

graph.rdaTest {sonarX}

R Documentation

Display the rdaTest output

Description

This is highly flexible drawing device

Usage

```
graph.rdaTest(  
  rdaTest, xax=1, yax=2, scaling=1, plot.type="notchosen",  
  binary=NULL, groups=NULL,  
  centroids=TRUE, ell=TRUE, sites=TRUE, stars=FALSE,  
  label.env=TRUE, label.spc=TRUE, label.site=TRUE,  
  alpha=0.95, saveplot=FALSE, path=NULL, width=6, height=6,  
  xlim=NULL, ylim=NULL, lwd=1, len=0.1,  
  lty.env=1, lty.spc=1, lty.ell=1, lty.stars=1, lty.axis=2,  
  col.env="blue", col.spc="red", col.groups=NULL, col.ell="black",  
  col.ctd="black",  
  pos.env=NULL, pos.bin=4, pos.site=NULL, pos.ctd=4,  
  cex.env=1, cex.spc=1, cex.site=1, cex.ctd=1, cex.axis=1,  
  pch.bin=21, pch.site=21, pch.ctd=23,  
  ell.axis=FALSE, mul.spc=0.90, mul.env=0.90, mul.text=0.10,  
  mai.perc=NULL)
```

Arguments

<code>rdaTest</code>	The rdaTest output.
<code>xax</code>	The matrix columns to be used as the abscissa in the rda graph. These value can be negative; if so, the plot axes will be reversed.
<code>yax</code>	The matrix columns to be used as the ordinate in the rda graph. These value can be negative; if so, the plot axes will be reversed.
<code>scaling</code>	Allows the user to choose between the two scaling types, 1 or 2.
<code>plot.type</code>	Either "F" = site scores, or "z" = fitted site scores.
<code>binary</code>	A vector specifying which of the environmental variables are binary. The data points possessing the characteristics in question will be averaged and represented by a symbol instead of an arrow, as in Canoco.
<code>groups</code>	A vector describing the group assignment of all objects.
<code>centroids</code>	A logical value (TRUE or FALSE) defining if centroids are to be drawn.
<code>ell</code>	A logical value (TRUE or FALSE) defining if confidence ellipses around groups of points are to be drawn. Group membership is described in "groups". See examples below.
<code>sites</code>	A logical value (TRUE or FALSE) defining if the sites are to be drawn on the plot.
<code>stars</code>	A logical value (TRUE or FALSE) defining if stars (lines linking each site to its respective

	group centroid) are to be drawn on the plot.
<code>label.env</code>	A logical value (TRUE or FALSE) defining if the environmental variable names are to be printed on the plot.
<code>label.spc</code>	A logical value (TRUE or FALSE) defining if the species names are to be printed on the plot.
<code>label.site</code>	A logical value (TRUE or FALSE) defining if the site names are to be printed on the plot.
<code>alpha</code>	The confidence region of the ellipses (e.g.: 0.90, 0.80, etc.) Default value: 0.95.
<code>saveplot</code>	The plot can be saved to a .pdf file. If set to FALSE, the plot will only be plotted in a window. If set to TRUE, no graph will be plotted on the screen, but a pdf file will be created.
<code>path</code>	Complete path to the file in which the plot will be saved; example: "~/Desktop/toto.pdf".
<code>width</code>	Plot width in inches.
<code>height</code>	Plot height in inches.
<code>xlim</code>	Vectors describing the x axes minimum and maximum of the plotted region.
<code>ylim</code>	Vectors describing the y axes minimum and maximum of the plotted region.
<code>lwd</code>	Line width for the axes, arrows, and ellipses.
<code>len</code>	Length of the arrow heads.
<code>lty.env</code>	The environmental variables arrow drawing type.
<code>lty.spc</code>	The species variables arrow drawing type.
<code>lty.ell</code>	The drawing type for ellipses; see <code>lty</code> in the help documentation file of par .
<code>lty.stars</code>	The line type for stars. See par .
<code>lty.axis</code>	The drawing type of axes, dotted lines, etc. See par .
<code>col.env</code>	The environmental variables arrow and text colour.
<code>col.spc</code>	The species variables arrow and text colour.
<code>col.groups</code>	By default, each group is assigned a different colour, so that sites and group centroids have an identifiable colour, which is a function of their group membership.
<code>col.ell</code>	Colour of the ellipses.
<code>col.ctd</code>	Colour of the centroids.
<code>pos.env</code>	Position offset of the environmental variable names: 1=bottom, 2=left, 3=top, 4=right.
<code>pos.bin</code>	Position offset of the binary environmental variables names: 1=bottom, 2=left, 3=top, 4=right.
<code>pos.site</code>	Position offset of the site names: 1=bottom, 2=left, 3=top, 4=right.
<code>pos.ctd</code>	Position offset of the group centroid names: 1=bottom, 2=left, 3=top, 4=right.
<code>cex.env</code>	Font size for environmental variable labels.
<code>cex.spc</code>	Font size for species names.
<code>cex.site</code>	Font size for sites labels.
<code>cex.ctd</code>	Font size for centroids labels.
<code>cex.axis</code>	Font size for axis names.
<code>pch.bin</code>	Symbol used to represent binary variables. See <code>pch</code> definition in the points function.
<code>pch.site</code>	Symbol used to represent sites. See <code>pch</code> definition in the points function.
<code>pch.ctd</code>	Symbol used to represent centroids. See <code>pch</code> definition in the points function.
<code>ell.axis</code>	If TRUE, draw the major and minor axes of the ellipses.

<code>mul.spc</code>	The species variable arrows are initially expanded or compressed to fit within the plot borders and be displayed with maximum spread. <code>mul.spc</code> is a multiplier used to modify the arrow proportions. By default, 0.90 is used to give some room for the arrow labels.
<code>mul.env</code>	The same as <code>mul.spc</code> but for the environmental variable.
<code>mul.text</code>	Text position at the end of each arrow, plus a percentage of the arrow length; the default value is 10%.
<code>mai.perc</code>	Allows to increase or decrease the margin size, using a percentage of the total width and height of the graph. That value is between 0 and 1.

Value

A plot is returned.

Author(s)

Sebastien Durand <Sebastien.Durand@UMontreal.ca>

See Also

[rdaTest](#)

Examples

```
#see this-is-escaped-codenormal-bracket138bracket-normal
data(rda.analyse)
graph.rdaTest(s$results[[1]], plot.type = "Z")

#Find groups
toto<-cascadeKM(s$results[[1]][[6]],2,5,iter=9,criterion="calinski")
toto2<-plotKM(toto)
#in this case we take the 5 group solution
graph.rdaTest(s$results[[1]], plot.type = "Z",groups=toto[[1]][,4],label.site=FALSE)

graph.rdaTest(s$results[[1]], plot.type = "Z",groups=toto[[1]][,4], lty.env=0,
  lty.spc=0, ell=FALSE, stars=TRUE, label.site=FALSE,xlim=c(-1,1),ylim=c(-1,1))
```

[Package *sonarX* version 0.1-6 [Index](#)]