University of Zurich[UZH]

Swiss Institute of Bioinformatics

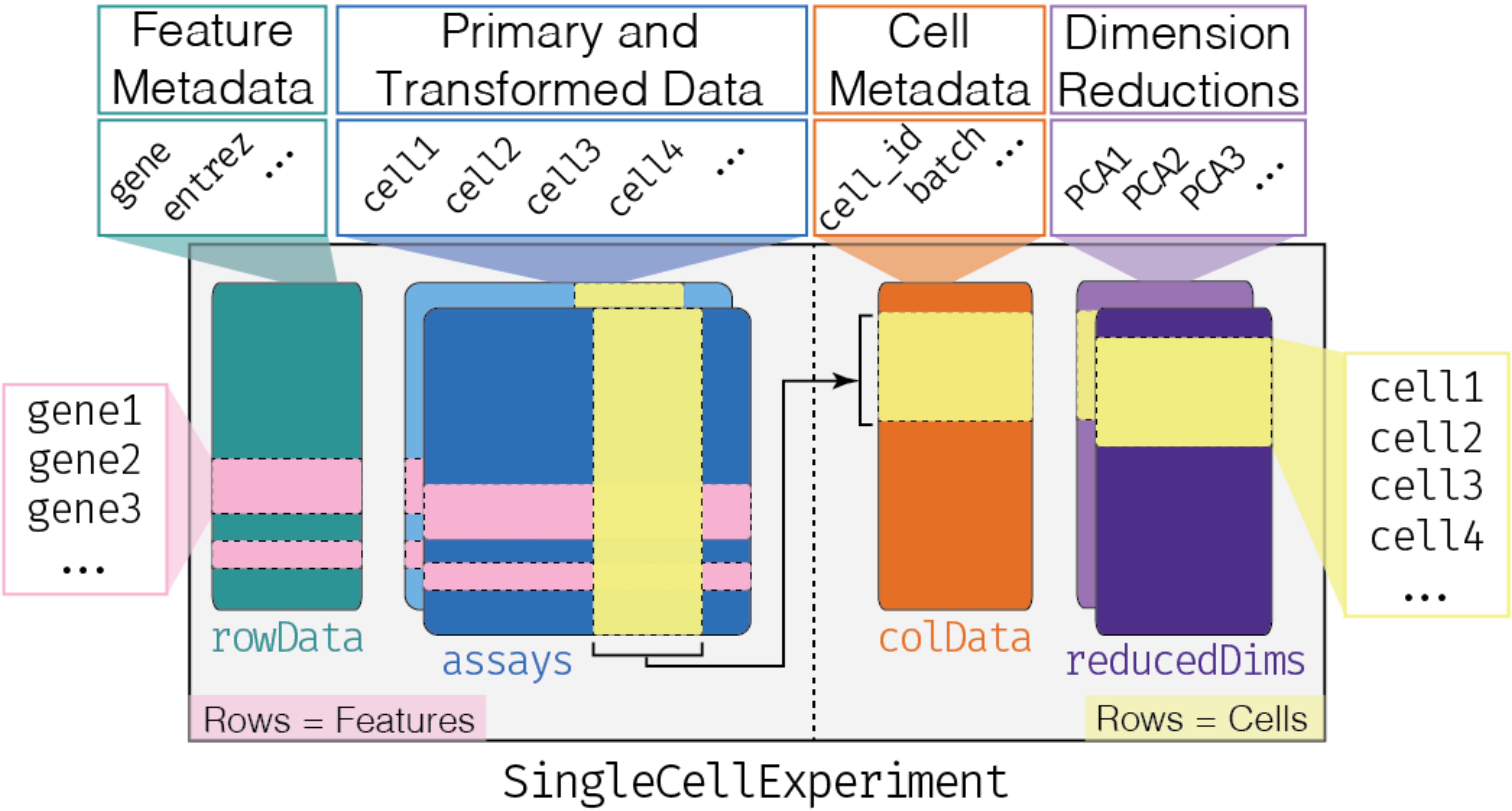
@CrowellHL

Analysing spatially resolved transcriptomics data with Bioconductor

Helena L. Crowell · PhD Student · Mark D. Robinson group · University of Zurich, Switzerland · Jan 31, 2022

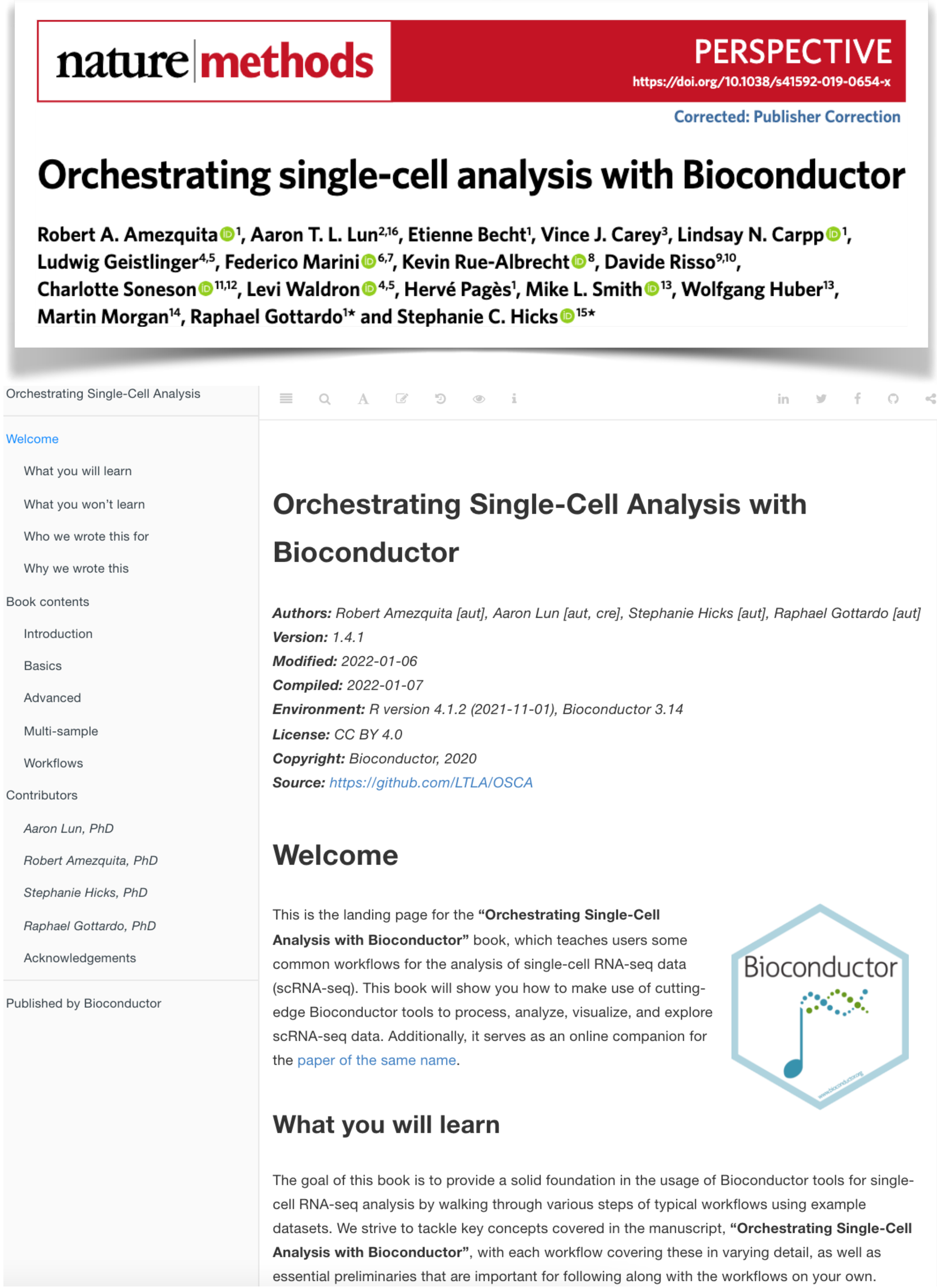
SingleCellExperiment: S4 class for storing data from single-cell experiments

- `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)



OSCA: Orchestrating Single-Cell Analysis with Bioconductor

- comprehensive online book
  "covering installation, sources of help, specialised topics pertaining to specific aspects of scRNA-seq analysis and complete workflows […]"


nature methods PERSPECTIVE
https://doi.org/10.1038/s41592-019-0654-x
Corrected: Publisher Correction

Orchestrating single-cell analysis with Bioconductor

Robert A. Amezquita[1], Aaron T. L. Lun[2,16], Etienne Becht[1], Vince J. Carey[3], Lindsay N. Carpp[1], Ludwig Geistlinger[4,5], Federico Marini[6,7], Kevin Rue-Albrecht[8], Davide Risso[9,10], Charlotte Soneson[11,12], Levi Waldron[4,5], Hervé Pagès[1], Mike L. Smith[13], Wolfgang Huber[13], Martin Morgan[14], Raphael Gottardo[1]* and Stephanie C. Hicks[15]*

Orchestrating Single-Cell Analysis

Welcome
What you will learn
What you won't learn
Who we wrote this for
Why we wrote this
Book contents
Introduction
Basics
Advanced
Multi-sample
Workflows
Contributors
Aaron Lun, PhD
Robert Amezquita, PhD
Stephanie Hicks, PhD
Raphael Gottardo, PhD
Acknowledgements
Published by Bioconductor

Orchestrating Single-Cell Analysis with Bioconductor

Authors: Robert Amezquita [aut], Aaron Lun [aut, cre], Stephanie Hicks [aut], Raphael Gottardo [aut]
Version: 1.4.1
Modified: 2022-01-06
Compiled: 2022-01-07
Environment: R version 4.1.2 (2021-11-01), Bioconductor 3.14
License: CC BY 4.0
Copyright: Bioconductor, 2020
Source: https://github.com/LTLA/OSCA

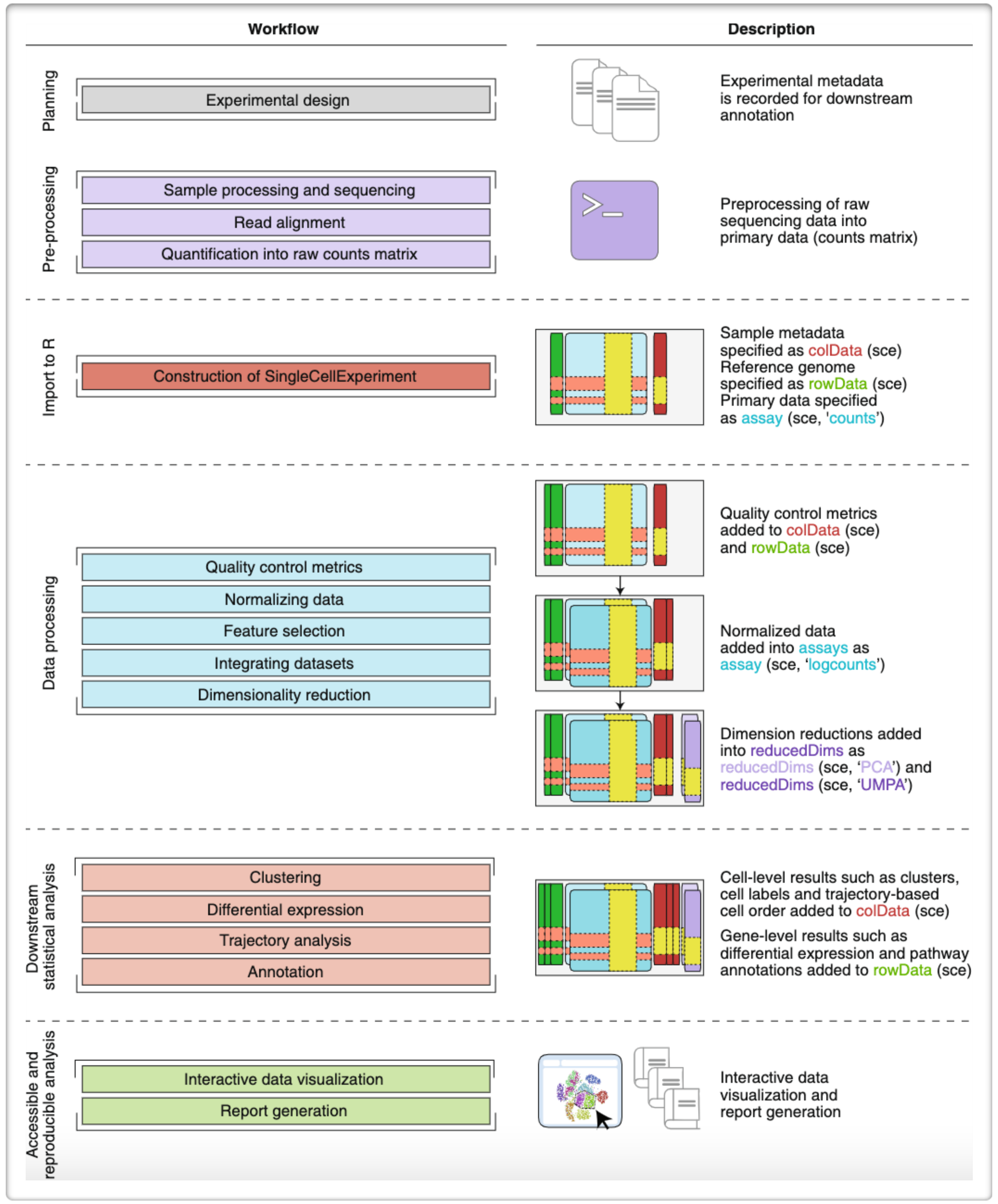
Welcome

This is the landing page for the **"Orchestrating Single-Cell Analysis with Bioconductor"** book, which teaches users some common workflows for the analysis of single-cell RNA-seq data (scRNA-seq). This book will show you how to make use of cutting-edge Bioconductor tools to process, analyze, visualize, and explore scRNA-seq data. Additionally, it serves as an online companion for the paper of the same name.

Bioconductor

What you will learn

The goal of this book is to provide a solid foundation in the usage of Bioconductor tools for single-cell RNA-seq analysis by walking through various steps of typical workflows using example datasets. We strive to tackle key concepts covered in the manuscript, **"Orchestrating Single-Cell Analysis with Bioconductor"**, with each workflow covering these in varying detail, as well as essential preliminaries that are important for following along with the workflows on your own.

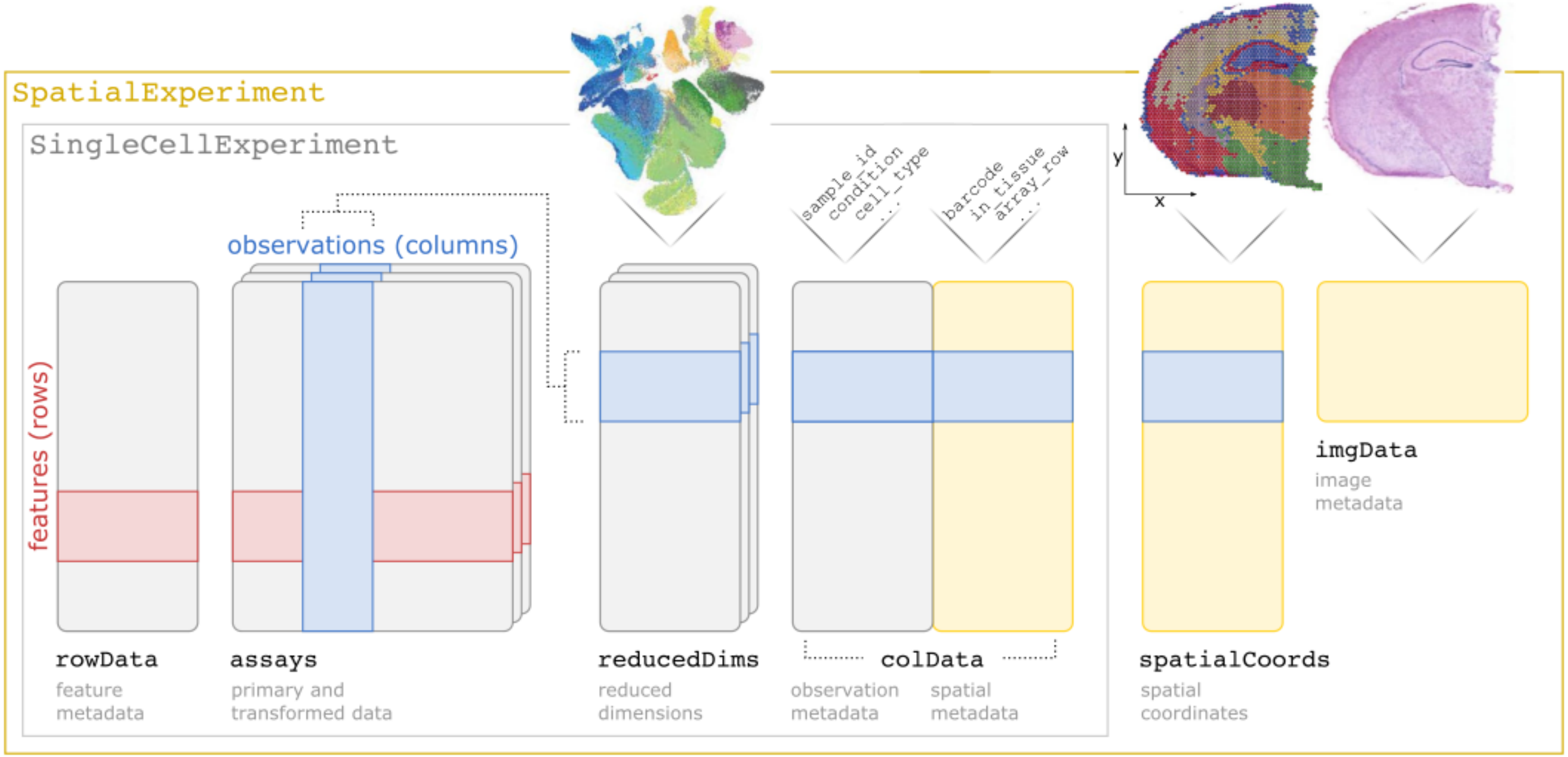


SpatialExperiment: infrastructure for spatially resolved transcriptomics data in R using Bioconductor

Dario Righelli, Lukas M. Weber, Helena L. Crowell, Brenda Pardo, Leonardo Collado-Torres, Shila Ghazanfar, Aaron T. L. Lun, Stephanie C. Hicks, Davide Risso

doi: https://doi.org/10.1101/2021.01.27.428431

- **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 <u>protected</u>
  (against removal & arbitrary replacement)
- **`imgData`**
  image metadata
  (e.g., image, scale factor)



imgData for storing image-related metadata

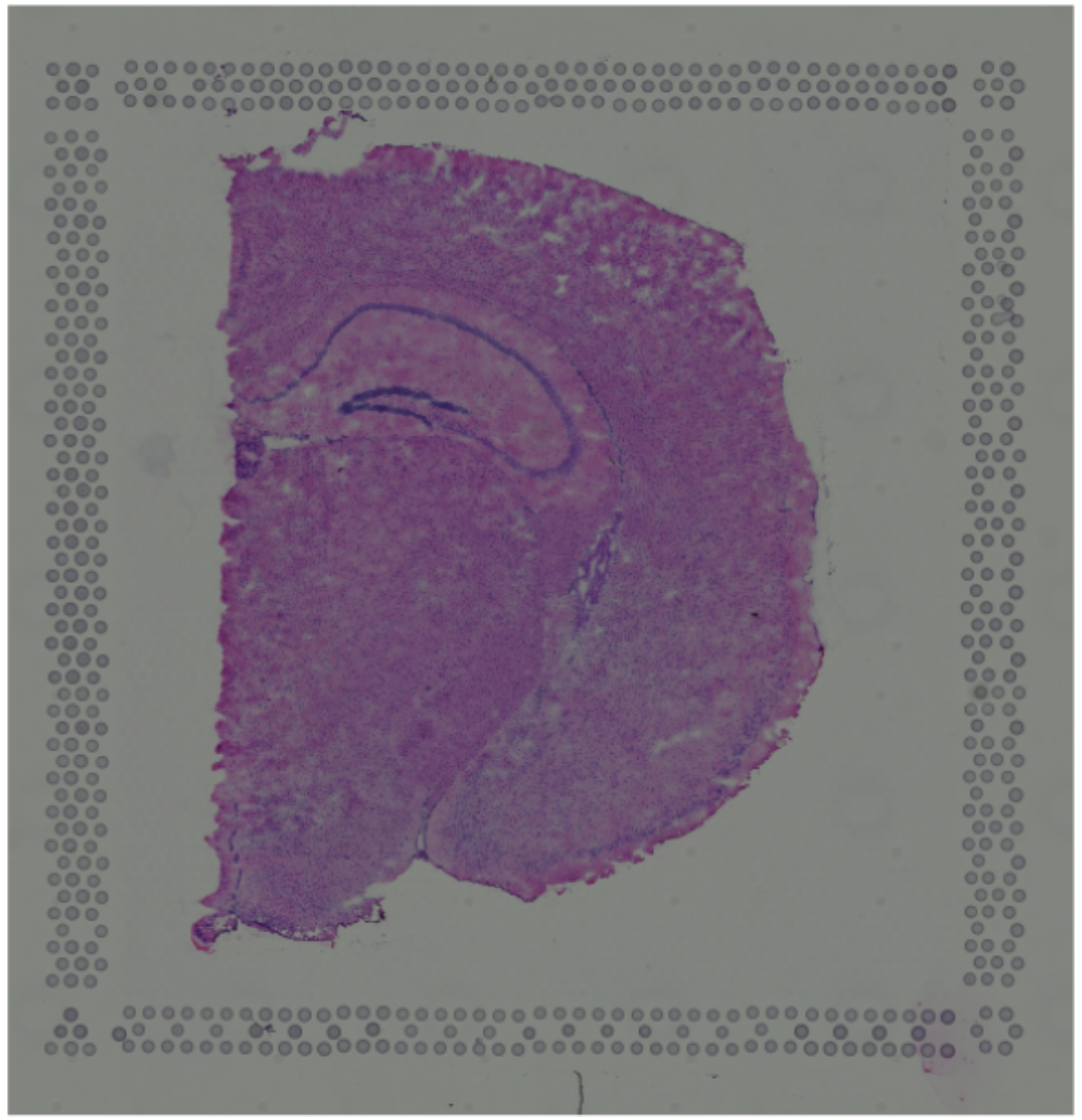
```
imgData(spe)
```

```
## DataFrame with 2 rows and 4 columns
##     sample_id    image_id   data scaleFactor
##   <character> <character> <list>   <numeric>
## 1    section1      lowres   ####   0.0510334
## 2    section2      lowres   ####   0.0510334
```

- `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

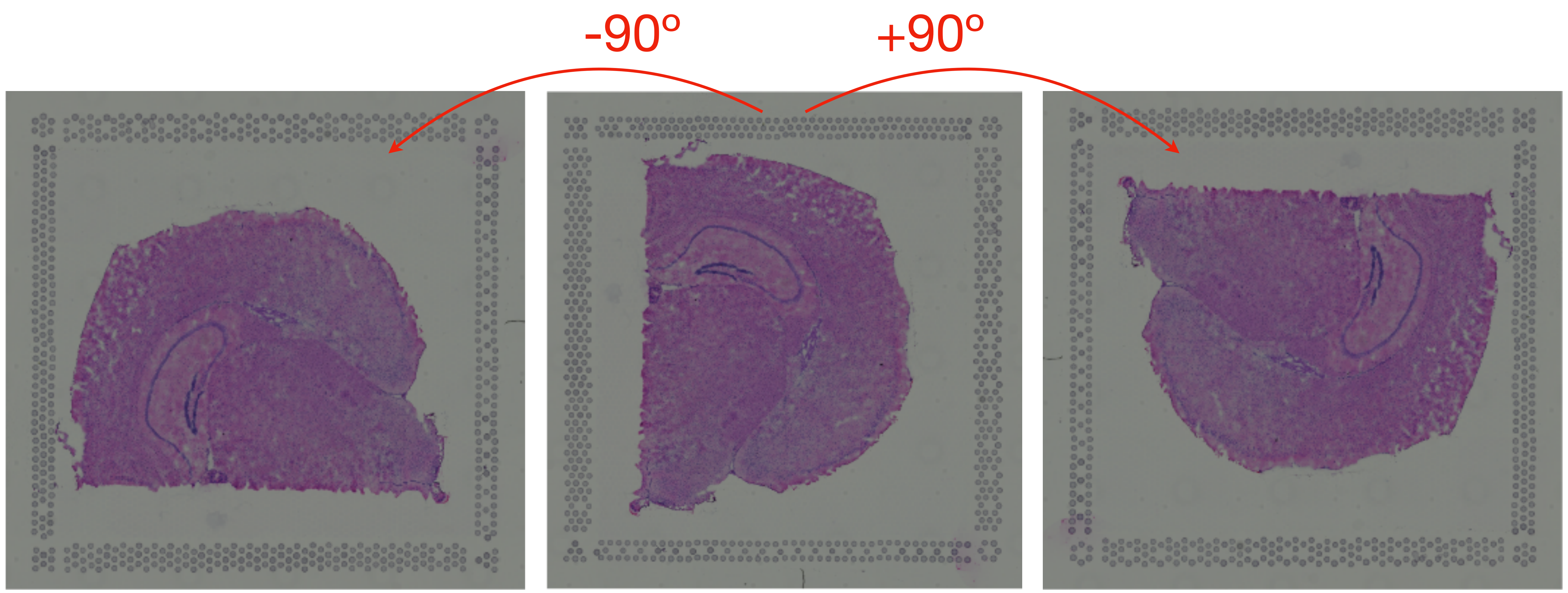
- `get/add/rmvImg`
  accession/addition/removal
  of `SpatialImage(s)`
- `imgRaster & -Source`
  accession of `SpatialImage`'s
  `raster` object & location/URL

```
url <- "https://i.redd.it/3pw5uah7xo041.jpg"
spe <- addImg(spe,
    sample_id = "section1",
    image_id = "pomeranian",
    imageSource = url,
    scaleFactor = NA_real_,
    load = TRUE)
img <- imgRaster(spe,
    sample_id = "section1",
    image_id = "pomeranian")
plot(img)
```

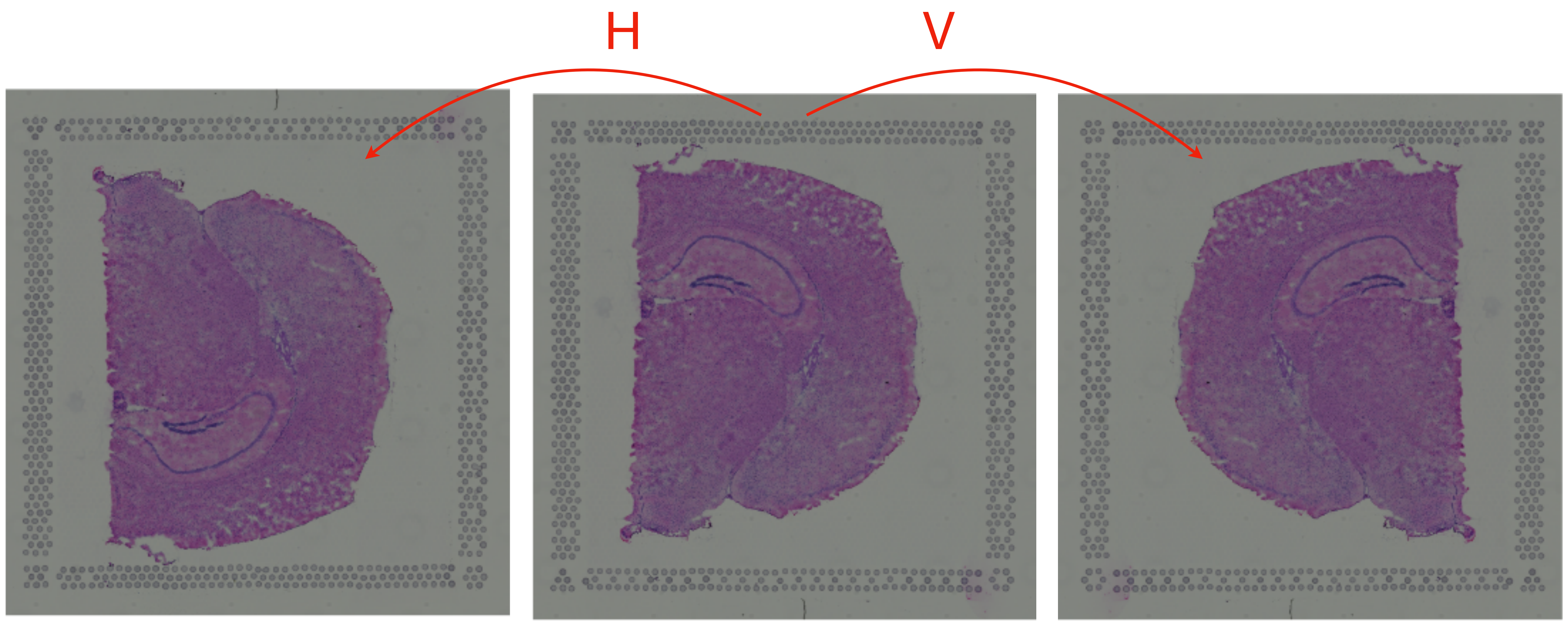


rotate- & mirrorImg for image transformation

```r
# 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))
```




```r
# extract first image
spi <- getImg(spe10x)
# mirror horizontally/vertically
spi1 <- mirrorImg(spi, "h")
spi2 <- mirrorImg(spi, "v")
# visual comparison
par(mfrow = c(1, 3))
plot(as.raster(spi))
plot(as.raster(spi1))
plot(as.raster(spi2))
```



Aaron T. Lun

- each feature-observation may be associated with multiple measurements

```
##     gene   cell          x          y
## 1 gene48 cell11 0.03539863 0.64465423
## 2 gene15 cell12 0.78304050 0.27896636
## 3 gene40  cell9 0.38427813 0.27187383
## 4 gene31  cell2 0.33349229 0.67031502
## 5 gene45  cell6 0.55584953 0.98536730
## 6 gene23 cell20 0.20483172 0.09803537
```

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

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

```
## 50 x 20 BumpyDataFrameMatrix
## rownames: gene1 gene2 ... gene49 gene50
## colnames: cell1 cell2 ... cell19 cell20
## preview [1,1]:
##   DataFrame with 0 rows and 2 columns
```

- SPE offers a designated `molecules()` accessor

SPE **construction** with *read10xVisium* & from scratch

- *Cell Ranger* output has a consistent structure…

```
sample
 . | – outs
 · · | – raw/filtered_feature_bc_matrix.h5
 · · | – raw/filtered_feature_bc_matrix
 · · · | – barcodes.tsv.gz
 · · · | – features.tsv.gz
 · · · | – matrix.mtx.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", "section1", "outs"),
   package = "SpatialExperiment")

# read in counts
fnm <- file.path(dir, "raw_feature_bc_matrix")
sce <- DropletUtils::read10xCounts(fnm)

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

# read in spatial coordinates
fnm <- file.path(dir, "spatial", "tissue_positions_list.csv")
xyz <- read.csv(fnm, header = FALSE,
    col.names = 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)$Symbol)

# construct 'SpatialExperiment'
(spe <- SpatialExperiment(
    assays = list(counts = assay(sce)),
    rowData = rd,
    colData = DataFrame(xyz),
    spatialCoordsNames = 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")

(spe10x <- read10xVisium(samples, sample_ids,
    type = "sparse", data = "raw",
    images = "lowres", load = FALSE))
```

```
## class: SpatialExperiment
## dim: 50 99
## metadata(0):
## assays(1): counts
## rownames(50): ENSMUSG00000051951 ENSMUSG00000089699 ...
##   ENSMUSG00000005886 ENSMUSG00000101476
## rowData names(1): symbol
## colnames(99): AAACAACGAATAGTTC-1 AAACAAGTATCTCCCA-1 ...
##   AAAGTCGACCCTCAGT-1 AAAGTGCCATCAATTA-1
## colData names(4): in_tissue array_row array_col sample_id
## reducedDimNames(0):
## mainExpName: NULL
## altExpNames(0):
## spatialCoords names(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)
isEmpty(spatialCoords(spe))
```

```
## [1] TRUE
```

- accepts `spatialCoords`

```r
xy <- as.matrix(cd[, c("x", "y")])

spe2 <- SpatialExperiment(
    assay = y,
    colData = cd["z"],
    spatialCoords = xy)
```

- or extracts them from `colData`

```r
n <- length(z <- letters)
y <- matrix(nrow = n, ncol = n)
cd <- DataFrame(x = seq(n), y = seq(n), z)

spe1 <- SpatialExperiment(
    assay = y,
    colData = cd,
    spatialCoordsNames = c("x", "y"))
```

- `imgData` may be provided or added downstream
- coercion from SCE also works!

TENxVisiumData & other data packages

- `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)

```r
library(ExperimentHub)
eh <- ExperimentHub()        # initialize hub instance
q <- query(eh, "TENxVisium") # retrieve 'TENxVisiumData' records
id <- q$ah_id[1]             # specify dataset ID to load
spe <- eh[[id]]              # load specified dataset
```

```
## ExperimentHub with 26 records
## # snapshotDate(): 2021-10-18
## # $dataprovider: 10X Genomics
## # $species: Homo sapiens, Mus musculus
## # $rdataclass: SpatialExperiment
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["EH6695"]]'
##
##           title
##   EH6695 | HumanBreastCancerIDC
##   EH6696 | HumanBreastCancerILC
##   EH6697 | HumanCerebellum
##   EH6698 | HumanColorectalCancer
##   EH6699 | HumanGlioblastoma
##   ...      ...
##   EH6739 | HumanSpinalCord_v3.13
##   EH6740 | MouseBrainCoronal_v3.13
##   EH6741 | MouseBrainSagittalPosterior_v3.13
##   EH6742 | MouseBrainSagittalAnterior_v3.13
##   EH6743 | MouseKidneyCoronal_v3.13
```

```r
spe
```

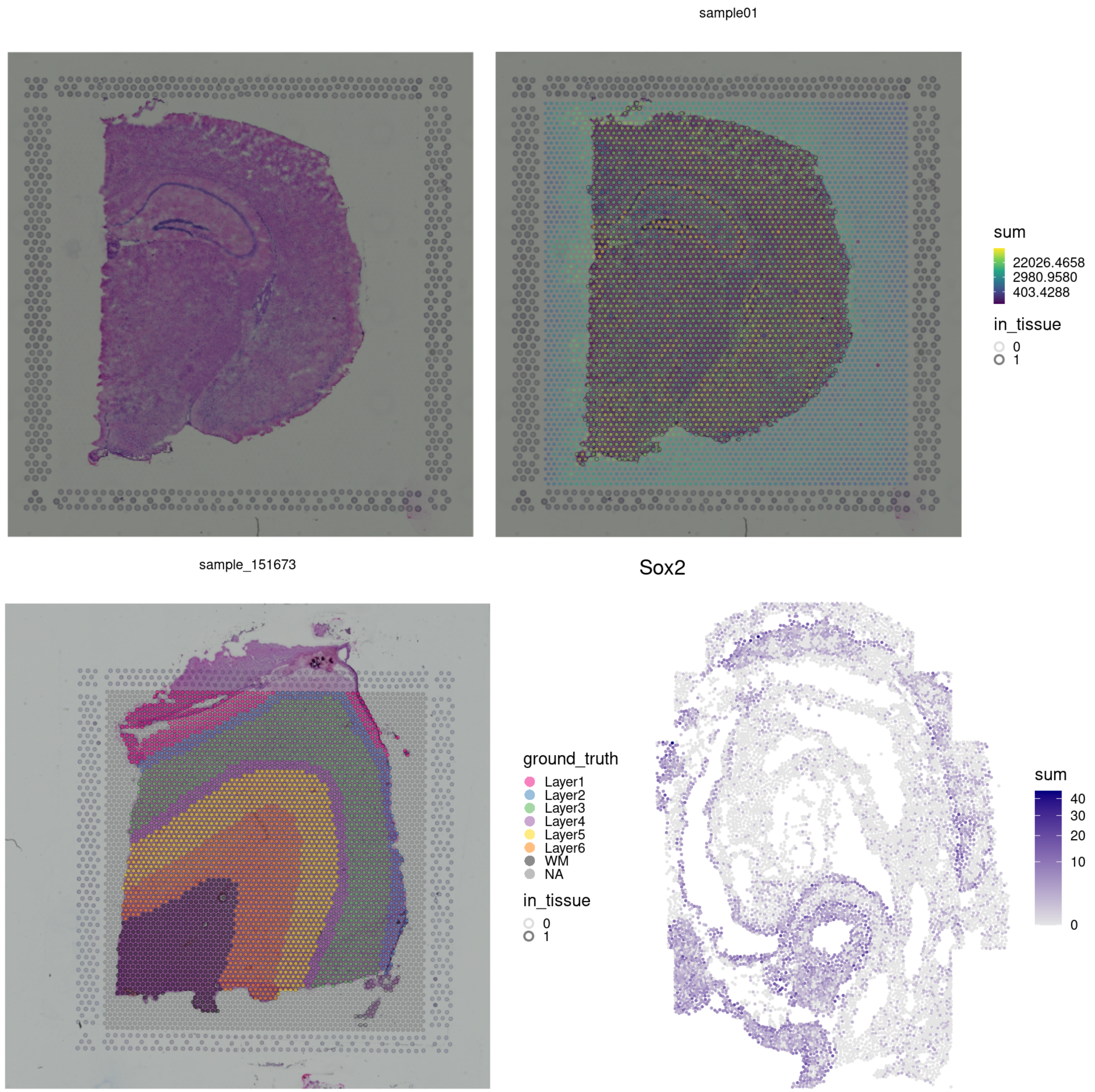
```
## class: SpatialExperiment
## dim: 36601 7785
## metadata(0):
## assays(1): counts
## rownames(36601): ENSG00000243485 ENSG00000237613 ... ENSG00000278817
##   ENSG00000277196
## rowData names(1): symbol
## colnames(7785): AAACAAGTATCTCCCA-1 AAACACCAATAACTGC-1 ...
##   TTGTTTGTATTACACG-1 TTGTTTGTGTAAATTC-1
## colData names(1): sample_id
## reducedDimNames(0):
## mainExpName: NULL
## altExpNames(0):
## spatialData names(3) : in_tissue array_row array_col
## spatialCoords names(2) : pxl_col_in_fullres pxl_row_in_fullres
## imgData names(4): sample_id image_id data scaleFactor
```

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

ggspavis for visualization

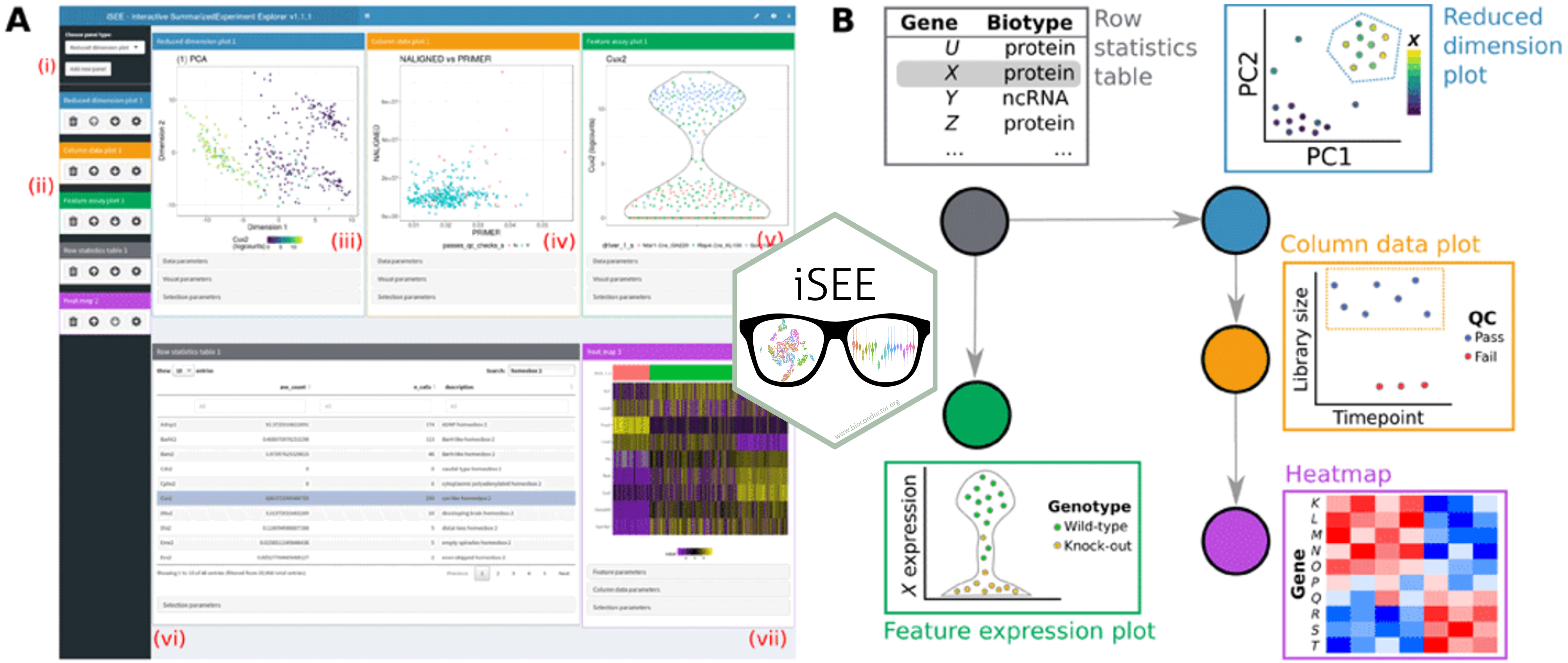
Lukas M. Weber

- designed to work with the `SpatialExperiment` class
- support for both spot- & molecule-based visualizations
- flexible colouring, highlighting & (optional) image overlaying



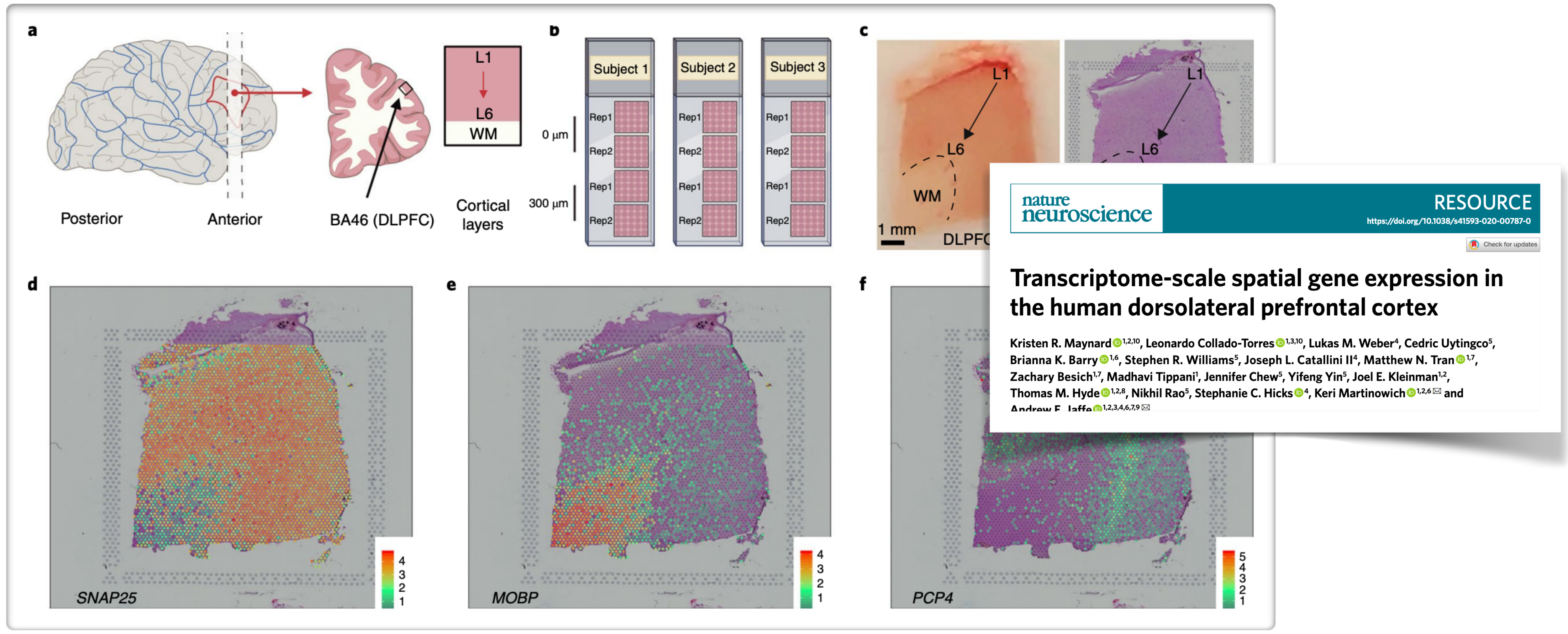
iSEE: interactive SummarizedExperiment explorer

- 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)

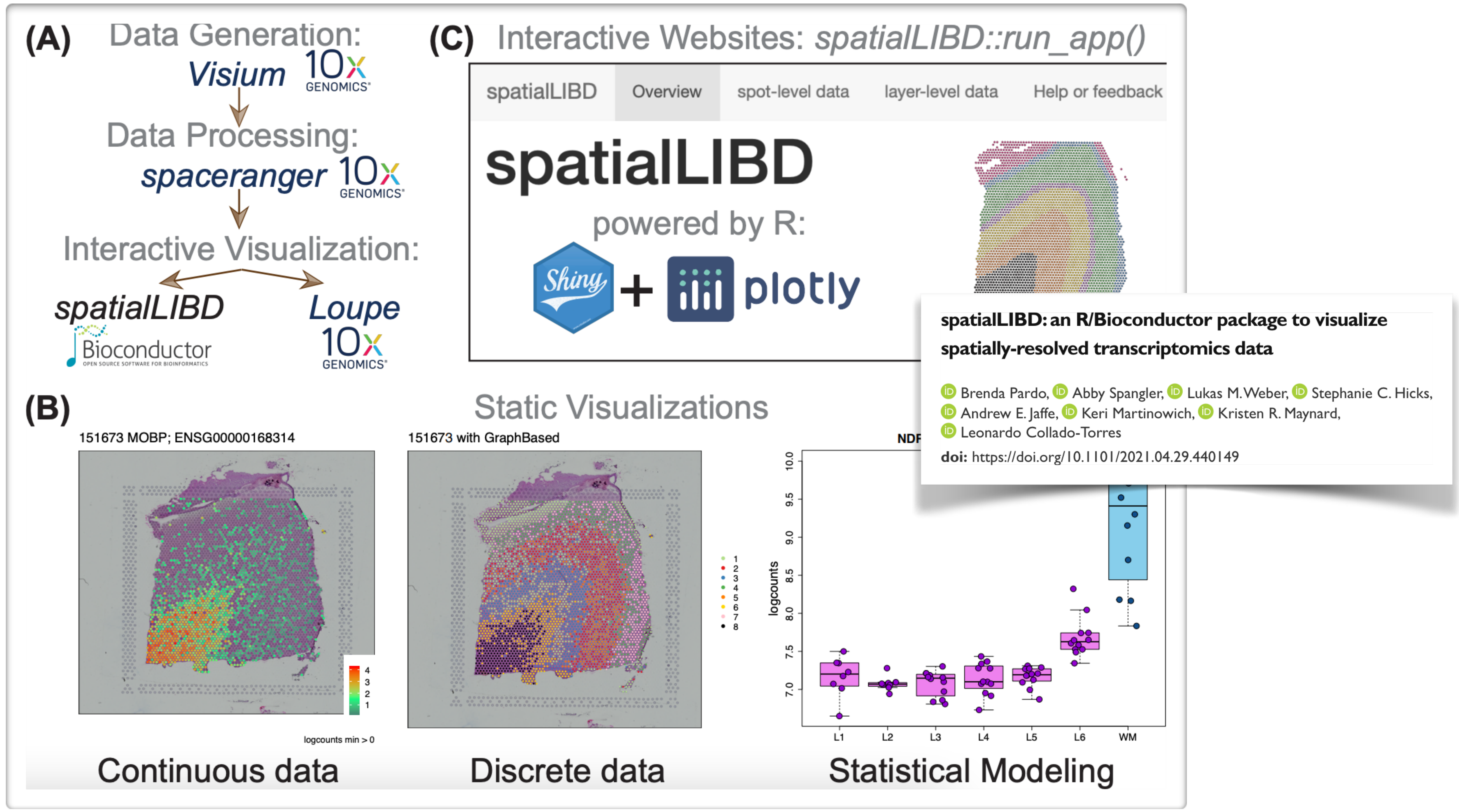


spatialLIBD for DLPFC study & visualising ST data

- 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)



Marc Elosua-Bayés

SPOTlight

Published online 5 February 2021 *Nucleic Acids Research, 2021, Vol. 49, No. 9 e50*
doi: 10.1093/nar/gkab043

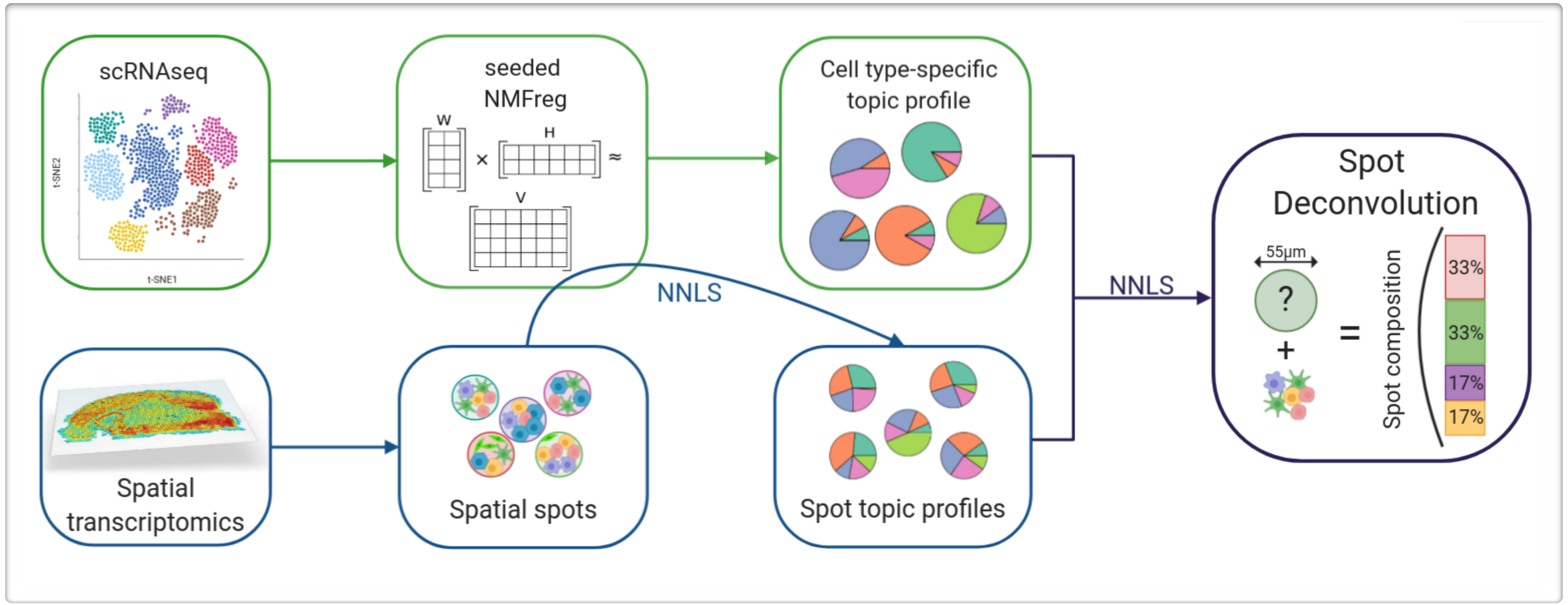
SPOTlight: seeded NMF regression to deconvolute spatial transcriptomics spots with single-cell transcriptomes

Marc Elosua-Bayes[1], Paula Nieto[1], Elisabetta Mereu[1], Ivo Gut[1,2] and Holger Heyn[1,2,*]

[1]CNAG-CRG, Centre for Genomic Regulation (CRG), Barcelona Institute of Science and Technology (BIST), Barcelona, Spain and [2]Universitat Pompeu Fabra (UPF), Barcelona, Spain

Received November 18, 2020; Revised January 04, 2021; Editorial Decision January 09, 2021; Accepted January 15, 2021

- originally only on GH & inter-operable with `Seurat` 😢
- recently submitted to Bioc & inter-operable with SCE 🥳



OSTA: Orchestrating Spatially Resolved Transcriptomics Analysis *with Bioconductor*

Lukas M. Weber

- early days
  (Bioc-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

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`)

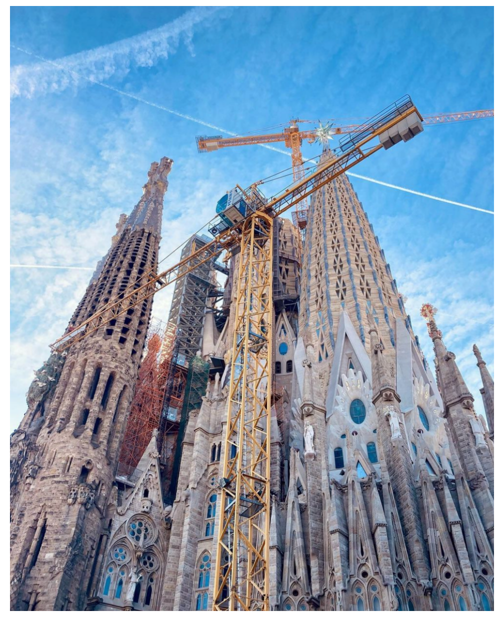

Lukas M. Weber


Aaron T. Lun

Bioc community


Stephanie Hicks


Davide Risso


Heyn lab (BCN, SPA)


Shila Ghazanfar


Leonardo Collado-Torres


Brenda Pardo