Analysing spatially resolved transcriptomics data with Bioconductor

Helena L. Crowell · PhD Student · Mark D. Robinson group · University of Zurich, Switzerland · Jan 31, 2022
**SingleCellExperiment**: S4 class for storing data from single-cell experiments

- **assays**: primary & transformed data (e.g., counts, expression-like values)
- **row/colData**: feature-/sample-level metadata
- **reducedDims**: sample-level reduced dimensionality embeddings
- **metadata**: experiment-wide metadata
- **altExps**: alternative experiments (same samples, different features)
comprehensive online book “covering installation, sources of help, specialised topics pertaining to specific aspects of scRNA-seq analysis and complete workflows [...]”
SpatialExperiment: S4 class for storing data from spatial experiments

- **spatial metadata**
  part of colData
  (e.g., array position, mapped to tissue?)

- **spatialCoords**
  slot à la reducedDims
  numeric matrix of spatial
  xy(z)-coordinates

- **slot sample_id in colData is protected**
  (against removal & arbitrary replacement)

- **imgData**
  image metadata
  (e.g., image, scale factor)
spatialImage (virtual class)

- loadedSpatialImage @image realised into memory stored as raster object

- storedSpatialImage @path local file (e.g., png, jpg, tif) loaded only on request

- remoteSpatialImage @url hosted remotely (under some URL) retrieved only on request

get/add/rmvImg accession/addition/removal of SpatialImage(s)

imgRaster & -Source accession of SpatialImage's raster object & location/URL
# extract first image
spi <- getImg(spe10x)

# apply counter-clockwise rotation
spi1 <- rotateImg(spi, -90)
spi2 <- rotateImg(spi, +90)

# visual comparison
par(mfrow = c(1, 3))
plot(as.raster(spi))
plot(as.raster(spi1))
plot(as.raster(spi2))
Each feature-observation may be associated with multiple measurements.

```
  gene cell  x  y
  1 gene48 cell11 0.0139863 0.6446542
  2 gene35 cell12 0.78304050 0.27896636
  3 gene40 cell19 0.38427813 0.27187383
  4 gene31 cell12 0.3349229 0.67031502
  5 gene45 cell16 0.5584953 0.88536730
  6 gene23 cell120 0.20483172 0.09803537
```

**bumpyMatrix** for holding non-scalar objects in each matrix entry.

```
# construct 'BumpyMatrix'
library(BumpyMatrix)
mo1 <- splitAsBumpyMatrix(
  df[, c("x", "y")],
  row = gene, col = cell)
```

### bumpyMatrix

- **features**
- **observations**

SPE offers a designated **molecules()** accessor.
SPE construction with `read10xVisium` & from scratch

**Cell Ranger output has a consistent structure...**

```r
sample
|   | outs
|---|---
|   | raw/filtered_feature_bc_matrix
|   | raw/feature_bc_matrix
|   | barcodes.tsv.gz
|   | features.ids.gz
|   | matrixmtx.gz
|   | spatial
|   | scalefactors.json.json
|   | tissue_lowres_image.png
|   | tissue_positions_list.csv

...but reading all data is cumbersome

```r
dir <- system.file(
  file.path("extdata", "10xvisium", "sections", "outs"),
  package = "SpatialExperiment"
)

# read in counts
fkm <- file.path(dir, "raw_feature_bc_matrix")
sce <- Drosophila::read10kCounts(fkm)

# read in image data
img <- readimgdata(
  path = file.path(dir, "spatial"),
  sample_id = "Foo"
)

# read in spatial coordinates
fkm <- file.path(dir, "spatial", "tissue_positions_list.csv")
xyz <- readCov(fkm, header = FALSE, colnames = c(
  "barcode", "in_tissue", "array_row", "array_col",
  "pxl_row_in_fullres", "pxl_col_in_fullres"))

# construct observation & feature metadata
rd <- S4Vectors::DataFrame(
  symbol = rowData(sce$symbs)
)

# construct `SpatialExperiment`
spc <- SpatialExperiment(
  assays = list(counts = assay(sce)),
  rawdata = rd,
  colData = data.frame(xyz),
  spatialcoordinates = c("pxl_col_in_fullres", "pxl_row_in_fullres"),
  imgData = img,
  sample_id = "Foo"
)
```

 designated `read10xVisium` constructor for Visium data

```r
dir <- system.file(
  file.path("extdata", "10xvisium"),
  package = "SpatialExperiment"
)
sample_ids <- c("section1", "section2")
samples <- file.path(dir, sample_ids, "outs")

(spcl <- read10xVisium(samples, sample_ids, type = "sparse", data = "raw",
  images = "tlowres", load = FALSE))

## class: SpatialExperiment
## dfm: 50 99
## metadatas(0):
## romnames(30): ENM22G000000012951 ENM22G000000089699 ...
## ENM22G0000000131476
## romdata names(1): symbol
## colnames(99): AAAAAACACTTACTTCTCTCCCA-1 ...
## AAGTGACCT(C)CATA-1 AAGTGCTCATCAAT-1
## coldata names(4): in_tissue array_row array_col sample_id
## reducedDimnames(0):
## mainExpName: NULL
## altExpName(0):
## spatialcoordinates(2): pxl_col_in_fullres pxl_row_in_fullres
## imgData names(4): sample_id image_id data scalefactor
```

works á la Summarized- & SingleCellExperiment

```r
spe <- SpatialExperiment(
  assays = y)

##[1] TRUE
```

accepts `spatialCoords` & from scratch

```r
xy <- as.matrix(cf, c("x", "y"))

spe2 <- SpatialExperiment(
  assay = y,
  coldata = cf[2],
  spatialcoords = xy)
```

or extracts them from `colData`

```r
c <- length(x = cletters)
y <- matrix(row = n, ncol = n)

cf <- data.frame(x = seq(x), y = seq(n), c)
spe <- SpatialExperiment(
  assay = y,
  coldata = cf,
  spatialcoordinates = c("x", "y")
)
```

```r
imgData may be provided or added downstream
```

coercion from SCE also works!
ExperimentHub package available since Bioc 3.14

collection of 10X Genomics Visium spatial gene expression datasets

13 datasets from 23 samples across two organisms (human & mouse) & 13 tissues

also: STexampleData & SingleCellMultiModal (seq-FISH + scRNA-seq of mouse visual cortex)

TENxVisiumData & other data packages

- HumanBreastCancerIDC
  - Human Breast Cancer (Block A Section 1)
  - Human Breast Cancer (Block A Section 2)

- HumanBreastCancerLC
  - Human Breast Cancer: Whole Transcriptome Analysis
  - Human Breast Cancer: Targeted, Immunology Panel

- HumanCerebellum
  - Human Cerebellum: Whole Transcriptome Analysis
  - Human Cerebellum: Targeted, Neuroscience Panel

- HumanColorectalCancer
  - Human Colorectal Cancer: Whole Transcriptome Analysis
  - Human Colorectal Cancer: Targeted, Gene Signature Panel

- HumanGlioblastoma
  - Human Glioblastoma: Whole Transcriptome Analysis
  - Human Glioblastoma: Targeted, Pan-Cancer Panel

- HumanHeart
  - Human Heart

- HumanLymphNode
  - Human Lymph Node

- HumanOvarianCancer
  - Human Ovarian Cancer: Whole Transcriptome Analysis
  - Human Ovarian Cancer: Targeted, Immunology Panel
  - Human Ovarian Cancer: Targeted, Pan-Cancer Panel

- HumanSpiralCord
  - Human Spinal Cord: Whole Transcriptome Analysis
  - Human Spinal Cord: Targeted, Neuroscience Panel

- MouseBrainCoronal
  - Mouse Brain Section (Coronal)

- MouseBrainSagittalAnterior
  - Mouse Brain Serial Section 1 (Sagittal-Anterior)
  - Mouse Brain Serial Section 2 (Sagittal-Anterior)

- MouseBrainSagittalPosterior
  - Mouse Brain Serial Section 1 (Sagittal-Posterior)
  - Mouse Brain Serial Section 2 (Sagittal-Posterior)

- MouseKidneyCoronal
  - Mouse Kidney Section (Coronal)
- designed to work with the `SpatialExperiment` class
- support for both spot- & molecule-based visualizations
- flexible colouring, highlighting & (optional) image overlaying
- Shiny-based GUI for interactive data exploration
- allows for deployment of custom panels
- SPE inherits from SCE
  inherits from SE
  - lots of room for extensions specific to spatially resolved data!
    (e.g., spatial plots)
spatialLIBD for DLPFC study & visualising ST data

- complex Visium dataset in SPE format
- human brain dorsolateral prefrontal cortex (DLPFC) region
- 2 pairs of spatially adjacent replicates across 3 subjects (12 samples in total)

- continuous & discrete spot-level visualization (including histology image)
- interactive data exploration (Shiny- & plotly-based)
- exportable visualizations (PDF/PNG) & results (CSV)
SPOTlight for deconvolution of cell-mixtures from single-cell reference data

- originally only on GH & inter-operable with Seurat 😞
- recently submitted to Bioc & inter-operable with SCE 😞
 OSTA: Orchestrating Spatially Resolved Transcriptomics Analysis with Bioconductor

- early days
  (Bioconductor-based packages dedicated to ST are virtually non-existent)

- will include...
  - theoretical introduction
    (e.g., technologies, SPE)
  - preprocessing
    (e.g., segmentation)
  - analysis walkthrough
    (some transferable from single-cell, other ST-specific)
  - exemplary workflows
    (e.g., SVG selection, spatial clustering)

OSTA

Welcome

1. Introduction
2. Spatially resolved transcriptomics
3. SpatialExperiment
4. Preprocessing steps
5. Image segmentation (Viaur)
6. Loupe Browser (Viaur)
7. Space Ranger (Viaur)
8. Analysis steps
9. Quality control
10. Normalization
11. Feature selection
12. Dimensionality reduction
13. Clustering
14. Marker genes
15. Spot-level deconvolution
16. Workflows and comparisons
17. Human DLPPC workflow

Orchestrating Spatially Resolved Transcriptomics Analysis with Bioconductor

2022-01-07

Welcome

This is the website for the online textbook
"Orchestrating Spatially Resolved Transcriptomics Analysis with Bioconductor" (OSTA).

This book describes the steps in a computational analysis pipeline for spatially resolved transcriptomics (ST) data, using the Bioconductor framework and R programming language. The analysis pipeline is built up as a series of steps, each described in a chapter, with complete examples and workflows using R code and datasets that can be run on your own laptop.

The book is organized into several parts, consisting of (i) introduction, (ii) preprocessing steps to prepare data for R, (iii) analysis steps, (iv) complete workflows, and (v) appendix.

- Lukas M. Weber, Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA
- Abby Spangler, Lieber Institute for Brain Development, Baltimore, MD, USA
- Madhavi Tippani, Lieber Institute for Brain Development, Baltimore, MD, USA
- Leonardo Collado-Torres, Lieber Institute for Brain Development, Baltimore, MD, USA
- Stephanie C. Hicks, Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA
Summary, outlook & acknowledgement

- **SpatialExperiment** for storage
- **ggspavis** for visualization
- **iSEE & spatialLIBD** for interactivity
- **TENxVisiumData & STexampleData** for; well, data
- **OSTA** for everything

Flipped altExps?
(same features, different samples)

- multiple binning resolutions
  (e.g., Visium HD, Stereo-seq)
- different segmentations
  (e.g., molecule-based ST data)
- sample-based geometric data
  (e.g., polygon vertices)
- ST-specific iSEE modules
- (Bioc-based) analysis packages dedicated to ST are lacking
  (e.g., basic spatial statistics)
- python-interoperability
  (e.g., AnnData for Squidpy)