Package 'SubcellularSpatialData'

July 24, 2025

Title Annotated spatial transcriptomics datasets from 10x Xenium, NanoString CosMx and BGI STOmics.

Version 1.5.0

Description This is a data package that hosts annotated sub-cellular localised datasets from the STOmics, Xenium and CosMx platforms. Specifically, it hosts datasets analysed in the publication Bhuva et. al, 2024 titled ``Library size confounds biology in spatial transcriptomics data". Raw transcript detections are hosted and functions to convert them to SpatialExperiment objects have been implemented.

biocViews ExperimentHub, SpatialData, Homo_sapiens_Data, Mus_musculus_Data, LungCancerData

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

URL https://davislaboratory.github.io/SubcellularSpatialData

BugReports https://github.com/DavisLaboratory/SubcellularSpatialData/issues

Imports dplyr, hexbin, Matrix, SpatialExperiment, stats

NeedsCompilation no

VignetteBuilder knitr

Suggests BiocStyle, ExperimentHub, ggplot2, knitr, prettydoc, rmarkdown, testthat (>= 3.0.0)

Depends R (>= 4.4)

LazyData false

Config/testthat/edition 3

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

git_branch devel

git_last_commit 53a10ce

git_last_commit_date 2025-04-15

Repository Bioconductor 3.22

Date/Publication 2025-07-24

Author Dharmesh D. Bhuva [aut, cre] (ORCID: <https://orcid.org/0000-0002-6398-9157>)

Maintainer Dharmesh D. Bhuva <dharmesh.bhuva@adelaide.edu.au>

Contents

`SubcellularSpatialData-package`
SubcellularSpatialData: Annotated spatial transcriptomics datasets from 10x Xenium, NanoString CosMx and BGI STOmics.

Description

This is a data package that hosts annotated sub-cellular localised datasets from the STOmics, Xenium and CosMx platforms. Specifically, it hosts datasets analysed in the publication Bhuva et. al, 2024 titled "Library size confounds biology in spatial transcriptomics data". Raw transcript detections are hosted and functions to convert them to SpatialExperiment objects have been implemented.

Details

The data in this package is published in the publication Bhuva et al., *Genome Biology*, 2024. Annotations for each dataset are obtained independently of the molecular measurements. The lowest level of measurement is annotated for each dataset. For BGI STOmics, this is each DNA nanoball, and for 10x Xenium and NanoString CosMx, these are the individual transcript detections.

The package provides the `tx2spe` function to convert these unit measurements into higher level summaries such as cells, regions, square bins, or hex bins for analysis purposes. These summaries are stored in SpatialExperiment objects.

Author(s)

Maintainer: Dharmesh D. Bhuva <dharmesh.bhuva@adelaide.edu.au> (ORCID)

See Also

Useful links:

- https://davislaboratory.github.io/SubcellularSpatialData
- Report bugs at https://github.com/DavisLaboratory/SubcellularSpatialData/issues

allocateHex *Allocate points to hexbins*

Description

Allocate points to hexbins

Usage

```
allocateHex(x, y, bins = 30, ...)
```

Arguments

x	a numeric, containing x-coordinates
y	a numeric, containing y-coordinates
bins	a numeric, stating the number of bins partitioning the range of x
...	additional args if needed

Value

a numeric, containing the index of bins to which each point was allocated

allocateRegion *Allocate points to annotated regions*

Description

Allocate points to annotated regions

Usage

```
allocateRegion(x, y, regions, ...)
```

Arguments

x	a numeric, containing x-coordinates
y	a numeric, containing y-coordinates
regions	a character, specifying the regions
...	additional args if needed

Value

a numeric, containing the index of bins to which each point was allocated

allocateSquare *Allocate points to a regular square grid*

Description

Allocate points to a regular square grid

Usage

```
allocateSquare(x, y, bins = 30, ...)
```

Arguments

x	a numeric, containing x-coordinates
y	a numeric, containing y-coordinates
bins	a numeric, stating the number of bins partitioning the range of x
...	additional args if needed

Value

a numeric, containing the index of bins to which each point was allocated

tx2spe *Compute cell/bin-level summaries and save to a SpatialExperiment object*

Description

This function summarises transcript detection (or DNA nanoball spot counts) into cells or hexbins and returns a SpatialExperiment object. Annotations of cells or hexbins are summarised such that the most frequent annotation is used.

Usage

```
tx2spe(x, bin = c("cell", "hex", "square", "region"), nbins = 100)
```

Arguments

x	a data.frame containing a sub-cellular localised dataset from the SubcellularSpatialData package.
bin	a character, stating whether transcripts should be binned into cells or hexbins. The default is 'cell', however, it is advisable to use 'hexbin' when cell identification is not accurate (e.g., when cell boundaries are inferred only from nuclei and not directly measured using cytoplasmic/cell membrane stains).
nbins	a numeric, stating the number of bins to create in the x and y axes

Details

Data description:

This function works on all data associated with this package. For new datasets, the transcript table MUST have the following columns:

- sample_id Unique identifier for the sample the transcript belongs to.
- cell Unique identifier for the cell the transcript belongs to (NA if it is not allocated to any cell).
- gene Gene name or identity of the transcript.
- genetype Type of target (e.g., gene or control probe). True genes should be annotated as "Gene". Other target types can be named as desired.
- x x-coordinate of the transcript.
- y y-coordinate of the transcript.
- counts Number of transcripts detected at this location (always 1 for NanoString CosMx and 10x Xenium as each row is an individual detection).
- technology Name of the platform the data was sequenced on. "STOmics" is treated differently as the number of spots as well as the number of nuclei/cells are counted. No specific format required for other technologies.

The "region" column is optional and only required if binning by regions. If present, it must be a character or factor. For datasets within this package, this column stored the independently annotated histological region.

Summarisation:

When transcript counts are aggregated (using any approach), aggregation of numeric columns is performed using the `mean(..., na.rm = TRUE)` function and aggregation of character/factor columns is performed such that the most frequent class becomes the representative class. As such, for a hex bin, the highest frequency region for the transcripts allocated to the bin becomes the bin's region annotation. New coordinates for cells, bins, or regions are computed using the mean function as well, therefore represent the center of mass for the object. For hex and square bins, the average coordinate is computed by default, however, x and y indices for the bin are stored in the colData under the *bin_x* and *bin_y* columns.

When aggregation is perfomed using bins or regions, an additional column, *ncells*, is computed that indicates how many unique cells are present within the bin/region. Do note that if a cell overlaps multiple bins/regions, it will be counted in each bin/region.

Please note that BGI only performs nuclear binning of counts therefore cellular counts obtained will represent nuclear counts only. We recommended that for fair evaluations during benchmarking, alternative binning algorithms that are fair are explored, or analysis be performed on square or hex binned counts.

Value

a SpatialExperiment object containing cell/bin/region-level expression quantification.

Examples

```
data(tx_small)
head(tx_small)
tx2spe(tx_small, "region")
```

tx_small *Sample annotated sub-cellular localised spatial 'omics dataset*

Description

This object stores a sample transcript table to demonstrate the format of data stored within this package. The sample dataset contains 100,000 transcript detections from 2 breast cancer samples profiled using the 10x Xenium platform.

Usage

```
tx_small
```

Format

A data.frame where each row represents a transcript detection. The full description of these transcript tables is available in the vignettes or in the documentation for the `tx2spe()` function.

Index