Package 'binovisualfields'

October 12, 2022

Type Package

Title Depth-Dependent Binocular Visual Fields Simulation

Version 0.1.1

Author Ping Liu [aut, cre],
Andrew Turpin [aut]

Maintainer Ping Liu <ping.liu@unimelb.edu.au>

URL https://people.eng.unimelb.edu.au/aturpin/opi/index.html

Description Simulation and visualization depth-dependent integrated visual fields. Visual fields are measured monocularly at a single depth, yet real-life activities involve predominantly binocular vision at multiple depths. The package provides functions to simulate and visualize binocular visual field impairment in a depth-dependent fashion from monocular visual field results based on Ping Liu, Allison McKendrick, Anna Ma-Wyatt, Andrew Turpin (2019) <doi:10.1167/tvst.9.3.8>. At each location and depth plane, sensitivities are linearly interpolated from corresponding locations in monocular visual field and returned as the higher value of the two. Its utility is demonstrated by evaluating DD-IVF defects associated with 12 glaucomatous archetypes of 24-2 visual field pattern in the included 'shiny' apps.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.1.0)

Imports plotrix, gtools, shiny

Suggests knitr, rmarkdown, testthat

RoxygenNote 6.1.1

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2020-03-10 10:20:02 UTC

R topics documented:

---

binovfcal | *Calculates an array of integrated visual fields*

Description

binovfcal calculates an array of integrated visual fields given required parameters.

Usage

```
binovfcal(leftvf, rghtvf, lefttheta, righttheta, distplanes, pd = NULL,
  gender = NULL, m_xs = seq(-27, 27, length.out = 10), m_ys = seq(21,
  -21, -6), c_xs = seq(-57, 57, 6), db_cutoff = 25)
```

Arguments

leftvf	A matrix of left monocular visual field
rghtvf	A matrix of right monocular visual field
lefttheta	A number angle of convergence for the left eye in radian
righttheta	A number angle of convergence for the right eye in radian
distplanes	A vector of object distances in mm.
pd	Pupil distance in mm
gender	A string of either "male" or "female"
m_xs	Horizontal coordinates for monocular visual field for the 24-2 pattern
m_ys	Vertical coordinates for monocular visual field for the 24-2 pattern
c_xs	Horizontal coordinates for integrated visual field (from -57 to 57 degree with 6 degree spacing)
db_cutoff	cutoff value default to 25 dB above which the simulated threshold value is returned (NA otherwise) when there threshold value is present only for one eye

Value

An array of binocular visual fields for the distances specified by distplanes vector.

Warning

the value of either pd or gender has to be provided the unit of pd and fixdist must be the same, default to mm.

Examples

```
rghtvf <- matrix(c(
NA, NA, NA, 30, 30, 30, 30, NA, NA, NA,
NA, NA, 30, 30, 30, 30, 30, 30, NA, NA,
NA, 30, 30, 30, 30, 30, 30, 30, 30, NA,
30, 30, 30, 30, 30, 30, 30, 30, 30, NA,
30, 30, 30, 30, 30, 30, 30,  0, 30, NA,
NA, 30, 30, 30, 30, 30, 30, 30, 30, NA,
NA, NA, 30, 30, 30, 30, 30, 30, NA, NA,
NA, NA, NA, 30, 30, 30, 30, NA, NA, NA
), ncol=10, byrow=TRUE)
leftvf <- rghtvf[, 10:1]
binovfcal(leftvf, rghtvf, -.045, .045, c(1000, 1010), pd=62)
```

---

caltheta   *Calculates the angle of convergence*

Description

caltheta calculates the angle of convergence (radians) for left or right eye to fixate at a designated fixation distance.

Usage

```
caltheta(fixdist, pd = NULL, gender = NULL, eye = c("left", "right"))
```

Arguments

fixdist   A two element vector of fixation distance in mm in Cartesian coordinates.

pd   A number of pupil distance in mm.

gender   A string of either "male" or "female"

eye   A string specifying either "left" or "right" eye.

Value

The angle of convergence in radians that respective eye with a pupil distance of pd rolls to fixate at fixation distance of fixdist.

Warning

the value of either pd or gender has to be provided the unit of pd and fixdist must be the same and is default to mm.

Examples

```
caltheta(c(600, 0), pd=65, eye="left")
caltheta(c(600, 0), gender="male", eye="left")
```

---

colfunc   *Creates a color scheme for visual field plots*

Description

colfunc creates a color scheme visualising the dB values in visual field matrices with darker color corresponding to lower dB values

Usage

```
colfunc(n = 35)
```

Arguments

n   A positive integer specifying the number of color gradients used in visual field plots and is default to 35

Value

A color mapping function

Examples

```
colfunc(35)
```

---

colorkey   *Generates a color legend for visual field plots*

Description

colorkey generates a color legend for dB values in visual field plots with darker colors corresponding to lower dB values (e.g., black = 0dB, bright yellow >= 35dB)

Usage

```
colorkey()
```

Value

a color legend for dB values in visual field plots

Examples

```
colorkey()
```

---

get_col   *Color-codes dB values*

Description

get_color returns the color of a visual field location given its dB value (e.g., black = 0dB, white = 35dB)

Usage

```
get_col(db)
```

Arguments

db   A number of sensitivity threshold in dB

Value

the color of a visual field location in hcl color space with darker colors corresponding to lower dB values

Examples

```
get_col(25)
```

---

get_inv_col   *Color-codes dB values*

Description

get_inv_col returns the color of a visual field location given its dB value, (e.g., white < 15dB, black > 15dB)

Usage

```
get_inv_col(db)
```

Arguments

db   A number of sensitivity threshold in dB

Value

either white or black color for a visual field location

Examples

```
get_inv_col(25)
```

---

makevf	*Makes a visual field matrix*

Description

`makevf` Makes a visual field matrix from a vector of 54 elements for the 24-2 test results

Usage

```
makevf(vfvector, eye = c("left", "right"))
```

Arguments

vfvector   A vector of length 54. The 54 data points for the 24-2 pattern have to be ordered from superior nasal to inferior temporal.

eye   A string of either "left" or "right"

Value

A matrix of 8 rows and 10 columns

Examples

```
vfvector <- rep(35, 54)
makevf(vfvector, 'left')
```

---

plotvf	*Plots visual field*

Description

`plotvf` plots a figure of a visual field matrix with sensitivity threshold values

Usage

```
plotvf(xs, vf, title = "")
```

Arguments

xs	A vector of horizontal coordinates. The length of it must be either 10 for a monocular or 25 for a binocular visual field plot.
vf	A matrix of either a left/right monocular visual field or a binocular visual field sensitivity values
title	A string for the name of the plot.

Value

A plot of a monocular or binocular visual field.

Warning

the length of xs and the number of columns of the vf must be the same and with a value of either 10 or 20.

Examples

```
m_xs <- seq(-27, 27, length.out = 10)
rghtvf <- matrix(c(
NA, NA, NA, 30, 30, 30, 30, NA, NA, NA,
NA, NA, 30, 30, 30, 30, 30, 30, NA, NA,
NA, 30, 30, 30, 30, 30, 30, 30, 30, NA,
30, 30, 30, 30, 30, 30, 30, 30, 30, NA,
30, 30, 30, 30, 30, 30, 30,  0, 30, NA,
NA, 30, 30, 30, 30, 30, 30, 30, 30, NA,
NA, NA, 30, 30, 30, 30, 30, 30, NA, NA,
NA, NA, NA, 30, 30, 30, 30, NA, NA, NA
), ncol=10, byrow=TRUE)
plotvf(m_xs, rghtvf, title='right visual field')
```

---

plotvfray	*Plots binocular visual field rays*

Description

`plotvfray` plots a figure showing how the left and right visual field sensitivity threshold data interact in the simulated binocular visual field.

Usage

```
plotvfray(leftvf, rghtvf, lefttheta, righttheta, fixdist, distplane)
```

Arguments

`leftvf`	An 8 by 10 matrix of sensitivity threshold data for the left visual field
`rghtvf`	An 8 by 10 matrix of sensitivity threshold data for the right visual field
`lefttheta`	A number left eye rotating angle in radian
`righttheta`	A number right eye rotating angle in radian
`fixdist`	A 2 element vector the coordinates of the fixation point in cartesian system in mm.
`distplane`	A number object distance in mm range from 0 to a maximum of 1500 mm.

Value

A plot of binocular visual field rays from a top view with left eye on the top.

Warning

the unit of fixdist, pd and distplane must be in mm.

Examples

```
rghtvf <- matrix(c(
NA, NA, NA, 30, 30, 30, 30, NA, NA, NA,
NA, NA, 30, 30, 30, 30, 30, 30, NA, NA,
NA, 30, 30, 30, 30, 30, 30, 30, 30, NA,
30, 30, 30, 30, 30, 30, 30, 30, 30, NA,
30, 30, 30, 30, 30, 30, 30,  0, 30, NA,
NA, 30, 30, 30, 30, 30, 30, 30, 30, NA,
NA, NA, 30, 30, 30, 30, 30, 30, NA, NA,
NA, NA, NA, 30, 30, 30, 30, NA, NA, NA
), ncol=10, byrow=TRUE)

leftvf <- rghtvf[, 10:1]

plotvfray(leftvf, rghtvf, -.05, .05, c(700, 0), 1000)
```

`plotvf_2`	*Plots visual field*

Description

`plotvf_2` plots a figure of a binocular visual field matrix with sensitivity threshold values with missing locations

Usage

```
plotvf_2(xs, vf, vf_norm, title = "")
```

Arguments

xs	A vector of binocular visual field horizontal coordinates. The length of it must be 20 for a binocular visual field plot
vf	A matrix of a binocular visual field sensitivity values
vf_norm	A matrix a binocular healthy visual field sensitivity values for a specified distance plane
title	A string for the name of the plot

Value

A plot of a monocular or binocular visual field

Warning

the length of xs and the number of columns of the vf must be the same and with a value of either 25.

Examples

```
c_xs <- seq(-57, 57, length.out = 20)
cvf <- matrix(c(
NA, NA, NA, NA, NA, NA, NA, NA, 30, 20, 20, 30, NA, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, 30, 20,  0,  0, 20, 30, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, 30, 30, 20,  0,  0, 20, 30, 30, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, 30, 30, 30, 20, 20, 30, 30, 30, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA, NA,
NA, NA, NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA, NA, NA
), ncol=20, byrow=TRUE)
cvf_norm <- matrix(c(
 NA, NA, NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA, NA, NA,
 NA, NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA, NA,
 NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA,
 NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA,
 NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA,
 NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA,
 NA, NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA, NA,
 NA, NA, NA, NA, NA, NA, NA, NA, 30, 30, 30, 30, NA, NA, NA, NA, NA, NA, NA, NA
), ncol=20, byrow=TRUE)
plotvf_2(c_xs, cvf, cvf_norm, title='integrated visual field')
```

---

rotate	*Rotates a point (x,y) by an angle*

Description

rotate calculates the coordinates for a point (x,y) after it rotates by an angle theta (radian)

Usage

```
rotate(xy, theta = 0)
```

Arguments

xy   A vector of length 2 representing the coordinates of a point in cartesian system

theta   A number rotating angle in radian

Value

Coordinates of the point after the rotation

Examples

```
rotate(c(1000, 0), theta=pi/6)
```

---

rundemo   *Runs shiny applications included in the package*

Description

rundemo(demo) runs a shiny application provided in the package

Usage

```
rundemo(demo)
```

Arguments

demo   A shiny application name. There are two applications i.e., "app", "app2" in-
                cluded in the package

Value

the called shiny application in a new browser

Examples

```
## Not run:
rundemo("app2.R")

## End(Not run)
```

Index