

Package ‘ViSCA’

August 13, 2018

Type Package

Title Cell-Movie Analysis and Visualization

Version 0.1.0

Author Victoria Stefanou

Maintainer Victoria Stefanou <vic.stefanou@gmail.com>

Description Routines for analysis of cell-movies at single-cell, colony and population level, visualization of lineage and division trees, tracking errors' correction and movie creation.

Depends R (>= 3.5.1), igraph (>= 1.2.1)

Imports data.table (>= 1.11.4), ggplot2 (>= 3.0.0), graphics, grDevices, stats, utils, grid, EBImage (>= 4.22.0), fitdistrplus (>= 1.0-9), gplots (>= 3.0.1), gridExtra (>= 2.3), nlsMicrobio (>= 0.0-1), pdist (>= 1.2), R.matlab (>= 3.6.1), scatterplot3d (>= 0.3-41), jsonlite (>= 1.5)

SystemRequirements FFmpeg (optional)

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

URL <https://github.com/vicstefanou/ViSCA>

R topics documented:

add_attr_growth_fit_pars	2
add_attr_roc	3
add_branch	4
createFDT	5
createFLT	7
create_cell_life	8
create_movie	9
extract_branch	11
get_attr_names	11
get_attr_stats	12
get_attr_vals	13
get_cand_merge_cells	13
get_cand_mother_cells	14

get_cells	15
get_cell_attr_vals	16
get_cell_fam	16
get_DT_cell	17
get_LT_cell	18
import_basca	18
import_json	21
import_oufti	23
import_ss	24
isConnected	26
isSubtree	26
merge_cells	27
plot_baranyi	29
plot_col_tree	30
plot_dot_attr2	31
plot_dot_attr2_gen2	33
plot_dot_attr3	35
plot_dot_attr_fam	37
plot_dot_time_attr	39
plot_growth_attr	41
plot_growth_attr_cell	42
plot_growth_attr_fit	43
plot_hist_attr	46
plot_Ncells	48
plot_pdf_attr	49
plot_time_attr	52
plot_tree	54
plot_viobox_attr	56
save_tree	57
select_subtree	58
split_cell	59
unite_trees	60
view_cell	61
ViSCA	62
Index	64

add_attr_growth_fit_pars

Compute growth curves of an attribute

Description

Computes the growth curve of a numeric attribute for each cell in a division tree, by fitting a *linear* or *exponential* model to the its attribute's time-series.

Usage

```
add_attr_growth_fit_pars(LT, DT, attr, model = c("lin", "exp"), frameR)
```

Arguments

LT	The lineage tree, an object of class "igraph".
DT	The corresponding division tree of the LT, an object of class "igraph".
attr	The name of the attribute in the LT, a character string. It can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation", "frame" and "age".
model	A character string naming the growth model to be fitted: <ul style="list-style-type: none"> "lin" for fitting a <i>linear</i> model $y = a \cdot t + b$ using the <i>linear least squares</i> method provided by lm "exp" for fitting an <i>exponential</i> model $y = y_0 \cdot e^{kt}$ using the <i>non-linear least squares</i> method provided by nls where y represents the attr and t the time in <i>hours</i> (starting from 0).
frameR	Frame rate of the movie in <i>frames per minute</i> , a non-zero positive numeric value.

Details

The estimated parameters as well as the RMSE of the fitted growth model are added as attributes to the DT:

- When model = "lin":
 - "<attr>_a", a non-zero positive numeric value in units of attr per *hour*
 - "<attr>_b", a positive numeric value in units of attr
 - "<attr>_linRMSE"
- When model = "exp":
 - "<attr>_k", a non-zero positive numeric value in units of attr per *hour*
 - "<attr>_0", a non-zero positive numeric value in units of attr
 - "<attr>_expRMSE"

NA values are stored for cells that failed to fit the selected model as well as for cells that are not included in the analysis, as returned from [get_cells](#). Information messages for cells that failed to fit the model are printed on the screen.

Value

The updated DT with the new attributes added, an object of class "igraph".

add_attr_roc	<i>Add ROC attribute</i>
--------------	--------------------------

Description

Calculates the instantaneous Rate Of Change (ROC) of a numeric attribute of a lineage tree. The ROC is calculated for each cell in the tree and can be positive or negative.

Usage

```
add_attr_roc(LT, attr, norm = TRUE, frameR)
```

Arguments

LT	The connected lineage tree, an object of class "igraph".
attr	The name of the attribute in the LT, a character string. It can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation", "frame" and "age".
norm	A logical value (TRUE or FALSE) indicating the type of the ROC that will be calculated. When the default value TRUE is used, ROC is normalized and represents the percentage of change relative to the previous frame. When value FALSE is used, ROC represents the change per hour.
frameR	Frame rate of the movie in <i>frames per minute</i> , a non-zero positive numeric value. This argument is ignored in case norm = FALSE.

Details

The calculated ROC is added as an attribute to the LT:

- "d<attr>_norm", a numeric value in the range [-1, 1] in arbitrary units, when norm = TRUE
- "d<attr>", a numeric value in units of attr per *hour*, when norm = FALSE

NA values are stored for cells that are just born as well as for cells that are not included in the analysis, as returned from [get_cells](#).

Value

The updated LT with the new attribute added, an object of class "igraph".

See Also

[isConnected](#) for checking if a tree is connected.

add_branch	<i>Add a branch to an LT</i>
------------	------------------------------

Description

Adds a lineage tree as a branch to another lineage tree at a specific position, defined by the cell which will be the mother of the root of the branch.

Usage

```
add_branch(LT, branch, cell, cell_list, col_list, Ncols)
```

Arguments

LT	The lineage tree to which the branch will be added, an object of class "igraph".
branch	The connected lineage tree (motherless branch) which will be added as a branch to the LT, an object of class "igraph".
cell	The label of the cell in the LT where the branch will be added, a character string. It can be any valid candidate mother cell, as returned from get_cand_mother_cells .
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.

Value

The new LT with the branch added, an object of class "igraph".

Prerequisites

See the *Prerequisites* section of [get_cand_mother_cells](#).

See Also

[extract_branch](#) for extracting a branch from a tree, [isConnected](#) for checking if a tree is connected.

createFDT	<i>Create cells' FDT</i>
-----------	--------------------------

Description

Creates the cells' Forest of Division Trees (FDT) given the corresponding Forest of Lineage Trees (FLT).

Usage

```
createFDT(LTmain, minLife = 5, frameR)
```

Arguments

LTmain	The main part of the overall FLT, a connected lineage tree containing the imaginary <i>root</i> cells (object of class "igraph").
minLife	Minimum life in <i>frames</i> for a cell to be included in the analysis, a positive integer value. The default value is 5. Use value 0 to exclude all leaf cells from the analysis.
frameR	Frame rate of the movie in <i>frames per minute</i> , a non-zero positive numeric value.

Details

A continuous segment (sequence) of LT nodes between two successive cell *divisions* represents the lifespan of a cell. A cell is *divided* if

- it gives birth to 2 daughter cells in the next frame,
- it disappears from the field of view in the next frame or
- it is not linked to any cell instant in the next frame due to tracking errors.

The function creates the Forest of Division Tree (FDT) by reducing all LT cell segments down to single nodes (DT nodes), except for the imaginary *root* cells.

Each node of the FDT represents a cell at its full lifespan, having as character string attributes the concatenation of all attributes of the *collapsed* LT nodes by ", ", with the following exceptions:

- The "colony" attribute is not concatenated, since the ID of the colony from which the cell emanated characterizes both cell and its instants. The "colony" attribute is again a non-zero positive integer number.

- Attributes "age" and/or "generation" are also not concatenated in case they exist, since they are (re)evaluated by the function and updated/stored in the FLT. See below for more details.
- Each boolean attribute in the FLT forms also a boolean attribute in the FDT, with value (TRUE or FALSE) based on the majority vote of the corresponding values of the *collapsed* LT nodes.

The concatenated "name" attribute represents the labels of the *collapsed* cells (instances of the cell) and is renamed to "cellInstances". The "name" attribute is again a non-zero positive integer number stored as a character string, denoting the label of the cell in the FDT. Value "1" corresponds to the main *root* cell. Values "1+<i>" correspond to the colonies' *root* cells, where "<i>" is the colony ID. The rest values correspond to the cells.

For each numeric attribute in the FLT (except for "colony" and "frame", plus "age" and/or "generation" in case they exist), the concatenation represents the cell's time-series of the attribute. Given each cell's time-series of an attribute "<attr>", the following numeric life attributes are estimated and stored as attributes in the corresponding FDT node:

- "<attr>_birth" is the "<attr>" value of the first instant of the cell
- "<attr>_division" is the "<attr>" value of the last instant of the cell
- "<attr>_mean" is the *mean* of "<attr>"
- "<attr>_sd" is the *standard deviation* of "<attr>", or NA in case the cell has only one instant
- "<attr>_min" is the minimum value of "<attr>"
- "<attr>_max" is the maximum value of "<attr>"

These numeric attributes are in units of "<attr>".

The following attributes are also life attributes and are stored in each DT node:

- "generation" is the ID of the generation of the cell, a positive integer value. This value is also updated/stored in the "generation" attribute of the instances of the cell in the FLT, since the ID of the generation characterizes both cell and its instances.
- "birthTime" is the ID of the frame at which the cell is firstly spotted (born), a non-zero positive integer value. This value is the "frame" value of the first instant of the cell.
- "divisionTime" is the ID of the frame at which the cell is lastly spotted (a frame before its *division*), a non-zero positive integer value. This value is the "frame" value of the last instant of the cell.
- "lifeFrames" is the duration of the cell life in *frames*, a non-zero positive integer value. This value is computed as

$$lifeFrames = divisionTime - birthTime + 1$$

- "lifeHours" is the duration of the cell life in *hours*, a non-zero positive numeric value. This value is computed as

$$lifeHours = \frac{lifeFrames}{60 \cdot frameR}$$

- "isConsidered" is a logical value (TRUE or FALSE) indicating whether the cell will be included in the analysis or not. This value is FALSE for the imaginary *root* cells. The value for the cells is computed based on the minLife argument. It is FALSE for all leaf cells and TRUE for the rest, in case minLife = 0, or FALSE for all leaf cells with "lifeFrames" <= minLife and TRUE for the rest, in case minLife != 0.

Value

A named list with the following components:

DTmain	The corresponding main part of the overall FDT, a connected division tree containing the imaginary <i>root</i> cells (object of class "igraph").
LTmain	The updated LTmain with the attributes "generation" and "age" updated/added, an object of class "igraph". The "age" attribute denotes the age of each cell instant in <i>frames</i> .
Ngens	Number of generations in the movie, a non-zero positive integer value. IDs of generations are in the range [0, Ngens-1].

See Also

[isConnected](#) for checking if a tree is connected, [save_tree](#) for saving a tree on disc.

createFLT

Create cells' FLT

Description

Creates the cells' Forest of Lineage Trees (FLT) given a list containing all the cell instants of the movie.

Usage

```
createFLT(cell_list, Ncols)
```

Arguments

cell_list	A list containing all the cell instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.

Details

Apart from the cell instants of the movie contained in the *cell_list*, FLT nodes include an imaginary main *root* cell as well as imaginary *root* cells for each colony. Colonies' *root* cells are daughters of the main *root* cell. Cell instants of the first frame of the movie are daughters of the corresponding colony's *root* cell. The imaginary *root* cells are used to facilitate the tree representation of the movie and the colony tracking and are automatically excluded from the analysis.

Each node of the FLT has as attributes all numeric and boolean values existing as components in the corresponding element of the *cell_list*. The imaginary *root* cells have value -1 in all numeric attributes and value FALSE in all boolean attributes. The following character string values also form attributes of each FLT node:

- "name" is the label of the cell in the FLT, a non-zero positive integer number stored as a character string. Value "1" corresponds to the main *root* cell. Values "1+<i>" correspond to the colonies' *root* cells, where "<i>" is the colony ID. The rest values correspond to the cell instants in the *cell_list* (1-1 correspondence).
- "cellName", as in the *cell_list*. Value "root" is used for the main *root* cell. Values "colony<i>" are used for the colonies' *root* cells, where "<i>" is the colony ID.

- "colId", as in the cell_list, but here stored as a character string. Value "-1" is used for the imaginary *root* cells.

NOTE: This attribute is stored iff it exists as component in the elements of the cell_list.

Value

A named list with the following components:

LTmain	The main part of the overall FLT, a connected lineage tree containing the imaginary <i>root</i> cells (object of class "igraph").
	Attribute "colony" in the LTmain depicts the starting colony of each cell instant (i.e. the colony of the corresponding colony's <i>root</i> cell from which a cell instant emanated). This approach is necessary in order to keep track of merging colonies.
branches	A list with the motherless branches of the overall FLT. Each branch (element of the list) is a connected lineage tree (object of class "igraph"). Motherless branches arise from tracking errors, in case a cell instant (root of the branch) fails to be connected to any cell instant of the previous frame, or when a cell instant (root of the branch) just entered the field of view.

See Also

[save_tree](#) for saving a tree on disc, [add_branch](#) for connecting a motherless branch to a lineage tree.

create_cell_life	<i>Visualize cell life</i>
------------------	----------------------------

Description

Creates the life images of a set of given cells.

Usage

```
create_cell_life(DT, cells = "all", cell_list, col_list, Ncols,
  saveT = c("sep", "comb"), savePath = getwd())
```

Arguments

DT	The division tree where the cells specified in cells belong, an object of class "igraph".
cells	The labels of the cells in the DT whose life images will be created, a vector of character strings. They can be any non-root cells, as returned from get_cells . The default value "all" stands for all non-root cells.
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.
saveT	A character string naming the options of image saving:

- "sep" saves a separate .png file for each frame of the cell's lifespan, named as "<cell>_f<first_frame>_<i>.png". Image files for each cell will be saved in a directory named after the cell.
- "comb" saves a single .png file with all frames of the cell's lifespan combined in *left-to-right, top-to-bottom* order, named as "<cell>_f<first_frame>.png".

"<cell>" is the label of the cell, "<first_frame>" is its birth frame and "<i>" is the instant of the cell starting from 1.

savePath A character string naming the absolute path of the directory where the cell life images will be saved (excluding the last "/"). The default value is the current working directory `getwd()`.

NOTE: The components should be separated by "/" on Windows.

Details

Separate life image(s) for each cell specified in `cells` are generated. The cell is viewed in its colony for every frame of its lifespan and is marked as red. The rest cells of the colony are marked as white.

Prerequisites

This function can be used by *BaSCA* users, importing the data with `import_basca`.

Users of *Oufti* or *SuperSegger* who imported the data with `import_oufti` or `import_ss`, respectively, are **excluded** from using this function, as no colony list was returned.

If `import_json` was used for importing the data, it is necessary that cell list elements have the `pixelList` and `colId` components and colony list elements have the `colImage` component. See `import_json` for more details. In other case, this function cannot be used (throws an error).

create_movie	Create movie
--------------	--------------

Description

Creates the movie of the segmented cells.

Usage

```
create_movie(overallLT, LT, cell_list, col_list, frameH, frameW, Ncols,
  attrC = "", unitC = "", NC = NULL, savePath = getwd())
```

Arguments

overallLT The lineage tree the cells of which will be visualized in the created movie, an object of class "igraph". Cells that do not belong to the LT are colored white.

For visualizing all cells extracted from the analysis of the movie, use the tree returned from `unite_trees` when called for all parts of the overall FLT (main part and possible motherless branches).

LT	The lineage tree the cells of which will be colored in the created movie, an object of class "igraph". This tree must be a subtree of the overallLT regarding the included cells in the analysis.
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
frameH	Frame image height in <i>pixels</i> , a non-zero positive integer value.
frameW	Frame image width in <i>pixels</i> , a non-zero positive integer value.
Ncols	Number of colonies in the movie, a non-zero positive integer value.
attrC	The name of the attribute in the LT by which the cells will be colored, a character string. It can be any numeric or boolean attribute, as returned from get_attr_names . Coloring is applied to all non-root cells of the LT, as returned from get_cells , except for cells with NA value in this attribute which are colored gray. When the default value "" (the empty character) is used, all cells of the LT are colored gray. Cells that belong to the overallLT but not to the LT are always colored white.
unitC	The unit of attrC, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attrC is in arbitrary units. This argument is ignored in case attrC = "".
NC	Number of colonies in the movie (if attrC = "colony") or number of generations in the movie (if attrC = "generation"), a non-zero positive integer value. This argument is ignored in case attrC != "colony" and attrC != "generation".
savePath	A character string naming the absolute path of the directory where the generated .png files will be saved (excluding the last "/"). The default value is the current working directory getwd(). Image files are named as "frame_<i>.png", where "<i>" is the frame ID. The movie file is named as "movie.mp4". All files will be saved in folder "<attrC>" (or "noColor" in case attrC = ""), created under the specified directory.

NOTE: The components should be separated by "/" on Windows.

Details

A separate image for each frame existing in the overallLT is generated.

Prerequisites

This function can be used by *BaSCA* users, importing the data with [import_basca](#).

Users of *Oufti* or *SuperSegger* who imported the data with [import_oufti](#) or [import_ss](#), respectively, are **excluded** from using this function, as no colony list was returned.

If [import_json](#) was used for importing the data, it is necessary that cell list elements have the pixelList and colId components and colony list elements have the ULcorner component. See [import_json](#) for more details. In other case, this function cannot be used (throws an error).

FFmpeg is **required** in the system in order to automatically convert the series of the generated .png files to an .mp4 file, at 5 fps (frames per second). Image files for more than two consecutive frames starting from frame 1 must have been generated.

See Also

[isSubtree](#) for checking if a tree is a subtree of another tree.

extract_branch	<i>Extract a branch from a tree</i>
----------------	-------------------------------------

Description

Extracts a branch from a lineage or division tree. The branch for extraction is defined by its root cell.

Usage

```
extract_branch(tree, cell)
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
cell	The label of the cell in the tree which is the root of the branch to be extracted, a character string. It can be any cell, as returned from get_cells .

Value

A named list with the following components:

treeNew	The new tree with the branch extracted, an object of class "igraph".
branch	The extracted connected motherless branch, an object of class "igraph".

See Also

[add_branch](#) for connecting a motherless branch to a lineage tree.

get_attr_names	<i>Get attributes' names</i>
----------------	------------------------------

Description

Returns the names of the attributes of a lineage or division tree.

Usage

```
get_attr_names(tree, type = c("n", "b", "c"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
type	A character string naming the type of attributes to be returned: <ul style="list-style-type: none"> • "n" for numeric attributes • "b" for boolean attributes • "c" for character string attributes

Value

The corresponding attributes' names, a vector of character strings.

get_attr_stats	<i>Get statistic of an attribute</i>
----------------	--------------------------------------

Description

Returns the statistic of a numeric attribute of a lineage or division tree.

Usage

```
get_attr_stats(tree, treeT = c("LT", "DT"), attr, stat = c("mean", "median",
  "sd", "min", "max"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr	The name of the attribute in the tree, a character string. It can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation" and "frame".
stat	A character string naming the statistic to be returned: <ul style="list-style-type: none"> • "mean" for the <i>mean</i> • "median" for the <i>median</i> • "sd" for the <i>standard deviation</i> • "min" for the minimum value • "max" for the maximum value

Details

The statistic is calculated considering all cells that are included in the analysis, as returned from [get_cells](#), except for cells with NA value in attr.

Value

A named list with the following components:

Ncells	Number of cells, a positive integer value.
value	The corresponding statistic of attribute attr, a numeric value, or NA.

get_attr_vals	<i>Get attribute's values</i>
---------------	-------------------------------

Description

Returns the values of an attribute for all cells of a lineage or division tree, as returned from [get_cells](#).

Usage

```
get_attr_vals(tree, attr)
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
attr	The name of the attribute in the tree, a character string. It can be any numeric, boolean or character string attribute, as returned from get_attr_names .

Value

A vector with the values of attribute attr.

get_cand_merge_cells	<i>Find candidate merge cells</i>
----------------------	-----------------------------------

Description

Returns the candidate cells for a given cell to be merged with, in the same or different lineage tree.

Usage

```
get_cand_merge_cells(LT, cell, LTcand = NULL, cell_list, col_list, Ncols,
  show = TRUE)
```

Arguments

LT	The lineage tree where the cell specified in cell belongs, an object of class "igraph".
cell	The label of the cell in the LT for which to find candidate merge cells, a character string. It can be any non-root cell, as returned from get_cells .
LTcand	The lineage tree where the candidate merge cells of cell will belong, an object of class "igraph". When the default value NULL is used, candidate merged cells will belong in the LT.
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.
show	A logical value (TRUE or FALSE) indicating whether view_cell will be called for the cell and the candidate merge cells. This capability is useful in order to choose the desired merge cell and call merge_cells . The default value is TRUE.

Details

Candidate merge cells are cells of the same colony and frame with least one common pixel with the given cell.

Value

The labels of the candidate merge cells, a vector of character strings. In case no candidate merge cells are found, NULL is returned.

Prerequisites

This function can be used by *BaSCA* users **only**, importing the data with [import_basca](#).

get_cand_mother_cells *Find candidate mothers*

Description

Returns the candidate mother cells for the root of a motherless branch in a lineage tree.

Usage

```
get_cand_mother_cells(branch, LT, Nd = 1, Ncands = 5, cell_list, col_list,
  Ncols, show = FALSE)
```

Arguments

branch	The connected lineage tree (motherless branch) for which to find candidate mothers, an object of class "igraph".
LT	The lineage tree where the candidate mothers of the branch will belong, an object of class "igraph".
Nd	Maximum number of daughters for a cell in the LT to be considered as a candidate mother, a positive integer value. The default value is 1.
Ncands	Maximum number of candidate mothers to be returned, a non-zero positive integer value. The default value is 5. Use value NULL to return all candidate mothers.
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.
show	A logical value (TRUE or FALSE) indicating whether view_cell will be called for the root of the branch and the candidate mothers. This capability is useful in order to choose the desired mother cell and call add_branch . The default value is FALSE. See the <i>Prerequisites</i> section of view_cell in order to be able to use value TRUE.

Value

The labels of the candidate mothers, a vector of character strings. Candidate mothers are sorted by ascending distance between their centroid and the centroid of the root of the branch, which implies best matching. In case no candidate mothers are found, NULL is returned.

Prerequisites

This function can be used by *BaSCA* users, importing the data with `import_basca`.

Users of *Oufiti* or *SuperSegger* who imported the data with `import_oufti` or `import_ss`, respectively, are **excluded** from using this function, as no colony list was returned.

If `import_json` was used for importing the data, it is necessary that cell list elements have the centroid and colId components and colony list elements have the ULcorner component. See `import_json` for more details. In other case, this function cannot be used (throws an error).

Candidate mothers are cells of the same colony in the previous frame.

See Also

`isConnected` for checking if a tree is connected.

get_cells	<i>Get cells of a tree</i>
-----------	----------------------------

Description

Returns the labels of the cells in a lineage or division tree.

Usage

```
get_cells(tree, treeT = c("LT", "DT"), type = c("all", "nr", "inc"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> "LT" if tree is a lineage tree "DT" if tree is a division tree This argument is ignored in case type = "all"
type	A character string naming the type of cells to be returned: <ul style="list-style-type: none"> "all" for all cells (including any existing imaginary <i>root</i> cells) "nr" for all non-root cells (excluding any existing imaginary <i>root</i> cells) "inc" for all cells included in the analysis

Value

The labels of the corresponding cells, a vector of character strings.

get_cell_attr_vals	<i>Get attributes' values of a cell</i>
--------------------	---

Description

Returns the values of the attributes of a cell in a lineage or division tree.

Usage

```
get_cell_attr_vals(tree, treeT = c("LT", "DT"), cell)
```

Arguments

tree	The lineage or division tree where the cell specified in cell belongs, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
cell	The label of the cell in the tree whose attributes' values will be returned, a character string. It can be any non-root cell, as returned from get_cells .

Value

The attributes' values of the cell, a named list.

get_cell_fam	<i>Get family cells of a cell</i>
--------------	-----------------------------------

Description

Returns the labels of the family cells of a cell in a lineage or division tree.

Usage

```
get_cell_fam(tree, treeT = c("LT", "DT"), cell, type = c("m", "gm", "d",
"gd", "s", "c"))
```

Arguments

tree	The connected lineage or division tree where the cell specified in cell belongs, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
cell	The label of the cell in the tree whose family cells will be returned, a character string. It can be any non-root cell, as returned from get_cells .
type	A character string naming the type of the family cells to be returned:

- "m" for mother
- "gm" for grandmother
- "d" for daughters
- "gd" for granddaughters
- "s" for sibling
- "c" for cousins

Value

The labels of the corresponding family cells of the `cell`, a vector of character strings. In case no family cells are found, `NULL` is returned.

See Also

[isConnected](#) for checking if a tree is connected.

get_DT_cell	<i>Get the label of the cell of a cell instant</i>
-------------	--

Description

Returns the label of a cell in a division tree, given the label of an instant of the cell in the corresponding lineage tree.

Usage

```
get_DT_cell(LT, cell, DT)
```

Arguments

LT	The lineage tree where the cell specified in <code>cell</code> belongs, an object of class "igraph".
cell	The label of the cell in the LT, a character string. It can be any non-root cell, as returned from get_cells .
DT	The corresponding division tree of the LT, an object of class "igraph".

Value

A named list with the following components:

cell	The label of the cell in the DT, a character string.
instant	The instant of the cell, a non-zero positive integer value.

In case `cell` is not found in the DT, `NULL` is returned.

See Also

[get_LT_cell](#) for the reverse.

get_LT_cell	<i>Get the label of an instant of a cell</i>
-------------	--

Description

Returns the label of an instant of a cell in a lineage tree, given the label of the cell in the corresponding division tree.

Usage

```
get_LT_cell(DT, cell, instant)
```

Arguments

DT	The division tree where the cell specified in <code>cell</code> belongs, an object of class "igraph".
cell	The label of the cell in the DT, a character string. It can be any non-root cell, as returned from get_cells .
instant	The instant of the cell specified in <code>cell</code> , a non-zero positive integer value.

Value

The label of the cell instant in the corresponding lineage tree of the DT, a character string. In case `cell` has less instants than the specified `instant`, NULL is returned.

See Also

[get_DT_cell](#) for the reverse.

import_basca	<i>Import BaSCA data</i>
--------------	--------------------------

Description

Imports a .mat file generated by *BaSCA* and converts it into a cell list and colony list containing all the cell instants and colony instants of the movie, respectively.

Usage

```
import_basca(file, pixelR, ringW = 10)
```

Arguments

file	A character string naming the .mat file generated by <i>BaSCA</i> (including the suffix ".mat") from which the data is to be imported. If it does not contain an absolute path, the file name is relative to the current working directory, <code>getwd()</code> .
pixelR	NOTE: The components should be separated by "/" on Windows. The pixel ratio in units of length, a non-zero positive numeric value.
ringW	The width of the ring (in <i>pixels</i>) which marks the boundary pixels of a colony, a non-zero positive integer value. This value is used to decide whether a cell is on the boundary of its colony or not. The default value is 10.

Value

A named list with the following components:

col_list A list containing all the colony instants of the movie. Each element of the list is a named list with the following components:

- **colName** is the name of the colony instant, a character string in the format "**f**<frame>_<colony>", where "<frame>" and "<colony>" is the ID of the frame and colony (in the frame) of the colony instant, respectively.
- **prev_colName** is a vector of character strings containing the **colName** of the corresponding colony instant(s) in the previous frame. For colony instants that do not have a corresponding colony instant in the previous frame, this is equal to "**f0_c0**".
- **next_colName** is a character string containing the **colName** of the corresponding colony instant in the next frame. For colony instants that do not have a corresponding colony instant in the next frame, this is equal to "**f00_c0**".
- **colImage** is the mask of the box surrounding the colony instant, a matrix of 0 and 1. 1s denote the pixels of cells and 0s the background pixels.
- **ULcorner** is an 1x2 matrix of non-zero integer values denoting the upper-left pixel of the box surrounding the colony instant in global (frame) coordinates. The first integer represents the row and the second the column of the pixel.
- **colBoundaryPixels** is a Nx2 matrix of non-zero integer values denoting the boundary pixels of the colony instant in colony coordinates (i.e. relative to the **colImage**). Each one of the N rows indicates a boundary pixel of the colony instant detected based on the **ringW** argument. The first column represents the row and the second the column of the boundary pixel.
- **colCentroid** is an 1x2 matrix of non-zero numeric values denoting the centroid (geometric center) of the colony instant in colony coordinates (i.e. relative to the **colImage**).

cell_list A list containing all the cell instants of the movie. Each element of the list is a named list with the following components:

- **cellName** is the name of the cell, a character string in the format "**x**<----->_**y**<----->_**f**<f
- **frame** is the ID of the frame of the cell, a non-zero positive integer number.
- **colony** is the ID of the colony of the cell in the frame, a non-zero positive integer number.
- **daughterIds** is a vector of character strings containing the **cellName** of the linked cell(s) in the next frame, or NULL in case no such cells exist.
- **colId** is a *pointer* to the corresponding colony instant of the cell in the **col_list**, a non-zero positive integer value.

Colonies that entered the field of view at a time point and did not exist from the beginning of the movie (i.e. from the first frame) should not have tracked cells, until they merge (if this is the case) with another existing colony. This means that no element should *point* to such colony instants.

- **pixelList** is a Nx2 matrix of non-zero integer values denoting the pixels of the cell in colony coordinates (i.e. relative to the **colImage** of the *colIdth* element in the **col_list**). Each one of the N rows indicates a pixel of the cell. The first column represents the row and the second the column of the pixel.

- `boundaryPixelList` is a Nx2 matrix of non-zero integer values denoting the boundary pixels of the cell in colony coordinates (i.e. relative to the `colImage` of the $colId^{th}$ element in the `col_list`). Each one of the N rows indicates a boundary pixel of the cell. The first column represents the row and the second the column of the boundary pixel.
- `centroid` is an 1x2 matrix of non-zero numeric values denoting the centroid (geometric center) of the cell in colony coordinates (i.e. relative to the `colImage` of the $colId^{th}$ element in the `col_list`). It is the *mean* of the `pixelList` by column.
- `length` is the length of the cell in units of length, a non-zero positive numeric value.
- `width` is the width of the cell in units of length, a non-zero positive numeric value.
- `LW` is the length-to-width ratio, a non-zero positive numeric value.
- `area` is the area of the cell in squared units of length, a non-zero positive numeric value.
- `perimeter` is the perimeter of the cell in units of length, a non-zero positive numeric value.
- `minorAxis` is the short axis of the ellipse surrounding the cell in units of length, a non-zero positive numeric value.
- `majorAxis` is the long axis of the ellipse surrounding the cell in units of length, a non-zero positive numeric value.
- `eccentricity` is a numeric value in the range $[0, 1]$ defining the eccentricity of the cell.
- `orientation` is a numeric value in the range $[0, 360)$ defining the orientation of the cell in *degrees*.
- `solidity` is a numeric value in the range $[0, 1]$ defining the solidity of the cell.
- `distFromCentroid` is the distance of the centroid of the cell from the centroid of its colony (i.e. the euclidean distance between the centroid and the `colCentroid` of the $colId^{th}$ element in the `col_list`) in units of length, a non-zero positive numeric value.
- `isOnBoundary` is a logical value (TRUE or FALSE) indicating whether the cell is on the boundary of its colony or not. It is TRUE in case there is more than 66 the `colBoundaryPixels` of the $colId^{th}$ element in the `col_list`, FALSE otherwise.
- `fluorescenceInt<i>` is a Nx1 matrix of positive numeric values indicating the cell fluorescence intensity of channel <i>. The values correspond to each one of the pixels in `pixelList` (1-1 correspondence).
- `fluorescenceInt<i>.mean` is the *mean* cell fluorescence intensity of channel <i>, a positive numeric value.
- `fluorescenceInt<i>.std` is the *standard deviation* of the cell fluorescence intensity of channel <i>, a positive numeric value.
- `fluorescenceInt<i>.coverage` is the percentage of the cell area covered by fluorescence of channel <i>, a numeric value in the range $[0, 1]$.

The `fluorescenceInt<i>*` components are included in case they were extracted by *BaSCA*.

Nframes

Number of frames in the movie, a non-zero positive integer value. IDs of frames are in the range $[1, Nframes]$.

Ncols	Number of colonies in the movie, a non-zero positive integer value. IDs of colonies are in the range [1, Ncols]. This value corresponds to the number of colonies at the start of the movie.
frameH	Frame image height in <i>pixels</i> , a non-zero positive integer value.
frameW	Frame image width in <i>pixels</i> , a non-zero positive integer value.

References

A. Balomenos, P. Tsakanikas, Z. Aspidou, A. Tampakaki, K. Koutsoumanis and E. Manolakis, “Image analysis driven single-cell analytics for systems microbiology”, BMC Systems Biology, vol. 11, no. 1, 2017.

import_json	<i>Import custom-made data</i>
-------------	--------------------------------

Description

Imports a .json file containing all the cell instants of the movie and another .json file containing all the colony instants of the movie.

Usage

```
import_json(file_cells, file_cols = NULL, frameH, frameW)
```

Arguments

file_cells, file_cols	Character strings naming the .json files (including the suffix ".json") containing all the cell and colony instants of the movie, respectively, from which the data is to be imported. If a string does not contain an absolute path, the file name is relative to the current working directory, getwd(). See the <i>Details</i> field for information about the format of these files.
	When the default value file_cols = NULL is used, no colony list is imported.
	NOTE: The components should be separated by "/" on Windows.
frameH	Frame image height in <i>pixels</i> , a non-zero positive integer value. This argument is ignored in case file_cols = NULL.
frameW	Frame image width in <i>pixels</i> , a non-zero positive integer value. This argument is ignored in case file_cols = NULL.

Details

The file_cols must be a JSON array of objects. Each object denotes a colony instant of the movie and should contain the following key-value pairs (otherwise an error is produced):

- colName is the name of the colony instant, a character string in the format "f<frame>_c<colony>", where "<frame>" and "<colony>" is the ID of the frame and colony (in the frame) of the colony instant, respectively.

- `prev_colName` is a JSON array of character strings containing the `colName` of the corresponding colony instant(s) in the previous frame. In case one such colony instant exists, it can also be a character string. For colony instants that do not have a corresponding colony instant in the previous frame, it should be equal to "`f0_c0`".
- `next_colName` is a character strings containing the `colName` of the corresponding colony instant in the next frame. For colony instants that do not have a corresponding colony instant in the next frame, it should be equal to "`f00_c0`".
- `colImage` is the mask of the box surrounding the colony instant, a JSON array of H arrays, where H is the height of the box. Each element of the array is a JSON array of W 0s and/or 1s, where W is the width of the box. 1s denote the pixels of cells and 0s the background pixels.

NOTE: This key-value pair is not necessary.

- `ULcorner` is a JSON array of 2 non-zero integer values denoting the upper-left pixel of the box surrounding the colony instant in global (frame) coordinates. The first integer represents the row and the second the column of the pixel.

NOTE: This key-value pair is not necessary unless the key `colImage` is contained (an error is produced).

The `file_cells` must be a JSON array of objects. Each object denotes a cell instant of the movie. Key-value pair(s) denoting numeric or boolean attribute(s) should be contained in every object. The following key-value pairs are also required to be contained in every object (otherwise an error is produced):

- `cellName` is the name of the cell, a character string in the format "`<cell>_f<frame>`".
- `frame` is the ID of the frame of the cell, a non-zero positive integer number.
- `colony` is the ID of the colony of the cell in the frame, a non-zero positive integer number. If the whole frame is treated as a single colony, value 1 must be used for all cells.
- `daughterIds` is a JSON array of character strings containing the `cellName` of the linked cell(s) in the next frame. In case one such cell exists, it can also be a character string. In case no such cells exist, it can either be NULL or omitted.
- `colId` is a *pointer* to the corresponding colony instant of the cell in the `file_cols`, a non-zero positive integer value.

Colonies that entered the field of view at a time point and did not exist from the beginning of the movie (i.e. from the first frame) should not have tracked cells, until they merge (if this is the case) with another existing colony. This means that no object should *point* to such colony instants.

NOTE: This key-value pair is not necessary unless `file_cols` != NULL (an error is produced) and should be omitted if `file_cols` = NULL (a warning is produced).

- `pixelList` is a JSON array of arrays. Each element of the array is a JSON array of 2 non-zero integer values, indicating a pixel of the cell in colony coordinates (i.e. relative to the `colImage` key of the `colIdth` object in `file_cols`). The first value represents the row and the second the column of the pixel.

NOTE: This key-value pair is not necessary. It should be omitted if `file_cols` = NULL or if objects in `file_cols` do not contain the key `colImage` (a warning is produced).

Value

A named list with the following components:

col_list	A list containing all the colony instants of the movie, or NULL if file_cols = NULL. Each element of the list is a named list having as components the corresponding key-value pairs in the file_cols.
cell_list	A list containing all the cell instants of the movie. Each element of the list is a named list having as components the corresponding key-value pairs in the file_cells.
	In case the key pixelList is contained, centroid is also computed as the <i>mean</i> of the pixelList by column. centroid is an 1x2 matrix of non-zero numeric values denoting the centroid (geometric center) of the cell in colony coordinates (i.e. relative to the colImage of the <i>colIdth</i> element in the col_list).
Nframes	Number of frames in the movie, a non-zero positive integer value. IDs of frames are in the range [1, Nframes].
Ncols	Number of colonies in the movie, a non-zero positive integer value. IDs of colonies are in the range [1, Ncols]. This value corresponds to the number of colonies at the start of the movie.
import_oufti	<i>Import Oufti data</i>

Description

Imports a .csv file generated by *Oufti* and converts it into a cell list containing all the cell instants of the movie.

Usage

```
import_oufti(file, pixelR)
```

Arguments

file A character string naming the .csv file generated by *Oufti* (including the suffix ".csv") from which the data is to be imported. If it does not contain an absolute path, the file name is relative to the current working directory, getwd().

NOTE: The components should be separated by "/" on Windows.

Before using this function, the user has to open the .csv file in a text editor and manually delete or comment (by adding the character "%" at the beginning) all lines of the file until the line with the "% parameter values" string.

pixelR The pixel ratio in units of length, a non-zero positive numeric value.

Details

Oufti treats the whole frame as a single colony.

Oufti does not export any information about either the pixels of each cell or the pixels each colony.

Oufti users importing the data with this function are **excluded** from using [plot_col_tree](#), [split_cell](#), [get_cand_merge_cells](#), [merge_cells](#), [get_cand_mother_cells](#), [add_branch](#), [create_movie](#), [create_cell_life](#) and [view_cell](#).

Value

A named list with the following components:

col_list	NULL
cell_list	<p>A list containing all the cell instants of the movie. Each element of the list is a named list with the following components:</p> <ul style="list-style-type: none"> • cellName is the name of the cell, a character string in the format "c<cellId>_f<frame>" • frame is the ID of the frame of the cell, a non-zero positive integer number • colony is the ID of the colony of the cell in the frame, equal to 1 • daughterIds is a vector of character strings containing the cellName of the linked cell(s) in the next frame, or NULL in case no such cells exist • length is the length of the cell in units of length, a non-zero positive numeric value • area is the area of the cell in squared units of length, a non-zero positive numeric value • polarity is a numeric value defining the polarity of the cell
Nframes	Number of frames in the movie, a non-zero positive integer value. IDs of frames are in the range [1, Nframes].
Ncols	Number of colonies in the movie, equal to 1. IDs of colonies are in the range [1, Ncols].
frameH	NULL
frameW	NULL

References

<http://oufti.org/>

import_ss	<i>Import SuperSegger data</i>
-----------	--------------------------------

Description

Imports the cell .mat data files generated by *SuperSegger* and converts them into a cell list containing all the cell instants of the movie.

Usage

```
import_ss(path, pixelR)
```

Arguments

path	<p>A character string naming the xy#/cell directory with the cell .mat data files generated by <i>SuperSegger</i> (excluding the last "/") from which the data is to be imported. If it does not contain an absolute path, it is relative to the current working directory, getwd().</p> <p>NOTE: The components should be separated by "/" on Windows.</p>
pixelR	The pixel ratio in units of length, a non-zero positive numeric value.

Details

SuperSegger treats the whole frame as a single colony.

SuperSegger users importing the data with this function do not import any information about either the pixels of each cell or the pixels each colony.

SuperSegger users importing the data with this function are **excluded** from using [plot_col_tree](#), [split_cell](#), [get_cand_merge_cells](#), [merge_cells](#), [get_cand_mother_cells](#), [add_branch](#), [create_movie](#), [create_cell_life](#) and [view_cell](#).

Value

A named list with the following components:

col_list	NULL
cell_list	<p>A list containing all the cell instants of the movie. Each element of the list is a named list with the following components:</p> <ul style="list-style-type: none"> • cellName is the name of the cell, a character string in the format "c<cellId>_f<frame>". • frame is the ID of the frame of the cell, a non-zero positive integer number. • colony is the ID of the colony of the cell in the frame, a non-zero positive integer number. • daughterIds is a vector of character strings containing the cellName of the linked cell(s) in the next frame, or NULL in case no such cells exist. • length is the length of the cell in units of length, a non-zero positive numeric value. • width is the width of the cell in units of length, a non-zero positive numeric value. • area is the area of the cell in squared units of length, a non-zero positive numeric value. • minorAxis is the short axis of the ellipse surrounding the cell in units of length, a non-zero positive numeric value. • majorAxis is the long axis of the ellipse surrounding the cell in units of length, a non-zero positive numeric value. • orientation is a numeric value in the range $[0, 360)$ defining the orientation of the cell in <i>degrees</i>. • cellDist is the distance of the cell to the edge of the colony in units of length, a positive numeric value. • edgeFlag is a logical value (TRUE or FALSE) indicating whether the cell is at the edge of the image or not. • contactHist is a logical value (TRUE or FALSE) indicating whether the cell is in contact with other cell or not. • fluorescenceInt<i>.mean is the <i>mean</i> cell fluorescence intensity of channel <i>, a positive numeric value. • fluorescenceInt<i>.std is the <i>standard deviation</i> of the cell fluorescence intensity of channel <i>, a positive numeric value. • fluorescenceInt<i>.coverage is the percentage of the cell area covered by fluorescence of channel <i>, a numeric value in the range $[0, 1]$.

The fluorescenceInt<i>* components are included in case they were extracted by *SuperSegger*.

Nframes	Number of frames in the movie, a non-zero positive integer value. IDs of frames are in the range [1, Nframes].
Ncols	Number of colonies in the movie, equal to 1. IDs of colonies are in the range [1, Ncols].
frameH	NULL
frameW	NULL

References

<http://mtshasta.phys.washington.edu/website/SuperSegger.php>

isConnected	<i>Check if a tree is connected</i>
-------------	-------------------------------------

Description

Checks if a lineage or division tree is connected. A tree is connected if only one motherless cell exists (i.e. the root of the tree).

Usage

```
isConnected(tree)
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
------	--

Value

A logical value (TRUE or FALSE) indicating whether the tree is connected or not.

isSubtree	<i>Check if a tree is a subtree of another tree</i>
-----------	---

Description

Checks if a lineage or division tree is a subtree of another lineage or division tree, regarding only the nodes of the trees.

Usage

```
isSubtree(subtree, tree, treeT = c("LT", "DT"), type = c("all", "nr",  
  "inc"))
```

Arguments

subtree	The lineage or division tree which is supposed to be the subtree of tree, an object of class "igraph".
tree	The lineage or division tree which is supposed to be the supertree of subtree, an object of class "igraph".
treeT	A character string naming the type of subtree and tree: <ul style="list-style-type: none"> • "LT" if subtree and tree are lineage trees • "DT" if subtree and tree are division trees
type	A character string naming the type of cells regarding which the statement will be checked: <ul style="list-style-type: none"> • "all" for all cells (including any existing imaginary <i>root</i> cells) • "nr" for all non-root cells (excluding any existing imaginary <i>root</i> cells) • "inc" for all cells included in the analysis

Value

A logical value (TRUE or FALSE) indicating whether the subtree is a subtree of the tree or not.

merge_cells	<i>Merge two cells</i>
-------------	------------------------

Description

Merges two cells from the same or different lineage trees.

Usage

```
merge_cells(LT1, LT2 = NULL, cell1, cell2, cell_list, col_list, Ncols, pixelR,
  matFolder, matFileName, exeFolder, mcrFolder, show = TRUE)
```

Arguments

LT1	The lineage tree where the cell specified in cell1 belongs, an object of class "igraph".
LT2	The lineage tree where the cell specified in cell12 belongs, an object of class "igraph". When the default value NULL is used, cell12 belongs to the LT.
cell1	The label of the first cell in the LT1 to be merged, a character string. It can be any non-root cell, as returned from get_cells .
cell2	The label of the second cell in the LT2 (or LT1 in case LT2 = NULL) to be merged, a character string. It can be any valid candidate merge cell, as returned from get_cand_merge_cells .
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.
pixelR	The pixel ratio in units of length, a non-zero positive numeric value.

matFolder	A character string naming the absolute path of the directory where the .mat file generated by <i>BaSCA</i> is saved (excluding the last "/"). The default value is the current working directory <code>getwd()</code> . NOTE: The components should be separated by "/" on Windows.
matFileName	A character string naming the .mat file generated by <i>BaSCA</i> (including the suffix ".mat"). The filename is relative to the <code>matFolder</code> .
exeFolder	A character string naming the absolute path of the installation folder of the MATLAB executable (excluding the last "/"). NOTE: The components should be separated by "/" on Windows.
mcrFolder	A character string naming the absolute path of the installation folder of the Matlab Compiler Runtime (MCR) (excluding the last "/"). NOTE: The components should be separated by "/" on Windows.
show	A logical value (TRUE or FALSE) indicating whether <code>view_cell</code> will be called for <code>cell1</code> and <code>cell2</code> before the merge operation and the resulting cell after the merge operation. This capability is useful in order to see the result of the function. The default value is TRUE.

Details

After the merge operation, `cell1` is replaced in both `LT1` and `cell_list` by the resulting cell. Daughter branches of `cell2` are extracted from `LT2` (or `LT1` in case `LT2 = NULL`), with successive calls of `extract_branch`. These motherless branches (lineage trees) are added as daughter branches to `cell1`, with successive calls of `add_branch`, until `cell1` has two daughters. Finally, `cell2` is deleted from `LT2` (or `LT1` in case `LT2 = NULL`) with `extract_branch`.

Value

A named list with the following components:

LT1	The updated <code>LT1</code> with <code>cell1</code> replaced and the possible daughter branches added/replaced, an object of class "igraph".
LT2	The updated <code>LT2</code> with <code>cell2</code> and its daughter branches deleted, an object of class "igraph". NULL is returned in case <code>LT2</code> was anyway NULL or if it ended up with no cells.
cell_list	The updated <code>cell_list</code> with <code>cell1</code> replaced.
branches	A list with the remaining motherless branches. Each branch (element of the list) is an object of class "igraph".

Prerequisites

This function can be used by *BaSCA* users **only**, importing the data with `import_basca`.

See Also

`split_cell` for the reverse.

plot_baranyi	<i>Plot growth curves of cell counts</i>
--------------	--

Description

Computes and plots the growth curves of cell counts (number of cells) in a lineage tree. The growth curve can be plotted for each colony or for the whole population. Each growth curve is computed by fitting the *Baranyi and Roberts* model to the corresponding data.

Usage

```
plot_baranyi(LT, DT, cols = -1, Ncols, Nframes, frameR, showRaw = FALSE,
  save = FALSE, savePars = list(w = 3000, h = 2000, res = 300, path =
  getwd(), name = "my_baranyi"))
```

Arguments

LT	The lineage tree, an object of class "igraph".
DT	The corresponding division tree of the LT, an object of class "igraph".
cols	The IDs of the colonies for which to fit the <i>Baranyi and Roberts</i> model, a vector of non-zero positive integer values. The default value -1 stands for all existing colonies in the LT. Use value -2 for fitting the model to the whole population.
Ncols	Number of colonies in the movie, a non-zero positive integer value. This argument is ignored in case cols = -2.
Nframes	Number of frames in the movie, a non-zero positive integer value.
frameR	Frame rate of the movie in <i>frames per minute</i> , a non-zero positive numeric value.
showRaw	A logical value (TRUE or FALSE) indicating whether the raw data (unfitted data points) will be shown on the plot or not, respectively. The default value is FALSE. This argument is ignored (regarded as FALSE) in case cols = -1.
save	A logical value (TRUE or FALSE) indicating whether the generated plot(s) will be saved in .png file(s) or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of each generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: w The width of the image file in <i>pixels</i> , a non-zero positive integer value. The default value is 3000. h The height of the image file in <i>pixels</i> , a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 300. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory getwd(). The default value is the current working directory getwd().

NOTE: The components should be separated by "/" on Windows.

name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_baranyi".

Details

The parameters of the *Baranyi and Roberts* model are found using the *non-linear least squares* method provided by `nls`, considering all non-root cells, as returned from `get_cells`.

When `cols = -1`, a common plot for all colonies is generated. In other cases, a separate plot for each colony specified in `cols` is generated.

x-axis represents the time in *hours* from the start (value 0) to end of the movie. y-axis represents the cell counts in logarithmic scale. The range of y-axis values depicted in each plot is common and is calculated as the range of values of all specified colonies, with minimum upper limit 2 (i.e. 20 cells).

Color denotes the corresponding colony.

Value

A dataframe with the following columns:

1. colony is the colony ID (a non-zero positive integer value) or -2 in case `cols = -2`.
2. lag is the λ parameter (*lag time*) of the *Baranyi and Roberts* model in *hours* (a non-zero positive numeric value) or NA in case the model failed to be fitted or colony has only cells which have not been divided.
3. mumax is the μ_{max} parameter (*maximum specific growth rate*) of the *Baranyi and Roberts* model in *1/hour* (a non-zero positive numeric value) or NA in case the model failed to be fitted or colony has only cells which have not been divided.

For groups with `lag = NA`, no plot is generated, except for the case that `showRaw = TRUE`. In case no cells exist, no plot is generated and NULL is returned.

plot_col_tree	<i>Plot colonies' lineage tree</i>
---------------	------------------------------------

Description

Plots the colonies' lineage tree.

Usage

```
plot_col_tree(col_list, Ncols, sizeV = 2, sizeE = 0.7, sizeL = 1,
  save = FALSE, savePars = list(w = 2000, h = 2000, res = 250, path =
  getwd(), name = "my_col_tree"))
```

Arguments

col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.
sizeV	Size of vertices, a non-zero positive numeric value. The default value is 2.
sizeE	Width of edges, a non-zero positive numeric value. The default value is 0.7.
sizeL	Size of explanatory legends and title, a non-zero positive numeric value. The default value is 1.

- save** A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
- savePars** A named list specifying the parameters of the generated image file. This argument is ignored in case `save = FALSE`. Elements of the list are the following:
- w** The width of the image file in *pixels*, a non-zero positive integer value. The default value is 2000.
 - h** The height of the image file in *pixels*, a non-zero positive integer value. The default value is 2000.
 - res** The resolution of the image file in *pixels per inch* (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 250.
 - path** A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory `getwd()`. The default value is the current working directory `getwd()`.

NOTE: The components should be separated by "/" on Windows.

name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_col_tree".

Prerequisites

This function can be used by *BaSCA* users, importing the data with `import_basca`.

Users of *Oufiti* or *SuperSegger* who imported the data with `import_oufti` or `import_ss`, respectively, are **excluded** from using this function, as no colony list was returned.

For other users, it is necessary that a colony list was imported with `import_json`. In other case, no colony list exists and this function cannot be used.

Nodes represent the colony instants of the movie and are colored based on the ID of the corresponding colony. The imaginary *root* colony instant is colored white. Gray nodes are colony instants which have arisen from merged colonies.

<code>plot_dot_attr2</code>	<i>Create scatter plot of two attributes</i>
-----------------------------	--

Description

Creates the X-Y scatter plot of two numeric attributes of a lineage or division tree.

Usage

```
plot_dot_attr2(tree, treeT = c("LT", "DT"), attr1, unit1 = "", attr2,
  unit2 = "", attrC = "", unitC = "", NC = NULL, attrS = "",
  Nbins = c(20, 20), save = FALSE, savePars = list(w = 2500, h = 2000, res
    = 350, path = getwd(), name = "my_dot_attr2"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr1, attr2	The names of the attributes in the tree, character strings. Each one can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation" and "frame".
unit1, unit2	The units of the corresponding attr, character strings. Each one should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value of each is the empty character "", which implies that the corresponding attr is in arbitrary units.
attrC	The name of the attribute in the tree by which the cells will be colored, a character string. It can be any numeric or boolean attribute, as returned from get_attr_names . Coloring is applied to all depicted cells, except for cells with NA value in this attribute which are colored gray. When the default value "" (the empty character) is used, data points' colors are the default.
unitC	The unit of attrC, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attrC is in arbitrary units. This argument is ignored in case attrC = "".
NC	Number of colonies in the movie (if attrC = "colony") or number of generations in the movie (if attrC = "generation"), a non-zero positive integer value. This argument is ignored in case attrC != "colony" and attrC != "generation".
attrS	The name of the attribute in the tree by which the cells will be shaped, a character string. It can be any boolean attribute, as returned from get_attr_names . Cells are represented by dots or squares if their value in this attribute is TRUE or FALSE, respectively. When the default value "" (the empty character) is used, data points' shape is the default.
Nbins	Number of equally spaced bins to be used for attr1 and attr2, a vector of two non-zero positive integer values >=2. The first value is used for attr1 and the second for attr2. These values are indicative and may change automatically depending on the values of attr1 and attr2, producing a warning message. The default value is c(20, 20). This argument is ignored in case attrC != "".
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of the generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 350.

path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory `getwd()`. The default value is the current working directory `getwd()`.

NOTE: The components should be separated by "/" on Windows.

name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_dot_attr2".

Details

Each data point (x,y) represents the value of attr1 and attr2 of a cell, respectively. The scatter plot is created for all cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in attr1 or attr2.

By default, all data points are dot points colored based on the density of the (X,Y) variable value. 2D binning is applied on the range of all attr1 and attr2 values. Density represents the counts (number of data points) in each bin.

The linear regression line is also drawn on the plot. X is the predictor variable and Y is the response. The parameters of the regression line are found using the *linear least squares* method provided by `lm`.

Value

A named list with the following components:

Ncells	Number of cells, a non-zero positive integer value.
r	The Pearson correlation coefficient (a numeric value in the range [-1, 1]) or NA in case less than 2 unique data points exist.
regression	<p>A named list with the following components:</p> <ul style="list-style-type: none"> • a is the slope of the regression line, a numeric value • b is the y-intercept of the regression line, a numeric value • r2 is the R-squared coefficient of the regression line as returned from <code>summary.lm</code>, a numeric value in the range [0, 1] <p>In case less than 2 unique data points exist, NULL is returned, instead.</p>

In case no cells exist, no plot is generated and NULL is returned.

plot_dot_attr2_gen2	<i>Create scatter plot of attribute(s) between two generations</i>
---------------------	--

Description

Creates the X-Y scatter plot of the same or different numeric attribute between cells of two specific generations of a division tree.

Usage

```
plot_dot_attr2_gen2(DT, gen1, gen2, attr1, unit1 = "", attr2 = "",
  unit2 = "", save = FALSE, savePars = list(w = 2500, h = 2000, res = 350,
  path = getwd(), name = "my_dot_attr2_gen2"))
```

Arguments

DT	The connected division tree, an object of class "igraph".
gen1	The ID of the first generation in the DT, a positive integer value.
gen2	The ID of the second generation in the DT, a positive integer value.
attr1	The name of the first attribute in the DT, a character string. It can be any numeric attribute, as returned from <code>get_attr_names</code> , except for "colony" and "generation". When <code>attr2 = ""</code> , this attribute is used for cells of both generations gen1 and gen2. In other case, this attribute is used for cells of generation gen1.
unit1	The unit of attr1, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr1 is in arbitrary units.
attr2	The name of the second attribute in the DT, a character string. When the default value "" (the empty string) is used, the same attribute attr1 is used for cells of both generations gen1 and gen2. In other case, it can be any numeric attribute, as returned from <code>get_attr_names</code> , except for "colony" and "generation", and it is used for cells of generation gen2.
unit2	The unit of attr2, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr2 is in arbitrary units. This argument is ignored in case <code>attr2 = ""</code> .
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of the generated image file. This argument is ignored in case <code>save = FALSE</code> . Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 350. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory <code>getwd()</code>. The default value is the current working directory <code>getwd()</code>.

NOTE: The components should be separated by "/" on Windows.

name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_dot_attr2_gen2".

Details

Each data point (x,y) represents the corresponding attribute value of the older and younger cell, respectively. The scatter plot is created for all specified cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in attr1 (or attr2 in case `attr2 != ""`).

Data points are purple dot points.

The linear regression line is also drawn on the plot. X is the predictor variable and Y is the response. The parameters of the regression line are found using the *linear least squares* method provided by [lm](#).

Value

A named list with the following components:

Ncells	Number of cells, a non-zero positive integer value.
r	The Pearson correlation coefficient (a numeric value in the range $[-1, 1]$) or NA in case less than 2 unique data points exist.
regression	A named list with the following components: <ul style="list-style-type: none"> • a is the slope of the regression line, a numeric value • b is the y-intercept of the regression line, a numeric value • r2 is the R-squared coefficient of the regression line as returned from summary.lm, a numeric value in the range $[0, 1]$ <p>In case less than 2 unique data points exist, NULL is returned, instead.</p>

In case no cells exist, no plot is generated and NULL is returned.

See Also

[isConnected](#) for checking if a tree is connected.

plot_dot_attr3	<i>Create scatter plot of three attributes</i>
----------------	--

Description

Creates the X-Y-Z scatter plot of three numeric attributes of a lineage or division tree.

Usage

```
plot_dot_attr3(tree, treeT = c("LT", "DT"), attr1, unit1 = "", attr2,
  unit2 = "", attr3, unit3 = "", attrC = "", unitC = "", NC = NULL,
  attrS = "", sizeL = 0.8, save = FALSE, savePars = list(w = 2500, h =
  2000, res = 350, path = getwd(), name = "my_dot_attr3"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr1, attr2, attr3	The names of the attributes in the tree, character strings. Each one can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation" and "frame".

unit1, unit2, unit3	The units of the corresponding attr, character strings. Each one should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value of each is the empty character "", which implies that the corresponding attr is in arbitrary units.
attrC	The name of the attribute in the tree by which the cells will be colored, a character string. It can be any numeric or boolean attribute, as returned from get_attr_names . Coloring is applied to all depicted cells, except for cells with NA value in this attribute which are colored gray. When the default value "" (the empty character) is used, data points' color is the default.
unitC	The unit of attrC, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attrC is in arbitrary units. This argument is ignored in case attrC = "".
NC	Number of colonies in the movie (if attrC = "colony") or number of generations in the movie (if attrC = "generation"), a non-zero positive integer value. This argument is ignored in case attrC != "colony" and attrC != "generation".
attrS	The name of the attribute in the tree by which the cells will be shaped, a character string. It can be any boolean attribute, as returned from get_attr_names . Cells are represented by dots or squares if their value in this attribute is TRUE or FALSE, respectively. When the default value "" (the empty character) is used, data points' shape is the default.
sizeL	Size of explanatory legends, a non-zero positive numeric value. The default value is 0.8. This argument is ignored in case attrC = "" or attrC is a boolean attribute or attrC = "colony" or attrC = "generation".
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of the generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 350. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory getwd(). The default value is the current working directory getwd(). <p>NOTE: The components should be separated by "/" on Windows.</p> <p>name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_dot_attr3".</p>

Details

Each data point (x,y,z) represents the value of attr1, attr2 and attr3 of a cell, respectively. The horizontal axis of the plot corresponds to the x-axis, the vertical to the z-axis and the diagonal to the y-axis. The scatter plot is created for all cells that are included in the analysis, as returned from [get_cells](#), except for cells with NA value in attr1 or attr2 or attr3.

By default, all data points are purple dot points.

The linear regression plane is also drawn on the plot. X, Y are the predictor variables and Z is the response. The parameters of the regression plane are found using the *linear least squares* method provided by [lm](#).

Value

A named list with the following components:

Ncells	Number of cells, a non-zero positive integer value.
regression	A named list with the following components: <ul style="list-style-type: none"> • a1 is the slope of the regression plane in the direction of x, a numeric value • a2 is the slope of the regression plane in the direction of y, a numeric value • b is the y-intercept of the regression plane, a numeric value • r2 is the R-squared coefficient of the regression plane as returned from summary.lm, a numeric value in the range [0, 1]

In case less than 3 unique data points exist, NULL is returned, instead.

In case no cells exist, no plot is generated and NULL is returned.

plot_dot_attr_fam	<i>Create scatter plot of an attribute between family cells</i>
-------------------	---

Description

Creates the X-Y scatter plot of a numeric attribute between family cells of a division tree. The scatter plot can be created for specific generation(s) or for the whole population.

Usage

```
plot_dot_attr_fam(DT, attr, unit = "", type = c("s", "c", "md", "gmgd"),
  gens = -1, Ngens, save = FALSE, savePars = list(w = 2500, h = 2000, res
    = 350, path = getwd(), name = "my_dot_attr_fam"))
```

Arguments

DT	The connected division tree, an object of class "igraph".
attr	The name of the attribute in the DT, a character string. It can be any numeric attribute, as returned from get_attr_names , except for "colony" and "generation".
unit	The unit of attr, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr is in arbitrary units.

type	<p>A character string naming the type of the family cells for which to create the scatter plot:</p> <ul style="list-style-type: none"> • "s" for siblings • "c" for cousins • "md" for mother and daughter • "gmgd" for grand-mother and grand-daughter
gens	<p>The IDs of the generations which will be included in the scatter plot, a vector of non-zero positive integer values. When type = "md" or type = "gmgd", each value denotes the generation of the younger cell of the family (i.e. daughter or grand-daughter, respectively). Acceptable values are in the range [1, Ngens-1] in case type = "s" or type = "md" and in the range [2, Ngens-1] in case type = "c" or type = "gmgd". The default value -1 stands for all existing generations in the DT (whole population).</p>
Ngens	<p>Number of generations in the movie, a non-zero positive integer value.</p>
save	<p>A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.</p>
savePars	<p>A named list specifying the parameters of the generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following:</p> <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 350. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory getwd(). The default value is the current working directory getwd(). <p>NOTE: The components should be separated by "/" on Windows.</p> <p>name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_dot_attr_fam".</p>

Details

A common scatter plot for all generations specified in gens is created. In the general case, each data point (x,y) represents the attribute value of the older and younger cell of the family, respectively. The scatter plot is created for all specified family cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in attr.

Data points are dot points colored based on the generation of the younger cell. Note that when type = "s" or type = "c", there is no distinction between older and younger cell, since both cells are of the same generation.

The linear regression line is also drawn on the plot. X is the predictor variable and Y is the response. The parameters of the regression line are found using the *linear least squares* method provided by `lm`.

Value

A named list with the following components:

Ncells	Number of cells, a non-zero positive integer value.
r	The Pearson correlation coefficient (a numeric value in the range [-1, 1]) or NA in case less than 2 unique data points exist.
regression	A named list with the following components: <ul style="list-style-type: none"> • a is the slope of the regression line, a numeric value • b is the y-intercept of the regression line, a numeric value • r2 is the R-squared coefficient of the regression line as returned from summary.lm, a numeric value in the range [0, 1] <p>In case less than 2 unique data points exist, NULL is returned, instead.</p>

In case no cells exist, no plot is generated and NULL is returned.

See Also

[isConnected](#) for checking if a tree is connected.

plot_dot_time_attr	<i>Create dot plot of an attribute by time</i>
--------------------	--

Description

Creates the dot plot of a numeric attribute of a lineage or division tree by time.

Usage

```
plot_dot_time_attr(tree, treeT = c("LT", "DT"), attr, unit = "",
  Nbins = 20, save = FALSE, savePars = list(w = 2500, h = 2000, res = 350,
  path = getwd(), name = "my_dot_time_attr"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr	The name of the attribute in the tree, a character string. In case treeT = "LT", it can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation" and "frame". In case treeT = "DT", it can be any numeric attribute, as returned from get_attr_names , having the suffix "_birth" or "_division".
unit	The unit of attr, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr is in arbitrary units.

Nbins	Number of equally spaced bins to be used for <code>attr</code> , a non-zero positive integer value ≥ 2 . This value is indicative and may change automatically depending on the values of <code>attr</code> , producing a warning message. The default value is 20.
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of the generated image file. This argument is ignored in case <code>save = FALSE</code> . Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 350. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory <code>getwd()</code>. The default value is the current working directory <code>getwd()</code>. <p>NOTE: The components should be separated by "/" on Windows.</p> <p>name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_dot_time_attr".</p>

Details

Each data point (x,y) represents the time in *frames* and the value of `attr` of a cell, respectively. The dot plot is created for all cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in `attr`.

Data points are dot points colored based on the density of the Y variable value at every time point x. Binning is applied on the range of all `attr` values. Density represents the counts (number of data points) in each bin at every time point x. Ultimately, the plot is a color representation of the histogram of `attr` in time.

Value

A named list with the following components:

Ncells	Number of cells, a non-zero positive integer value.
r	The Pearson correlation coefficient (a numeric value in the range [-1, 1]) or NA in case less than 2 unique data points exist.
regression	A named list with the following components: <ul style="list-style-type: none"> a is the slope of the regression line, a numeric value b is the y-intercept of the regression line, a numeric value r2 is the R-squared coefficient of the regression line as returned from <code>summary.lm</code>, a numeric value in the range [0, 1]

In case less than 2 unique data points exist, NULL is returned, instead.

In case no cells exist, no plot is generated and NULL is returned.

plot_growth_attr	<i>Plot raw growth curves of an attribute</i>
------------------	---

Description

Plots the raw single-cell growth curves (unfitted time-series data) of a numeric attribute for a set of given cells in a division tree.

Usage

```
plot_growth_attr(DT, LT, attr, unit = "", cells = "all", Nframes,
  save = FALSE, savePars = list(w = 2500, h = 2000, res = 300, path =
  getwd(), name = "my_growth_attr"))
```

Arguments

DT	The division tree where the cells specified in <code>cells</code> belong, an object of class "igraph".
LT	The corresponding lineage tree of the DT, an object of class "igraph".
attr	The name of the attribute in the LT, a character string. It can be any numeric attribute, as returned from <code>get_attr_names</code> , except for "colony", "generation", "frame" and "age".
unit	The unit of <code>attr</code> , a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that <code>attr</code> is in arbitrary units.
cells	The labels of the cells in the DT whose growth curve will be plotted, a vector of character strings. They can be any cells that are included in the analysis, as returned from <code>get_cells</code> . The default value "all" stands for all cells in the DT.
Nframes	Number of frames in the movie, a non-zero positive integer value.
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of the generated image file. This argument is ignored in case <code>save = FALSE</code> . Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 300. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory <code>getwd()</code>. The default value is the current working directory <code>getwd()</code>.

NOTE: The components should be separated by "/" on Windows.

name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_growth_attr".

Details

A common plot for all cells specified in `cells` is generated.

x-axis represents the time in *frames*, from the start (value 1) to end of the movie (value `Nframes`).

Each single-cell growth curve is randomly colored.

`plot_growth_attr_cell` *Plot single-cell growth curve of an attribute*

Description

Plots the raw (and fitted) single-cell growth curve (time-series data) of a numeric attribute for a set of given cells in a division tree.

Usage

```
plot_growth_attr_cell(DT, LT, attr, unit = "", cells = "all",
  model = c("", "lin", "exp", "both"), frameR, save = FALSE,
  savePars = list(w = 1500, h = 1000, res = 150, path = getwd(), name =
    "my_growth_attr_cell"))
```

Arguments

<code>DT</code>	The division tree where the cells specified in <code>cells</code> belong, an object of class "igraph".
<code>LT</code>	The corresponding lineage tree of the <code>DT</code> , an object of class "igraph".
<code>attr</code>	The name of the attribute in the <code>LT</code> , a character string. It can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation", "frame" and "age".
<code>unit</code>	The unit of <code>attr</code> , a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that <code>attr</code> is in arbitrary units.
<code>cells</code>	The labels of the cells in the <code>DT</code> whose growth curve will be plotted, a vector of character strings. They can be any non-root cells, as returned from get_cells . The default value "all" stands for all cells in the <code>DT</code> .
<code>model</code>	A character string naming the optional fitted growth curve(s) to be plotted, if the corresponding parameters have been already successfully estimated using add_attr_growth_fit_pars : <ul style="list-style-type: none"> • "" (the empty string) for plotting just the raw growth curve • "lin" for plotting the fitted <i>linear</i> model • "exp" for plotting the fitted <i>exponential</i> model • "both" for plotting both fitted <i>linear</i> and <i>exponential</i> model
<code>frameR</code>	Frame rate of the movie in <i>frames per minute</i> , a non-zero positive numeric value. This argument is ignored in case <code>model = ""</code> .

save	A logical value (TRUE or FALSE) indicating whether the generated plot(s) will be saved in .png file(s) or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of each generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 1500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 1000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 150. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory getwd(). The default value is the current working directory getwd().

NOTE: The components should be separated by "/" on Windows.

name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_growth_attr_cell".

Details

A separate plot for each cell specified in cells is generated.

x-axis represents the life of the cell in *frames*, starting from its birth (value 1).

plot_growth_attr_fit *Plot fitted growth curves of an attribute*

Description

Plots the fitted single-cell growth curves (fitted time-series data) of a numeric attribute for all cells in a division tree. The growth curves can be plotted per colony or generation or for the whole population. The average growth curve of each group is also computed and plotted separately.

Usage

```
plot_growth_attr_fit(DT, LT, attr, unit = "", model = c("lin", "exp"),
  grouped = c("col", "gen", "pop"), groups = -1, Nggroups, attrC = "",
  unitC = "", NC = NULL, dur = 1, sizeL = 0.7, save = FALSE,
  savePars = list(w = 1500, h = 1000, res = 150, path = getwd(), name =
    "my_growth_attr_fit"))
```

Arguments

DT	The division tree, an object of class "igraph".
LT	The corresponding lineage tree of the DT, an object of class "igraph".

attr	The name of the attribute in the LT, a character string. It can be any numeric attribute, as returned from <code>get_attr_names</code> , except for "colony", "generation", "frame" and "age".
unit	The unit of attr, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr is in arbitrary units..
model	A character string naming the type of the fitted growth models to be plotted, <ul style="list-style-type: none"> • "lin" for plotting the fitted <i>linear</i> models • "exp" for plotting the fitted <i>exponential</i> models Parameters of the model must have already been estimated using <code>add_attr_growth_fit_pars</code> .
grouped	A character string naming the grouping method: <ul style="list-style-type: none"> • "col" for grouping by colony • "gen" for grouping by generation • "pop" for no grouping (whole population)
groups	The IDs of the groups for which to create the plot, a vector of positive integer values. This argument is ignored in case grouped = "pop". The default value -1 stands for all existing groups in the DT.
Ngroups	Number of colonies in the movie (if grouped = "col") or number of generations in the movie (if grouped = "gen"), a non-zero positive integer value. This argument is ignored in case grouped = "pop".
attrC	The name of the attribute in the DT by which the cells' growth curves will be colored, a character string. It can be any numeric or boolean attribute, as returned from <code>get_attr_names</code> . Coloring is applied to all depicted cells of all groups specified in groups, except for cells with NA value in this attribute which are colored gray. When the default value "" (the empty character) is used, cells' colors are the default.
unitC	The unit of attrC, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attrC is in arbitrary units. This argument is ignored in case attrC = "".
NC	Number of colonies in the movie (if attrC = "colony") or number of generations in the movie (if attrC = "generation"), a non-zero positive integer value. This argument is ignored in case attrC != "colony" and attrC != "generation".
dur	The time span in hours, a non-zero positive numeric value. The default value is 1.
sizeL	Size of explanatory legends, a non-zero positive numeric value. The default value is 0.7.
save	A logical value (TRUE or FALSE) indicating whether the generated plot(s) will be saved in .png file(s) or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of each generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 1500.

- h The height of the image file in *pixels*, a non-zero positive integer value. The default value is 1000.
- res The resolution of the image file in *pixels per inch* (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 150.
- path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory `getwd()`. The default value is the current working directory `getwd()`.

NOTE: The components should be separated by "/" on Windows.

- name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_growth_attr_fit".

Details

For each group specified in `groups`, two types of plots are generated. The first plot depicts the single-cell growth curves of all cells that are included in the analysis, as returned from `get_cells`, except for cells with NA values in the attributes/parameters of the `model`, as estimated by `add_attr_growth_fit_pars`. By default, each single-cell growth curve is randomly colored.

The second plot depicts the average growth curve of the group, +/- one standard deviation. When `model = "lin"`, the following curves are drawn:

- $y = aMean \cdot t + bMean$
- $y = (aMean + aSD) \cdot t + (bMean + bSD)$
- $y = (aMean - aSD) \cdot t + (bMean - bSD)$

When `model = "exp"`, the following curves are drawn:

- $y = y0Mean \cdot e^{kMean \cdot t}$
- $y = (y0Mean + y0SD) \cdot e^{(kMean + kSD) \cdot t}$
- $y = (y0Mean - y0SD) \cdot e^{(kSD - kSD) \cdot t}$

The parameters of these curves are computed based on the corresponding parameters of the single-cell growth curves. See the *Value* field for more details. Color of the area between the curves denotes the corresponding group.

In both types of plots, x-axis represents the time in *hours* in the range $[0, dur]$. Value 0 is considered to be the birth time of each cell. The range of y-axis values is common among the plots of the same type and is calculated as the range of the corresponding values of all specified groups (excluding the outliers).

Value

When `model = "lin"`, a dataframe with the following columns is returned:

1. `group` is the ID of the group (a positive integer value) or -2 in case `grouped = "pop"`
2. `Ncells` is the number of cells, a positive integer value
3. `aMean` is the *mean* of "<attr>_a" (a non-zero positive numeric value in units of attr per hour), or NA in case `Ncells = 0`

4. aSD is the *standard deviation* of "<attr>_a" (a non-zero positive numeric value in units of attr per hour), or NA in case Ncells = 0 or Ncells = 1
5. bMean is the *mean* of "<attr>_b" (a positive numeric value in units of attr) or NA in case Ncells = 0
6. bSD is the *standard deviation* of "<attr>_b" (a positive numeric value in units of attr) or NA in case Ncells = 0 or Ncells = 1

When model = "exp", a dataframe with the following columns is returned:

1. group is the ID of the group (a positive integer value) or -2 in case grouped = "pop"
2. Ncells is the number of cells, a positive integer value
3. kMean is the *mean* of "<attr>_k" (a non-zero positive numeric value in units of attr per hour), or NA in case Ncells = 0
4. kSD is the *standard deviation* of "<attr>_k" (a non-zero positive numeric value in units of attr per hour), or NA in case Ncells = 0 or Ncells = 1
5. y0Mean is the *mean* of "<attr>_0" (a non-zero positive numeric value in units of attr) or NA in case Ncells = 0
6. y0SD is the *standard deviation* of "<attr>_0" (a non-zero positive numeric value in units of attr) or NA in case Ncells = 0 or Ncells = 1

For groups with Ncells = 0, no plot of first or second type is generated. For groups with Ncells = 1, no plot of second type is generated. In case no cells exist, no plot is generated and NULL is returned.

plot_hist_attr	<i>Create histogram of an attribute</i>
----------------	---

Description

Creates the histogram of a numeric attribute of a lineage or division tree. The histogram can be created for each colony or generation or for the whole population.

Usage

```
plot_hist_attr(tree, treeT = c("LT", "DT"), attr, unit = "",
  grouped = c("col", "gen", "pop"), groups = -1, Nggroups, Nbins = 20,
  save = FALSE, savePars = list(w = 2000, h = 2000, res = 250, path =
  getwd(), name = "my_hist_attr"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr	The name of the attribute in the tree, a character string. It can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation" and "frame".

unit	The unit of attr, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr is in arbitrary units.
grouped	A character string naming the grouping method: <ul style="list-style-type: none"> • "col" for grouping by colony • "gen" for grouping by generation • "pop" for no grouping (whole population)
groups	The IDs of the groups for which to create the histogram, a vector of positive integer values. This argument is ignored in case grouped = "pop". The default value -1 stands for all existing groups in the tree.
Ngroups	Number of colonies in the movie (if grouped = "col") or number of generations in the movie (if grouped = "gen"), a non-zero positive integer value. This argument is ignored in case grouped = "pop".
Nbins	Number of equally spaced bins to be used for attr, a non-zero positive integer value ≥ 2 . This value is indicative and may change automatically depending on the values of attr, producing a warning message. The default value is 20.
save	A logical value (TRUE or FALSE) indicating whether the generated plot(s) will be saved in .png file(s) or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of each generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: <p>w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000.</p> <p>h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000.</p> <p>res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 250.</p> <p>path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory getwd(). The default value is the current working directory getwd().</p> <p>NOTE: The components should be separated by "/" on Windows.</p> <p>name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_hist_attr".</p>

Details

A separate plot for each group specified in groups is generated. Each histogram is created considering all cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in attr.

y-axis represents the cell counts. The range of x-axis values depicted in each plot is common and is calculated as the range of attr values of all specified groups. Binning is applied on the same common range.

Color denotes the corresponding group.

Value

A dataframe with the following columns:

1. group is the ID of the group (a positive integer value) or -2 in case grouped = "pop".
2. Ncells is the number of cells, a positive integer value.

For groups with Ncells = 0, no plot is generated. In case less than 2 unique cells exist, no plot is generated. In case no cells exist, NULL is returned.

plot_Ncells	<i>Plot cell counts</i>
-------------	-------------------------

Description

Calculates and plots the number of cells (cell counts) in a lineage or division tree. Cell counts can be plotted for each colony, generation or frame.

Usage

```
plot_Ncells(tree, treeT = c("LT", "DT"), grouped = c("col", "gen", "frame"),
  groups = -1, Nggroups, save = FALSE, savePars = list(w = 4000, h = 2000,
  res = 250, path = getwd(), name = "my_Ncells"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
grouped	A character string naming the grouping method: <ul style="list-style-type: none"> • "col" for grouping by colony • "gen" for grouping by generation • "frame" for grouping by frame, which is acceptable in case treeT = "LT".
groups	The IDs of the groups for which to plot the cell counts, a vector of positive integer values. The default value -1 stands for all existing groups in the tree.
Nggroups	Number of colonies in the movie (if grouped = "col") or number of generations in the movie (if grouped = "gen") or number of frames in the movie (if grouped = "frame"), a non-zero positive integer value.
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of the generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 4000. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000.

res The resolution of the image file in *pixels per inch* (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 250.

path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory `getwd()`. The default value is the current working directory `getwd()`.

NOTE: The components should be separated by "/" on Windows.

name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_Ncells".

Details

A common plot for all groups specified in `groups` is generated. Cell counts are calculated considering all cells that are included in the analysis, as returned from `get_cells`.

Color denotes the corresponding group.

Value

A dataframe with the following columns:

1. `group` is the ID of the group, a positive integer value.
2. `Ncells` is the number of cells, a positive integer value.

In case no cells exist, no plot is generated and `NULL` is returned.

<code>plot_pdf_attr</code>	<i>Plot the PDF of an attribute</i>
----------------------------	-------------------------------------

Description

Computes and plots the Probability Density Function (PDF) of a numeric attribute of a lineage or division tree. The PDF can be plotted for each colony or generation or for the whole population. Each PDF is computed by fitting a distribution model (*Normal*, *Gamma* or *Lognormal*) to the corresponding data.

Usage

```
plot_pdf_attr(tree, treeT = c("LT", "DT"), attr, unit = "",
  grouped = c("col", "gen", "pop"), groups = -1, Nggroups,
  model = c("norm", "gamma", "lnorm", "auto"), plot3D = TRUE,
  save = FALSE, savePars = list(w = 2000, h = 2000, res = 250, path =
  getwd(), name = "my_pdf_attr"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr	The name of the attribute in the tree, a character string. It can be any numeric attribute, as returned from <code>get_attr_names</code> , except for "colony", "generation" and "frame".
unit	The unit of attr, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr is in arbitrary units.
grouped	A character string naming the grouping method: <ul style="list-style-type: none"> • "col" for grouping by colony • "gen" for grouping by generation • "pop" for no grouping (whole population)
groups	The IDs of the groups for which to plot the PDF, a vector of positive integer values. This argument is ignored in case grouped = "pop". The default value -1 stands for all existing groups in the tree.
Nggroups	Number of colonies in the movie (if grouped = "col") or number of generations in the movie (if grouped = "gen"), a non-zero positive integer value. This argument is ignored in case grouped = "pop".
model	A character string naming the distribution model to be fitted: <ul style="list-style-type: none"> • "norm" is for fitting the <i>Normal</i> distribution. • "gamma" is for fitting the <i>Gamma</i> distribution. • "lnorm" is for fitting the <i>Lognormal</i> distribution. • "auto" is for finding the best-fit distribution. This is accomplished by fitting separately the <i>Normal</i>, <i>Gamma</i> and <i>Lognormal</i> distribution. The best-fit distribution is then chosen using the Bayesian Inference Criterion (BIC), according to which the best model is the one with the lowest numeric BIC value. <p>Each model is fitted using the <i>maximum likelihood estimation (MLE)</i> method provided by <code>fitdist</code>. Note that the <i>Gamma</i> and <i>Lognormal</i> distributions can be fitted to attributes with non-zero positive numeric values. Zero values are automatically replaced by value 1e-6. For negative values, an error is produced.</p>
plot3D	A logical value (TRUE or FALSE) indicating whether a 3D or 2D plot will be generated, respectively. When the default value TRUE is used, a common 3D plot for all groups specified in groups is generated. When the value FALSE is used, a separate 2D plot for each group specified in groups is generated. This argument is ignored (regarded as FALSE) in case grouped = "pop" or if only one group is specified in groups.
save	A logical value (TRUE or FALSE) indicating whether the generated plot(s) will be saved in .png file(s) or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of each generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following:

- w The width of the image file in *pixels*, a non-zero positive integer value. The default value is 2000.
- h The height of the image file in *pixels*, a non-zero positive integer value. The default value is 2000.
- res The resolution of the image file in *pixels per inch* (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 250.
- path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory `getwd()`. The default value is the current working directory `getwd()`.

NOTE: The components should be separated by "/" on Windows.

- name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_pdf_attr".

Details

Each PDF is computed considering all cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in `attr`.

The range of x-axis (attribute) values depicted in each 2D plot is common and is calculated as the range of values of all groups specified in `groups`.

Color denotes the corresponding group.

Value

A dataframe with the following columns:

1. group is the ID of the group (a positive integer value) or -2 in case `grouped = "pop"`.
2. Ncells is the number of cells, a positive integer value.
3. distr is a character string naming the distribution model that was fitted: "norm" for *Normal*, "gamma" for *Gamma* and "lnorm" for *Lognormal* distribution or NA if no distribution was fitted (less than 2 unique values of `attr` exist).
4. mean is the μ parameter (*mean*) of the *Normal* distribution (a numeric value), or NA in case `distr != "norm"`.
5. sd is the σ parameter (*standard deviation*) of the *Normal* distribution (a non-zero positive numeric value), or NA in case `distr != "norm"`.
6. shape is the α parameter (*shape*) of the *Gamma* distribution (a non-zero positive numeric value), or NA in case `distr != "gamma"`.
7. rate is the β parameter (*rate*) of the *Gamma* distribution (a non-zero positive numeric value), or NA in case `distr != "gamma"`.
8. meanlog is the μ parameter of the *Lognormal* distribution (a numeric value), or NA in case `distr != "lnorm"`.
9. sdlog is the σ parameter of the *Lognormal* distribution (a non-zero positive numeric value), or NA in case `distr != "lnorm"`.
10. BIC is the BIC value of the fitted distribution (a numeric value), or NA in case `distr = NA`.

11. dBIC is a character string summarizing the strength of the chosen distribution model specified in `distr` against the other models with higher BIC values. Value is NA in case `model != "auto"` or if `distr = NA`.

The format of the string is "<dBIC_norm>, <dBIC_gamma>, <dBIC_lnorm>". Each <dBIC_model> value is rounded. The larger a <dBIC_model> value, the stronger the evidence that attribute `attr` of the group follows the chosen `distr` distribution against the <model> distribution. Values >10 typically indicate strong preference to the chosen distribution.

For groups with `distr = NA`, no plot is generated. In case no cells exist, no plot is generated and NULL is returned.

plot_time_attr	<i>Plot mean and sd of an attribute per frame</i>
----------------	---

Description

Computes and plots the *mean* and *standard deviation* of a numeric attribute of a lineage or division tree per frame. The plot can be created for each colony or generation or for the whole population.

Usage

```
plot_time_attr(tree, treeT = c("LT", "DT"), attr, unit = "",
  grouped = c("col", "gen", "pop"), groups = -1, Nggroups, save = FALSE,
  savePars = list(w = 2500, h = 2000, res = 350, path = getwd(), name =
    "my_time_attr"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr	The name of the attribute in the tree, a character string. In case <code>treeT = "LT"</code> , it can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation" and "frame". In case <code>treeT = "DT"</code> , it can be any numeric attribute, as returned from get_attr_names , having the suffix "_birth" or "_division".
unit	The unit of <code>attr</code> , a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that <code>attr</code> is in arbitrary units.
grouped	A character string naming the grouping method: <ul style="list-style-type: none"> • "col" for grouping by colony • "gen" for grouping by generation • "pop" for no grouping (whole population)
groups	The IDs of the groups for which to create the plot, a vector of positive integer values. This argument is ignored in case <code>grouped = "pop"</code> . The default value -1 stands for all existing groups in the tree.

Ngroups	Number of colonies in the movie (if grouped = "col") or number of generations in the movie (if grouped = "gen"), a non-zero positive integer value. This argument is ignored in case grouped = "pop".
save	A logical value (TRUE or FALSE) indicating whether the generated plot(s) will be saved in .png file(s) or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of each generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following: <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2500. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 2000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 350. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory getwd(). The default value is the current working directory getwd(). <p>NOTE: The components should be separated by "/" on Windows.</p> <p>name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_time_attr".</p>

Details

A separate plot for each group specified in groups is generated. Each plot is created considering all cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in attr.

x-axis represents the time in *frames*. The range of x and y-axis values depicted in each plot is common and is calculated as the range of the corresponding values of all specified groups.

Color denotes the corresponding group.

Value

A named list with the following components:

group	The ID of the group (a positive integer value) or -2 in case grouped = "pop".
data	A dataframe with the following columns: <ol style="list-style-type: none"> 1. frame is the frame ID, a non-zero positive integer value. 2. Ncells is the number of cells, a positive integer value. 3. mean is the <i>mean</i> of attr, a numeric value. 4. sd is the <i>standard deviation</i> of attr, a positive numeric value or NA in case Ncells = 1. <p>For groups with no cells, no plot is generated and NULL is returned.</p>

plot_tree

*Plot a tree***Description**

Plots a lineage or division tree.

Usage

```
plot_tree(tree, treeT = c("LT", "DT"), attrC = "", unitC = "",
  NC = NULL, attrS = "", cellsC = "", colorCol = FALSE, Ncols,
  showLabels = FALSE, sizeLabel = 0.1, circular = TRUE,
  showLegends = TRUE, sizeL = 1, sizeV = 0.5, sizeE = 0.1,
  save = FALSE, savePars = list(w = 6000, h = 6000, res = 600, path =
  getwd(), name = "my_tree"))
```

Arguments

tree	The connected lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attrC	The name of the attribute in the tree by which the cells will be colored, a character string. It can be any numeric or boolean attribute, as returned from get_attr_names . Coloring is applied to all non-root cells, as returned from get_cells , except for cells with NA value in this attribute which are colored gray. Any existing imaginary <i>root</i> cells are always colored white. When the default value "" (the empty character) is used, nodes' color is the default.
unitC	The unit of attrC, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attrC is in arbitrary units. This argument is ignored in case attrC = "".
NC	Number of colonies in the movie (if attrC = "colony") or number of generations in the movie (if attrC = "generation"), a non-zero positive integer value. This argument is ignored in case attrC != "colony" and attrC != "generation".
attrS	The name of the attribute in the tree by which the cells will be shaped, a character string. It can be any boolean attribute, as returned from get_attr_names . Cells are represented by dots or squares if their value in this attribute is TRUE or FALSE, respectively. When the default value "" (the empty character) is used, nodes' shape is the default.
cellsC	The labels of the cells in the tree which will be colored red, a vector of character strings. They can be any cells, as returned from get_cells . The default value is the empty character "", which stands for no cell.
colorCol	A logical value (TRUE or FALSE) indicating whether the edges of the tree will be colored based on the colony or not, respectively. The value TRUE can be used only if tree contains the imaginary main <i>root</i> cell. When the default value FALSE is used, edges' color is the default.

Ncols	Number of colonies in the movie, a non-zero positive integer value. This argument is ignored in case colorCol = FALSE.
showLabels	A logical value (TRUE or FALSE) indicating whether the labels of the cells will be shown on the plot or not, respectively. The default value is FALSE.
sizeLabel	Size of cell labels, a non-zero positive numeric value. The default value is 0.1. This argument is ignored in case showLabels = FALSE.
circular	A logical value (TRUE or FALSE) indicating whether the tree will be plotted in circular or classical tree layout, respectively. The default value is FALSE.
showLegends	A logical value (TRUE or FALSE) indicating whether the explanatory legends and title will be shown on the plot or not, respectively. The default value is FALSE.
sizeL	Size of explanatory legends and title, a non-zero positive numeric value. The default value is 1. This argument is ignored in case showLegends = FALSE.
sizeV	Size of vertices, a non-zero positive numeric value. The default value is 0.5.
sizeE	Width of edges, a non-zero positive numeric value. The default value is 0.1.
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	<p>A named list specifying the parameters of the generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following:</p> <ul style="list-style-type: none"> w The width of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 6000. h The height of the image file in <i>pixels</i>, a non-zero positive integer value. The default value is 6000. res The resolution of the image file in <i>pixels per inch</i> (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 600. path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory getwd(). The default value is the current working directory getwd(). <p>NOTE: The components should be separated by "/" on Windows.</p> <p>name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_tree".</p>

Details

Nodes represent all cells, as returned from [get_cells](#). By default, all nodes are white dot points and edges are colored darkgray.

See Also

[isConnected](#) for checking if a tree is connected.

plot_viobox_attr	Create violin plot or boxplot for an attribute
------------------	--

Description

Creates the violin plot or boxplot of a numeric attribute of a lineage or division tree. The plot can be created for each colony or generation or for the whole population.

Usage

```
plot_viobox_attr(tree, treeT = c("LT", "DT"), attr, unit = "",
  grouped = c("col", "gen", "pop"), groups = -1, Nggroups, type = c("vio",
    "box"), save = FALSE, savePars = list(w = 2500, h = 2000, res = 300, path
    = getwd(), name = "my_viobox_attr"))
```

Arguments

tree	The lineage or division tree, an object of class "igraph".
treeT	A character string naming the type of tree: <ul style="list-style-type: none"> • "LT" if tree is a lineage tree • "DT" if tree is a division tree
attr	The name of the attribute in the tree, a character string. It can be any numeric attribute, as returned from get_attr_names , except for "colony", "generation" and "frame".
unit	The unit of attr, a character string. It should be in the format "<string>,<number>", where "<number>" represents the power and is optional (e.g. "m" for meters and "cm,3" for cubic centimeters). The default value is the empty character "", which implies that attr is in arbitrary units.
grouped	A character string naming the grouping method: <ul style="list-style-type: none"> • "col" for grouping by colony • "gen" for grouping by generation • "pop" for no grouping (whole population)
groups	The IDs of the groups for which to create the violin plot or boxplot, a vector of positive integer values. This argument is ignored in case grouped = "pop". The default value -1 stands for all existing groups in the tree.
Nggroups	Number of colonies in the movie (if grouped = "col") or number of generations in the movie (if grouped = "gen"), a non-zero positive integer value. This argument is ignored in case grouped = "pop".
type	A character string naming the type of the plot to be created: <ul style="list-style-type: none"> • "vio" for violin plot. The black dot represents the <i>median</i>. • "box" for boxplot. The black dots represent the outliers (i.e. datapoints out of the 1st/3rd quantile).
save	A logical value (TRUE or FALSE) indicating whether the generated plot will be saved in a .png file or displayed in the Plots Pane of RStudio, respectively. The default value is FALSE.
savePars	A named list specifying the parameters of the generated image file. This argument is ignored in case save = FALSE. Elements of the list are the following:

- w The width of the image file in *pixels*, a non-zero positive integer value. The default value is 2500.
- h The height of the image file in *pixels*, a non-zero positive integer value. The default value is 2000.
- res The resolution of the image file in *pixels per inch* (ppi), a non-zero positive integer value. The smaller this value, the larger the plot area in inches, and the smaller the text relative to the graph itself. The default value is 300.
- path A character string naming the directory where the image file will be saved (excluding the last "/"). If it does not contain an absolute path, the image file will be saved relative to the current working directory `getwd()`. The default value is the current working directory `getwd()`.

NOTE: The components should be separated by "/" on Windows.

- name The image file name, a character string. The suffix ".png" is added automatically. The default value is "my_viobox_attr".

Details

A common plot for all groups specified in `groups` is generated. Each violin plot or boxplot is created considering all cells that are included in the analysis, as returned from `get_cells`, except for cells with NA value in `attr`.

Color denotes the corresponding group.

Value

A dataframe with the following columns:

1. `group` is the ID of the group (a positive integer value) or -2 in case `grouped = "pop"`.
2. `Ncells` is the number of cells, a positive integer value.

For groups with `Ncells < 3`, no plot is generated. In case no cells exist, no plot is generated and NULL is returned.

save_tree

Save a tree on disc

Description

Saves a lineage or division tree on disc in nodes and edges .csv files.

Usage

```
save_tree(tree, path = getwd(), prefix = "", sep = "\t")
```

Arguments

tree	The lineage or division tree to be saved, an object of class "igraph".
path	A character string naming the directory where the "nodes.csv" and "edges.csv" files will be saved (excluding the last "/"). If it does not contain an absolute path, the files will be saved relative to the current working directory, getwd(). The default value is the current working directory getwd().
	NOTE: The components should be separated by "/" on Windows.
prefix	A prefix that will be added to the name of the "nodes.csv" and "edges.csv" files, a character string. The default value is the empty string "".
sep	The field separator character. Values on each line of the "nodes.csv" and "edges.csv" files will be separated by this character. The default value is the tab separator "\t".

select_subtree	<i>Select a subtree</i>
----------------	-------------------------

Description

Selects a subtree of a lineage or division tree based on multiple selection criteria combined with logical AND operator(s).

Usage

```
select_subtree(tree, criteria)
```

Arguments

tree	The lineage or division tree from which to select a subtree, an object of class "igraph".
criteria	A list containing the selection criteria. Each criterion (element of the list) should be a named list with the following components: <ul style="list-style-type: none"> attr A character string naming the attribute in the tree on which the selection will be made. It can be any numeric or boolean attribute, as returned from get_attr_names. val The value of comparison. It should be a numeric value when attr is numeric and a logical value (TRUE or FALSE) when attr is boolean. op The comparison operator. It should be "==", "!=", "<", "<=", ">" or ">=" when attr is numeric and "==" or "!=" when attr is boolean.

Details

The selection is made on the nodes of the tree. Only edges whose endpoints are both included in the resulting subtree are included. Therefore, the resulting subtree may be disconnected.

Value

The selected subtree, an object of class "igraph".

See Also

[isConnected](#) for checking if a tree is connected, [unite_trees](#) for combining multiple selection criteria with logical OR operator(s).

split_cell	<i>Split a cell</i>
------------	---------------------

Description

Splits a cell in a lineage tree.

Usage

```
split_cell(LT, cell, Nsplit = 2, cell_list, col_list, Ncols, pixelR,
          matFolder = getwd(), matFileName, exeFolder, mcrFolder, show = TRUE)
```

Arguments

LT	The lineage tree where the cell specified in cell belongs, an object of class "igraph".
cell	The label of the cell in the LT to be splitted, a character string. It can be any non-root cell, as returned from get_cells .
Nsplit	Number of cells for the cell to be splitted, an integer value >=2. The default value is 2.
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.
pixelR	The pixel ratio in units of length, a non-zero positive numeric value.
matFolder	A character string naming the absolute path of the directory where the .mat file generated by <i>BaSCA</i> is saved (excluding the last "/"). The default value is the current working directory <code>getwd()</code> .
	NOTE: The components should be separated by "/" on Windows.
matFileName	A character string naming the .mat file generated by <i>BaSCA</i> (including the suffix ".mat"). The filename is relative to the matFolder.
exeFolder	A character string naming the absolute path of the installation folder of the MATLAB executable (excluding the last "/").
	NOTE: The components should be separated by "/" on Windows.
mcrFolder	A character string naming the absolute path of the installation folder of the Matlab Compiler Runtime (MCR) (excluding the last "/").
	NOTE: The components should be separated by "/" on Windows.
show	A logical value (TRUE or FALSE) indicating whether view_cell will be called for the mother of the cell (if it exists), the cell before the split operation and the resulting cells after the split operation. This capability is useful in order to see the result of the function. The default value is TRUE.

Details

There is no guarantee that `cell` will be indeed splitted into `Nsplit` cells. After the split operation, the resulting cells are sorted by ascending distance between their centroid and the centroid of their mother, if it exists. Otherwise, the order of the resulting cells is random. The original cell is replaced in both `LT` and `cell_list` by the first listed cell, which implies best matching. For each one of the remaining resulting cells a single-node (connected) lineage tree (motherless branch) is automatically created.

Value

A named list with the following components:

<code>LT</code>	The updated <code>LT</code> with the <code>cell</code> replaced, an object of class "igraph".
<code>cell_list</code>	The updated <code>cell_list</code> with the <code>cell</code> replaced and the rest resulting cells added.
<code>branches</code>	A list with the single-node (connected) lineage trees for each one of the rest resulting cells Each branch (element of the list) is an object of class "igraph".

Prerequisites

This function can be used by *BaSCA* users **only**, importing the data with [import_basca](#).

See Also

[merge_cells](#) for the reverse, [add_branch](#) for connecting a motherless branch to a lineage tree.

<code>unite_trees</code>	<i>Unite multiple trees</i>
--------------------------	-----------------------------

Description

Unites multiple lineage or division trees from the same movie in one. This function is useful for combining multiple selection criteria with logical OR operator(s).

Usage

```
unite_trees(trees)
```

Arguments

<code>trees</code>	A list with the trees to be united. Each element of the list is a lineage or division tree (an object of class "igraph").
--------------------	---

Details

Nodes and edges of the resulting tree are the union of nodes and edges of the trees, respectively. Therefore, the resulting tree is disconnected. Any missing attributes among the trees are filled with NA values in the resulting tree.

Value

The united tree, an object of class "igraph".

See Also

[select_subtree](#) for combining multiple selection criteria with logical AND operator(s).

view_cell	<i>View a cell</i>
-----------	--------------------

Description

Displays the images of a set of given cells in the Plots Pane of RStudio.

Usage

```
view_cell(LT, cells, cell_list, col_list, Ncols)
```

Arguments

LT	The lineage tree where the cells specified in <code>cells</code> belong, an object of class "igraph".
cells	The labels of the cells in the LT to be viewed, a vector of character strings. They can be any non-root cells, as returned from get_cells .
cell_list	A list containing all the cell instants of the movie.
col_list	A list containing all the colony instants of the movie.
Ncols	Number of colonies in the movie, a non-zero positive integer value.

Details

A separate image for each cell specified in `cells` is generated. The cell is viewed in its colony and is marked as red. The rest cells of the colony are marked as white.

Prerequisites

This function can be used by *BaSCA* users, importing the data with [import_basca](#).

Users of *Oufiti* or *SuperSegger* who imported the data with [import_oufti](#) or [import_ss](#), respectively, are **excluded** from using this function, as no colony list was returned.

If [import_json](#) was used for importing the data, it is necessary that cell list elements have the `pixelList` and `colId` components and colony list elements have the `colImage` component. See [import_json](#) for more details. In other case, this function cannot be used (throws an error).

Description

ViSCA is an R package for the statistical analysis and visualization of single-cell data derived from the analysis of time-lapse cell-movies. The package enables users to explore the spatiotemporal trends of single-cell attributes, discover possible epigenetic effects across generations and even identify and correct errors.

Details

ViSCA was initially developed to extend the *BaSCA* pipeline with analytics, visualization and error correction capabilities. However, most of these capabilities are general and independent. Therefore, the package can be used for data derived from any other software, such as *SuperSegger* and *Oufiti*.

Data import

ViSCA supports various input file formats for the analyzed cell-movies.

Use `import_basca`, `import_oufti` or `import_ss` to import data exported by the named software. These functions automatically convert the input file(s) into a cell list (and colony list), containing all the cell instants (and colony instants) of the movie, respectively.

Such structures can be also directly imported from custom-made `.json` files, provided that they have the appropriate format. See `import_json` for details.

FLT/FDT representation of cell-movies

Once the cell list of the movie is loaded, use `createFLT` to transform it into a Forest of Lineage Trees (FLT) data structure. A lineage tree (LT) node represents a cell instant (i.e. cell at a specific frame/instant of its lifespan). A continuous segment (sequence) of LT nodes between two successive cell divisions represents the lifespan of a cell.

If one reduces LT cell segments down to a single node, he obtains the Forest of Division Trees (FDT) data structure, capturing only each cell's division event and summarizing its lifespan. A division tree (DT) node represents a cell (i.e. cell at its full lifespan). See `createFDT` for details.

These tree data structures are objects of class `"igraph"` and are the core structures of the package.

Attributes

Single-cell attributes are divided into two broad categories:

- cell instant attributes, that may change value at each time point. These attributes are extracted by the software, loaded into the cell list and finally stored as node attributes in the FLT by `createFLT`.

Some other values are also stored as node attributes in the FLT when `add_attr_roc` or `createFDT` are called. See the documentation of each function for more details.

- cell life attributes characterize a cell's whole lifespan. These attributes are estimated by [createFDT](#) and are stored as node attributes in the FDT.

Some other values are also stored as node attributes in the FDT when [add_attr_growth_fit_pars](#) is called. See the documentation of the function for more details.

Analytics

ViSCA allows users to perform statistical analysis of single-cell attributes at multiple levels of organization (whole community, sub-population, colonies, generations, subtrees of individual colonies, etc.). Analytics capabilities are categorized into:

- statistics ([get_attr_stats](#), [plot_hist_attr](#), [plot_pdf_attr](#), [plot_violbox_attr](#), [plot_time_attr](#))
- scatterplots for correlating attributes ([plot_dot_attr2](#), [plot_dot_attr3](#), [plot_dot_time_attr](#), [plot_dot_attr2_gen2](#), [plot_dot_attr_fam](#))
- estimation of growth curves ([plot_baranyi](#), [add_attr_growth_fit_pars](#), [plot_growth_attr_fit](#), [plot_growth_attr](#), [plot_growth_attr_cell](#))

Visualization

ViSCA provides two different ways for visualization:

1. [plot_tree](#) for visualizing a lineage or generation tree
2. [create_movie](#) for animating the segmented cells by creating videos

Color can be used to map a cell instant/life attribute and capture its variability across cells, colonies, frames or generations.

The user can also monitor how the life of a selected cell evolves in the movie using [create_cell_life](#).

Error correction

ViSCA allows users to correct

- tracking errors with [extract_branch](#), [get_cand_mother_cells](#) and [add_branch](#).
- segmentation errors with [split_cell](#), [get_cand_merge_cells](#) and [merge_cells](#). This capability is offered to *BaSCA* users **only**.

Notes

Some functions have prerequisites in order to be used. See the *Prerequisites* field of each function for more details.

References

A. Balomenos, P. Tsakanikas, Z. Aspidou, A. Tampakaki, K. Koutsoumanis and E. Manolakis, “Image analysis driven single-cell analytics for systems microbiology”, BMC Systems Biology, vol. 11, no. 1, 2017.

<http://oufti.org/>

<http://mtshasta.phys.washington.edu/website/SuperSegger.php>

Index

add_attr_growth_fit_pars, 2, 42, 44, 45, 63
add_attr_roc, 3, 62
add_branch, 4, 8, 11, 14, 23, 25, 28, 60, 63

create_cell_life, 8, 23, 25, 63
create_movie, 9, 23, 25, 63
createFDT, 5, 62, 63
createFLT, 7, 62

extract_branch, 5, 11, 28, 63

fitdist, 50

get_attr_names, 3, 4, 10, 11, 12, 13, 32, 34–37, 39, 41, 42, 44, 46, 50, 52, 54, 56, 58
get_attr_stats, 12, 63
get_attr_vals, 13
get_cand_merge_cells, 13, 23, 25, 27, 63
get_cand_mother_cells, 4, 5, 14, 23, 25, 63
get_cell_attr_vals, 16
get_cell_fam, 16
get_cells, 3, 4, 8, 10–13, 15, 16–18, 27, 30, 33, 34, 37, 38, 40–42, 45, 47, 49, 51, 53–55, 57, 59, 61
get_DT_cell, 17, 18
get_LT_cell, 17, 18

import_basca, 9, 10, 14, 15, 18, 28, 31, 60–62
import_json, 9, 10, 15, 21, 31, 61, 62
import_oufti, 9, 10, 15, 23, 31, 61, 62
import_ss, 9, 10, 15, 24, 31, 61, 62
isConnected, 4, 5, 7, 15, 17, 26, 35, 39, 55, 59
isSubtree, 11, 26

lm, 3, 33, 35, 37, 38

merge_cells, 13, 23, 25, 27, 60, 63

nls, 3, 30

plot_baranyi, 29, 63
plot_col_tree, 23, 25, 30
plot_dot_attr2, 31, 63
plot_dot_attr2_gen2, 33, 63
plot_dot_attr3, 35, 63
plot_dot_attr_fam, 37, 63
plot_dot_time_attr, 39, 63
plot_growth_attr, 41, 63
plot_growth_attr_cell, 42, 63
plot_growth_attr_fit, 43, 63
plot_hist_attr, 46, 63
plot_Ncells, 48
plot_pdf_attr, 49, 63
plot_time_attr, 52, 63
plot_tree, 54, 63
plot_vibox_attr, 56, 63

save_tree, 7, 8, 57
select_subtree, 58, 61
split_cell, 23, 25, 28, 59, 63
summary.lm, 33, 35, 37, 39, 40

unite_trees, 9, 59, 60

view_cell, 13, 14, 23, 25, 28, 59, 61
ViSCA, 62
ViSCA-package (ViSCA), 62