

Package ‘hoodscanR’

May 24, 2024

Title Spatial cellular neighbourhood scanning in R

Version 1.3.0

Description hoodscanR is an user-friendly R package providing functions to assist cellular neighborhood analysis of any spatial transcriptomics data with single-cell resolution. All functions in the package are built based on the SpatialExperiment object, allowing integration into various spatial transcriptomics-related packages from Bioconductor. The package can result in cell-level neighborhood annotation output, along with functions to perform neighborhood colocalization analysis and neighborhood-based cell clustering.

biocViews Spatial, Transcriptomics, SingleCell, Clustering

License GPL-3 + file LICENSE

URL <https://github.com/DavisLaboratory/hoodscanR>,
<https://davislaboratory.github.io/hoodscanR/>

BugReports <https://github.com/DavisLaboratory/hoodscanR/issues>

Encoding UTF-8

LazyData false

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports knitr, rmarkdown, SpatialExperiment, SummarizedExperiment, circlize, ComplexHeatmap, scico, rlang, utils, ggplot2, grid, methods, stats, RANN, Rcpp (>= 1.0.9)

LinkingTo Rcpp

Suggests testthat (>= 3.0.0), BiocStyle

Config/testthat/edition 3

Depends R (>= 4.3)

VignetteBuilder knitr

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

git_branch devel

git_last_commit 9a3a032

git_last_commit_date 2024-04-30

Repository Bioconductor 3.20

Date/Publication 2024-05-24

Author Ning Liu [aut, cre] (<<https://orcid.org/0000-0002-9487-9305>>),
Jarryd Martin [aut]

Maintainer Ning Liu <liu.n@wehi.edu.au>

Contents

| | |
|-----------------------------|-----------|
| hoodscanR-package | 2 |
| calcMetrics | 3 |
| clustByHood | 4 |
| findNearCells | 5 |
| mergeByGroup | 6 |
| mergeHoodSpe | 7 |
| plotColocal | 7 |
| plotHoodMat | 9 |
| plotProbDist | 10 |
| plotTissue | 12 |
| readHoodData | 13 |
| scanHoods | 14 |
| spe_test | 14 |
| Index | 16 |

| | |
|-------------------|---------------------------------------------------------------------------------------------------------|
| hoodscanR-package | <i>Method to identify cellular spatial neighbourhood from single cell spatial transcriptomics data.</i> |
|-------------------|---------------------------------------------------------------------------------------------------------|

Description

hoodscanR implements a novel method to scan for cell neighbourhood from spatial transcriptomics data at single cell level, such as CosMx and MERFISH etc. hoodscanR takes the cellular position and cell type annotations as inputs, allowing cellular spatial neighbourhood analysis.

Details

Key neighborhood analysis functions include [findNearCells](#), [scanHoods](#), [mergeByGroup](#), [calcMetrics](#), [clustByHood](#).

Key visualisation functions include [plotTissue](#), [plotHoodMat](#), [plotColocal](#), [plotProbDist](#).

Author(s)

Ning Liu <liu.n@wehi.edu.au>

| | |
|-------------|-------------------------------------------------|
| calcMetrics | <i>Calculate metrics for probability matrix</i> |
|-------------|-------------------------------------------------|

Description

Calculate metrics for probability matrix

Usage

```
calcMetrics(spe, pm = NA, pm_cols = NA, val_names = c("entropy", "perplexity"))
```

Arguments

| | |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------------|
| spe | A SpatialExperiment object. |
| pm | Optional. The probability matrix. |
| pm_cols | The colnames of probability matrix. This is requires for SpatialExperiment input. Assuming that the probability is stored in the colData. |
| val_names | Character vector with length of 2. Column names used to store calculated entropy and perplexity. |

Value

A SpatialExperiment object. Calculated entropy and perplexity are saved as columns in the colData of the SpatialExperiment object. Entropy and perplexity are calculated based on information theory:

$P(x)$ is the probability calculated from the scanHoods function.

Entropy $H(x) = -P(x)\log_2(P(x))$

Perplexity $P(x) = 2^{H(x)}$

By default, the calculated entropy and perplexity will be stored in the colData of the input spe, with column name as entropy and perplexity.

Examples

```
data("spe_test")

spe <- readHoodData(spe, anno_col = "celltypes")

fnc <- findNearCells(spe, k = 100)

pm <- scanHoods(fnc$distance)

pm2 <- mergeByGroup(pm, fnc$cells)

spe <- mergeHoodSpe(spe, pm2)

spe <- calcMetrics(spe, pm_cols = colnames(pm2))
```

clustByHood

Cluster the probability matrix with K-means

Description

Cluster the probability matrix with K-means

Usage

```
clustByHood(object, ...)
```

```
## S4 method for signature 'matrix'
```

```
clustByHood(object, k = 2^ncol(object) - 1, iter_max = 1000, nstart = 5)
```

```
## S4 method for signature 'SpatialExperiment'
```

```
clustByHood(
  object,
  pm_cols,
  k = 0,
  iter_max = 1000,
  nstart = 5,
  algo = "Hartigan-Wong",
  val_names = "clusters"
)
```

Arguments

| | |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------------|
| object | A probability matrix or a SpatialExperiment. |
| ... | Ignore parameter. |
| k | The number of clusters. By default is $2^{\text{ncol}(\text{object})}-1$. |
| iter_max | the maximum number of iterations allowed. |
| nstart | how many random sets should be chosen. |
| pm_cols | The colnames of probability matrix. This is requires for SpatialExperiment input. Assuming that the probability is stored in the colData. |
| algo | Algorithm to be used. Options include Hartigan-Wong, Lloyd, and MacQueen. |
| val_names | Character. Column names used to store the clusters. |

Value

A probability matrix or a SpatialExperiment object. For latter, the clustering results are saved in the colData of the SpatialExperiment object.

Examples

```
m <- matrix(abs(rnorm(1000 * 100)), 1000, 100)

clust <- clustByHood(m, k = 3)
```

| | |
|---------------|--------------------------------------------------|
| findNearCells | <i>Find the k-th nearest cells for each cell</i> |
|---------------|--------------------------------------------------|

Description

Find the k-th nearest cells for each cell

Usage

```
findNearCells(  
  dat,  
  k = 100,  
  targetCell = FALSE,  
  reportCellID = FALSE,  
  reportDist = TRUE,  
  anno_col = 0  
)
```

Arguments

| | |
|--------------|---------------------------------------------------------------------------|
| dat | A SpatialExperiment object, can be generated using function readHoodData. |
| k | The maximum number of nearest cells to compute. |
| targetCell | Specify the cells to be the target cell for finding nearest cells. |
| reportCellID | Logical. Set to TRUE to report cell id instead of cell types. |
| reportDist | Logical. Set to TRUE to report the distance matrix. |
| anno_col | Character vector. The name of annotation column to use. |

Details

The findNearCells function uses the nn2 function from the RANN package, which uses the Approximate Near Neighbor (ANN) C++ library. For more information on the ANN library please see <http://www.cs.umd.edu/~mount/ANN/>.

Value

A list includes a data.frame and a matrix, describing the cell types and distances of the k-th nearest cells of each cell.

Examples

```
data("spe_test")  
  
spe <- readHoodData(spe, anno_col = "celltypes")  
  
fnc <- findNearCells(spe, k = 100)
```

| | |
|--------------|------------------------------------------------------|
| mergeByGroup | <i>Merge probability matrix based on annotations</i> |
|--------------|------------------------------------------------------|

Description

Merge probability matrix based on annotations

Usage

```
mergeByGroup(pm, group_df)
```

Arguments

pm A numeric matrix. Probability matrix generated by the `soft_max` function.
group_df A character matrix. Annotation of the neighboring cells to be used.

Value

A probability matrix, describing the probability of each cell being in each cellular neighborhood.

Examples

```
data("spe_test")  
  
spe <- readHoodData(spe, anno_col = "celltypes")  
  
fnc <- findNearCells(spe, k = 100)  
  
pm <- scanHoods(fnc$distance)  
  
pm2 <- mergeByGroup(pm, fnc$cells)
```

| | |
|--------------|----------------------------------------------------------------|
| mergeHoodSpe | <i>Merge probability matrix into SpatialExperiment object.</i> |
|--------------|----------------------------------------------------------------|

Description

Merge probability matrix into SpatialExperiment object.

Usage

```
mergeHoodSpe(spe, pm, val_names = NULL)
```

Arguments

| | |
|-----------|-------------------------------------------------------------------|
| spe | A SpatialExperiment object. |
| pm | Probability matrix. Can be obtained by the function mergeByGroup. |
| val_names | Character vector with length of the ncol of pm. |

Value

A SpatialExperiment object. Cell-level neighborhood information are saved in the colData of the SpatialExperiment object.

Examples

```
data("spe_test")  
  
spe <- readHoodData(spe, anno_col = "celltypes")  
  
fnc <- findNearCells(spe, k = 100)  
  
pm <- scanHoods(fnc$distance)  
  
pm2 <- mergeByGroup(pm, fnc$cells)  
  
spe <- mergeHoodSpe(spe, pm2)
```

| | |
|-------------|------------------------------------------------|
| plotColocal | <i>Plot heatmap for neighbourhood analysis</i> |
|-------------|------------------------------------------------|

Description

Plot heatmap for neighbourhood analysis

Usage

```

plotColocal(object, ...)

## S4 method for signature 'matrix'
plotColocal(object, hm_width = 5, hm_height = 5)

## S4 method for signature 'SpatialExperiment'
plotColocal(
  object,
  pm_cols,
  self_cor = TRUE,
  by_group = NULL,
  hm_width = 5,
  hm_height = 5,
  cluster_row = TRUE,
  cluster_col = TRUE,
  return_matrix = FALSE
)

```

Arguments

| | |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>object</code> | A probability matrix or <code>SpatialExperiment</code> . |
| <code>...</code> | Ignore parameter. |
| <code>hm_width</code> | Integer. The width of heatmap. |
| <code>hm_height</code> | Integer. The height of heatmap. |
| <code>pm_cols</code> | The colnames of probability matrix. This is requires for <code>SpatialExperiment</code> input. Assuming that the probability is stored in the <code>colData</code> . |
| <code>self_cor</code> | Logical. By default is <code>TRUE</code> , indicating running a correlation between neighbourhoods to perform a simple co-localization analysis. When this set to <code>FALSE</code> , it will plot the average probability of each neighbourhood by group using the <code>by_group</code> parameter. |
| <code>by_group</code> | Character. This is required when <code>self_cor</code> is set to <code>FALSE</code> . |
| <code>cluster_row</code> | Logical. Cluster rows. |
| <code>cluster_col</code> | Logical. Cluster columns. |
| <code>return_matrix</code> | Logical. Export a numeric matrix . |

Value

A `ComplexHeatmap` plot. When `return_matrix` is set to `TRUE`, return a matrix Object.

Examples

```

data("spe_test")

spe <- readHoodData(spe, anno_col = "celltypes")

```



```
fnc <- findNearCells(spe, k = 100)
pm <- scanHoods(fnc$distance)
pm2 <- mergeByGroup(pm, fnc$cells)
spe <- mergeHoodSpe(spe, pm2)
plotColocal(spe, pm_cols = colnames(pm2))
plotColocal(spe, pm_cols = colnames(pm2), self_cor = FALSE, by_group = "cell_annotation")
```

| | |
|-------------|---------------------------------------------|
| plotHoodMat | <i>Plot probability matrix as a heatmap</i> |
|-------------|---------------------------------------------|

Description

Plot probability matrix as a heatmap

Usage

```
plotHoodMat(object, ...)
```

```
## S4 method for signature 'matrix'
plotHoodMat(
  object,
  targetCells = NA,
  n = 30,
  hm_width = 4,
  hm_height = 15,
  clusterRows = TRUE,
  clusterCols = TRUE,
  title = "Probability of neighborhoods"
)
```

```
## S4 method for signature 'SpatialExperiment'
plotHoodMat(
  object,
  pm_cols,
  targetCells = NA,
  n = 30,
  hm_width = 4,
  hm_height = 15,
  clusterRows = TRUE,
  clusterCols = TRUE,
  title = "Probability of neighborhoods"
)
```

Arguments

| | |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| object | A probability matrix or SpatialExperiment. |
| ... | Ignore parameter. |
| targetCells | Character. Optional. Can specify one or more cells to be plotted. |
| n | Integer. The number of randomly selected cells to be plotted. This parameter will be used when targetCells is not specify. |
| hm_width | Integer. The width of heatmap. |
| hm_height | Integer. The height of heatmap. |
| clusterRows | Logical. Cluster rows or not. |
| clusterCols | Logical. Cluster columns or not. |
| title | Title of the heatmap. |
| pm_cols | The colnames of probability matrix. This is requires for SpatialExperiment input. Assuming that the probability is stored in the colData. |

Value

A ComplexHeatmap plot.

Examples

```
data("spe_test")

spe <- readHoodData(spe, anno_col = "celltypes")

fnc <- findNearCells(spe, k = 100)

pm <- scanHoods(fnc$distance)

pm2 <- mergeByGroup(pm, fnc$cells)

spe <- mergeHoodSpe(spe, pm2)

plotHoodMat(spe, pm_cols = colnames(pm2))
```

plotProbDist

Plot probability distribution

Description

Plot probability distribution

Usage

```

plotProbDist(object, ...)

## S4 method for signature 'matrix'
plotProbDist(object, targetCells = NA, ...)

## S4 method for signature 'SpatialExperiment'
plotProbDist(
  object,
  pm_cols,
  targetCells = NA,
  by_cluster = FALSE,
  show_clusters = as.character(seq(6)),
  plot_all = FALSE,
  sample_size = 2,
  ...
)

```

Arguments

| | |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>object</code> | A probability matrix or <code>SpatialExperiment</code> . |
| <code>...</code> | aesthetic mappings to pass to <code>ggplot2::aes_string()</code> . |
| <code>targetCells</code> | Character. Optional. Can specify one or more cells to be plotted. |
| <code>pm_cols</code> | The colnames of probability matrix. This is required for <code>SpatialExperiment</code> input. Assuming that the probability is stored in the <code>colData</code> . |
| <code>by_cluster</code> | Logical. By default is <code>TRUE</code> , to plot distribution by each cluster. |
| <code>show_clusters</code> | Character. The cluster to be plotted, by default is 1 to 6. |
| <code>plot_all</code> | Logical. By default is <code>FALSE</code> , set this to <code>true</code> to plot box plot instead of bar plot to show all cells in each cluster. |
| <code>sample_size</code> | Integer. By default is 2, sampling two cells from each cluster to be plotted. |

Value

A `ggplot` object.

Examples

```

data("spe_test")

spe <- readHoodData(spe, anno_col = "celltypes")

fnc <- findNearCells(spe, k = 100)

pm <- scanHoods(fnc$distance)

pm2 <- mergeByGroup(pm, fnc$cells)

```

```
spe <- mergeHoodSpe(spe, pm2)

plotProbDist(spe, pm_cols = colnames(pm2))
```

plotTissue

Plot cells based on cell position on tissue.

Description

Plot cells based on cell position on tissue.

Usage

```
plotTissue(
  spe,
  targetcell = FALSE,
  k_near = 100,
  targetsize = 3,
  targetshape = 1,
  targetcolor = "red",
  scaleFactor = 1,
  reverseY = TRUE,
  ...
)
```

Arguments

| | |
|-------------|-----------------------------------------------------------------------------------------------|
| spe | SpatialExperiment object. |
| targetcell | Optional. Can input ONE specific cell id to zoom-in on the region of a specific cell. |
| k_near | Optional. If targetcell is specified, the k_near cells around the targetcell will be plotted. |
| targetsize | Dot size of the targetcell. |
| targetshape | Shape of the targetcell. |
| targetcolor | Colour of the targetcell. |
| scaleFactor | Scale factor to align with the image. |
| reverseY | Reverse y coordinates. |
| ... | aesthetic mappings to pass to <code>ggplot2::aes_string()</code> . |

Value

A ggplot object.

Examples

```
data("spe_test")

plotTissue(spe, color = celltypes)
```

| | |
|--------------|-----------------------------------------------------------------------|
| readHoodData | <i>Read cellular position and annotation data into a list object.</i> |
|--------------|-----------------------------------------------------------------------|

Description

Read cellular position and annotation data into a list object.

Usage

```
readHoodData(  
  spe = NA,  
  anno_col = NA,  
  cell_pos_dat = NA,  
  cell_anno_dat = NA,  
  pos_col = NA  
)
```

Arguments

| | |
|---------------|------------------------------------------------------------------------------------------------------------------|
| spe | SpatialExperiment object. |
| anno_col | Character. The column name of the annotation to be used in the following neighbourhood analysis. |
| cell_pos_dat | data.frame object contains the cellular positions. |
| cell_anno_dat | data.frame object contains the cell annotations. |
| pos_col | Character. If the x and y are in the colData instead of in the SpatialCoords of spe, can specify this parameter. |

Value

A SpatialExperiment object.

Examples

```
data("spe_test")

spe <- readHoodData(spe, anno_col = "celltypes")
```

scanHoods *Scan cellular neighbourhoods.*

Description

Scan cellular neighbourhoods.

Usage

```
scanHoods(
  m,
  mode = c("proximityFocused", "smoothFadeout"),
  tau = NA,
  t_init = NA
)
```

Arguments

| | |
|--------|----------------------------------------------------------------------------------------------|
| m | Distance matrix. Can be obtained from function findNearCells. |
| mode | Character. Either proximityFocused or smoothFadeout. By default is proximityFocused. |
| tau | The hyperparameter tau, by default is $\text{median}(m^{**2})/5$ |
| t_init | An initial tau. In the smoothFadeout mode, user can provide an initial tau for optimization. |

Value

A probability matrix.

Examples

```
m <- matrix(abs(rnorm(1000 * 100)), 1000, 100)

pm <- scanHoods(m)
```

spe_test *Example test spatial transcriptomics data*

Description

hoodscanR-package has 1 datasets:

- spe_test Example test spatial transcriptomics data in SpatialExperiment format. This test data is randomly subsetting from the publicly available CosMx non-small cell lung cancer data. Source data: <https://nanosttring.com/products/cosmx-spatial-molecular-imager/nsclc-ffpe-dataset/>.

spe_test

15

Usage

```
data("spe_test")
```

Format

A SpatialExperiment object

Value

A SpatialExperiment object

Examples

```
data(spe_test)
```

Index

* **internal**

- hoodscanR-package, 2
- spe_test, 14

calcMetrics, 2, 3

clustByHood, 2, 4

clustByHood, matrix-method
(clustByHood), 4

clustByHood, SpatialExperiment-method
(clustByHood), 4

findNearCells, 2, 5

hoodscanR (hoodscanR-package), 2

hoodscanR-package, 2

mergeByGroup, 2, 6

mergeHoodSpe, 7

plotColocal, 2, 7

plotColocal, matrix-method
(plotColocal), 7

plotColocal, SpatialExperiment-method
(plotColocal), 7

plotHoodMat, 2, 9

plotHoodMat, matrix-method
(plotHoodMat), 9

plotHoodMat, SpatialExperiment-method
(plotHoodMat), 9

plotProbDist, 2, 10

plotProbDist, matrix-method
(plotProbDist), 10

plotProbDist, SpatialExperiment-method
(plotProbDist), 10

plotTissue, 2, 12

readHoodData, 13

scanHoods, 2, 14

spe (spe_test), 14

spe_test, 14