

Package: CancerEvolutionVisualization (via r-universe)

September 11, 2024

Title Publication Quality Phylogenetic Tree Plots

Version 3.0.0

Date 2024-08-28

Description Generates tree plots with precise branch lengths, gene annotations, and cellular prevalence. The package handles complex tree structures (angles, lengths, etc.) and can be further refined as needed by the user.

License GPL-2

URL <https://github.com/uclahs-cds/package-CancerEvolutionVisualization>

BugReports

<https://github.com/uclahs-cds/package-CancerEvolutionVisualization/issues>

Depends R (>= 3.5.0), graphics, grid, gridExtra, gtable,

Imports grDevices, utils, BoutrosLab.plotting.general

Suggests testthat, knitr

VignetteBuilder knitr

LazyLoad yes

LazyData yes

Repository <https://uclahs-cds.r-universe.dev>

RemoteUrl <https://github.com/uclahs-cds/package-cancerevolutionvisualization>

RemoteRef HEAD

RemoteSha b27c69365fa35311be67998fcfa707a7e6c2688f

Contents

colours	2
create.ccf.heatmap	2
create.ccf.summary.heatmap	3
create.clone.genome.distribution.plot	5

create.cluster.heatmap	6
GRCh37	7
GRCh38	8
SRCGrob	8

Index	13
--------------	-----------

colours	<i>Colour scheme vector</i>
---------	-----------------------------

Description

Default colours used by CP polygons, etc.

Format

Character vector

create.ccf.heatmap	<i>Subclone Tree Plot</i>
--------------------	---------------------------

Description

Creates a heatmap of cancer cell fraction (CCF) distribution across tumour samples. The function is a wrapper around `BoutrosLab.plotting.general::create.heatmap()` with some changes in the default parameters. All parameter description are the same as in `BoutrosLab.plotting.general::create.heatmap()` except for `ccf.thres`.

Usage

```
create.ccf.heatmap(
  x,
  ccf.thres = NULL,
  cluster.dimensions = 'both',
  clustering.method = 'complete',
  distance.method = 'euclidean',
  xaxis.lab = '',
  xlab.label = 'Mutations',
  print.colour.key = FALSE,
  colour.scheme = c('white', 'blue'),
  ...
)
```

Arguments

<code>x</code>	Either a data-frame or a matrix from which the heatmap is to created
<code>ccf.thres</code>	CCF threshold to be applied to the heatmap. Values below the threshold will be set to 0. Defaults to NULL
<code>cluster.dimensions</code>	Defaults to “both”.
<code>clustering.method</code>	Defaults to “complete”.
<code>distance.method</code>	Defaults to “euclidean”.
<code>xaxis.lab</code>	Defaults to an empty string.
<code>xlab.label</code>	Defaults to “Mutations”.
<code>print.colour.key</code>	Defaults to FALSE.
<code>colour.scheme</code>	Defaults to <code>c('white', 'blue')</code> .
<code>...</code>	Pass through argument. See <code>BoutrosLab.plotting.general::create.heatmap()</code> for further details.

Value

A ‘grob’ object of the heatmap.

Author(s)

Helena Winata

`create.ccf.summary.heatmap`
Subclone Tree Plot

Description

Creates a heatmap of cancer cell fraction (CCF) distribution across tumour samples with clone IDs as a covariate beneath the heatmap. Subplot parameters controls the appearance of the heatmap and barplots. See `BoutrosLab.plotting.general::create.barplot()` or `BoutrosLab.plotting.general::create.heatmap()` for parameter description. Legend parameters are passed to `BoutrosLab.plotting.general::legend.grob()`.

Usage

```
create.ccf.summary.heatmap(
  DF,
  ccf.thres = NULL,
  median.col = 'median.ccf.per.sample',
  clone.order = NULL,
  sample.order = NULL,
```

```

hm.col.scheme = c('white', 'blue'),
subplot.xlab.cex = 1.2,
subplot.xaxis.cex = 1,
subplot.xaxis.fontface = 'bold',
subplot.xaxis.rot = 90,
subplot.ylab.cex = 1.2,
subplot.yaxis.cex = 1,
subplot.yaxis.fontface = 'bold',
hm.xaxis.rot = 90,
legend.size = 3,
legend.title.cex = 1.2,
legend.label.cex = 1,
...
);

```

Arguments

DF	A data-frame with the following column names: 'ID', 'SNV.id', 'clone.id', 'CCF'.
ccf.thres	CCF threshold to be applied to the heatmap. Values below the threshold will be set to 0. Defaults to NULL
median.col	Defaults to "median.ccf.per.sample"
clone.order	Defaults to NULL
sample.order	Defaults to NULL
hm.col.scheme	Heatmap colour scheme. Defaults to c('white', 'blue')
subplot.xlab.cex	Subplot parameter. Defaults to 1.2
subplot.xaxis.cex	Subplot parameter. Defaults to 1
subplot.xaxis.fontface	Subplot parameter. Defaults to "bold"
subplot.xaxis.rot	Subplot parameter. Defaults to 90
subplot.ylab.cex	Subplot parameter. Defaults to 1.2
subplot.yaxis.cex	Subplot parameter. Defaults to 1
subplot.yaxis.fontface	Subplot parameter. Defaults to "bold"
hm.xaxis.rot	Subplot parameter. Defaults to 90
legend.size	Legend parameter. Defaults to 3
legend.title.cex	Legend parameter. Defaults to 1.2
legend.label.cex	Legend parameter. Defaults to 1
...	Pass through argument. See BoutrosLab.plotting.general::create.multipanelplot() for further details.

Value

A ‘grob’ object of the summary plot.

Author(s)

Helena Winata

```
create.clone.genome.distribution.plot
```

Create Clone Genome Distribution Plot

Description

This function creates a plot showing the distribution of clones across the genome. It generates a scatter plot of the SNVs colored by clone ID and a density plot showing the density of each clone across the genome. The function can handle both single and multi-sample inputs.

Usage

```
create.clone.genome.distribution.plot(
  snv.df,
  genome.build = 'GRCh37',
  clone.order = NULL,
  clone.colours = NULL,
  filename = NULL,
  multi.sample = FALSE,
  ...
)
```

Arguments

snv.df	A data frame containing the SNV data. It must have columns 'chr', 'pos', and 'clone.id'. If multi.sample = TRUE, it must also have a column 'ID' specifying the sample ID for each SNV.
genome.build	The genome build to use. Defaults to "GRCh37".
clone.order	The order in which to plot the clones. If NULL, clones will be sorted alphabetically.
clone.colours	A named vector specifying the color to use for each clone. If NULL, colors will be automatically assigned.
filename	Directory or filepath to save the plot in. If multi.sample = TRUE, this must be a directory. if multi.sample = FALSE, this must be a filepath. If NULL, the plot will not be saved.
multi.sample	Logical indicating whether the input data contains multiple samples. Defaults to FALSE.
...	Additional arguments to be passed to BoutrosLab.plotting.general::create.multipanelplot().

Details

This function preprocesses the input data frame, extracts chromosome information, and iterates over each sample to create a clone genome distribution plot. For each sample, it calculates the density of each clone across the genome and creates a scatter plot of the SNVs colored by clone ID and a density plot showing the density of each clone.

Value

A 'grob' object.

Author(s)

Helena Winata, Selina Wu

create.cluster.heatmap

Subclone Tree Plot

Description

Creates a heatmap of cancer cell fraction (CCF) distribution across tumour samples with clone IDs as a covariate beneath the heatmap.

Usage

```
create.cluster.heatmap(  
  DF,  
  clone.colours = NULL,  
  height = 6,  
  width = 11,  
  xaxis.col = NULL,  
  legend.size = 3,  
  legend.title.cex = 1.2,  
  legend.label.cex = 1,  
  filename = NULL,  
  xlab.cex = 1.2,  
  xaxis.cex = 1,  
  xaxis.fontface = 'bold',  
  y.spacing = 1,  
  colour.scheme = c('white', 'blue'),  
  ...  
);
```

Arguments

DF	A data-frame with the following column names: 'ID', 'SNV.id', 'clone.id', 'CCF'.
clone.colours	Named list to provide a colour scheme for the clone ID covariate bar. If NULL, colours will be randomly generated. Defaults to NULL.
height	Defaults to 6
width	Defaults to 11
xaxis.col	Column in DF to extract x-axis labels from. Defaults to NULL.
legend.size	Width of the legend boxes in 'character' units. Defaults to 3
legend.title.cex	Size of titles in the legends. Defaults to 1.2
legend.label.cex	Size of text labels in the legends. Defaults to 1
filename	Filename for tiff output, or if NULL returns the trellis object itself. Defaults to NULL.
xlab.cex	Defaults to 1.2
xaxis.cex	Defaults to 1
xaxis.fontface	Defaults to "bold".
y.spacing	Spacing between heatmap and clone covariate bar. Defaults to 1
colour.scheme	Colour scheme for the heatmap. Defaults to c('white', 'blue').
...	Pass through argument. See BoutrosLab.plotting.general::create.heatmap() for further details.

Value

A 'grob' object of the heatmap.

Author(s)

Helena Winata

See Also

[create.ccf.heatmap](#)

GRCh37

GRCh37 Chromosom Information

Description

Chromosome information for the GRCh37 genome build. Used for plotting.

Format

data.frame

GRCh38

GRCh38 Chromosom Information

Description

Chromosome information for the GRCh38 genome build. Used for plotting.

Format

data.frame

SRCGrob

Subclone Tree Plot

Description

Creates a phylogenetic tree using subclonal reconstruction data. An optional node text data.frame can be used to annotate notable genes alongside branches.

Usage

```
SRCGrob(  
  tree,  
  node.text = NULL,  
  main = NULL,  
  horizontal.padding = 0.1,  
  scale1 = 1,  
  scale2 = 1,  
  yat = NULL,  
  yaxis1.label = NULL,  
  yaxis2.label = NULL,  
  xlab.cex = 1.55,  
  ylab.cex = 1.55,  
  xaxis.cex = 1.45,  
  yaxis.cex = 1.45,  
  xaxis.label = "CP",  
  label.cex = NA,  
  node.text.cex = 0.85,  
  main.y = NULL,  
  main.cex = 1.7,  
  node.radius = 0.1,  
  node.text.line.dist = 0.1,  
  colour.scheme = CancerEvolutionVisualization::colours,  
  add.normal = FALSE,  
  use.radians = FALSE,
```



```

normal.cex = 1,
sig.shape = 3,
label.nodes = TRUE,
disable.polygons = FALSE,
length.from.node.edge = TRUE,
size.units = "npc"
);

```

Arguments

tree	Tree structure data.frame
node.text	Dataframe for text labels to be displayed next to nodes
main	Main plot title
horizontal.padding	Increase/reduce the plot's horizontal padding proportionally. A positive value will expand the padding, and a negative value will reduce it.
scale1	Proportionally scale the values of the first branch length column in the tree input.
scale2	Proportionally scale the values of the second branch length column in the tree input.
yat	Specific values to be used for the y-axis ticks. A list is required, with each element corresponding to an axis.
yaxis1.label	Text label for the first, leftmost y-axis
yaxis2.label	Text label for the second, rightmost y-axis
xlab.cex	Font size for the x-axis label
ylab.cex	Font size for the y-axis labels
xaxis.cex	Font size for the x-axis tick labels
yaxis.cex	Font size for the y-axis tick labels
xaxis.label	Text label for the x-axis
label.cex	Font size for the node labels
node.text.cex	Font size for the node text
main.y	Move the main plot title position up or down
main.cex	Font size for the main plot title
node.radius	Node size
node.text.line.dist	Distance between node text and tree branches (as a value between 0 and 1)
colour.scheme	Vector of colour values to be used for CP polygons
add.normal	Adds a normal
use.radians	Unit to be used for "angle" column (degrees or radians)
normal.cex	Font size within the normal "box"
sig.shape	Changes the shape of the CP shading. Lower values are smoother.
label.nodes	Enable/disable node labels

`disable.polygons` Disables CP polygon drawing (even when CP values are provided)

`length.from.node.edge` Sets the branch length to be calculated from the edge of the node instead of the centre

`size.units` Grid units to be used for all specific size/length parameters

Value

A 'grob' of class "SRCGrob"

Author(s)

Dan Knight

Examples

```
# Simple Tree Plot
simple.tree <- data.frame(
  parent = c(NA, 1, 1)
);

SRCGrob(simple.tree);

# Specify Branch Lengths
branch.lengths.tree <- data.frame(
  simple.tree,
  length1 = c(10, 20, 15)
);

SRCGrob(branch.lengths.tree);

# Cellular Prevalence
CP.tree <- data.frame(
  branch.lengths.tree,
  CP = c(1, 0.3, 0.5)
);

SRCGrob(CP.tree);

# Scaled Branch Lengths
SRCGrob(
  branch.lengths.tree,
  scale1 = 0.8
);

# Override Branch Angles in Degrees
degrees.tree <- data.frame(
```

```
    parent = c(NA, 1, 2),
    angle = c(NA, NA, 90)
  );

SRCGrob(degrees.tree);

# Override Branch Angles in Radians
radians.tree <- data.frame(
  parent = c(NA, 1, 2),
  angle = c(NA, NA, pi / 2)
);

SRCGrob(
  radians.tree,
  use.radians = TRUE
);

# Plot Title
SRCGrob(
  simple.tree,
  main = 'Example Plot'
);

# Y-Axis Label
SRCGrob(
  branch.lengths.tree,
  yaxis1.label = 'SNVs'
);

# Y-Axis Ticks
yaxis1.ticks <- seq(0, 10, 2);

SRCGrob(
  branch.lengths.tree,
  yat = list(yaxis1.ticks)
);

# Normal
SRCGrob(
  simple.tree,
  add.normal = TRUE
);

# Nodeless Mode
nodeless.tree <- data.frame(
  parent = c(NA, 1, 2, 2),
  draw.node = c(TRUE, FALSE, TRUE, TRUE)
```

```
);  
SRCGrob(nodeless.tree);
```

Index

colours, [2](#)
create.ccf.heatmap, [2](#), [7](#)
create.ccf.summary.heatmap, [3](#)
create.clone.genome.distribution.plot,
[5](#)
create.cluster.heatmap, [6](#)

GRCh37, [7](#)
GRCh38, [8](#)

SRCGrob, [8](#)