Package 'scider'

April 28, 2025

Type Package

Title Spatial cell-type inter-correlation by density in R

Version 1.6.0

Description scider is a user-friendly R package providing functions to model the global density of cells in a slide of spatial transcriptomics data. All functions in the package are built based on the SpatialExperiment object, allowing integration into various spatial transcriptomics-related packages from Bioconductor. After modelling density, the package allows for serveral downstream analysis, including colocalization analysis, boundary detection analysis and differential density analysis.

biocViews Spatial, Transcriptomics

License GPL-3 + file LICENSE

URL https://github.com/ChenLaboratory/scider, https://chenlaboratory.github.io/scider/

BugReports https://github.com/ChenLaboratory/scider/issues

Encoding UTF-8

LazyData false

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports SpatialExperiment, SummarizedExperiment, spatstat.geom, spatstat.explore, sf, lwgeom, ggplot2, stats, pheatmap, plotly, shiny, igraph, janitor, knitr, methods, utils, rlang, isoband, S4Vectors, grDevices

Suggests edgeR, testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 4.3)

VignetteBuilder knitr

git_url https://git.bioconductor.org/packages/scider

git_branch RELEASE_3_21

git_last_commit f53c7b3

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-04-27

Author Ning Liu [aut] (ORCID: <https://orcid.org/0000-0002-9487-9305>), Mengbo Li [aut] (ORCID: <https://orcid.org/0000-0002-9666-5810>), Yunshun Chen [aut, cre] (ORCID: <https://orcid.org/0000-0003-4911-5653>)

Maintainer Yunshun Chen <yuchen@wehi.edu.au>

Contents

scider-package . . . 3
allocateCells . . . 3
cellsInRegion . . . 4
computeDensity . . . 5
contour2sf . . . 6
corDensity . . . 6
findROI . . . 7
getContour . . . 8
getContourRegions . . . 9
grid2sf . . . 9
gridDensity . . . 10
mergeROI . . . 11
plotCellCompo . . . 12
plotContour . . . 13
plotContourRegion . . . 14
plotCorHeatmap . . . 14
plotDensCor . . . 15
plotDensity . . . 16
plotROI . . . 17
plotSpatial . . . 18
postSelRegion . . . 18
selectRegion . . . 19
spe2PB . . . 20
xenium_bc_spe . . . 21

Index **22**

scider-package *Spatial cell-type inter-correlation by density in R.*

Description

scider implements functions to analyse spatial transcriptomics data with cell type annotations by performing cell type correlation via density estimation and cell type co-localization via real number distance. Functions include density estimation, statistical modelling and visualizations.

Details

scider uses SpatialExperiment objects as the main infrastructure, which can easily be integrated with a wide variety of Bioconductor packages.

Author(s)

Ning Liu <liu.n@wehi.edu.au>, Mengbo Li <li.me@wehi.edu.au>, Yunshun Chen <yuchen@wehi.edu.au>

allocateCells *Annotate all cells with contour level of cell type-specific density.*

Description

Annotate all cells with contour level of cell type-specific density.

Usage

```
allocateCells(spe, to.roi = TRUE, to.contour = TRUE)
```

Arguments

spe	A SpatialExperiment object.
to.roi	Logical. Whether to allocate cells to ROIs.
to.contour	Logical. Whether to allocate cells to contour levels.

Value

A SpatialExperiment object. An extra column is added to the colData.

Examples

```
data("xenium_bc_spe")

spe <- gridDensity(spe)

coi <- "Breast cancer"

spe <- findROI(spe, coi = coi)

spe <- allocateCells(spe)
```

`cellsInRegion` *Check which cells are in which regions*

Description

Check which cells are in which regions

Usage

```
cellsInRegion(spe, region, name_to, NA_level = "0", levels = NULL)
```

Arguments

`spe`	A SpatialExperiment object.
`region`	List or an sf object that represents a region or an ROI.
`name_to`	Colname in colData(spe) to store the annotation.
`NA_level`	Label for cells not falling in any of the regions. Default to 0.
`levels`	Factor levels.

Value

A SpatialExperiment object. The region information of each cell is stored in the colData.

computeDensity *Perform kernel density estimation on SpatialExperiment*

Description

Perform kernel density estimation on SpatialExperiment

Usage

```
computeDensity(
  spe,
  mode = "pixels",
  kernel = "gaussian",
  bandwidth = NULL,
  weights = NULL,
  ngrid.x = 100,
  ngrid.y = NULL,
  grid.length.x = NULL,
  grid.length.y = NULL,
  xlim = NULL,
  ylim = NULL,
  diggle = FALSE
)
```

Arguments

spe	A SpatialExperiment object.
mode	Choose either points or pixels. Specifying whether to compute the density at a grid pixel location or at at point.
kernel	The smoothing kernel. Options are gaussian, epanechnikov, quartic or disc.
bandwidth	The smoothing bandwidth. By default performing automatic bandwidth selection using cross-validation using function spatstat.explore::bw.diggle.
weights	Optional weights to be attached to the points.
ngrid.x	Number of grids in the x-direction. Default to 100.
ngrid.y	Number of grids in the y-direction.
grid.length.x	Grid length in the x-direction.
grid.length.y	Grid length in the y-direction.
xlim	The range of the x-coordinates of the image.
ylim	The range of the y-coordinates of the image.
diggle	Logical. If TRUE, use the Jones-Diggle improved edge correction. See spatstat.explore::density.ppp() for details.

Value

Output from spatstat.explore::density.ppp.

contour2sf *Draw a contour region on some density level*

Description

Draw a contour region on some density level

Usage

```
contour2sf(spe, contour, coi, cutoff)
```

Arguments

spe	A SpatialExperiment object.
contour	Name in metadata.
coi	A character vector of cell types of interest (COIs).
cutoff	A numeric scalar specifying the density cutoff.

Value

An sf object of the contour region of the specified level.

corDensity *Test for density correlation between two cell types.*

Description

Test for density correlation between two cell types.

Usage

```
corDensity(spe, by.roi = TRUE)
```

Arguments

spe	A SpatialExperiment object.
by.roi	Logical. If TRUE (default), then return the testing results at ROI level. If FALSE, then combine the testing results across all ROIs.

Value

A DataFrame containing the testing results.

Examples

```
data("xenium_bc_spe")

coi <- c("Breast cancer", "Fibroblasts", "B cells", "T cells")

spe <- gridDensity(spe, coi = coi)

spe <- findROI(spe, coi = coi, method = "walktrap")

result <- corDensity(spe)
```

`findROI` *Find ROIs based on cell type-specific densities via graph-based method.*

Description

Find ROIs based on cell type-specific densities via graph-based method.

Usage

```
findROI(
  spe,
  coi,
  probs = 0.85,
  ngrid.min = 20,
  method = "walktrap",
  diag.nodes = FALSE,
  sequential.roi.name = TRUE,
  ...
)
```

Arguments

`spe`	A SpatialExperiment object.
`coi`	A character vector of cell types of interest (COIs).
`probs`	A numeric scalar. The threshold of proportion that used to filter grid by density. Default to 0.85.
`ngrid.min`	An integer. The minimum number of grids required for defining a ROI. Default to 20.
`method`	The community dectection method to be used, either walktrap or connected. Default to walktrap.
`diag.nodes`	Logical. Set this to TRUE to allow diagonal grid points to be adjacent nodes.
`sequential.roi.name`	Logical. Set this to FALSE if you want the original ROI name before filtering are retained.
`...`	Other parameters that passed to walktrap.community.

Value

A SpatialExperiment object.

Examples

```
data("xenium_bc_spe")

coi <- c("Breast cancer", "Fibroblasts")

spe <- gridDensity(spe, coi = coi)

spe <- findROI(spe, coi = coi, method = "walktrap")
```

---

getContour *Get contour from density*

Description

Get contour from density

Usage

```
getContour(spe, coi, bins = NULL, binwidth = NULL, breaks = NULL)
```

Arguments

spe	A SpatialExperiment object.
coi	A character vector of cell types of interest (COIs).
bins	An integer. Number of contour levels.
binwidth	A numeric scale of the smoothing bandwidth.
breaks	A numeric scale referring to the breaks in `ggplot2:::contour_breaks`.

Value

A SpatialExperiment object. An sf object of the contour region of the specified level is stored in the metadata of the SpatialExperiment object.

Examples

```
data("xenium_bc_spe")

spe <- gridDensity(spe)

coi <- "Breast cancer"

spe <- getContour(spe, coi = coi)
```

getContourRegions *Calculate areas between every two density levels*

Description

Calculate areas between every two density levels

Usage

```
getContourRegions(spe, coi)
```

Arguments

spe	A SpatialExperiment object.
coi	A character vector of cell types of interest (COIs).

Value

A list of sf objects, each representing the region between two contour density levels.

grid2sf *Combine grids in each ROI to a sf region*

Description

Combine grids in each ROI to a sf region

Usage

```
grid2sf(spe)
```

Arguments

spe	A SpatialExperiment object.

Value

List of ROIs saved as sf objects.

gridDensity *Perform kernel density estimation on SpatialExperiment for cell types of interest*

Description

Perform kernel density estimation on SpatialExperiment for cell types of interest

Usage

```
gridDensity(
  spe,
  coi = NULL,
  id = "cell_type",
  kernel = "gaussian",
  bandwidth = NULL,
  ngrid.x = 100,
  ngrid.y = NULL,
  grid.length.x = NULL,
  grid.length.y = NULL,
  diggle = FALSE
)
```

Arguments

spe	A SpatialExperiment object.
coi	A character vector of cell types of interest (COIs). Default to all cell types.
id	A character. The name of the column of colData(spe) containing the cell type identifiers. Set to cell_type by default.
kernel	The smoothing kernel. Options are "gaussian", "epanechnikov", "quartic" or "disc".
bandwidth	The smoothing bandwidth. By default performing automatic bandwidth selection using cross-validation using function spatstat.explore::bw.diggle.
ngrid.x	Number of grids in the x-direction. Default to 100.
ngrid.y	Number of grids in the y-direction.
grid.length.x	Grid length in the x-direction.
grid.length.y	Grid length in the y-direction.
diggle	Logical. If TRUE, use the Jones-Diggle improved edge correction. See spatstat.explore::density.ppp() for details.

Value

A SpatialExperiment object. Grid density estimates for all cell type of interest are stored in spe@metadata$grid_density. Grid information is stored in spe@metadata$grid_info

Examples

```
data("xenium_bc_spe")

spe <- gridDensity(spe)
```

`mergeROI` *Manually merge ROIs*

Description

Manually merge ROIs

Usage

```
mergeROI(spe, merge.list, id = "component", rename = FALSE)
```

Arguments

`spe`	A SpatialExperiment object.
`merge.list`	A (named) list of vectors of ROI ids to be merged. Each vector in the list should be of length greater than or equal to 2. If no name is specified, the merged ROI will be named by concatenating ROIs being merged.
`id`	Character. The name of the column in `spe@metadata$roi` that stores the ROIs to be merged. Default is "component".
`rename`	Logical. If TRUE, names of merge.list are ignored. ROIs will be given a new name. For the unmerged ROIs, their new names are not necessarily the same as those before merging.

Value

A SpatialExperiment object.

Examples

```
data("xenium_bc_spe")

coi <- c("Breast cancer", "Fibroblasts")

spe <- gridDensity(spe, coi = coi)

spe <- findROI(spe, coi = coi, method = "walktrap")

spe <- mergeROI(spe, list("1-2" = 1:2))
```

plotCellCompo *Plot cell type composition in each density level of cell of interest.*

Description

Plot cell type composition in each density level of cell of interest.

Usage

```
plotCellCompo(
  spe,
  coi,
  id = "cell_type",
  level.name = paste0(janitor::make_clean_names(coi), "_contour"),
  by.roi = FALSE
)
```

Arguments

spe	A SpatialExperiment object.
coi	A character vector of cell types of interest (COIs).
id	A character. The name of the column of colData(spe) containing the cell type identifiers. Set to cell_type by default.
level.name	The column name that generated by cellAssign.
by.roi	Logical. Default is FALSE, set to TRUE to allow plotting by ROI.

Value

A ggplot object.

Examples

```
data("xenium_bc_spe")
spe <- gridDensity(spe, coi = c("Breast cancer", "Fibroblasts"))
spe <- findROI(spe, coi = c("Breast cancer", "Fibroblasts"))
spe <- getContour(spe, coi = "Breast cancer")
spe <- allocateCells(spe)
plotCellCompo(spe, coi = "Breast cancer")
plotCellCompo(spe, coi = "Breast cancer", by.roi = TRUE)
```

plotContour *Plot contour lines.*

Description

Plot contour lines.

Usage

```
plotContour(
  spe,
  coi,
  overlay = c("cell", "density"),
  id = "cell_type",
  sub.level = NULL,
  ...
)
```

Arguments

spe	A SpatialExperiment object.
coi	A character vector of length 1 of the cell type of interest.
overlay	Character vector. Either plot overlay on density or cell. By default is cell.
id	A character. The name of the column of colData(spe) containing the cell type identifiers. Set to cell_type by default.
sub.level	Character vector. Subset on specific level.
...	Aesthetic mappings to pass to `ggplot2::aes_string()`.

Value

A ggplot object.

Examples

```
data("xenium_bc_spe")

spe <- gridDensity(spe)

coi <- "Breast cancer"

spe <- getContour(spe, coi = coi)

plotContour(spe, coi = coi, size = 0.3, alpha = 0.2)
```

plotContourRegion *Visualising an sf object (for internal use only at the moment)*

Description

Visualising an sf object (for internal use only at the moment)

Usage

```
plotContourRegion(
  spe,
  coi,
  id = "cell_type",
  overlay = c("density", "cell"),
  sub.level
)
```

Arguments

spe	A SpatialExperiment object.
coi	A character vector of length 1 of the cell type of interest.
id	A character. The name of the column of colData(spe) containing the cell type identifiers. Set to cell_type by default.
overlay	Character vector. Either plot overlay on density or cells.
sub.level	Numeric vector of length 1 or 2, identifies which density level to plot. When length is 1, plot the density region above this level. When length is 2, plot the density region between the two levels.

Value

A ggplot object.

plotCorHeatmap *Plot model statistics using heatmap.*

Description

Plot model statistics using heatmap.

Usage

```
plotCorHeatmap(
  model.result,
  stats = c("cor.coef", "t", "p.Pos", "p.Neg"),
  roi = "all",
  cell.type = "all"
)
```

Arguments

model.result	A data.frame object.
stats	Character value. Choose either coefficient or t. Coefficient by default.
roi	Character value. By default is all. The specific ROIs to be plotted.
cell.type	Character value. By default is all. The cell types to be plotted.

Value

A pheatmap object.

Examples

```
data("xenium_bc_spe")

coi <- c("Breast cancer", "Fibroblasts", "B cells", "T cells")

spe <- gridDensity(spe, coi = coi)

spe <- findROI(spe, coi = coi, method = "walktrap")

model_result <- corDensity(spe)

plotCorHeatmap(model_result)
```

---

plotDensCor *Plot density correlation between two cell types*

Description

Plot density correlation between two cell types

Usage

```
plotDensCor(
  spe,
  celltype1 = NULL,
  celltype2 = NULL,
  by.roi = TRUE,
  fit = c("spline", "linear"),
  df = 3,
  ...
)
```

Arguments

spe	A SpatialExperiment object.
celltype1	Cell type 1 to compare.
celltype2	Cell type 2 to compare.
by.roi	Logical. Plot facet by ROIs or not.
fit	Character. Options are "spline" and "linear".
df	Integer. Degrees of freedom of the spline fit. Default to 3 (i.e., a cubic spline fit).
...	aesthetic mappings to pass to `ggplot2::aes()`.

Value

A ggplot object.

Examples

```
data("xenium_bc_spe")

coi <- c("Breast cancer", "Fibroblasts")

spe <- gridDensity(spe, coi = coi)

spe <- findROI(spe, coi = coi, method = "walktrap")

plotDensCor(spe, celltype1 = "Breast cancer", celltype2 = "Fibroblasts")
```

---

plotDensity *Plot grid-based density.*

Description

Plot grid-based density.

Usage

```
plotDensity(spe, coi, probs = 0.8)
```

Arguments

spe	A SpatialExperiment object.
coi	A character vector of cell types of interest (COIs).
probs	Numeric value between 0 and 1, used for filtering uninformative grid, default is 0.8.

Value

A ggplot object.

Examples

```
data("xenium_bc_spe")

spe <- gridDensity(spe)

plotDensity(spe, coi = "Breast cancer")

plotDensity(spe, coi = "Fibroblasts")
```

---

plotROI *Plot ROIs on spatial.*

Description

Plot ROIs on spatial.

Usage

```
plotROI(spe, id = "cell_type", show.legend = FALSE, ...)
```

Arguments

spe	A SpatialExperiment object.
id	Character. The name of the column of colData(spe) containing the cell type identifiers. Set to cell_type by default.
show.legend	Logical. Show legend or not.
...	Aesthetic mappings pass for point.

Value

A ggplot object.

Examples

```
data("xenium_bc_spe")

coi <- c("Breast cancer", "Fibroblasts")

spe <- gridDensity(spe, coi = coi)

spe <- findROI(spe, coi = coi, method = "walktrap", steps = 5)

plotROI(spe, size = 0.3, alpha = 0.2)
```

plotSpatial *Plot cells based on spatial coordinates.*

Description

Plot cells based on spatial coordinates.

Usage

```
plotSpatial(spe, reverseY = FALSE, n = 30, ...)
```

Arguments

spe	A SpatialExperiment object.
reverseY	Reverse y coordinates.
n	Integer value. The number of distinct color to be generated, default is 30.
...	Aesthetic mappings to pass to ggplot2::aes_string().

Value

A ggplot object.

Examples

```
data("xenium_bc_spe")

plotSpatial(spe, shape = ".", color = cell_type, size = 0.3, alpha = 0.2)
```

postSelRegion *Merge sel_region from the selectRegion function to SpatialExperiment.*

Description

Merge sel_region from the selectRegion function to SpatialExperiment.

Usage

```
postSelRegion(spe, sel_region)
```

Arguments

spe	A SpatialExperiment object.
sel_region	A dataframe object. Can be generated from function selectRegion.

Value

A SpatialExperiment object.

Examples

```
data("xenium_bc_spe")

coi <- c("Breast cancer", "Fibroblasts", "B cells", "T cells")

spe <- gridDensity(spe, coi = coi)

sel_region <- data.frame(
    "node" = seq(10),
    "node_x" = seq(10),
    "node_y" = seq(10)
)

spe1 <- postSelRegion(spe, sel_region)
```

---

selectRegion *Select region of interest from plot*

Description

Select region of interest from plot

Usage

```
selectRegion(data, x.col = "x", y.col = "y")
```

Arguments

data	A data.frame object.
x.col	Column name of the x coordinates.
y.col	Column name of the y coordinates.

Value

A data.frame object in the global environment.

Examples

```
data("xenium_bc_spe")

spe_b <- spe[, SummarizedExperiment::colData(spe)$cell_type == "B cells"]

dat <- as.data.frame(SpatialExperiment::spatialCoords(spe_b))
```

```
# selectRegion(dat, x.col = "x_centroid", y.col = "y_centroid")
```

---

spe2PB *Given a 'SpatialExperiment' data object, create pseudo-bulk samples using the colData information and return a DGEList object*

Description

Given a 'SpatialExperiment' data object, create pseudo-bulk samples using the colData information and return a DGEList object

Usage

```
spe2PB(
  spe,
  by.group = TRUE,
  group.id = "cell_type",
  by.roi = TRUE,
  roi.only = TRUE,
  contour = NULL
)
```

Arguments

spe	A SpatialExperiment object.
by.group	Logical. Whether to perform pseudo-bulking by group. TRUE by default.
group.id	Character. The column name of the colData(spe) that contains the group information. Default to 'cell_type'.
by.roi	Logical. Whether to perform pseudo-bulking by ROI. TRUE by default.
roi.only	Logical. Whether to filter out pseudo-bulk samples formed by cells not in any ROIs. TRUE by default.
contour	Character. The name of the group or cell type on which the contour level is computed. If NULL, then no pseudo-bulking will be performed based on contour level. Default to NULL.

Value

An edgeR::DGEList object where each library (column) is a pseudo-bulk sample.

Examples

```
data("xenium_bc_spe")

spe <- gridDensity(spe)

coi <- "Breast cancer"

spe <- findROI(spe, coi = coi)

spe <- allocateCells(spe)

y <- spe2PB(spe)
```

---

xenium_bc_spe *Description of the scider example datasets*

Description

scider-package has 1 datasets:

- xenium_bc_spe Example test spatial transcriptomics data in SpatialExperiment format. This test data is randomly subsetting from a publicly available 10X Xenium breast cancer data. Source data: https://www.10xgenomics.com/resources/datasets/xenium-ffpe-human-breast-with-custom-add-on-panel-1-standard

Usage

```
data("xenium_bc_spe")
```

Format

A SpatialExperiment object

Value

A SpatialExperiment object

Examples

```
data(xenium_bc_spe)
```

Index

∗ **internal**
scider-package, 3
xenium_bc_spe, 21

allocateCells, 3

cellsInRegion, 4
computeDensity, 5
contour2sf, 6
corDensity, 6

findROI, 7

getContour, 8
getContourRegions, 9
grid2sf, 9
gridDensity, 10

mergeROI, 11

plotCellCompo, 12
plotContour, 13
plotContourRegion, 14
plotCorHeatmap, 14
plotDensCor, 15
plotDensity, 16
plotROI, 17
plotSpatial, 18
postSelRegion, 18

scider *(*scider-package*)*, 3
scider-package, 3
selectRegion, 19
spe *(*xenium_bc_spe*)*, 21
spe2PB, 20

xenium_bc_spe, 21