Package ‘textBoxPlacement’

March 16, 2023

Version 1.0

Date 2023-03-15

Title Compute a Non-Overlapping Layout of Text Boxes to Label Multiple Overlain Plots

Maintainer Barry Zeeberg <barryz2013@gmail.com>

Author Barry Zeeberg [aut, cre]

Depends R (>= 4.2.0)

LazyData true

Imports graphics, stats, grDevices

Description Compute a non-overlapping layout of text boxes to label multiple overlain curves. For each curve, iteratively search for an adjacent x,y position for the text box that does not overlap with the other curves. If this process fails, then offsets are computed to add to the y values for each curve, that results in sufficient space to add all of the text labels.

License GPL (>= 2)

Encoding UTF-8

VignetteBuilder knitr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

RoxygenNote 7.2.3

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2023-03-16 19:20:02 UTC

R topics documented:

`axesRange` *axesRange*

Description

compute correct axes ranges for a set of overlain plots

Usage

```
axesRange(xList)
```

Arguments

`xList`	list of numerical vectors containing axis points

Value

xlim

Examples

```
axesRange(yList1)
```

`labs` *textBoxPlacement data sets*

Description

textBoxPlacement data sets

Usage

```
data(labs)
```

`permInd` *permInd*

Description

compute the indices of sorting yrange in decreasing order

Usage

```
permInd(yrange)
```

Arguments

`yrange`	numeric vector (max-min) for vector of y values

Value

returns the integer vector indices of sorting yrange in decreasing order

Examples

```
permInd(yrange(yList1))
```

`permuteCurves` *permuteCurves*

Description

reorder multiple curves so that the curve with the largest y range is on the bottom of the staggered graph

Usage

```
permuteCurves(xList, yList, tList, textBoxHeights, permInd)
```

Arguments

`xList`	list whose components are numeric vectors of the x values for overlaid curves
`yList`	list whose components are numeric vectors of the y values for overlaid curves
`tList`	list of character string texts to insert in plot
`textBoxHeights`	return value of textBoxUserUnits()
`permInd`	return value of permInd()

Value

returns a list whose (re-ordered) components are:

- xList a list of numeric vector for x values
- yList a list of numeric vector for y values - re-ordered and offset-adjusted
- tList a list of character strings for text boxes to label the curves
- offset a numeric vector offset to add to each staggered curve
- ystart a numeric vector of starting positions

Examples

```
units<-textBoxUserUnits(textList,yrange(yList1),verbose=TRUE)
permuteCurves(xList1,yList1,textList,units,permInd(yrange(yList1)))
```

`positionTextBox` *positionTextBox*

Description

compute x and y coordinates for placement of text box based upon the values of the function to avoid running into the graph line

Usage

```
positionTextBox(text, x, y, xPos, adj, nApprox = 10, reallyText)
```

Arguments

`text`	character string text to insert in plot
`x`	numeric vector of x values
`y`	numeric vector of y values
`xPos`	numeric x position for text box
`adj`	numeric vector param passed to text()
`nApprox`	integer number of intervals to interpolate between x data points
`reallyText`	Boolean if TRUE then execute text() command

Value

returns a list c(ymin,ymax,strheight(text),xPos1)

Examples

```
x<-1:10
y<-1:10
plot(x,y,type="l")
positionTextBox(text="TEXT ME",x=x,y=y,xPos=1,
  adj=c(0,0),nApprox=10,reallyText=TRUE)
```

`positionTextBoxDriver` *positionTextBoxDriver*

Description

Driver to compute x and y coordinates for placement of text box based upon the values of the function to avoid running into the graph line and avoid overlapping with other overlay curves

Usage

```
positionTextBoxDriver(
  text,
  x,
  y,
  xPos,
  nApprox = 10,
  xList,
  yList,
  stag = FALSE,
  offset = 0,
  verbose
)
```

Arguments

`text`	character string text to insert in plot
`x`	numeric vector of x values
`y`	numeric vector of y values
`xPos`	numeric x position for text box
`nApprox`	integer number of intervals to interpolate between x data points
`xList`	list whose components are numeric vectors of the x values for overlaid curves
`yList`	list whose components are numeric vectors of the y values for overlaid curves
`stag`	Boolean TRUE if this plot has staggering added to curves
`offset`	numeric vector of offsets added to each curve
`verbose`	Boolean if TRUE print informative or diagnostic messages to console

Value

returns a numeric vector c(yReal,yAdj) where xTry is an x value at which the text box will fit without overlapping another overlay curve, or returns -1000000 for failure

Examples

```
positionTextBoxDriver(text="TEXT ME",x=xList1[[1]],y=yList1[[1]],
  xPos=1,nApprox=10,xList=xList1[-1],yList=yList1[-1],stag=FALSE,offset=0,verbose=TRUE)
```

`positionTextBoxDriverDriver`

positionTextBoxDriverDriver

Description

Driver to compute x and y coordinates for placement of text box based upon the y values of the function to avoid running into the graph line and avoid overlapping with other overlay curves

Usage

```
positionTextBoxDriverDriver(
  xList,
  yList,
  textList,
  xPos,
  nApprox = 10,
  labs,
  stag = FALSE,
  offset = 0,
  ystart,
  ylim,
  verbose
)
```

Arguments

`xList` list whose components are numeric vectors of the x values for overlaid curves

`yList` list whose components are numeric vectors of the y values for overlaid curves

`textList` list of character string texts to insert in plot

`xPos` numeric vector x position for text box

`nApprox` integer number of intervals to interpolate between x data points

`labs` list of labels annotating a graph

- main character string main title
- xlab character string x axis label
- ylab character string y axis label

`stag` Boolean TRUE if this plot has staggering added to curves

`offset` numeric vector of offsets added to each curve

`ystart` numeric vector of starting positions

`ylim` numeric vector ylim parameter for plot()

`verbose` Boolean if TRUE print informative or diagnostic messages to console

Details

if the length of the return value is not 0, then additional processing might be needed for the bad curves, such as adding an offset to their y values, plotting them in a different color or symbol, and keying them to a second y axis on the right of the graph

Value

returns a vector of integers indicating curves whose text box could not be drawn

Examples

```
# the text box for the second curve cannot fit,
# as it is sandwiched between two curves that are too close

plot(xList1[[1]],yList1[[1]],type="l")
positionTextBoxDriverDriver(xList=xList1,yList=yList1,
  textList=textList,xPos=c(1,1,1),nApprox=10,labs=labs,
  stag=FALSE,offset=0,ystart=0,ylim=axesRange(yList1),verbose=TRUE)
```

positionTextBoxDriverDriverDriver

positionTextBoxDriverDriverDriver

Description

Driver invoke positionTextBoxDriverDriver() using the original user input data If that fails to produce an uncluttered plot, then invokes stagger() to reshape the data before re-running positionTextBoxDriverDriver().

Usage

```
positionTextBoxDriverDriverDriver(
  xList,
  yList,
  textList,
  xPos,
  nApprox = 10,
  labs,
  sortB,
  verbose = FALSE
)
```

Arguments

xList (optional) list whose components are numeric vectors of the x values for overlaid curves

yList list whose components are numeric vectors of the y values for overlaid curves

textList (optional) list of character string texts to insert in plot

xPos (optional) numeric vector x position for text box

nApprox (optional) integer number of intervals to interpolate between x data points

labs (optional) list of labels annotating a graph

- main character string main title
- xlab character string x axis label
- ylab character string y axis label

sortB Boolean if TRUE staggered curves reordered, with largest range curve on bottom of graph

verbose Boolean if TRUE print informative or diagnostic messages to console

Details

if xList or textList is missing, it is constructed from elements in yList

hint: to prevent conflicts, run the following line manually before running positionTextBoxDriverDriverDriver()

rm(list=ls())

see https://stackoverflow.com/questions/27253376/different-results-from-rscript-and-r-cmd-batch

Value

returns no values, but has side effect of generating a graph.

Examples

```
# There is not enough space for text boxes in original graph.
# The package automatically adds offsets to the curves,
# keeping the curves in the original order,
# and successfully adds text boxes

positionTextBoxDriverDriverDriver(xList=xList1,yList=yList1,
  textList=textList,xPos=c(1,1,1),nApprox=10,labs=labs,sortB=FALSE)

# data set contains some negative values

positionTextBoxDriverDriverDriver(xList=xList2,yList=yList2,
  textList=textList,xPos=c(1,1,1),nApprox=10,labs=labs,sortB=FALSE)

# show the difference when we sort the order of the curves
# to position the curve with the largest range on the bottom

positionTextBoxDriverDriverDriver(xList=xList2,yList=yList2,
  textList=textList,xPos=c(1,1,1),nApprox=10,labs=labs,sortB=TRUE,verbose=TRUE)
```

stagger *stagger*

Description

unclutter the overlay plot by staggering the graphs

Usage

```
stagger(xList, yList, tList, sortB = FALSE, verbose)
```

Arguments

`xList`	list whose components are numeric vectors of the x values for overlaid curves
`yList`	list whose components are numeric vectors of the y values for overlaid curves
`tList`	list of character string texts to insert in plot
`sortB`	Boolean if TRUE staggered curves reordered, with largest range curve on bottom of graph
`verbose`	Boolean if TRUE print informative or diagnostic messages to console

Details

In order to unclutter the overlay plot, we need to stagger the graphs the offset for each graph will be the sum of the max values for all of the preceding graphs. So the stack of staggered graphs will have max y (ie, ymax) equal to the sum of the max's.

Value

returns a list whose components are:

- textBoxHeights return value of textBoxUserUnits()
- permInd return value of permInd()
- xlim numeric vector parameter for plot()
- ylim numeric vector parameter for plot()

Examples

```
# demonstrate effect of sorting the curves

plot.new()

stagger(xList2,yList2,textList,sortB=FALSE,verbose=TRUE)

stagger(xList2,yList2,textList,sortB=TRUE,verbose=TRUE)
```

textBoxUserUnits *textBoxUserUnits*

Description

compute the heights of the text boxes in user units

Usage

```
textBoxUserUnits(tList, yrange, verbose)
```

Arguments

`tList`	a list of character strings for text boxes to label the curves
`yrange`	numeric vector (max-min) for vector of y values
`verbose`	Boolean if TRUE print informative or diagnostic messages to console

Value

numeric vector of the heights of the text boxes in user units

Examples

```
textBoxUserUnits(textList,yrange(yList1),verbose=TRUE)
```

textList *textBoxPlacement data sets*

Description

textBoxPlacement data sets

Usage

```
data(textList)
```

`xlim` *xlim*

Description

compute the numeric vector xlim for a set of curves

Usage

```
xlim(xList)
```

Arguments

`xList`	list whose components are numeric vectors of the x values for overlaid curves

Value

numeric vector xlim

Examples

```
xlim(xList1)
```

`xList1` *textBoxPlacement data sets*

Description

textBoxPlacement data sets

Usage

```
data(xList1)
```

`xList2` *textBoxPlacement data sets*

Description

textBoxPlacement data sets

Usage

```
data(xList2)
```

xPosCheck *xPosCheck*

Description

is the value of xPos within a valid range?

Usage

```
xPosCheck(xPos, xList, verbose)
```

Arguments

`xPos`	integer specifying x position to try to place text box
`xList`	list whose components are numeric vectors of the x values for overlaid curves
`verbose`	Boolean if TRUE print informative or diagnostic messages to console

Value

numeric vector valid values of xPos

Examples

```
# replace incorrect xPos with reasonable value

xPosCheck(c(1,1,-5),xList2,verbose=TRUE)
```

ylim *ylim*

Description

compute the numeric vector ylim

Usage

```
ylim(yList, yrange, textBoxHeights, sortB, permInd)
```

Arguments

`yList`	list whose components are numeric vectors of the y values for overlaid curves
`yrange`	numeric vector (max-min) for vector of y values
`textBoxHeights`	return value of textBoxUserUnits()
`sortB`	Boolean if TRUE staggered curves are reordered, with largest range curve on bottom of graph
`permInd`	return value of permInd()

Value

returns a numeric vector ylim

Examples

```
# demonstrate effect of sorting the curves

plot.new()
ylim(yList1,yrange(yList1),textBoxUserUnits(textList,yrange(yList1),verbose=TRUE),
  FALSE,permInd(yrange(yList1)))
```

yList1 *textBoxPlacement data sets*

Description

textBoxPlacement data sets

Usage

```
data(yList1)
```

yList2 *textBoxPlacement data sets*

Description

textBoxPlacement data sets

Usage

```
data(yList2)
```

`yrange` *yrange*

Description

compute the staggered y values for the overlay plot

Usage

```
yrange(yList)
```

Arguments

`yList`	list whose components are numeric vectors of the y values for overlaid curves

Value

numeric vector yrange (max-min) for vector of y values

Examples

```
yrange(yList1)
```

Index