

visHexPattern

February 12, 2015

visHexPattern	<i>Function to visualise codebook matrix or input patterns within a supra-hexagonal grid</i>
---------------	--

Description

visHexPattern is supposed to codebook matrix or input patterns within a supra-hexagonal grid.

Usage

```
visHexPattern(sObj, plotType = c("lines", "bars", "radars"), pattern =  
NULL,  
height = 7, margin = rep(0.1, 4), colormap = c("customized", "bwr",  
"jet", "gbr", "wyr", "br", "yr", "rainbow", "wb"), customized.color =  
"red",  
zeropattern.color = "gray", legend.cex = 0.8, newpage = T)
```

Arguments

sObj	an object of class "sMap" or "sTopol" or "sInit"
plotType	the plot type, can be "lines" for line/point graph, "bars" for bar graph, "radars" for radar graph
pattern	By default, it sets to "NULL" for the codebook matrix. It is intended for the user-input patterns, i.e., a matrix with the dimension of nHex x nPattern, where nHex is the number of hexagons and nPattern is the number of elements for each pattern
height	a numeric value specifying the height of device
margin	margins as units of length 4 or 1
colormap	short name for the predefined colormap, and "customized" for custom input (see the next 'customized.color'). The predefined colormap can be one of "jet" (jet colormap), "bwr" (blue-white-red colormap), "gbr" (green-black-red colormap), "wyr" (white-yellow-red colormap), "br" (black-red colormap), "yr" (yellow-red colormap), "wb" (white-black colormap), and "rainbow" (rainbow colormap, that is, red-yellow-green-cyan-blue-magenta). Alternatively, any hyphen-separated HTML color names, e.g. "blue-black-yellow", "royalblue-white-sandybrown", "darkgreen-white-darkviolet". A list of standard color names can be found in http://html-color-codes.info/color-names

customized.color	the customized color for pattern visualisation
zeropattern.color	the color for zero horizontal line
legend.cex	a numerical value giving the amount by which legend text should be magnified relative to the default (i.e., 1)
newpage	logical to indicate whether to open a new page. By default, it sets to true for opening a new page

Value

invisible

Note

The "plotType" includes:

- "lines": line plot. If multiple colors are given, the points are also plotted. When the pattern involves both positive and negative values, zero horizontal line is also shown
- "bars": bar plot. When the pattern involves both positive and negative values, the zero horizontal line is in the middle of the hexagon; otherwise at the top of the hexagon for all negative values, and at the bottom for all positive values
- "radars": radar plot. Each radar diagram represents one pattern, wherein each element value is proportional to the distance from the center. Note, it starts on the right and wind counter-clockwise around the circle

See Also

[sPipeline](#), [visColormap](#)

Examples

```
# 1) generate data with an iid matrix of 1000 x 9
data <- cbind(matrix(rnorm(1000*3,mean=0,sd=1), nrow=1000, ncol=3),
matrix(rnorm(1000*3,mean=0.5,sd=1), nrow=1000, ncol=3),
matrix(rnorm(1000*3,mean=-0.5,sd=1), nrow=1000, ncol=3))
colnames(data) <- c("S1","S1","S1","S2","S2","S2","S3","S3","S3")

# 2) sMap resulted from using by default setup
sMap <- sPipeline(data=data)

# 3) plot codebook patterns using different types
# 3a) line plot
visHexPattern(sMap, plotType="lines")
# 3b) bar plot
visHexPattern(sMap, plotType="bars")
# 3c) radar plot
visHexPattern(sMap, plotType="radars")
# 4) plot user-input patterns using different types
# 4a) generate pattern data with two different groups "S" and "T"
nHex <- sMap$nHex
pattern <- cbind(matrix(runif(nHex*3,min=0,max=1), nrow=nHex, ncol=3),
matrix(runif(nHex*3,min=1,max=2), nrow=nHex, ncol=3))
colnames(pattern) <- c("S1","S2","S3","T1","T2","T3")
```

```
# 4b) for line plot
visHexPattern(sMap, plotType="lines", pattern=pattern,
customized.color="red", zeropattern.color="gray")
# 4c) for bar plot
visHexPattern(sMap, plotType="bars", pattern=pattern,
customized.color=rep(c("red", "green"), each=3))
# 4d) for radar plot
visHexPattern(sMap, plotType="radars", pattern=pattern,
customized.color=rep(c("red", "green"), each=3))
```