

Package ‘SpatialCPie’

October 16, 2019

Title Cluster analysis of Spatial Transcriptomics data

Version 1.0.0

Description SpatialCPie is an R package designed to facilitate cluster evaluation for spatial transcriptomics data by providing intuitive visualizations that display the relationships between clusters in order to guide the user during cluster identification and other downstream applications.

The package is built around a shiny “gadget” to allow the exploration of the data with multiple plots in parallel and an interactive UI. The user can easily toggle between different cluster resolutions in order to choose the most appropriate visual cues.

biocViews Transcriptomics, Clustering, RNASeq, Software

Depends R (>= 3.6)

Imports colorspace (>= 1.3-2), dplyr (>= 0.7.6), ggiraph (>= 0.5.0), ggplot2 (>= 3.0.0), ggrepel (>= 0.8.0), grid (>= 3.5.1), igraph (>= 1.2.2), lpSolve (>= 5.6.13), methods (>= 3.5.0), miniUI (>= 0.1.1.1), purrr (>= 0.2.5), readr (>= 1.1.1), rlang (>= 0.2.2), scatterpie (>= 0.1.2), shiny (>= 1.1.0), stats (>= 3.5.0), SummarizedExperiment (>= 1.10.1), tibble (>= 1.4.2), tidyr (>= 0.8.1), tidyselect (>= 0.2.4), utils (>= 3.5.0), zeallot (>= 0.1.0)

Suggests BiocStyle (>= 2.8.2), jpeg (>= 0.1-8), knitr (>= 1.20), rmarkdown (>= 1.10), testthat (>= 2.0.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 6.1.0

VignetteBuilder knitr

git_url <https://git.bioconductor.org/packages/SpatialCPie>

git_branch RELEASE_3_9

git_last_commit 3949e9a

git_last_commit_date 2019-05-02

Date/Publication 2019-10-15

Author Joseph Bergenstraahle [aut, cre]

Maintainer Joseph Bergenstraahle <joseph.bergenstrahle@gmail.com>

R topics documented:

SpatialCPie-package	2
parseSpotFile	2
runCPie	3
Index	5

SpatialCPie-package *SpatialCPie: Cluster analysis of Spatial Transcriptomics data*

Description

SpatialCPie is an R package designed to facilitate cluster evaluation for spatial transcriptomics data by providing intuitive visualizations that display the relationships between clusters in order to guide the user during cluster identification and other downstream applications. The package is built around a shiny "gadget" to allow the exploration of the data with multiple plots in parallel and an interactive UI. The user can easily toggle between different cluster resolutions in order to choose the most appropriate visual cues.

Author(s)

Maintainer: Joseph Bergenstrahle <joseph.bergenstrahle@gmail.com>

parseSpotFile *Parse spot detector output*

Description

Parses the output from the ST spot detector tool for use with SpatialCPie.

Usage

```
parseSpotFile(file)
```

Arguments

file spot file

Value

`data.frame` with columns "x" and "y" specifying the pixel coordinates of each spot

Examples

```
## Create spot file
data <- rbind(
  c(7, 18, 7.00, 18.07, 563.2, 947.0),
  c(8, 11, 8.00, 11.04, 612.5, 627.7)
)
filename <- tempfile()
write.table(
  data,
  file = filename,
  sep = "\t",
  quote = FALSE,
  col.names = c("x", "y", "new_x", "new_y", "pixel_x", "pixel_y")
)

## Parse spot file
parseSpotFile(filename)

## Delete spot file
unlink(filename)
```

runCPie

Run SpatialCPie

Description

Runs the SpatialCPie gadget.

Usage

```
runCPie(counts, image = NULL, spotCoordinates = NULL,
  margin = "spot", resolutions = 2:4,
  assignmentFunction = function(k, x) kmeans(x, centers = k)$cluster,
  view = NULL)
```

Arguments

counts	gene count matrix or a SummarizedExperiment-class object containing count values.
image	image to be used as background to the plot.
spotCoordinates	data.frame with pixel coordinates. The rows should correspond to the columns (spatial areas) in the count file.
margin	which margin to cluster.
resolutions	numeric vector specifying the clustering resolutions.
assignmentFunction	function to compute cluster assignments.
view	viewer object.

Value

a list with the following items:

- "clusters": Cluster assignments (may differ from assignments)
- "treePlot": The cluster tree ggplot object
- "piePots": The pie plot ggplot objects

Examples

```
if (interactive()) {
  options(device.ask.default = FALSE)

  ## Set up coordinate system
  coordinates <- as.matrix(expand.grid(1:10, 1:10))

  ## Generate data set with three distinct genes generated by three
  ## distinct cell types
  profiles <- diag(rep(1, 3)) + runif(9)
  centers <- cbind(c(5, 2), c(2, 8), c(8, 2))
  mixes <- apply(coordinates, 1, function(x) {
    x <- exp(-colSums((centers - x) ^ 2) / 50)
    x / sum(x)
  })
  means <- 100 * profiles %*% mixes
  counts <- matrix(rpois(prod(dim(means)), means), nrow = nrow(profiles))
  colnames(counts) <- apply(
    coordinates,
    1,
    function(x) do.call(paste, c(as.list(x), list(sep = "x"))))
  )
  rownames(counts) <- paste("gene", 1:nrow(counts))

  ## Run SpatialCPie
  runCPie(counts)
}
```

Index

`data.frame`, [2](#), [3](#)

`numeric`, [3](#)

`parseSpotFile`, [2](#)

`runCPie`, [3](#)

`SpatialCPie` (`SpatialCPie-package`), [2](#)

`SpatialCPie-package`, [2](#)

`viewer`, [3](#)