Package 'rcell2.examples'

October 13, 2022

Title Example Rmd notebooks and sample data for rcell2 (CellID Data Analysis Inside the Tidyverse)

Version 0.0.3

Description Analyse cellID data in the tidyverse framework. This package also has utility functions to generate, filter and preview data from fluorescence microscopy experiments (and maybe general cell cytometry).
Included are the tools to process CellID output (from tidyCell). Also, rcell2 interfaces directly with two other packages. The rcell2.cellid package bundles the CellID program source, and wraps it in a single R function. The rcell2.magick package offers R-Shiny apps and magick-based functions to help users preview cell images and filter data graphically.
This group of packages is meant to succeed the older RCell package.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.1

Depends R (>= 3.6)

biocViews

Imports stats

Suggests rcell2,
rcell2.cellid,
rcell2.magick,
tidyverse

R topics documented:

list_examples *list_examples: List example notebooks from rcell2 example notebooks*

Description

Notebooks with several examples for different kinds of CellID data analyses.

Usage

```
list_examples()
```

Details

Workflow examples are available as Rmd templates in their corresponding packages (rcell2, rcell2.cellid, and rcell2.magkc).

rcell2.examples *Examples for Yeast Cell Cytometry Suite for CellID in R.*

Description

It is a rewrite and revamp of the previously awesome Rcell package, by Dr. Alan Bush. Plotting functions from that package have been excluded in the name of minimalism. Thoughfully, ggplot2 definitions for all those "cplots" are available in our vignettes (happy copy-pasting!).

Details

The cellMagick package provides three categories of important functions: cellMagick, tidyCell and shinyCell.

tidyCell functions

The tidyCell functions run CellID and/or manage it's output. Also useful to turn custom data into compatible dataframes for the other functions in this package.

cellMagick functions

Renders images from individual cells, based on original images, user defined filters and data from cells. It should not be required that the data comes from images processed by CellID. Requirement of the imagemagick library might bother some, so this awesome feature is optional.

shinyCell functions

R-Shiny based graphical interface to filter cells by arbitrary variables, and inspect and annotate cells. It should not be required that the data comes from images processed by CellID.

See Also

magick

`recenter.vector` *Pone el origen de un vector en otro lado, moviendo los elementos despues del borde al principio y los de antes del borde quedan al final*

Description

Pone el origen de un vector en otro lado, moviendo los elementos despues del borde al principio y los de antes del borde quedan al final

Usage

```
recenter.vector(v, i)
```

`smooth.spline.poly` *whuber's spline.poly adapted to stats::smooth.spline*

Description

See: https://gis.stackexchange.com/a/24929

Usage

```
smooth.spline.poly(xy, k = 3, dof = 5, length.out = nrow(xy), ...)
```

Arguments

`xy`	A data.frame with ordered X/Y pairs representing the polygon
`k`	The amount of points used to "close" the path, and prevent discontinuous derivatives at the end-points.
`dof`	The degrees of freedom used by `stats::smooth.spline`
`length.out`	Set to an integer length for the smoothed output (defaults to input length).

Details

Splining a polygon with `stats::smooth.spline`

The rows of 'xy' give coordinates of the boundary vertices, in order. 'vertices' is the number of spline vertices to create. (Not all are used: some are clipped from the ends.) 'k' is the number of points to wrap around the ends to obtain a smooth periodic spline.

Value

Returns a data.frame of smooth points.

`spline.poly` *Splining a polygon with* `stats::spline`

Description

whuber's spline.poly as shared in StackExchange

Usage

```
spline.poly(xy, vertices, k = 3, ...)
```

Arguments

`xy`	A data.frame with ordered X/Y pairs representing the polygon
`vertices`	The amount of vertices in the final smoothing/interpolation.
`k`	The amount of points used to "close" the path, and prevent discontinuous derivatives at the end-points.

Details

See: https://gis.stackexchange.com/a/24929

The rows of 'xy' give coordinates of the boundary vertices, in order. 'vertices' is the number of spline vertices to create. (Not all are used: some are clipped from the ends.) 'k' is the number of points to wrap around the ends to obtain a smooth periodic spline.

Value

Returns an array of points.

Index