Package ‘patchPlot’

February 20, 2015

Version 0.1.5
Date 2013-03-06
Author Pierrick Bruneau
Maintainer Pierrick Bruneau <pbruneau@gmail.com>
Title Scatterplots of image patches
Description Functions to generate scatterplots with images patches
             instead of usual glyphs, with associated utilities.
Depends R (>= 2.10.0), grid, datautils
LazyLoad yes
LazyData yes
License LGPL-3
NeedsCompilation no
Repository CRAN
Date/Publication 2013-03-11 11:55:30

R topics documented:

animatePatches ......................................................... 2
checkGmm .................................................................. 2
colorPlot ................................................................. 3
drawGradient ............................................................ 4
drawPatches ............................................................... 5
drawSinglePatch ........................................................ 6
genHatchData ............................................................. 7
getInterRGB .............................................................. 8
Lab2RGB ................................................................. 9
plotGmmOverlay ........................................................ 10
plotGreyPatches ......................................................... 11
RGB2Lab ................................................................. 12

Index 13

1
animatePatches

Description
Tentative function for animations using grid: animate a black square on a circular path.

Usage
animatePatches(nsecs)

Arguments
nsecs number of seconds to animate. Set with +Inf for infinite loop (interrupt with CTRL+C in command prompt).

Value
NULL

Author(s)
Pierrick Bruneau

See Also
datautils::getTimestamp datautils::getElapsed

Examples
animatePatches()

checkGmm

Description
Checks that the parametrized GMM objects has the expected structure.

Usage
checkGmm(obj)

Arguments
obj GMM object to be checked (see VBmix::newGmm for format definition)
**Value**

TRUE if the correct structure is found, FALSE else.

**Author(s)**

Pierrick Bruneau

**See Also**

plotGmmOverlay

**Examples**

```r
# set a gaussian mixture model
gaussmod <- list()
gaussmod$w <- c(0.5, 0.5)
gaussmod$mean <- list(c(-3,0), c(3,0))
gaussmod$cov <- list(diag(2), diag(2))
checkGmm(gaussmod)
```

**Description**

Plot a data.frame with color cells.

**Usage**

`colorPlot(x, zeroColor, oneColor)`

**Arguments**

- `x`: data.frame to plot, taking values in [0,1] : the associated colors are then taken on the gradient from zeroColor to oneColor.
- `zeroColor`: the "zero bound" color, in any valid R color format (e.g. "yellow", "#00FF00", ...) 
- `oneColor`: the "one bound" color, in any valid R color format (e.g. "yellow", "#00FF00", ...) 

**Value**

NULL

**Author(s)**

Pierrick Bruneau
References

Inspired from myImagePlot (http://www.phaget4.org/R/image_matrix.html)

See Also

gInterRGB

Examples

dat <- matrix(runif(16), nrow=4)
dat <- as.data.frame(dat)
rownames(dat) <- colnames(dat) <- c("A", "B", "C", "D")
colorPlot(dat, "green", "red")

drawGradient

drawGradient

drawGradient

drawGradient

drawGradient

drawGradient

Description

Draw the gradient between Lab colors, as a large color patch on white background. The gradient is interpolated in Lab space, i.e. is perceptually uniform (see reference).

Usage

drawGradient(col1, col2, rgb=TRUE, numcols=200)

Arguments

col1 numeric vector of 3 values, either specifying RGB in [0,1] values, or Lab. Starting color for the gradient.
col2 numeric vector of 3 values, either specifying RGB in [0,1] values, or Lab. Ending color for the gradient.rgb logical, determines if col1 and col2 are in RGB or Lab color space.
numcols number of steps in the gradient (linear) interpolation.

Value

NULL

Author(s)

Pierrick Bruneau

References

Ware, C. (2004) Information visualization, from cognitive, perceptive, and visual cues point of view.
**drawPatches**

**See Also**

drawPatches

**Examples**

drawGradient(c(1,0,0), c(0,1,0)) # gradient of one the human perceptuals channels (see reference)

drawPatches

drawPatches

**Description**

Draw a scatterplot with images patches as placeholders for the points.

**Usage**

drawPatches(data, patches, patchSize=0.05, alpha=0.5, patchNpix=20, highlight=numeric(0), labels=rep("1", length(highlight)))

**Arguments**

data nx2 matrix of points positions in the 2D space.
patches rgbpatch object, with n rows in each channel specifying n elements. The values of a row-element are mapped to pixel intensities. Patch width is deduced from patchNpix argument.
patchSize patch size as a ratio of the whole window.
alpha If highlight is empty, this is the alpha value for all patches. Else if a set is highlighted, highlighted elements have this alpha value, and other elements have alpha/20.
patchNpix expected size of the image patches sides. Should match the specification of patches object.
highlight numeric vector taking values in 1..n, indicating elements in data to highlight. Patches are associated to alpha=1 if highlighted, 0.04 if not. If no highlight is specified, all patches have alpha=1.
labels text labels associated to highlighted elements. Defaults to "1" for all highlighted elements.

**Value**

NULL

**Author(s)**

Pierrick Bruneau
See Also
drawSinglePatch drawGradient

Examples
data <- matrix(runif(20), ncol=2)
coldata <- matrix(runif(30), ncol=3)
hatchdata <- matrix(runif(30), ncol=3)
patches <- genHatchData(coldata, hatchdata)
drawPatches(data, patches)

drawSinglePatch  drawSinglePatch

Description
Probes one RGB or Lab color, by diisplaying a large colored patch on a white background.

Usage
drawSinglePatch(vec, rgb=TRUE)

Arguments
vec    numeric vector of the color to draw (RGB in [0,1], or Lab values).
rgb    logical value that indicates RGB values if TRUE (Lab otherwise).

Value
NULL, and opens a new X11 window.

Author(s)
Pierrick Bruneau

See Also
drawGradient drawPatches

Examples
drawSinglePatch(c(0.3, 0.05, 0.7))
Description

Generate a collection of n RGB patches (square images with uniform background and hatching).

Usage

```r
genHatchData(rgbcol, rgbhatch, side=20)
```

Arguments

- `rgbcol`: nx3 matrix of RGB values in [0,1]. Colors of the background of patches.
- `rgbhatch`: nx3 matrix of RGB values in [0,1]. Colors of hatchings.
- `side`: side of each image patch, in pixels.

Value

`rgbpatch` object, structured as follows:

- `rchannel`: n x squared(side) matrix, containing R values in [0,1] for the n patches. Each line, ie patch, is intended to be restored column-wise for drawing.
- `gchannel`: same for G values.
- `bchannel`: same for B values.

Author(s)

Pierrick Bruneau

See Also

drawPatches

Examples

```r
coldata <- matrix(runif(30), ncol=3)
hatchdata <- matrix(runif(30), ncol=3)
patches <- genHatchData(coldata, hatchdata)
```
getInterRGB

Description

Interpolate colors in RGB-specified bounds.

Usage

getInterRGB(vals, zeroColor, oneColor)

Arguments

vals values in [0,1]. The resulting interpolated color will be value*oneColor + (1-value)*zeroColor, in Lab space.

zeroColor the "zero bound" color, in any valid R color format (e.g. "yellow", "#00FF00", ...)

oneColor the "one bound" color, in any valid R color format (e.g. "yellow", "#00FF00", ...)

Value

a vector of RGB colors in string format, e.g. "#BBFF00".

Author(s)

Pierrick Bruneau

See Also
drawGradient

Examples

res <- getInterRGB(runif(10), "yellow", "red")
Description

Convert a matrix of points in Lab color space to RGB points.

Usage

Lab2RGB(mat)

Arguments

mat matrix of row-elements (points in Lab color space, L should scale in [0,100], and (a,b) in [-110,110]).

Value

matrix of row-elements (points with R, G and B values in [0,1]).

Author(s)

Pierrick Bruneau

References

This transform is based on ITU-R BT 709, using the D65 white point reference. See http://fr.wikipedia.org/wiki/Rec.709 for details. also see http://www.easyrgb.com/index.php?X=MATH

See Also

Lab2RGB

Examples

dat <- matrix(runif(30), ncol=3)
labdat <- RGB2Lab(dat)
rgbdat <- Lab2RGB(labdat)
plotGmmOverlay

Description

Overlays a Gaussian mixture over an existing plot as output by plotGreyPatches.

Usage

plotGmmOverlay(gmm)

Arguments

- **gmm**: GMM object (see VBmix::newGmm for format definition)

Value

NULL

Author(s)

Pierrick Bruneau

See Also

plotGreyPatches

Examples

```r
maingrey <- runif(20)
auxgrey <- runif(20)
xvals <- c(rnorm(20, -3), rnorm(20, 3))
yvals <- rnorm(40)

# set a gaussian mixture model
gaussmod <- list()
gaussmod$w <- c(0.5, 0.5)
gaussmod$mean <- list(c(-3,0), c(3,0))
gaussmod$cov <- list(diag(2), diag(2))

# overlay the output of plotGreyPatches
plotGreyPatches(cbind(xvals, yvals), maingrey, auxgrey)
plotGmmOverlay(gaussmod)
```
Description

Draw a scatterplot with grey patches as placeholders for the points. Each patch may display two
distinct grey nuances (upper left and bottom right halves).

Usage

plotGreyPatches(data, maingrey, auxgrey=NULL, patchSize=0.05, alpha=0.5, highlight=numeric(0), labels=rep("1", length=length(data)))

Arguments

data : nx2 matrix of points positions in the 2D space.
maingrey : Vector of gray intensities in [0,1]
auxgrey : If NULL, each patch displays only 1 grey intensity. If not NULL, its length
should match maingrey, so that maingrey[i] and auxgrey[i] are the 2 grey nu-
ances displayed by the i-th patch
patchSize : patch size as a ratio of the whole window.
alpha : If highlight is empty, this is the alpha value for all patches. Else if a set is
highlighted, highlighted elements have this alpha value, and other elements have
alpha/20.
highlight : numeric vector taking values in 1..n, indicating elements in data to highlight.
Patches are associated to alpha=1 if highlighted, 0.04 if not. If no highlight is
specified, all patches have alpha=1.
labels : text labels associated to highlighted elements. Defaults to "1" for all highlighted
elements.

Value

NULL

Author(s)

Pierrick Bruneau

See Also

drawPatches
Examples

maingrey <- runif(20)
auxgrey <- runif(20)
xvals <- c(rnorm(20, -3), rnorm(20, 3))
yvals <- rnorm(40)
plotgreypatches(cbind(xvals, yvals), maingrey, auxgrey) # with 2 nuances per patch
plotgreypatches(cbind(xvals, yvals), maingrey) # only one nuance

Description

Convert a matrix of RGB points to Lab color space.

Usage

RGB2Lab(mat)

Arguments

mat matrix of row-elements (points with R, G and B values in [0,1]).

Value

matrix of row-elements (points in Lab color space, L should scale in [0,100], and (a,b) in [-110,110]).

Author(s)

Pierrick Bruneau

References

This transform is based on ITU-R BT 709, using the D65 white point reference. See http://fr.wikipedia.org/wiki/Rec.(underscore)709 for details. also see http://www.easyrgb.com/index.php?X=MATH

See Also

Lab2RGB

Examples

dat <- matrix(runif(30), ncol=3)
labdat <- RGB2Lab(dat)
Index

animatePatches, 2
checkGmm, 2
colorPlot, 3
drawGradient, 4
drawPatches, 5
drawSinglePatch, 6
genHatchData, 7
getInterRGB, 8
Lab2RGB, 9
plotGmmOverlay, 10
plotGreyPatches, 11
RGB2Lab, 12