights below bottom row of column contents to print the phrase "Reference State:" (or "Reference <placeType>:") as a label for bottom row of columns plot.
tikzText	Logical, for whether or not to format text for tikz plotting.

Details

This function is currently hardcoded to give a table with four columns, with given column names. Users may wish to modify this code and write their own table function, which can be swapped into tableFunction within [RankPlotWithTable](). Be aware that RankPlotWithTable uses [layout]() to arrange the table and plot side-by-side, so layout cannot be used within a new tableFunction.

See Also

[RankPlotWithTable]() and [RankPlot]().

Examples

```
# Table of US states' mean travel times to work, from the 2011 ACS
data(TravelTime2011)
# Just as inside RankPlotWithTable(),
# we have to set par(xpd=TRUE)
# and adjust the plotting margins
oldpar <- par(no.readonly = TRUE)
oldmar <- par('mar')
par(xpd=TRUE, mar=c(oldmar[1],0,oldmar[3],0))
with(TravelTime2011,
     RankTable(ranks = Rank, names = State,
               est = Estimate.2dec, se = SE.2dec,
               placeType = "State"))
par(oldpar)
```

TravelTime2011	*Mean travel times to work, from 2011 ACS.*

Description

A dataset containing the estimated mean travel time (in minutes) to work of workers 16 years and over who did not work at home (henceforth "mean travel time to work"), and its estimated standard error, for each of the 51 states (including Washington, D.C.), from the 2011 American Community Survey.

Usage

```
TravelTime2011
```

Format

A data frame with 51 rows and 7 variables:

Rank state rank, by estimated mean travel time, where 1 is lowest travel time and 51 is highest

State full name of the state

Estimate.2dec estimated mean travel time, in minutes

SE.2dec estimated standard error of the estimated mean travel time, in minutes

Abbreviation postal abbreviation of the state

Region factor variable for geographic region of the state: Northeast, South, Midwest, West, Pacific

FIPS Federal Information Processing Standard (FIPS) code of the state; may be useful for linking with other datasets

Source

https://www.census.gov/

TravelTime2011.1dec	*Mean travel times to work, from 2011 ACS, rounded to 1 decimal place.*

Description

A dataset containing the estimated mean travel time (in minutes) to work of workers 16 years and over who did not work at home (henceforth "mean travel time to work"), and its estimated Margin of Error at the 90% confidence level, for each of the 51 states (including Washington, D.C.), from the 2011 American Community Survey.

Usage

```
TravelTime2011.1dec
```

Format

A data frame with 51 rows and 7 variables:

Rank state rank, by estimated mean travel time, where 1 is lowest travel time and 51 is highest

State full name of the state

Estimate.1dec estimated mean travel time, in minutes

MOE.1dec estimated Margin of Error (at the 90% confidence level) of the estimated mean travel time, in minutes

Abbreviation postal abbreviation of the state

Region factor variable for geographic region of the state: Northeast, South, Midwest, West, Pacific

FIPS Federal Information Processing Standard (FIPS) code of the state; may be useful for linking with other datasets

Details

Due to rounding, some ranks are tied in this version of the data. Also note that this dataset reports Margins of Error (MoEs) instead of standard errors.

Source

https://www.census.gov/

Index