

Package ‘SOFIA’

October 12, 2022

Title Making Sophisticated and Aesthetical Figures in R

Version 1.0

Author Luis Diaz-Garcia

Maintainer Luis Diaz-Garcia <diazgarcia@wisc.edu>

Depends R (>= 2.14)

Imports png, grid

Description Software that leverages the capabilities of Circos by manipulating data, preparing configuration files, and running the Perl-native Circos directly from the R environment with minimal user intervention. Circos is a novel software that addresses the challenges in visualizing genetic data by creating circular ideograms composed of tracks of heatmaps, scatter plots, line plots, histograms, links between common markers, glyphs, text, and etc. Please see <<http://www.circos.ca>>.

License GPL-3

URL <http://cggl.horticulture.wisc.edu>

NeedsCompilation no

Repository CRAN

SystemRequirements Circos

Date/Publication 2017-01-22 12:38:27

R topics documented:

mark0	2
SOFIA	3

Index	10
--------------	-----------

 mark0

Genetic maps to be used as examples for generating Circos figures

Description

mark0 is a dataframe that is used to construct the input "data". mark0 has four variables, \$map, \$chr, \$pos and \$locus, which are the minimum requirement to run SOFIA. The information contained in mark0 corresponds to two different genetic maps.

Usage

```
data(mark0)
```

References

Luis Diaz-Garcia, Giovanni Covarrubias-Pazaran, Brandon Schlautman and Juan Zalapa. 2016. SOFIA: an R package for enhancing genetic visualization with Circos. 2016. Journal of Heredity. Submitted.

Examples

```
# loading data contained in mark0
# data(mark0)
# str(mark0)

# making the dataframe data with the variables map, chr, pos and locus
# data1<-data.frame(map=mark0$map,chr=mark0$lg,pos=mark0$consensus,locus=mark0$marker)

##### adding variables to plot
# plotting text
# data1$someNames<-data1$locus
# data1$someNames[sample(1:10000,9000)]<-NA

# plotting a heatmap with only two colors (black and white for marker presence and absence,
# respectively).
# This plot gives a karyotype plot
# data1$kar<-data1$pos

# more random data
# data1$lod1<--log10(mark0$b)
# data1$lod2<-sample(data1$lod1)
# data1$lod3<-rev(data1$lod1)

# defining the location of each plot (5 plots total)
# plotLocation<-data.frame(r0=c(0.90,.88,.78,.68,.58),r1=c(.99,.9,.87,.77,.67))
# in this case, the first plot is going to be in the position 0.90-0.99, the second in the
# position 0.88-0.90, and so on.

# defining the background for the 5 plots. In this case, the first two plots are not going to have
# background. All plot are going to have horizontal axes spaced at 5 units.
```

```

# plotBackground<-data.frame(backgroundShow=c(FALSE,FALSE,TRUE,TRUE,TRUE),
# backgroundColor=rep('vvlgrey',5),axisShow=rep(TRUE,5),axisSep=rep(4,5))

# defining the overall configuration for the figure. In this case, there are two maps,
# the chromosomes for map 1 are going to be ordered from 1 to 12, however map 2 is going
# to have its chromosomes in order 9 to 1. In addition, the argument rev is going to
# reverse chromosomes for map 1. Finally, map 1 is going to have two different
# kind of blues (lblue_a3 and vvdblue_a3, which mieans light blue for the first one,
# and very very dark blue for the second; a3 is a degree of transparency) for their
# chromosomes, while the map two is going to be all dark green.
# chromoConfiguration<-data.frame(order=c(1:12,9:1),map=c(rep(1,12),rep(2,9)),
# rev=c(rep(FALSE,12),rep(TRUE,9)),color=c(rep(c('vvdblue_a3','lblue_a3'),6),rep('dgreen',9)),
# radius=rep(1,21))

# defining plot types
# plotType<-c('text','heatmap',rep('scatter',2),'line')

# defining marker colors. For the first plot, the letters are going to be black,
# for the second plot, the presence of numeric data is going to be colored with
# black (for traditional heatmaps with multiple colors, a color palette has to be defined),
# for the third plot, the color 'chr' tells SOFIA to use the color of the chromosome where
# the marker is located, for the forth plot, the markers are going to be colored using the
# palette piyg-11-div (higher and lower values are going to have different colors), and
# the fifth line plot is going to be colored with dark red.
# plotColor<-c('black','black','chr','piyg-11-div','dred_a3')

# definifn marker size. For the text plot, the marker size defines the font size.
# For scatter, it defines the circle size while for line plots, it defines the line
# thickness. For heatmaps, any random number can be included since it is not used at all.
# markerSize<-c(8,10,16,16,1)

# please change the argument circosLocation
# SOFIA(data=data1,linkColor='chr',linkGeometry=c(.001,.1),linkRadius=c(.57,.57),
# linksFlag=TRUE,chromoConfiguration=chromoConfiguration,plotBackground=plotBackground,
# plotLocation=plotLocation,plotType=plotType,plotColor=plotColor,markerSize=markerSize,
# circosLocation='/circos/bin',tickSuffix='cM',returnConf=TRUE,circosDisplay=FALSE)

```

SOFIA

Making Sofisticated and Aesthetical Figures in R

Description

Automatically prepares all necessary configuration files and then runs Perl-based Circos directly from R.

Usage

```

SOFIA(data,
chromoConfiguration=NULL,

```

```

dataColorFlag=FALSE,
dataColor=NULL,
plotType=NULL,
plotColor=NULL,
markerSize=NULL,
plotLocation=NULL,
plotBackground=NULL,
plotImportance=NULL,
plotOrientation=NULL,
density=NULL,
linksFlag=FALSE,
linkColor='red',
linkGeometry=c(.2, .2),
linkRadius=c(.7, .7),
blocksFlag=FALSE,
blocksData=NULL,
tilesFlag=FALSE,
tilesData=NULL,
tilesLocation=NULL,
blocksColor=rbind(c('blue', 'red'), c('green', 'purple')),
blocksLocation=c(.5, .01),
gaps=NULL,
ideogramThickness=20,
generalPlotConfFlag=FALSE,
generalPlotConf=NULL,
chrPrefixFont='upper',
chrPrefix='LG',
tickSeparation=50,
tickSuffix='cM',
circosLocation=NULL,
returnConf=FALSE,
circosDisplay=FALSE,
figureDisplay=TRUE,
deleteData=TRUE,
runCircos=TRUE,
confName='circos.conf')

```

Arguments

data	Genetic map(s) to be plotted into the Circos figure. As minimum, it has to have four variables named \$locus, \$chr, \$pos and \$map. \$locus corresponds to the names of all the positions in the map (this can be marker names for example), \$chr is the chromosome number for each locus, \$pos is the position in the map for each locus, and \$map defines if multiple maps are plotted (if a single map is going to be used, \$map has to be 1's for all the rows (loci) in the dataframe). After this four variables (or columns), all the data to be plotted has to be integrated in additional variables. Specific names for these columns/variables are not required. NA's are omitted when constructing plots. There are six basic types of plots: 1) scatter plot, 2) heatmap, 3) heatmap_interval, 4) line, 5) text
------	---

and 6) glyphs. In addition, four more types of specialized plots are included, these are 1) tiles, 2) links, 3) blocks and 4) density. These four plot types are controlled differently than the basic types, and the data corresponding to these plots are not contained into the dataframe "data".

chromoConfiguration	Dataframe with properties to configure the plot. It has to have as many rows as chromosomes in the dataframe "data". Five variables are required: the first two have to be called \$map and \$order, and they describe the order in which the chromosomes are going to be displayed in the Circos figure. Then, the 3rd variable must be named \$rev and specifies if a specific chromosome is going to be reversed in the figure. Variable 4 has to be called \$color and specifies the color of the bars that represent the chromosomes in the figure. Finally, the last variable is called \$radius and specifies the radius of the chromosomes. As default, a chromoConfiguration dataframe is created in which the chromosome order is as found in the dataframe "data", all the chromosomes are oriented clockwise and colored grey.
dataColor	This option is used if the dataframe "data" posses more than 4 columns (when plots are going to be included in the figure), and specific colors want to be used for specific points in the plot. In that is the case, dataColor has to have the same number of rows and (columns+4) as the dataframe "data". Each column of dataColor correspond to each plot defined in the dataframe "data" after column 5.
dataColorFlag	When TRUE, the dataColor option for coloring single markers is used.
plotType	A vector defining the type of plot for each column anter column 4 in the dataframe "data". The option are 1) scatter plot, 2) heatmap, 3) heatmap_interval, 4) line, 5) text and 6) glyphs. If omitted, all the plots are going to be plotted as scatter plots, unless text is found.
plotColor	A vector defining the color of the plot for each column anter column 4 in the dataframe "data". If omitted, random colors are assigned.
markerSize	A vector defining the marker size to be used in the plots for each column anter column 4 in the dataframe "data". If omitted, random sizes are assigned.
plotLocation	A dataframe with as many rows as number of plots. Column 1 has to be named r0, it specifies the initial location of the plot (y0). Column 2 has to be named r1, it specifies the final location of the plot (y1). y1 must be > than y0 and y1 < chromosome radius (see chromoConfiguration)
plotBackground	A dataframe with as many rows as number of plots. Column 1 has to be named backgroundShow, it specifies if a background is going to be displayed for every plot. Column 2 has to be named backgroundColor, it specifies the background color for every plot. Column 3 has to be called axisShow and specifies if horizontal axis is going to be display for every plot. Finally, column 4 has to be called axisSep and specifies the separation of the axis to be displayed.
plotImportance	A vector of numbers that control the overlapping of plots if their locations are the same. Larger numbers are plotted in the top.
plotOrientation	A vector containt "in" or "out" for each of the plots. This determines the orientation in the plot within the figure.

density	When used, it draws a histogram with the marker density. This has to be a dataframe with rows=1 and the following variables. \$show (TRUE or FALSE) specifies if the density plot is showed or not. \$bins (integer >0) controls in how many bins each chromosome is broken for counting markers. \$backgroundShow (TRUE or FALSE) determines if a background is displayed while \$backgroundShow determines the color. \$axisShow (TRUE or FALSE) determines if horizontal axis is displayed while \$axisSep (numeric > 0) determines the separation between axes.
linksFlag	When TRUE and two different maps are plotted, links connecting common markers are displayed.
linkColor	Controls the color of the links when linksFlag=TRUE.
linkGeometry	A vector of length=2 which determines the bezier_radius and the crest (from 0 to 1).
linkRadius	A vector of length=2 which determine the location of both ends of links. The first number in the vector corresponds to map 1 while the position 2 corresponds to the map 2.
blocksFlag	When TRUE, recombination blocks are displayed.
blocksData	blocksData must be a matrix with rows equal to the number of markers and columns equal to the number of individuals in the population. This genetic data must correspond to phased genotypes codified as 0's and 1's.
blocksColor	If a single map is used, a vector containing colors that represent 0's and 1's. If two maps are used, four colors are required.
blocksLocation	A vector of length=2. The first number corresponds to the outer position of the recombination blocks for plant 1. The second number corresponds to the thickness of each block.
tilesFlag	When TRUE, tiles are displayed. Tiles are little horizontal bars drawn at specific positions in the maps. Color, location and length can be controlled.
tilesData	tilData is a dataframe with as many rows as desired tiles, and the following variables. \$map and \$chr specifies the map and chromosome in which the tile is going to be plotted. \$pos1 and \$pos2 specifies the initial and final position of the tile. \$color specifies the color of the tile.
tilesLocation	tilesLocation is a vector of length=2 in which the location of the tiles are drawn (as r0 and r1). Only one set of tiles are allowed.
gaps	gaps is a dataframe with as many rows as desired gaps in the figure. The following variables are required. \$mapA and \$mapB corresponds to the maps in which a gap is going to be displayed. If only one map is used, both variables must be 1. \$chrA (for mapA) and \$chrB (for mapB) corresponds to the specific chromosomes in which the gap is going to be displayed.
ideogramThickness	A single number controlling the thickness of the bars that represent the chromosomes.
generalPlotConf	When TRUE, a vector of parameters for general configuration of plots is used. See next argument.

<code>generalPlotConfFlag</code>	A vector of parameters to be applied to all the plots. If you are familiar with Circos (if not, probably you do not this parameter), these arguments are located in the block plots-plots but outside each specific block plotplot. Synthesis is similar to Circos.
<code>chrPrefixFont</code>	'upper' or 'lower'.
<code>chrPrefix</code>	A prefix to name chromosome labels.
<code>tickSeparation</code>	Separation of the ticks for each chromosome. Default = 20.
<code>tickSuffix</code>	A suffix to name tick labels. Default = 'cM'.
<code>circosLocation</code>	Directory that contains the circos file.
<code>returnConf</code>	When TRUE, return the circos.conf into the console.
<code>circosDisplay</code>	When TRUE, return Circos output when constructing the figure.
<code>figureDisplay</code>	When TRUE, Circos figure is read and displayed into R.
<code>deleteData</code>	When TRUE, SOFIA deletes all the files, circos.conf, ideogram.conf and data files, used for making a figure.
<code>runCircos</code>	When FALSE, SOFIA only creates the files required to make a figure but does not run Circos. This is highly useful if you only want to make manual modification in the circos.conf configuration file. If you set this FALSE, you may want to set deleteData=FALSE.
<code>confName</code>	Name of the Circos configuration file. Default = 'circos.conf'.

Details

Although many arguments can be defined in SOFIA, most of them are optional and figures can be generated with minimal configuration. Assuming that a dataframe "data" with the basic four variables \$map \$locus \$pos and \$chr, plus x number of additional variables with numeric/text data, a Circos figure can be made by only using the arguments 1) data and 2) circosLocation (for example `>SOFIA(data=data,circosLocation='yourCircosLocation/bin/')`). If this is the case, all the data will be plotted as scatter plot and the plots are going to be distributed uniformly across the Circos figure. This is a good start.

References

Luis Diaz-Garcia, Giovanni Covarrubias-Pazaran, Brandon Schlautman and Juan Zalapa. 2016. SOFIA: an R package for enhancing genetic visualization with Circos. 2016. Journal of Heredity. Submitted. Martin I Krzywinski, Jacqueline E Schein, Inanc Birol, Joseph Connors, Randy Gascoyne, Doug Horsman, Steven J Jones, and Marco A Marra. 2009. Circos: An information aesthetic for comparative genomics. Genome Res.

Examples

```
# Loading data contained in mark0
data(mark0)
str(mark0)

# Making the dataframe data with the variables map, chr, pos and locus
```

```

data1<-data.frame(map=mark0$map,chr=mark0$lg,pos=mark0$consensus,locus=mark0$marker)

#### Adding numeric and text variables to plot in the figure

# text plot
data1$someNames<-data1$locus
data1$someNames[sample(1:10000,9000)]<-NA

# Plotting a heatmap with only two colors (black for marker presence and white for marker absence.
# The colors are defined later).
# By using this trick, a karyotype-like plot can be obtained.
data1$kar<-data1$pos

# Random numeric data
data1$lod1<--log10(mark0$b)
data1$lod2<-sample(data1$lod1)
data1$lod3<-rev(data1$lod1)

# Until here, we added five columns with numeric and text data to the dataframe data1. Now, we
# have to work in other arguments related with general and specific figure configuration.

# Defining the location of each plot (5 plots total).
# In this case, the first plot is going to be in the position 0.90-0.99, the second in the
# position 0.88-0.90, and so on.

plotLocation<-data.frame(r0=c(0.90,.88,.78,.68,.58),r1=c(.99,.9,.87,.77,.67))

# Defining the background for the 5 plots.
# In this case, the first two plots are not going to have a
# background. All plots are going to have horizontal lines spaced at 5 units and colored vvlgrey.

plotBackground<-data.frame(backgroundShow=c(FALSE,FALSE,TRUE,TRUE,TRUE),
backgroundColor=rep('vvlgrey',5),axisShow=rep(TRUE,5),axisSep=rep(4,5))

# Defining the overall configuration for the figure.
# In this example, Figure 2 has two genetic maps; the chromosomes for map 1 are
# going to be ordered
# from 1 to 12, while map 2 is going to have its chromosomes in order 9 to 1.
# In addition, the argument rev is going to flag
# all chromosomes in map 2 to be reversed. Finally, chromosomes in map 1 are going
# to be colored with two different
# blue colors (vvdblue_a3 and lblue_a3), while map two is going to have all
# chromosomes colored with a single color (dgreen).

chromoConfiguration<-data.frame(order=c(1:12,9:1),map=c(rep(1,12),rep(2,9)),
rev=c(rep(FALSE,12),rep(TRUE,9)),color=c(rep(c('vvdblue_a3','lblue_a3'),6),rep('dgreen',9)),
radius=rep(1,21))

```

```
# Defining marker colors.
# For the first plot, the text labels are going to be black colored; for the second plot,
# the presence of numeric data is going to be colored with black (for traditional
# heatmaps with multiple colors, a color palette,
# such as reds-9-seq, has to be defined); for the third plot,
# the color 'chr' tells SOFIA to use the color of the chromosome bar corresponding
# to the markers location; for the fourth plot, the markers are
# going to be colored using the palette piyg-11-div (markers are colored
# based on their values); the fifth line plot is going to be red colored (dred_a3).

plotType<-c('text','heatmap',rep('scatter',2),'line')

# defining marker colors. For the first plot, the letters are going to be black,
# for the second plot, the presence of numeric data is going to be colored with
# black (for traditional heatmaps with multiple colors, a color palette has to be defined),
# for the third plot, the color 'chr' tells SOFIA to use the color of the chromosome where
# the marker is located, for the fourth plot, the markers are going to be colored using the
# palette piyg-11-div (higher and lower values are going to have different colors), and
# the fifth line plot is going to be colored with dark red.

plotColor<-c('black','black','chr','piyg-11-div','dred_a3')

# Defining marker size.
# For the text plot, the marker size defines the text size.
# For scatter, it defines the circle size while for line plots, it defines the line thickness.
# For heatmaps, any random number can be included since it is not used at all.

markerSize<-c(8,10,16,16,1)

# Please change the argument circosLocation

SOFIA(data=data1,linkColor='chr',linkGeometry=c(.001,.1),linkRadius=c(.57,.57),
linksFlag=TRUE,chromoConfiguration=chromoConfiguration,plotBackground=plotBackground,
plotLocation=plotLocation,plotType=plotType,plotColor=plotColor,markerSize=markerSize,
circosLocation=NULL,tickSuffix='cM',returnConf=TRUE,circosDisplay=TRUE,
deleteData=TRUE,runCircos=TRUE,confName='circos.conf')
```

Index

mark0, 2

SOFIA, 3