Package 'vfprogression'

October 12, 2022

Type Package

Title Visual Field (VF) Progression Analysis and Plotting Methods

Version 0.7.1

Date 2019-05-21

Author Tobias Elze, Dian Li (documentation), Eun Young Choi (QC)

Maintainer Dian Li <lidian@zju.edu.cn>

Description Realization of published methods to analyze visual field (VF) progression. Introduction to the plotting methods (designed by author TE) for VF output visualization. A sample dataset for two eyes, each with 10 follow-ups is included. The VF analysis methods could be found in -- Musch et al. (1999) <doi:10.1016/S0161-6420(99)90147-1>, Nouri-Mahdavi et at. (2012) <doi:10.1167/iovs.11-9021>, Schell et at. (2014) <doi:10.1016/j.ophtha.2014.02.021>, Aptel et al. (2015) <doi:10.1111/aos.12788>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Depends R (>= 2.10)

Imports stats, grDevices, graphics

NeedsCompilation no

Repository CRAN

Date/Publication 2019-05-24 15:00:02 UTC

R topics documented:

`plotComponentMatrix` *General plotting function for multiple 24-2 or 30-2 visual field measurements together:*

Description

`plotComponentMatrix` plots the following 24-2 or 30-2 visual field measurement: sensitivity, TD, TD prob, PD, and PD prob:

Usage

```
plotComponentMatrix(componentmatrix, ncomp = ncol(componentmatrix),
  plot.ncols = 5, plot.nrows = NULL,
  plot.annot.topleft.function = toString,
  plot.annot.bottomleft.function = function(i) NULL,
  globaltitle = sprintf("k = %i", ncol(componentmatrix)),
  globalannotright = NULL,
  zmin = -ceiling(max(abs(c(min(componentmatrix),
  max(componentmatrix))))), zmax = -zmin,
  color.pal = colorRampPalette(c("red", "white", "blue"), space =
  "Lab")(256), td.probabilities = FALSE,
  show.colorbar = !td.probabilities, titleheight = 0.2, ...)
```

Arguments

`componentmatrix`
: a matrix or data frame, column represents different eyes and rows are the VF measurements of the same type (sensitivity, TD, TD prob, PD, or PD prob).

`ncomp`
: a numeric variable defines the number of components to be plotted (default: all).

`plot.ncols`
: a numeric variable defines the number of columns to be plotted (default: 5).

`plot.nrows`
: a numeric variable defines the number of rows to be plotted (default: NULL (automatically calculated)).

`plot.annot.topleft.function`
: a function(i) that is given to any subplot i to create its top left annotation.

`plot.annot.bottomleft.function`
: a function(i) that is given to any subplot i to create its bottom left annotation (default: returns NULL).

`globaltitle`
: a string for global title (default: k = ncomp; set to NULL to suppress global title).

`globalannotright`
: a string annotation to the right of the global title (default: NULL).

`zmin`
: minimum value of the color scale (default: auto defined).

`zmax`
: maximum value of the color scale (default: auto defined).

`color.pal`
: an object that defines color scale theme (default: colorRampPalette(c("red", "white", "blue"), space = "Lab")(256)).

`td.probabilities`
: a logic variable indicates whether to plot TD probability symbols instead of TD colors (default: FALSE).

`show.colorbar`
: a logic variable indicates whether to show a global colorbar (default: !td.probabilities).

`titleheight`
: a numeric variable defines the height of the title relative to height of row one.

`...`
: other variables to be added.

Value

heatmap for sensitivity, TD and PD input. Value plot for TD prob and PD prob input.

Examples

```
data(vfseries)
componentmatrix = t(vfseries[1:10, grepl('^s[0-9]+', colnames(vfseries))])
globaltitle = paste("Sensitivities, k = ", ncol(componentmatrix), sep = '')
plotComponentMatrix(componentmatrix, globaltitle = globaltitle)
componentmatrix = t(vfseries[1:10, grepl('^td[0-9]+', colnames(vfseries))])
globaltitle = paste("TDs, k = ", ncol(componentmatrix), sep = '')
plotComponentMatrix(componentmatrix, globaltitle = globaltitle)
componentmatrix = t(vfseries[1:10, grepl('^pd[0-9]+', colnames(vfseries))])
globaltitle = paste("PDs, k = ", ncol(componentmatrix), sep = '')
plotComponentMatrix(componentmatrix, globaltitle = globaltitle)
componentmatrix = t(vfseries[1:10, grepl('^tdp[0-9]+', colnames(vfseries))])
globaltitle = paste("TD Probs, k = ", ncol(componentmatrix), sep = '')
plotComponentMatrix(componentmatrix, globaltitle = globaltitle, td.probabilities = TRUE)
componentmatrix = t(vfseries[1:10, grepl('^pdp[0-9]+', colnames(vfseries))])
globaltitle = paste("PD Probs, k = ", ncol(componentmatrix), sep = '')
plotComponentMatrix(componentmatrix, globaltitle = globaltitle, td.probabilities = TRUE)
```

plotfield.normalized *Single plotting function for one 24-2 or 30-2 visual field measurement:*

Description

plotfield.normalized plots the following 24-2 or 30-2 visual field measurement: sensitivity, TD, TD prob, PD, and PD prob:

Usage

```
plotfield.normalized(eigenfields, component = 1,
  zmin = -max(abs(c(min(eigenfields), max(eigenfields)))),
  zmax = max(abs(c(min(eigenfields), max(eigenfields)))),
  color.pal = colorRampPalette(c("red", "white", "blue"), space =
  "Lab")(256), show.colorbar = TRUE, topleftannotation = NULL,
  bottomleftannotation = NULL, labelcex = 2, ...)
```

Arguments

eigenfields	a vector contains Sensitivity/TD/PD measurement. For 24-2 VF eigenfields should have 52 or 54 elements. For 30-2 VF, eigenfields should have 74 or 76 elements.
component	Number of components to be plotted (default: 1).
zmin	minimum value of the color scale (default: auto defined).
zmax	maximum value of the color scale (default: auto defined).
color.pal	an object that defines color scale theme (default: colorRampPalette(c("red", "white", "blue"), space = "Lab")(256)).
show.colorbar	a logic value to show colorbar (default: TRUE).
topleftannotation	a string annotation shown on the top left side of the plot (default: NULL).
bottomleftannotation	a string annotation shown on the bottom left side of the plot (default: NULL).
labelcex	a numeric variable for label size (default: 2).
...	other variables to be added.

Value

heatmap for sensitivity, TD and PD input

Examples

```
data(vfseries)
eigenfields = t(vfseries[1, grepl('^s[0-9]+', colnames(vfseries))])
plotfield.normalized(eigenfields)
title(main = "Sensitivity", line = 3)
```

```
eigenfields = t(vfseries[1, grepl('^td[0-9]+', colnames(vfseries))])
plotfield.normalized(eigenfields)
title(main = "Total Deviation", line = 3)
eigenfields = t(vfseries[1, grepl('^pd[0-9]+', colnames(vfseries))])
plotfield.normalized(eigenfields)
title(main = "Pattern Deviation", line = 3)
```

`plotTdProbabilities` *Value plotting function for 24-2 or 30-2 visual field measurement:*

Description

`plotTdProbabilities` plots the following 24-2 or 30-2 visual field measurement: TD probs, and PD probs:

Usage

```
plotTdProbabilities(tdprob, cex = 2, rectangle.color = "black",
  rectangle.width = 0.16, margins = c(2, 1, 2, 2) + 0.1, ...)
```

Arguments

`tdprob`	a vector contains TD probs/PD probs measurement. For 24-2 VF `tdprob` should have 52 or 54 elements. For 30-2 VF, `tdprob` should have 74 or 76 elements.
`cex`	a numeric variable for label size (default: 2).
`rectangle.color`	a string variable defines label color (default: 'black').
`rectangle.width`	a numeric variable defines label width (default: '0.16').
`margins`	a vector define the plot margins (default: c(2, 1, 2, 2)+0.1).
`...`	other variables to be added.

Value

value plot for TD prob and PD prob input.

Examples

```
data(vfseries)
tdprob = t(vfseries[1, grepl('^tdp[0-9]+', colnames(vfseries))])
plotTdProbabilities(tdprob)
title(main = "Total Deviation Probability", line = 3)
tdprob = t(vfseries[1, grepl('^pdp[0-9]+', colnames(vfseries))])
plotTdProbabilities(tdprob)
title(main = "Pattern Deviation Probability", line = 3)
```

plotTDvalues *Value plotting function for 24-2 or 30-2 visual field measurement:*

Description

plotTDvalues plots the following 24-2 or 30-2 visual field measurement: sensitivity, TD, and PD:

Usage

```
plotTDvalues(tds, cex.tds = 1, textcolor = function(x) "black",
  show.lines = T, ...)
```

Arguments

tds	a vector contains sensitivity/TD/PD measurement. For 24-2 VF tds should have 52 or 54 elements. For 30-2 VF, tds should have 74 or 76 elements.
cex.tds	a numeric variable for label size (default: 1).
textcolor	a function defines the label color.
show.lines	a logical variable indicates whether to show the horizontal and vertical lines.
...	other variables to be added.

Value

value plot for sensitivity, TD and PD input.

Examples

```
data(vfseries)
tds = t(vfseries[1, grepl('^s[0-9]+', colnames(vfseries))])
plotTDvalues(tds)
title(main = "Sensitivity", line = 3)
tds = t(vfseries[1, grepl('^td[0-9]+', colnames(vfseries))])
plotTDvalues(tds)
title(main = "Total Dviation", line = 3)
tds = t(vfseries[1, grepl('^pd[0-9]+', colnames(vfseries))])
plotTDvalues(tds)
title(main = "Pattern Dviation", line = 3)
```

progression *general progression function*

Description

progression returns the progression criterion with four methods. plr.nouri.2012, vfi, schell2014, cigts

Usage

```
progression(vfseries, method = c("plr.nouri.2012", "vfi", "schell2014",
  "cigts"))
```

Arguments

vfseries	is a data frame. MUST contain the following columns: yearsfollowed', and 'eyeid'. Rows represent the single measurements. Other requirements, such as number of minimum measurements (rows), and necessary VF measurements could be found in each progression method's documentation
method	selected from one or more from: plr.nouri.2012, vfi, schell2014, cigts. Default it ...

Value

"stable", "worsening", or "improving" of measurements in measmatrix

See Also

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4495761/

Examples

```
data(vfseries)
progression(vfseries)
progression(vfseries[vfseries$eyeid == 1,])
progression(vfseries[vfseries$eyeid == 2,])
progression(vfseries, method=c("cigts"))
```

progression.cigts *CIGTS VF progression*

Description

progression.cigts returns the progression of visual field test based on 52 or 54 total deviation probabilities (tdp). CIGTS VF progression (Musch et al., 1999).

Usage

```
progression.cigts(measmatrix)
```

Arguments

measmatrix	is a data frame. MUST contain the following columns: 52/54 TD probs (column names MUST be 'tdp1' ~ 'tdp52' or 'tdp1' ~ 'tdp54'), 'yearsfollowed', and 'eyeid'. Rows represent the single measurements. The minimum measurements (rows) is 5.

Value

"stable", "worsening", or "improving" of measurements in measmatrix. Note: If a VF series is temporarily improving and temporarily worsening, it is assumed to be "stable" overall

References

http://www.aaojournal.org/article/S0161-6420(99)90147-1/abstract

Examples

```
data(vf.cigts)
colnames(vf.cigts)
progression.cigts(vf.cigts)
progression.cigts(vf.cigts[vf.cigts$eyeid == 1,])
progression.cigts(vf.cigts[vf.cigts$eyeid == 2,])
```

progression.plr.nouri.2012

Nouri-Mahdavi 2012 VF progression

Description

progression.plr.nouri.2012 returns the progression criterion, using Pointwise Linear Regression (PLR) progression detection method according to Nouri-Mahdavi et al. (2012).

Usage

```
progression.plr.nouri.2012(measmatrix)
```

Arguments

measmatrix	is a data frame. MUST contain the following columns: 52/54 TD (column names MUST be 'td1' ~ 'td52' or 'td1' ~ 'td54'), 'yearsfollowed', and 'eyeid'. Rows represent the single measurements. The minimum measurements (rows) is 3.

Value

"stable", "worsening", or "improving" of measurements in `measmatrix`

See Also

https://www.ncbi.nlm.nih.gov/pubmed/22427560/

Examples

```
data(vf.plr.nouri.2012)
colnames(vf.plr.nouri.2012)
progression.plr.nouri.2012(vf.plr.nouri.2012)
progression.plr.nouri.2012(vf.plr.nouri.2012[vf.plr.nouri.2012$eyeid == 1,])
progression.plr.nouri.2012(vf.plr.nouri.2012[vf.plr.nouri.2012$eyeid == 2,])
```

---

progression.schell2014

Schell 2014 VF progression

Description

`progression.schell2014` returns the progression criterion after Schell et al. 2014, which is essentially like CIGTS but with MD, and only one follow-up is enough to confirm progression.

Usage

```
progression.schell2014(measmatrix)
```

Arguments

measmatrix	is a data frame. MUST contain the following columns: 'md' (mean deviation) and 'eyeid'. Rows represent the single measurements. The minimum measurements (rows) is 4.

Value

"stable", "worsening", or "improving" of measurements in `measmatrix`. Note: If a VF series is temporarily improving and temporarily worsening, it is assumed to be "stable" overall

See Also

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4495761/

Examples

```
data(vf.schell2014)
colnames(vf.schell2014)
progression.schell2014(vf.schell2014)
progression.schell2014(vf.schell2014[vf.schell2014$eyeid == 1,])
progression.schell2014(vf.schell2014[vf.schell2014$eyeid == 2,])
```

---

progression.vfi *progression according to VFI (significant slope, p<=0.05)*

Description

`progression.vfi` returns the progression criterion used in Aptel et al. (2015).

Usage

```
progression.vfi(measmatrix)
```

Arguments

`measmatrix`	is a data frame. MUST contain the following columns: 'vfi' (visual field index), 'yearsfollowed', and 'eyeid'. Rows represent the single measurements. The minimum measurements (rows) is 3.

Value

"stable", "worsening", or "improving" of measurements in `timepoints`

See Also

https://www.ncbi.nlm.nih.gov/pubmed/26095771/

Examples

```
data(vf.vfi)
colnames(vf.vfi)
progression.vfi(vf.vfi)
progression.vfi(vf.vfi[vf.vfi$eyeid == 1,])
progression.vfi(vf.vfi[vf.vfi$eyeid == 2,])
```

vf.cigts *Combined Visual Field Series for General Progression Method*

Description

Data

Usage

```
data(vf.cigts)
```

Format

A data frame sample for CIGTS progression method, which includes visual field related measurement for two eyes, each with 10 follow-ups. Rows represent the single measurements.

Source

eyeid eyeid, labeled as 1,2... for different eyes.

yearsfollowed follow-up years. The minimum measurements /rows for one eye is 5.

tdp1-tdp54 52 total deviation probability, or 'tdp' measurements. The minimum measurements, or rows for one eye is 5. ...

Examples

```
data(vf.cigts)
colnames(vf.cigts)
progression.cigts(vf.cigts)
progression.cigts(vf.cigts[vf.cigts$eyeid == 1,])
progression.cigts(vf.cigts[vf.cigts$eyeid == 2,])
```

vf.plr.nouri.2012 *Combined Visual Field Series for General Progression Method*

Description

Data

Usage

```
data(vf.plr.nouri.2012)
```

Format

A data frame sample for Pointwise Linear Regression (PLR) method according to Nouri-Mahdavi 2012 progression, which includes visual field related measurement for two eyes, each with 10 follow-ups. Rows represent the single measurements.

Source

eyeid eyeid, labeled as 1,2... for different eyes

yearsfollowed follow-up years. The minimum measurements, or rows, for one eye is 3

td1-td54 52 total deviation, or 'td' measurements. The minimum measurements, or rows, for one eye is 3 ...

Examples

```
data(vf.plr.nouri.2012)
colnames(vf.plr.nouri.2012)
progression.plr.nouri.2012(vf.plr.nouri.2012)
progression.plr.nouri.2012(vf.plr.nouri.2012[vf.plr.nouri.2012$eyeid == 1,])
progression.plr.nouri.2012(vf.plr.nouri.2012[vf.plr.nouri.2012$eyeid == 2,])
```

---

vf.schell2014 *Combined Visual Field Series for General Progression Method*

Description

Data

Usage

```
data(vf.schell2014)
```

Format

A data frame sample for progression method by Schell et al. 2014, which includes visual field related measurement for two eyes, each with 10 follow-ups. Rows represent the single measurements.

Source

eyeid eyeid, labeled as 1,2... for different eyes.

md mean deviation measurements. The minimum measurements, or rows, for one eye is 4. ...

Examples

```
data(vf.schell2014)
colnames(vf.schell2014)
progression.schell2014(vf.schell2014)
progression.schell2014(vf.schell2014[vf.schell2014$eyeid == 1,])
progression.schell2014(vf.schell2014[vf.schell2014$eyeid == 2,])
```

vf.vfi *Combined Visual Field Series for General Progression Method*

Description

Data

Usage

```
data(vf.vfi)
```

Format

A data frame for CIGTS progression example, which includes visual field related measurement for two eyes each with 10 follow-ups.

Source

eyeid eyeid, labeled as 1,2... for different eye groups.

yearsfollowed follow-up years. The minimum measurements, or rows, for one eye is 3.

vfi visual field index. The minimum measurements, or rows, for one eye is 3. ...

Examples

```
data(vf.vfi)
colnames(vf.vfi)
progression.vfi(vf.vfi)
progression.vfi(vf.vfi[vf.vfi$eyeid == 1,])
progression.vfi(vf.vfi[vf.vfi$eyeid == 2,])
```

vfseries *Combined Visual Field Series for General Progression Method*

Description

Data

Usage

```
data(vfseries)
```

Format

A data frame sample including the following visual field related measurement for two eyes, each with 10 follow-ups.

Source

eyeid eyeid, labeled as 1,2... for different eyes.

nvisit number of visits.

yearsfollowed follow-up years.

distprev to be updated.

age in years.

righteye 1 as right eye, 0 as left eye.

malfixrate VF test malfixation rate.

ght glaucoma hemifield test result.

vfi visual field index.

md mean deviation.

mdprob mean deviation probability.

psd pattern standard deviation.

psdprob pattern standard deviation probability.

s1-s54 52 sensitivity measurements.

td1-td54 52 total deviation measurements.

tdp1-tdp54 52 total deviation probability measurements.

pdp1-pdp54 52 pattern deviation probability measurements. ...

Examples

```
data(vfseries)
progression(vfseries)
progression(vfseries[vfseries$eyeid == 1,])
progression(vfseries[vfseries$eyeid == 2,])
progression(vfseries, method=c("cigts"))
progression.cigts(vfseries)
progression(vfseries, method=c('plr.nouri.2012', 'schell2014', 'vfi'))
```

Index