UCI
Genomics Research and Technology Hub
ROBERT R. SPRAGUE FOUNDATION HALL

Introduction to Spatial Transcriptomic (ST) Data Analysis

Jenny Wu

Director of Bioinformatics

Genomics Research and Technology Hub

Chao Family Comprehensive Cancer Center

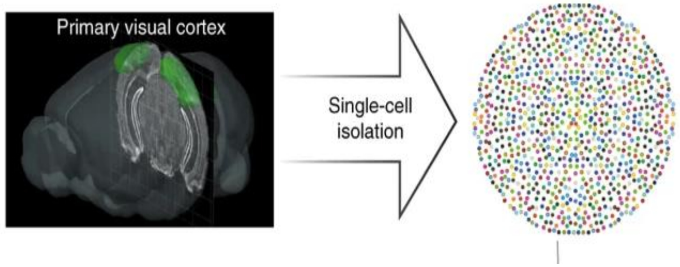
UC Irvine

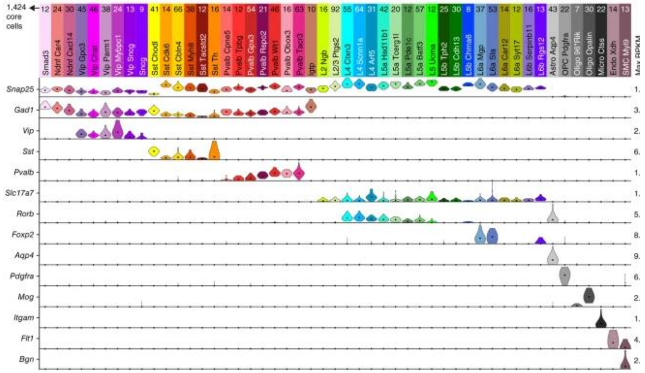
Outline

- Why spatial omics
- Spatial transcriptomic technologies: Sequencing vs Imaging based
- Data analysis pipeline and applications
  - ✓ Pre-processing: platform dependent
  - ✓ Downstream analysis and visualization
- Pre-processing with Space Ranger and Xenium onboard analysis Ranger
- Downstream analysis pipeline - Seurat Workflow
  - ✓ Data import
  - ✓ QC, filtering and feature selection
  - ✓ Dimension reduction and clustering
  - ✓ Data visualization and integration
- Advanced topics: cell segmentation, cell type deconvolution, integration with scRNA data, and inter cellular communication

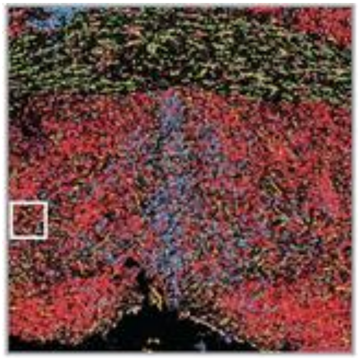
Why Spatial Biology

- Single cell sequencing reveals cellular heterogeneity



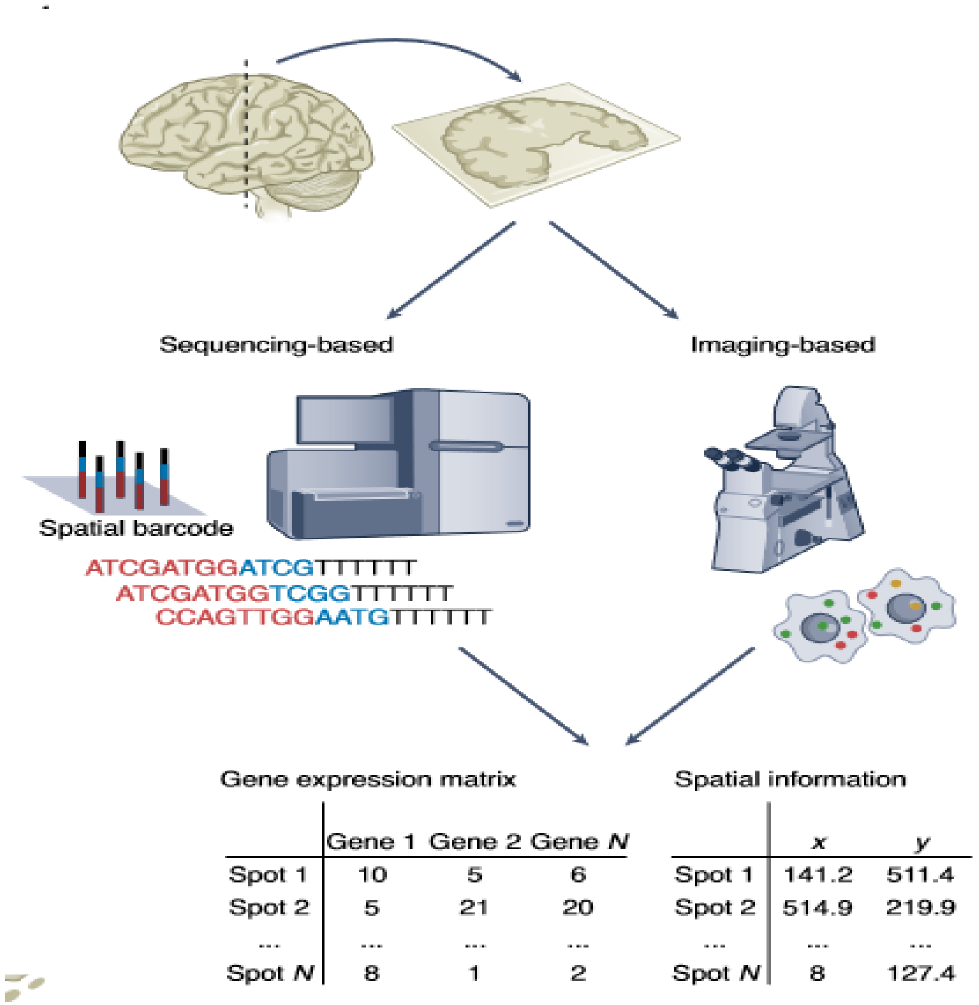


- Spatial information is important



Two Classes of ST Technology

- Sequencing Based (SST) gene detection
    - *10x Visium, Slide-seq2, Stereo-seq etc*
    - Whole transcriptome but not true single cell resolution
- Imaging based (IST) gene detection
    - *10x Xenium, MERFISH, Resolve etc*
    - Subcellular resolution but limited gene throughput



Gene expression matrix

	Gene 1	Gene 2	Gene *N*
Spot 1	10	5	6
Spot 2	5	21	20
...	...	...	...
Spot *N*	8	1	2

Spatial information

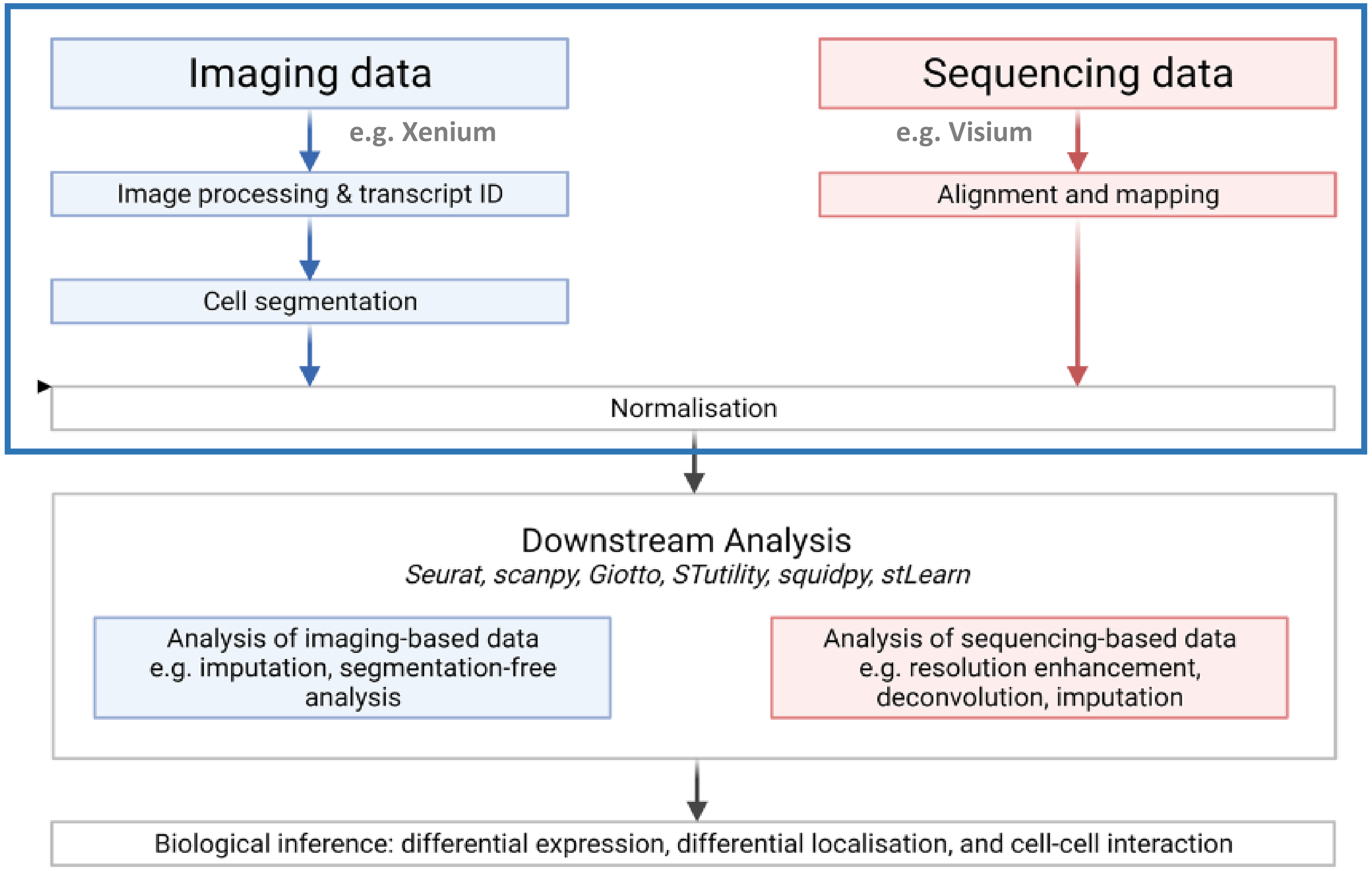
	x	*y*
Spot 1	141.2	511.4
Spot 2	514.9	219.9
...	...	...
Spot *N*	8	127.4

Tian *et al*, 2023

General Data Analysis Workflow

- Pre-processing is handled by proprietary software such as *Space Ranger or* on Xenium Analyzer instrument.

- Downstream analysis may be performed with a wide range of transcriptomics analysis packages: *Seurat, Scanpy, Giotto* etc.



Imaging data

e.g. Xenium

Image processing & transcript ID

Cell segmentation

Sequencing data

e.g. Visium

Alignment and mapping

Normalisation

Downstream Analysis

Seurat, scanpy, Giotto, STutility, squidpy, stLearn

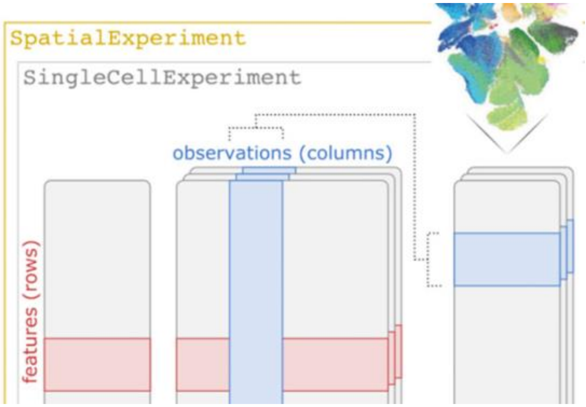
Analysis of imaging-based data e.g. imputation, segmentation-free analysis

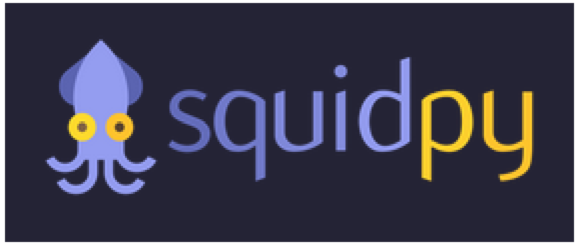
Analysis of sequencing-based data e.g. resolution enhancement, deconvolution, imputation

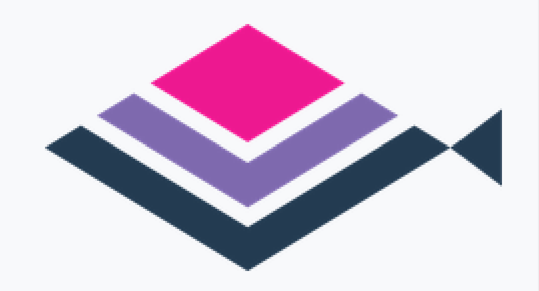
Biological inference: differential expression, differential localisation, and cell-cell interaction

Williams *et al*, 2022

Analytical tools for ST Downstream Analysis



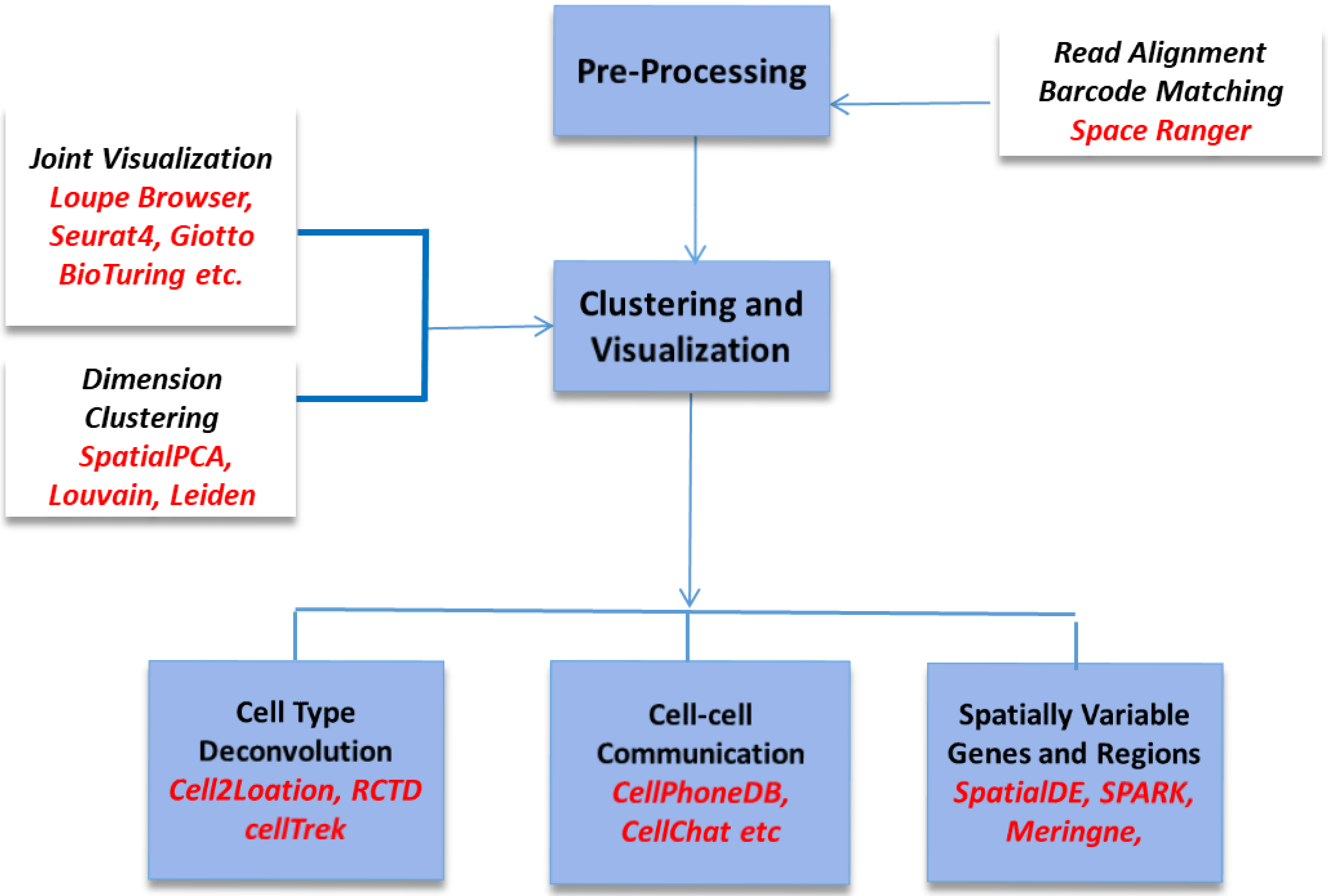




Giotto

QuPath

Visium Data Analysis Work Flow



Pre-Processing

Read Alignment
Barcode Matching
Space Ranger

Clustering and Visualization

Joint Visualization
Loupe Browser, Seurat4, Giotto BioTuring etc.

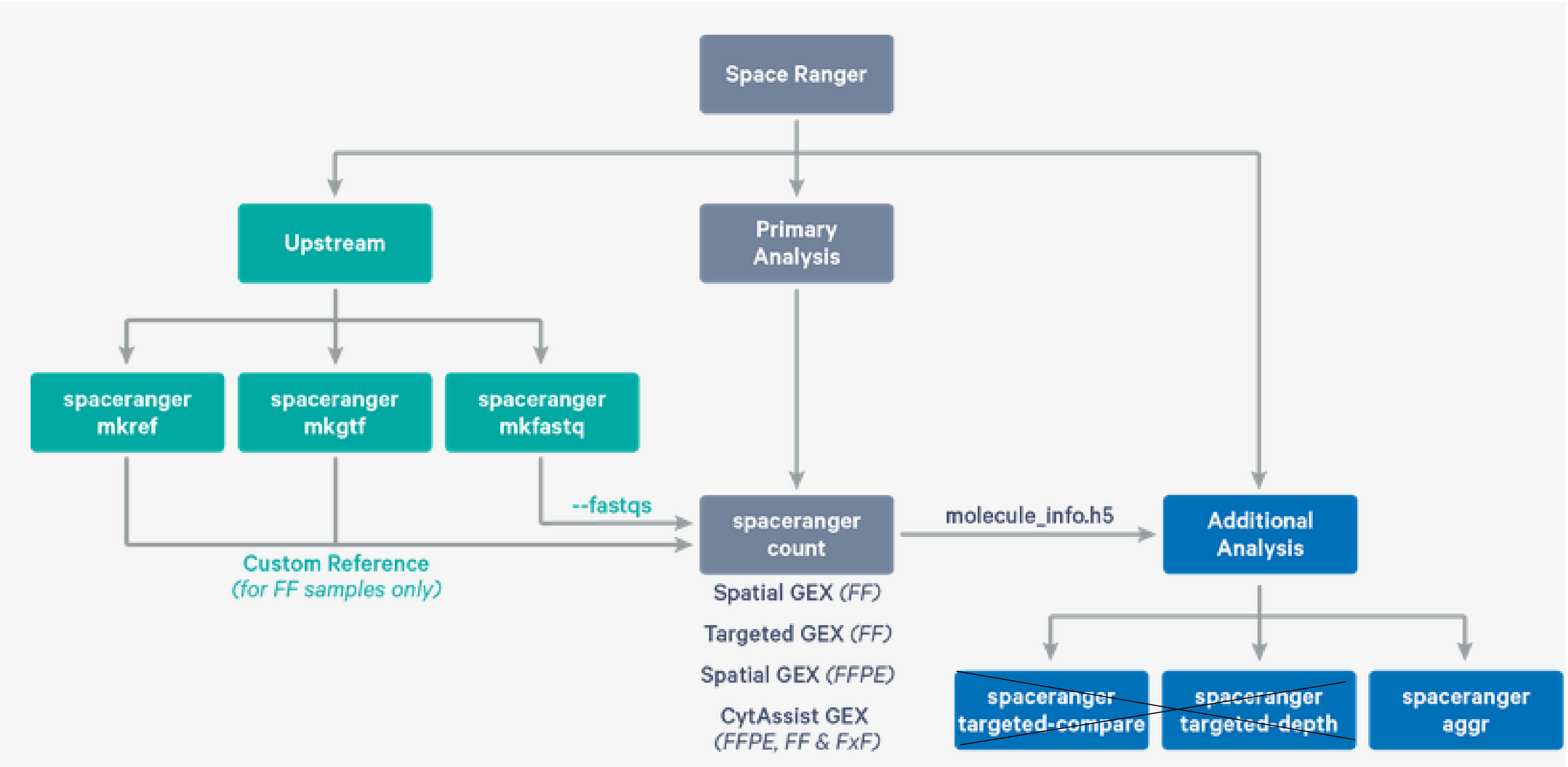
Dimension Clustering
SpatialPCA, Louvain, Leiden

Cell Type Deconvolution
Cell2Loation, RCTD cellTrek

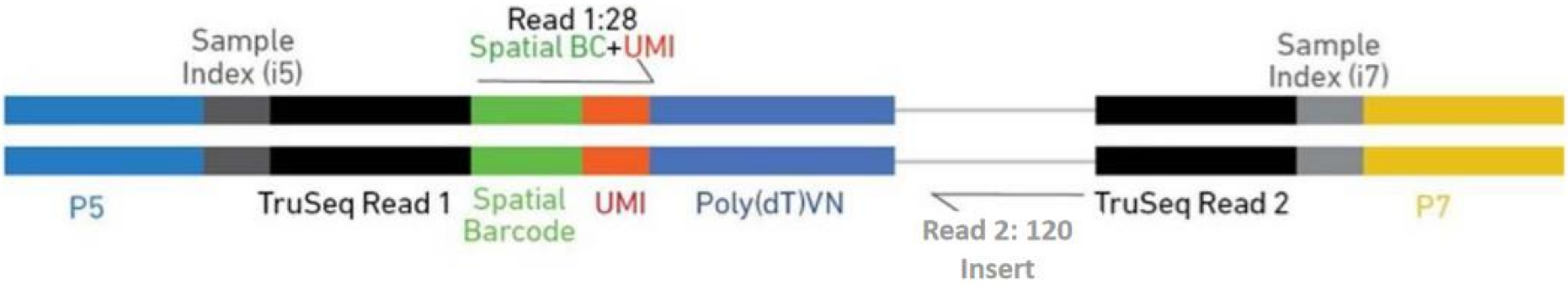
Cell-cell Communication
CellPhoneDB, CellChat etc

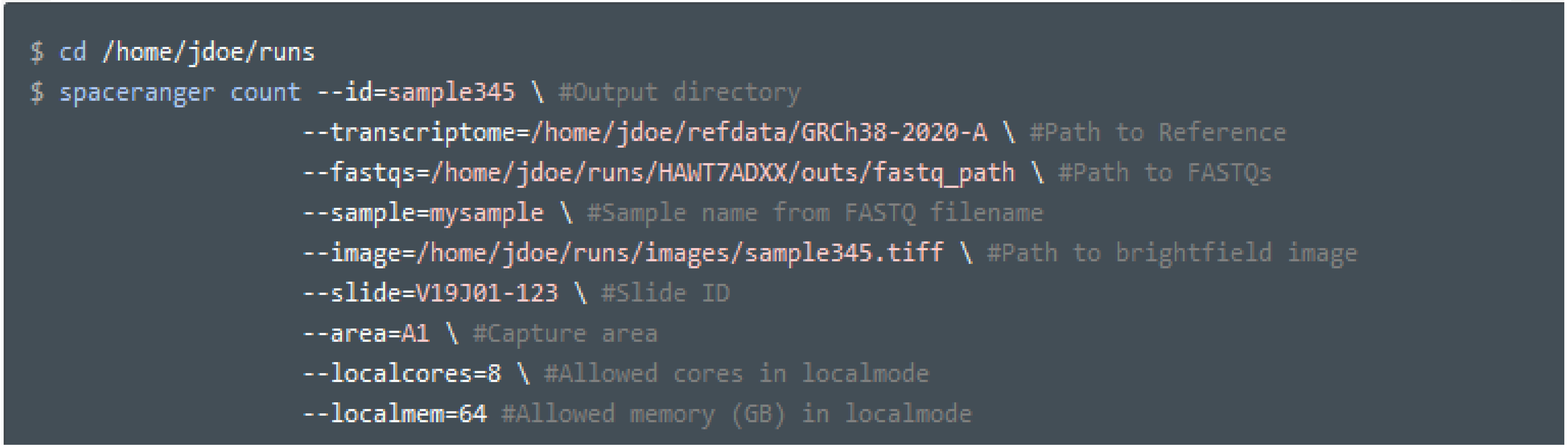
Spatially Variable Genes and Regions
SpatialDE, SPARK, Meringne,

Primary Analysis with 10x Space Ranger



10x Visium Data and Space Ranger Count

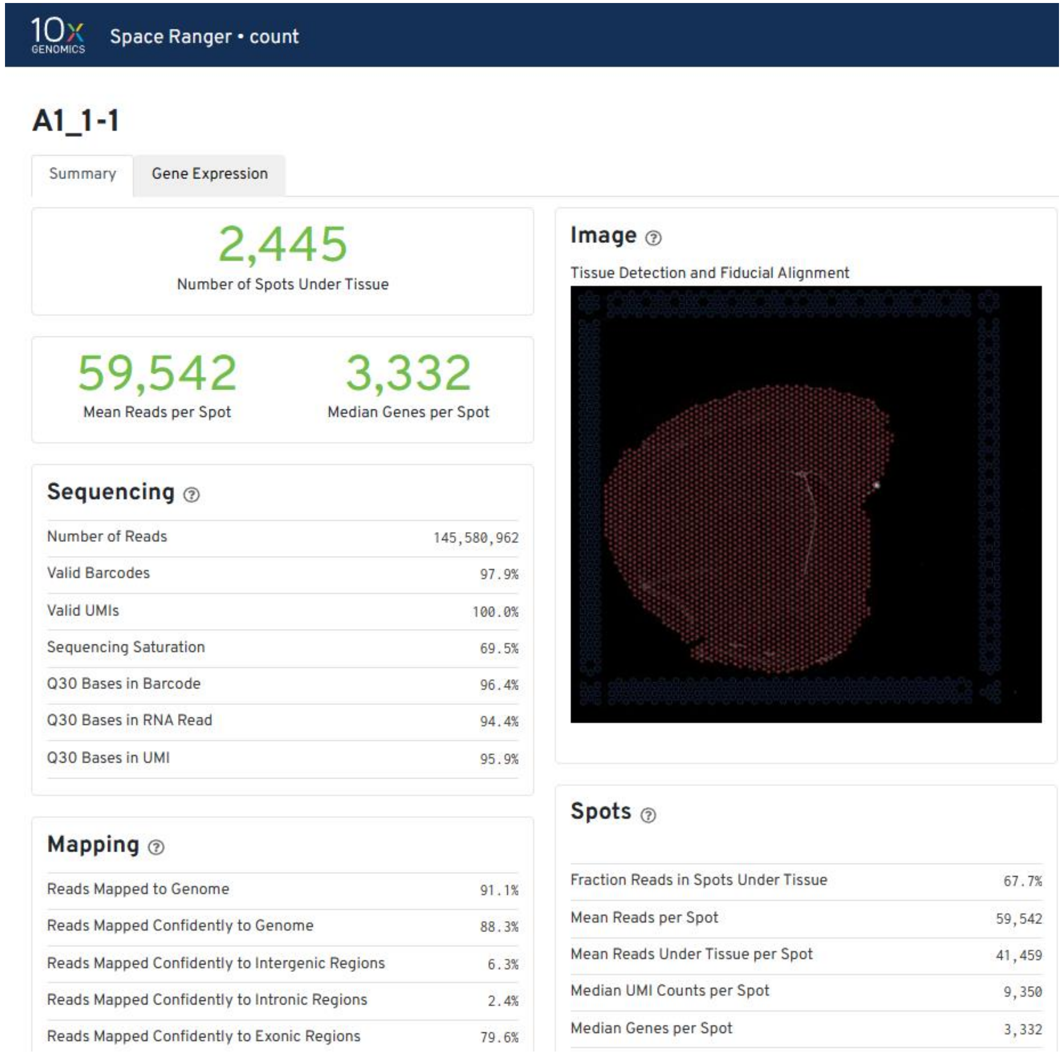




```
$ cd /home/jdoe/runs
$ spaceranger count --id=sample345 \ #Output directory
                   --transcriptome=/home/jdoe/refdata/GRCh38-2020-A \ #Path to Reference
                   --fastqs=/home/jdoe/runs/HAWT7ADXX/outs/fastq_path \ #Path to FASTQs
                   --sample=mysample \ #Sample name from FASTQ filename
                   --image=/home/jdoe/runs/images/sample345.tiff \ #Path to brightfield image
                   --slide=V19J01-123 \ #Slide ID
                   --area=A1 \ #Capture area
                   --localcores=8 \ #Allowed cores in localmode
                   --localmem=64 #Allowed memory (GB) in localmode
```

You can use SpaceRanger mkref command to build custom reference genome index

Space Ranger Output: Web Summary



10x GENOMICS Space Ranger • count

A1_1-1

Summary | Gene Expression

2,445
Number of Spots Under Tissue

59,542
Mean Reads per Spot

3,332
Median Genes per Spot

Sequencing ?

Number of Reads	145,580,962
Valid Barcodes	97.9%
Valid UMIs	100.0%
Sequencing Saturation	69.5%
Q30 Bases in Barcode	96.4%
Q30 Bases in RNA Read	94.4%
Q30 Bases in UMI	95.9%

Mapping ?

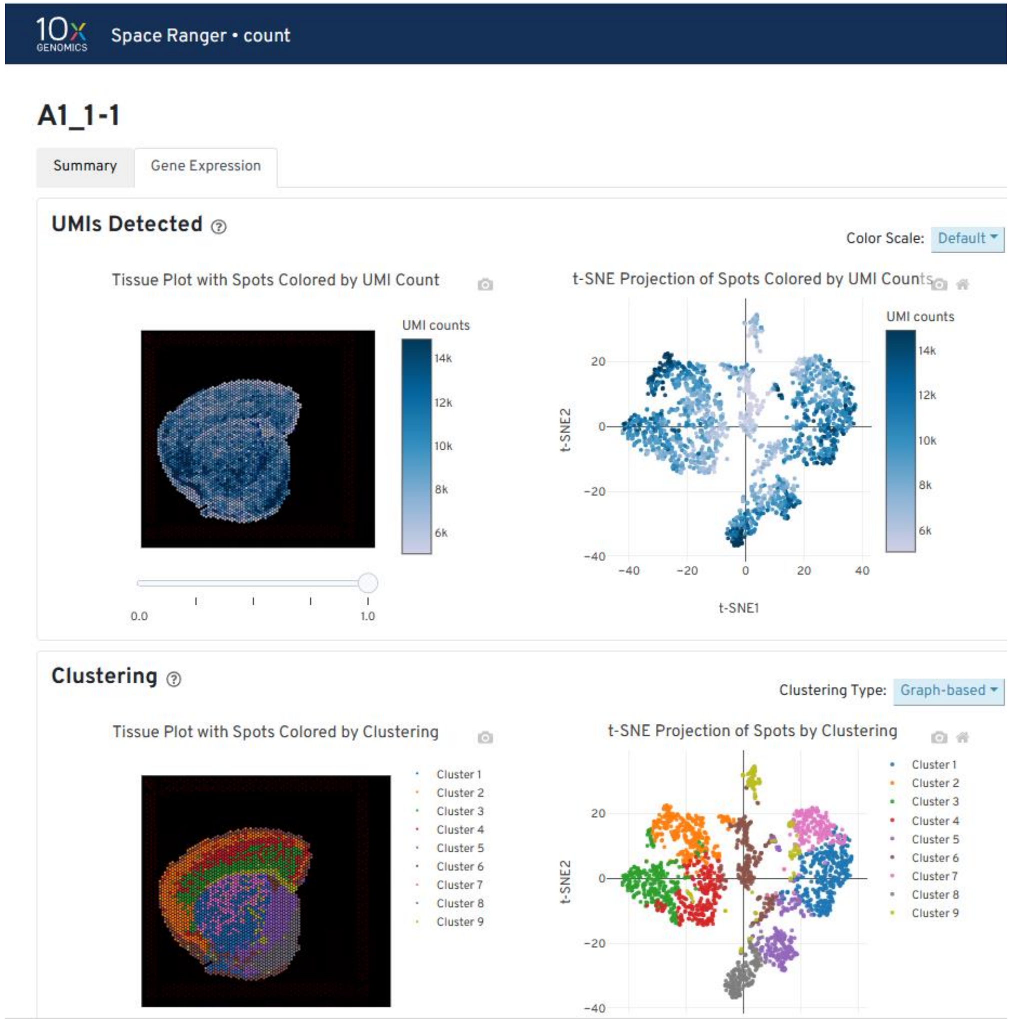
Reads Mapped to Genome	91.1%
Reads Mapped Confidently to Genome	88.3%
Reads Mapped Confidently to Intergenic Regions	6.3%
Reads Mapped Confidently to Intronic Regions	2.4%
Reads Mapped Confidently to Exonic Regions	79.6%

Image ?

Tissue Detection and Fiducial Alignment

Spots ?

Fraction Reads in Spots Under Tissue	67.7%
Mean Reads per Spot	59,542
Mean Reads Under Tissue per Spot	41,459
Median UMI Counts per Spot	9,350
Median Genes per Spot	3,332



10x GENOMICS Space Ranger • count

A1_1-1

Summary | Gene Expression

UMIs Detected ?

Color Scale: Default

Tissue Plot with Spots Colored by UMI Count

t-SNE Projection of Spots Colored by UMI Counts

Clustering ?

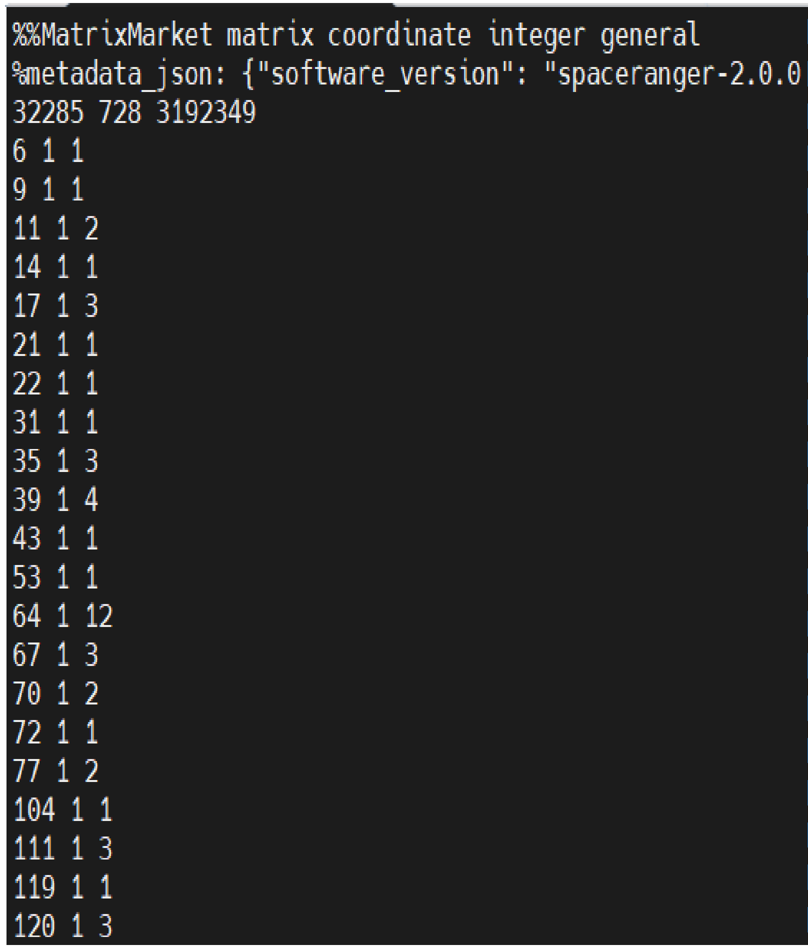
Clustering Type: Graph-based

Tissue Plot with Spots Colored by Clustering

t-SNE Projection of Spots by Clustering

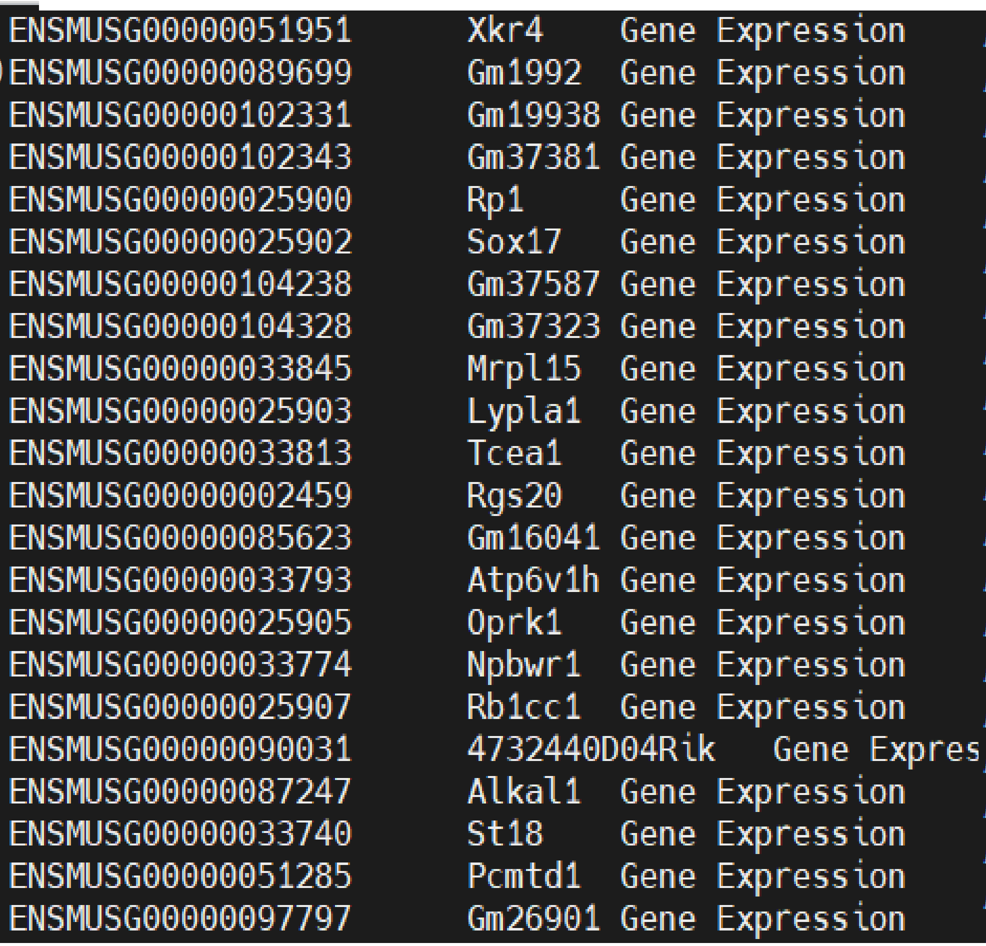
MatrixMarket Format (MEX) for Count Matrix

Sparse Feature Count Matrix (matrix.mtx)



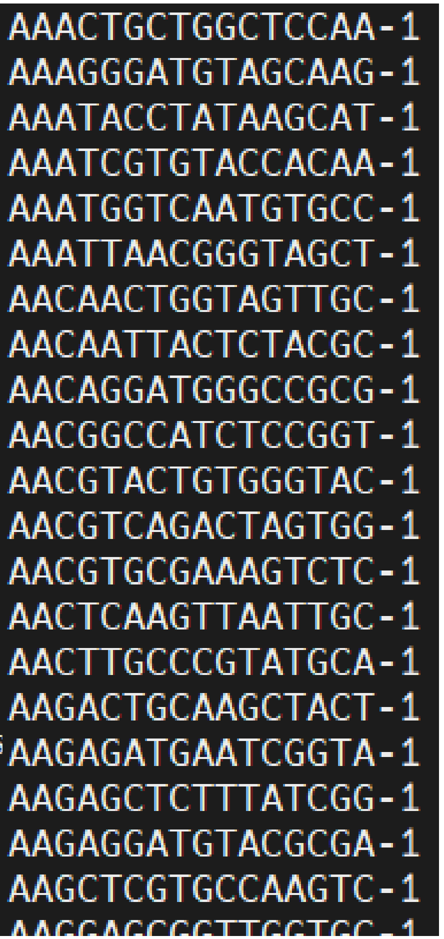
```
%%MatrixMarket matrix coordinate integer general
%metadata_json: {"software_version": "spaceranger-2.0.0
32285 728 3192349
6 1 1
9 1 1
11 1 2
14 1 1
17 1 3
21 1 1
22 1 1
31 1 1
35 1 3
39 1 4
43 1 1
53 1 1
64 1 12
67 1 3
70 1 2
72 1 1
77 1 2
104 1 1
111 1 3
119 1 1
120 1 3
```

Gene Info: features.tsv



```
ENSMUSG00000051951      Xkr4    Gene Expression
ENSMUSG00000089699      Gm1992  Gene Expression
ENSMUSG00000102331      Gm19938 Gene Expression
ENSMUSG00000102343      Gm37381 Gene Expression
ENSMUSG00000025900      Rp1     Gene Expression
ENSMUSG00000025902      Sox17   Gene Expression
ENSMUSG00000104238      Gm37587 Gene Expression
ENSMUSG00000104328      Gm37323 Gene Expression
ENSMUSG00000033845      Mrpl15  Gene Expression
ENSMUSG00000025903      Lypla1  Gene Expression
ENSMUSG00000033813      Tcea1   Gene Expression
ENSMUSG00000002459      Rgs20   Gene Expression
ENSMUSG00000085623      Gm16041 Gene Expression
ENSMUSG00000033793      Atp6v1h Gene Expression
ENSMUSG00000025905      Oprk1   Gene Expression
ENSMUSG00000033774      Npbwr1  Gene Expression
ENSMUSG00000025907      Rb1cc1  Gene Expression
ENSMUSG00000090031      4732440D04Rik   Gene Expres
ENSMUSG00000087247      Alkal1  Gene Expression
ENSMUSG00000033740      St18    Gene Expression
ENSMUSG00000051285      Pcmtd1  Gene Expression
ENSMUSG00000097797      Gm26901 Gene Expression
```

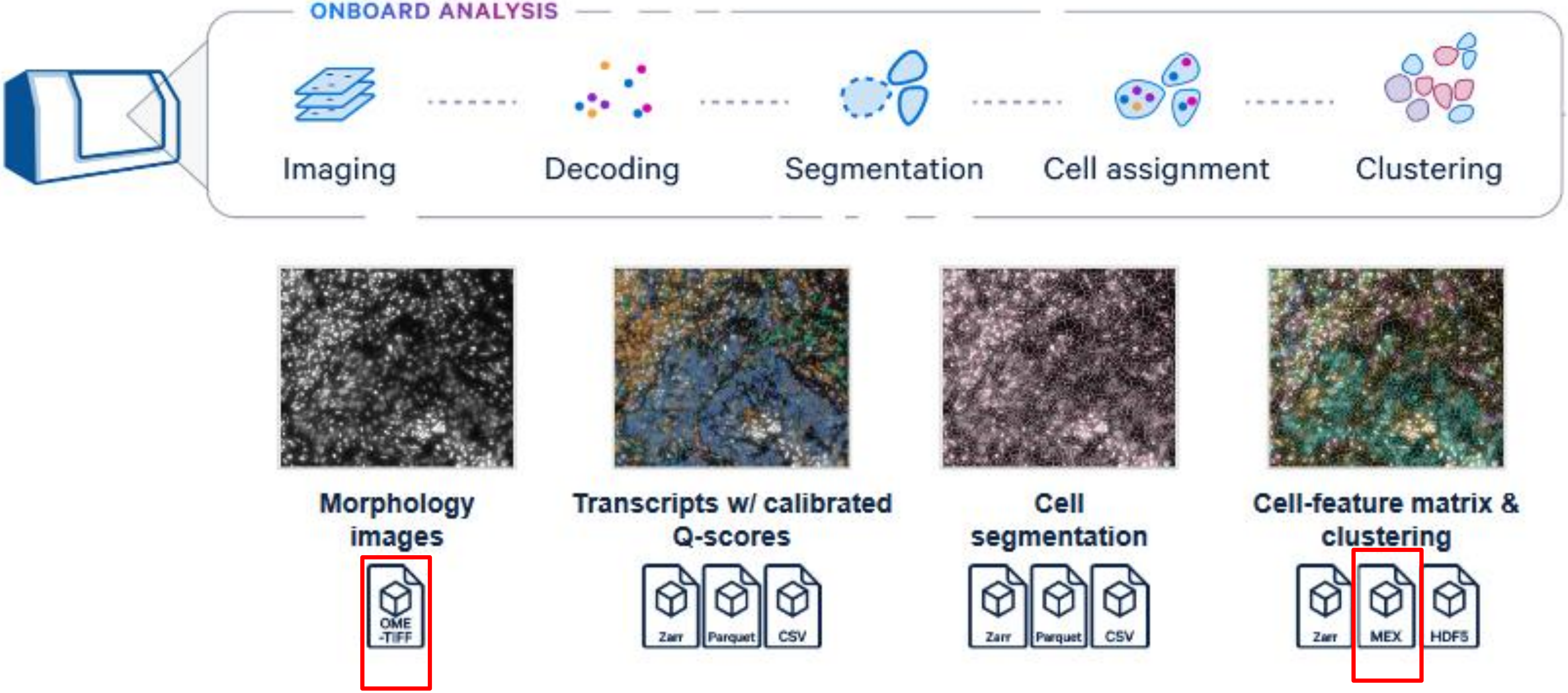
Spot Info (barcodes.tsv)



```
AAACTGCTGGCTCCAA-1
AAAGGGATGTAGCAAG-1
AAATACCTATAAGCAT-1
AAATCGTGTACCACAA-1
AAATGGTCAATGTGCC-1
AAATTAACGGGTAGCT-1
AACAACTGGTAGTTGC-1
AACAATTACTCTACGC-1
AACAGGATGGGCCGCG-1
AACGGCCATCTCCGGT-1
AACGTACTGTGGGTAC-1
AACGTCAGACTAGTGG-1
AACGTGCGAAAGTCTC-1
AACTCAAGTTAATTGC-1
AACTTGCCCGTATGCA-1
AAGACTGCAAGCTACT-1
AAGAGATGAATCGGTA-1
AAGAGCTCTTTATCGG-1
AAGAGGATGTACGCGA-1
AAGCTCGTGCCAAGTC-1
AAGGAGCGGTTGGTGC-1
```

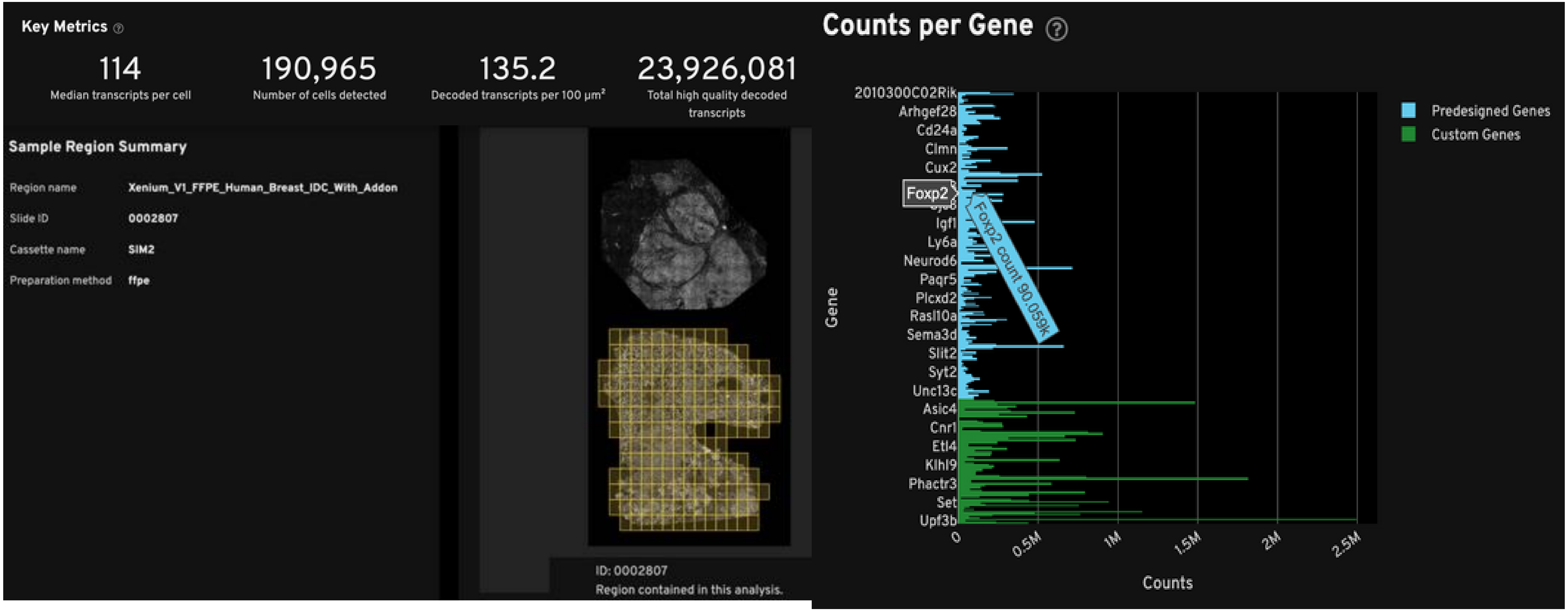
These are the three files in the Space Ranger omics output folder that can be input for downstream Seurat analysis

Xenium Onboard Analysis Output Formats



https://www.10xgenomics.com/support/software/xenium-onboard-analysis/latest/analysis/

Xenium Web Summary



Key Metrics

114	190,965	135.2	23,926,081
Median transcripts per cell	Number of cells detected	Decoded transcripts per 100 µm²	Total high quality decoded transcripts

Sample Region Summary

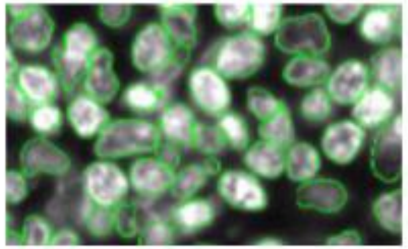
Region name	Xenium_V1_FFPE_Human_Breast_IDC_With_Addon
Slide ID	0002807
Cassette name	SIM2
Preparation method	ffpe

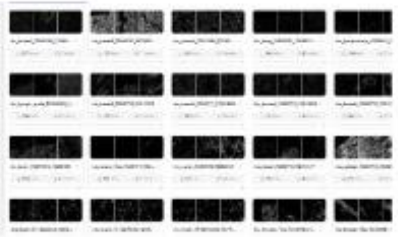
ID: 0002807
Region contained in this analysis.

The format of the Xenium analysis summary file may change frequently as new features and improvements are added.

Imaging Based ST: Cell Segmentation

- Many of the downstream analyses and interpretations of the spatially resolved data depend on the ability to resolve individual cells
  - *Cellpose*, *Baysor*
  - *Stardist*, *deepcell*
  - Spot-based Spatial cell-type Analysis by Multidimensional mRNA density estimation (*SSAM*)
  - *DL* can be computational intensive

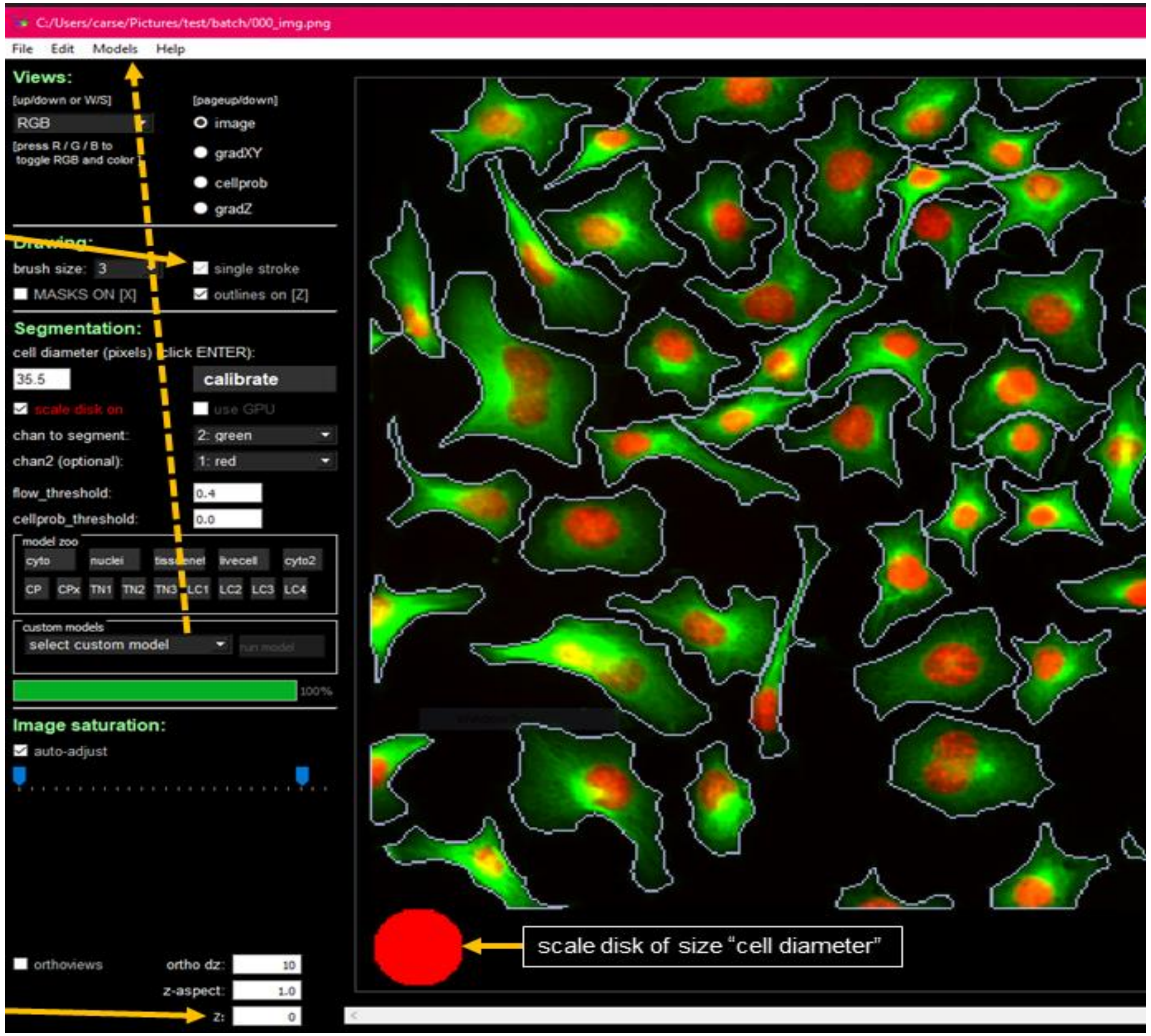




Optimizing Xenium Segmentation

Cellpose + Baysor

- Xenium's current segmentation: Cookie cutter style, 2D, nuclei segmentation based on DAPI, followed by an expansion of the segmentation masks with 15 micron
- Nuclei segmentation using Cellpose and the assignment of reads to individual cells using Baysor



C:/Users/carse/Pictures/test/batch/000_img.png

Views:

[up/down or W/S] RGB

[press R / G / B to toggle RGB and color]

[pageup/down] ○ image ● gradXY ● cellprob ● gradZ

Drawing:

brush size: 3

☑ single stroke

□ MASKS ON [X]

☑ outlines on [Z]

Segmentation:

cell diameter (pixels) (click ENTER): 35.5

☑ scale disk on

□ use GPU

chan to segment: 2: green

chan2 (optional): 1: red

flow_threshold: 0.4

cellprob_threshold: 0.0

100%

Image saturation:

☑ auto-adjust

□ orthoviews

ortho dz: 10

z-aspect: 1.0

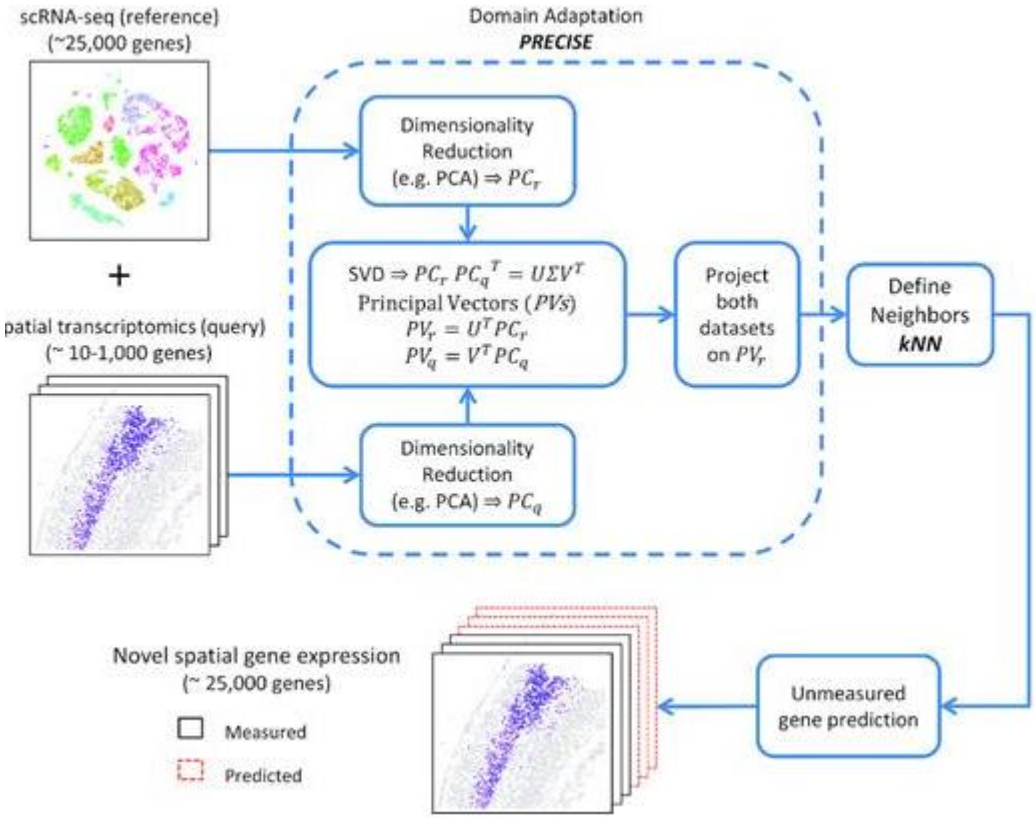
z: 0

scale disk of size "cell diameter"

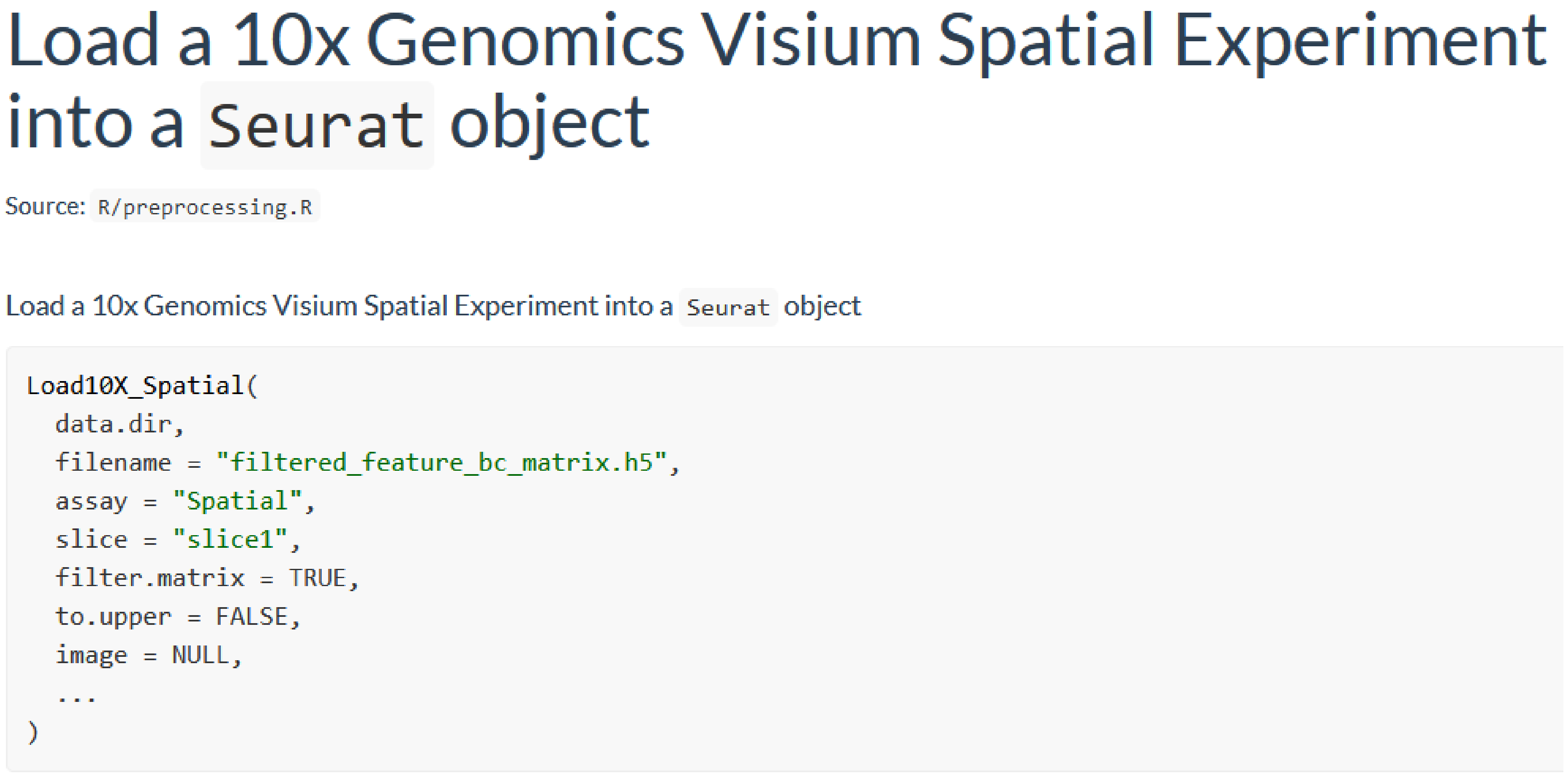
Xenium: Gene Imputation

- Predicting gene expression from a reference scRNA-seq onto the cellular-resolution ST dataset

- *gimVI, <u>SpaGE,</u> Tangram, SpaOTsc; RCTD, Seurat integration, Liger …*


scRNA-seq (reference)
(~25,000 genes)
+
patial transcriptomics (query)
(~ 10-1,000 genes)
Domain Adaptation
PRECISE
Dimensionality Reduction (e.g. PCA) ⇒ PC_r
SVD ⇒ $PC_r\ PC_q{}^T = U\Sigma V^T$
Principal Vectors (PVs)
$PV_r = U^T PC_r$
$PV_q = V^T PC_q$
Dimensionality Reduction (e.g. PCA) ⇒ PC_q
Project both datasets on PV_r
Define Neighbors kNN
Unmeasured gene prediction
Novel spatial gene expression
(~ 25,000 genes)
□ Measured
□ Predicted

Seurat: Data Import: Visium



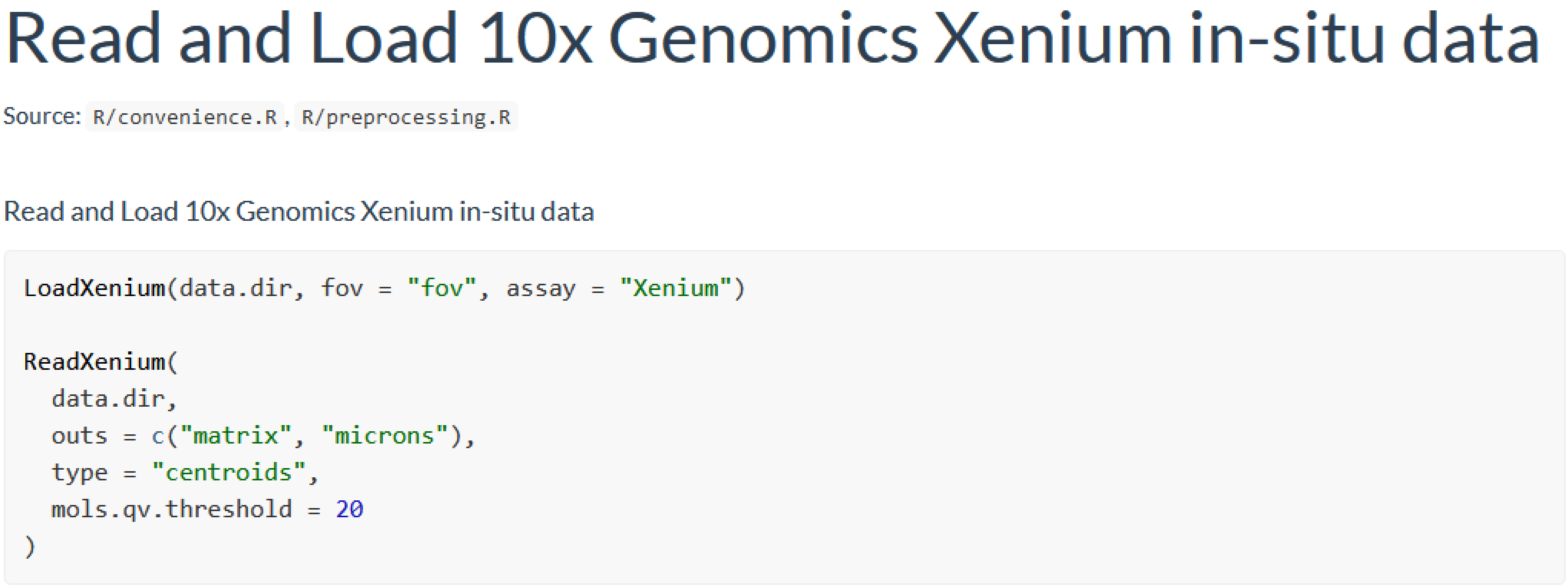
Load a 10x Genomics Visium Spatial Experiment into a `Seurat` object

Source: `R/preprocessing.R`

Load a 10x Genomics Visium Spatial Experiment into a `Seurat` object

```
Load10X_Spatial(
  data.dir,
  filename = "filtered_feature_bc_matrix.h5",
  assay = "Spatial",
  slice = "slice1",
  filter.matrix = TRUE,
  to.upper = FALSE,
  image = NULL,
  ...
)
```

Seurat: Data Import: Xenium



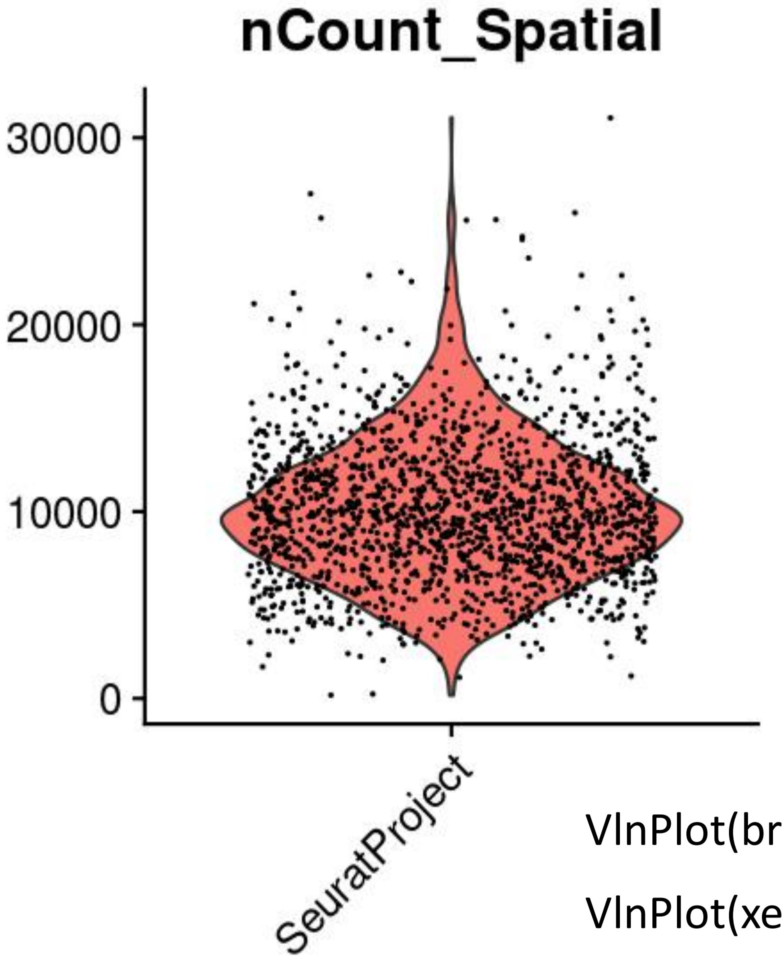
Read and Load 10x Genomics Xenium in-situ data

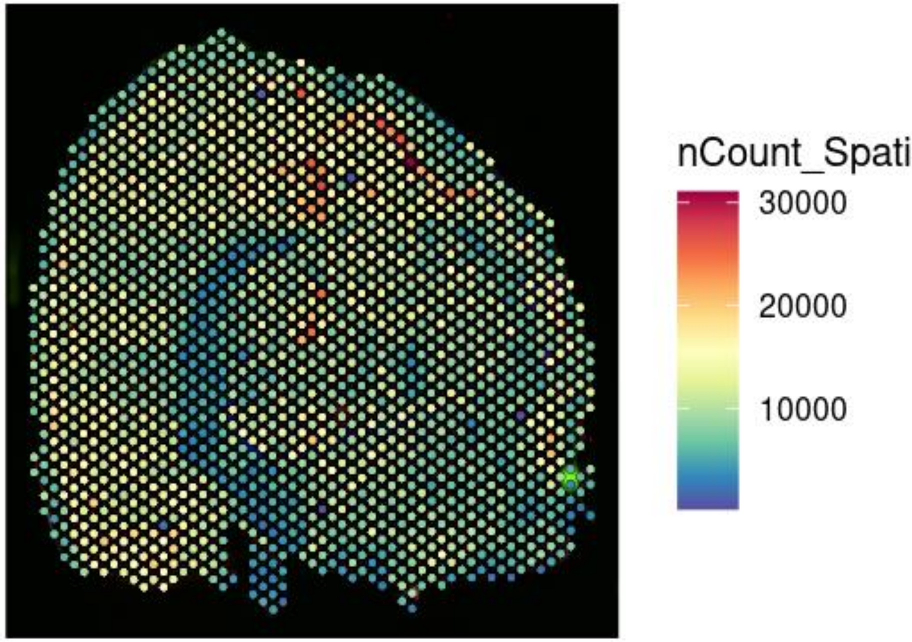
Source: `R/convenience.R`, `R/preprocessing.R`

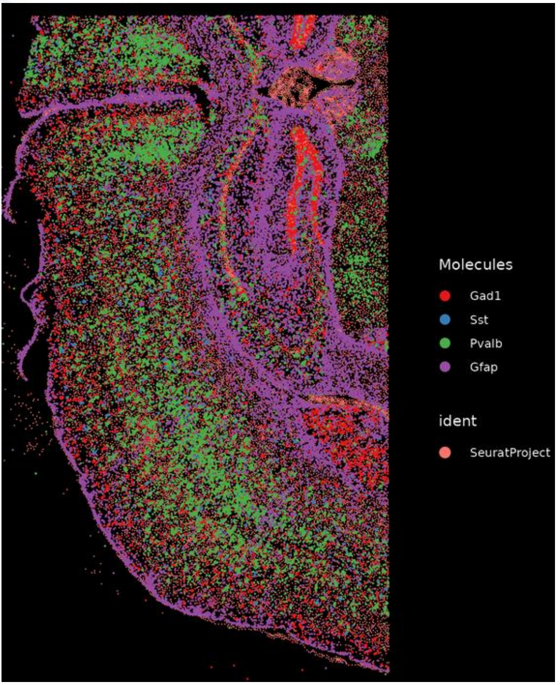
Read and Load 10x Genomics Xenium in-situ data

```r
LoadXenium(data.dir, fov = "fov", assay = "Xenium")

ReadXenium(
  data.dir,
  outs = c("matrix", "microns"),
  type = "centroids",
  mols.qv.threshold = 20
)
```

Seurat: QC and Filtering

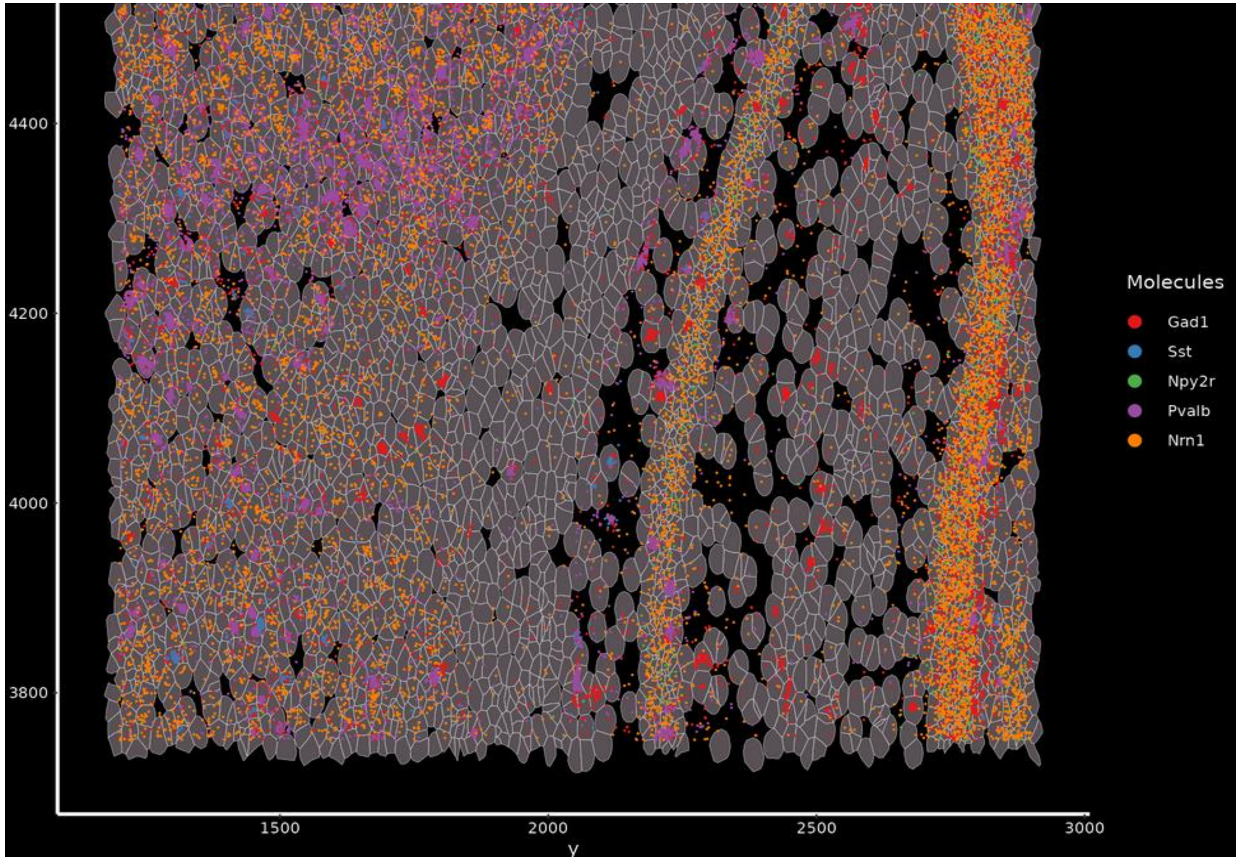






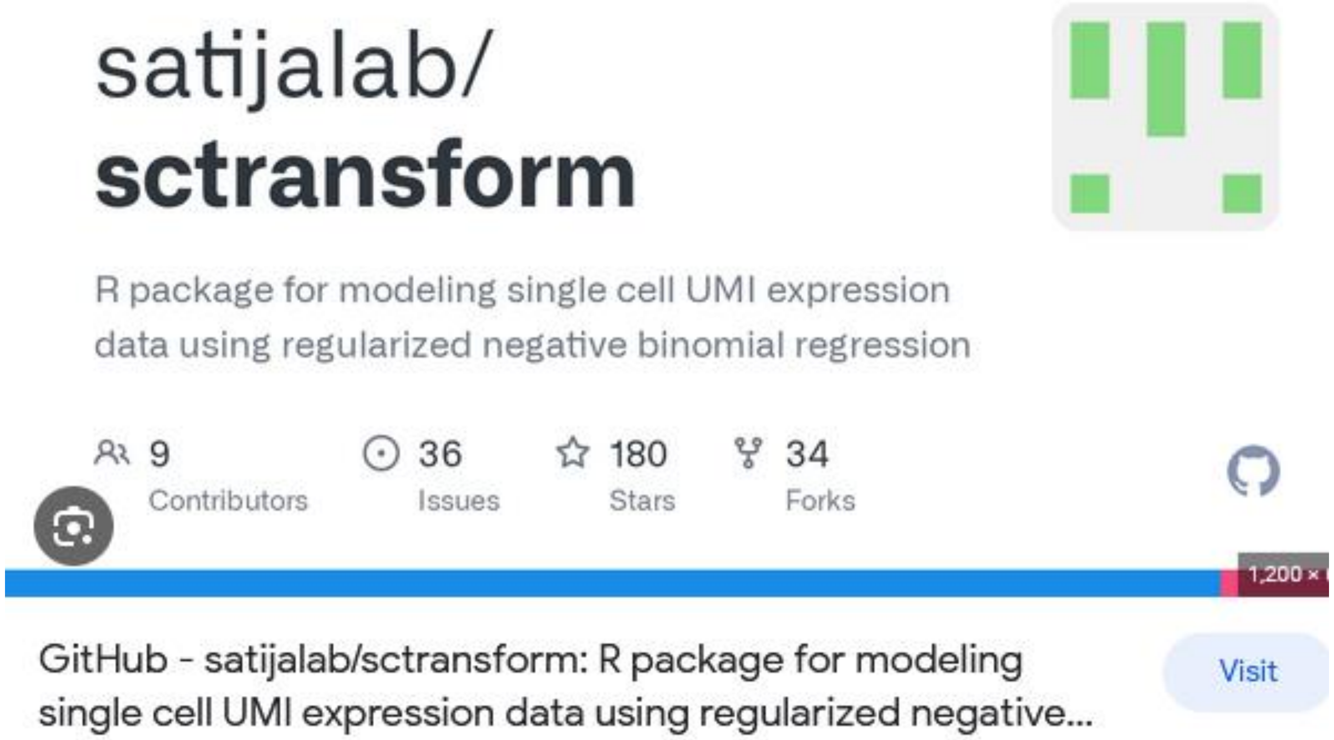
```
VlnPlot(brain, features = "nCount_Spatial", pt.size = 0.1) + NoLegend()

VlnPlot(xenium.obj, features = c("nFeature_Xenium", "nCount_Xenium"), ncol = 2)

SpatialFeaturePlot(brain, features = "nCount_Spatial") + theme(legend.position = "right")

ImageDimPlot(xenium.obj, fov = "fov", molecules = c("Gad1", "Sst", "Pvalb", "Gfap"), nmols = 20000)
```

Crop Xenium Images



```
cropped.coords <- Crop(xenium.obj[["fov"]], x = c(1200,
2900), y = c(3750, 4550), coords = "plot")

xenium.obj[["zoom"]] <- cropped.coords

# visualize cropped area with cell segmentations &
selected molecules

DefaultBoundary(xenium.obj[["zoom"]]) <- "segmentation"

ImageDimPlot(xenium.obj, fov = "zoom", axes = TRUE,
border.color = "white", border.size = 0.1, cols =
"polychrome", coord.fixed = FALSE, molecules = c("Gad1",
"Sst", "Npy2r", "Pvalb", "Nrn1"), nmols = 10000)
```

Normalization, Feature Selection: SCTransform



satijalab/
sctransform

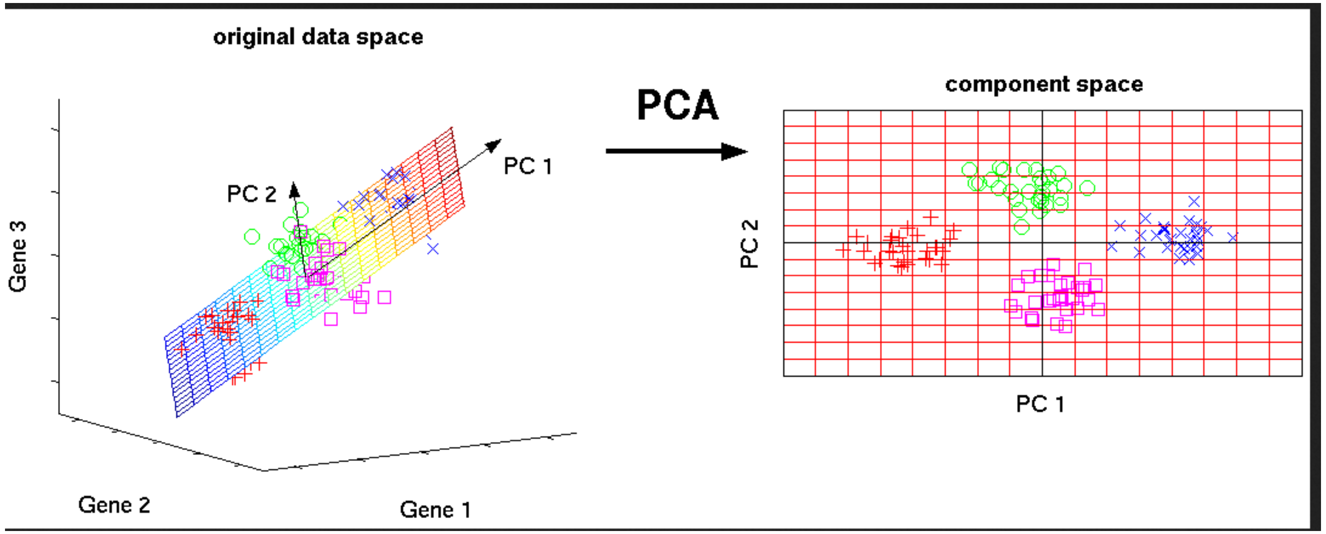
R package for modeling single cell UMI expression data using regularized negative binomial regression

GitHub - satijalab/sctransform: R package for modeling single cell UMI expression data using regularized negative...

```
brain <- SCTransform(brain, assay = "Spatial", verbose = FALSE")
```

Linear Dimension Reduction

- Principle Component Analysis (PCA) is a standard technique for visualizing high dimensional data and for data pre-processing.



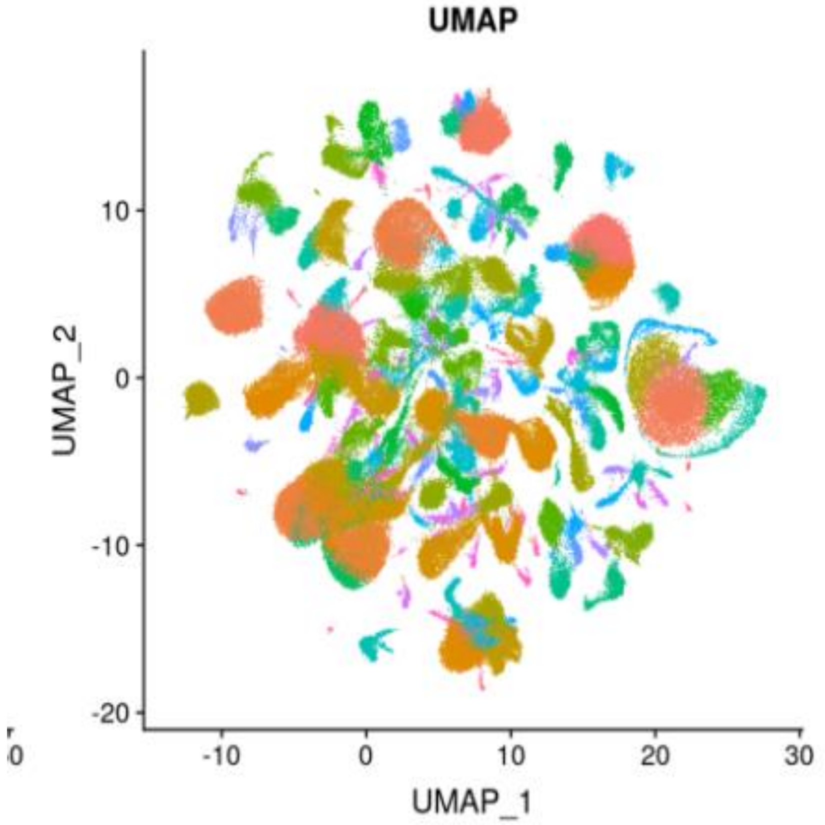
Matthias Scholz, 2015

```
brain <- RunPCA(brain, assay = "SCT", verbose = FALSE)
```

HVGs from SCTransform go into PCA

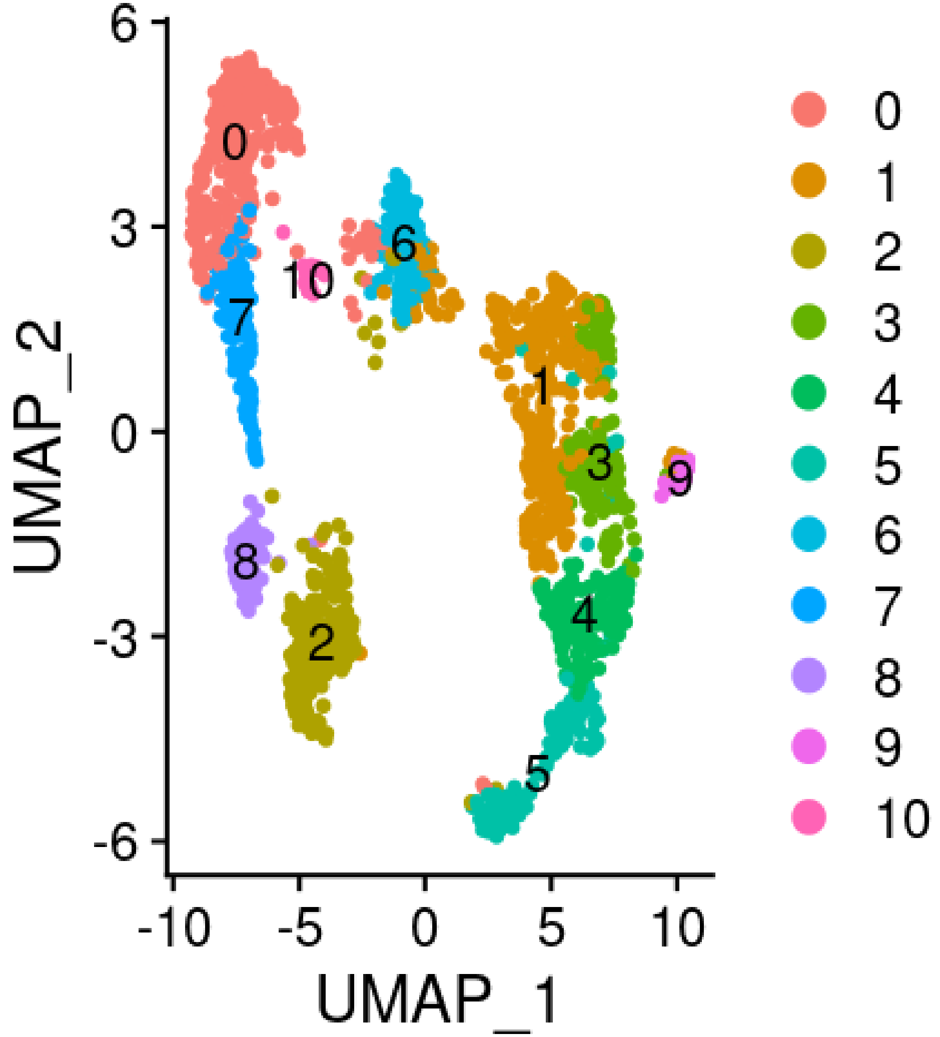
Clustering

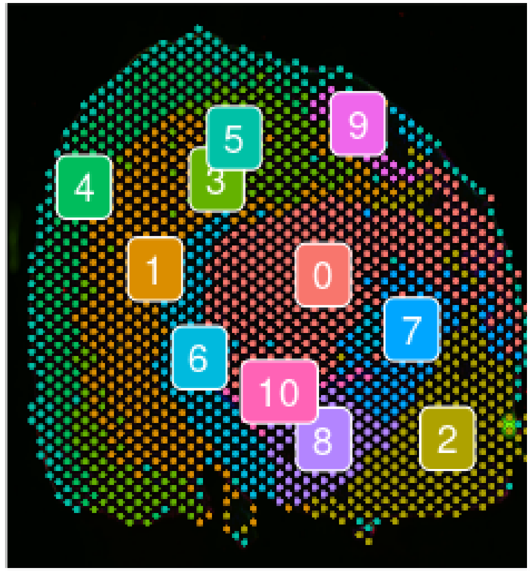
- Distance/similarity metrics
- Algorithm choices
    - K means
    - Hierarchical
    - Density based
    - Graph partition
    - …

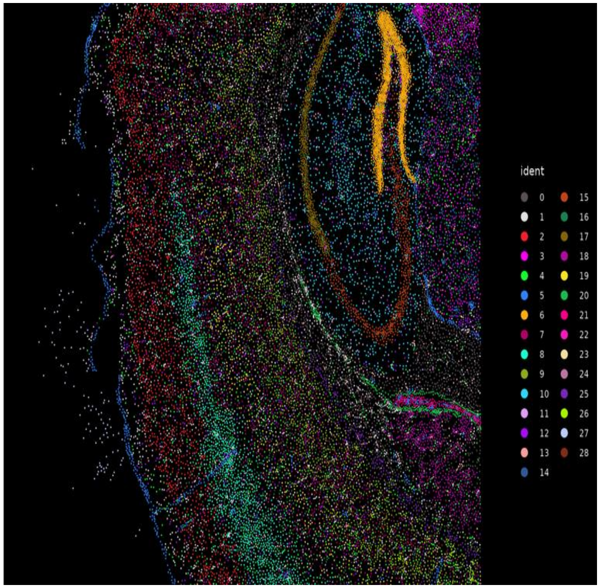


```
brain <- FindNeighbors(brain, reduction = "pca", dims = 1:30)
brain <- FindClusters(brain, verbose = FALSE)
```

Non Linear Dimension Reduction : UMAP







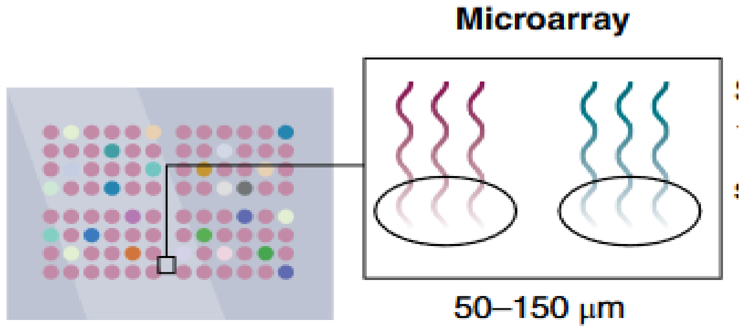
```
brain <- RunUMAP(brain, reduction = "pca", dims = 1:30)
```

UMAP is not clustering!

Sequencing Based ST Data Analysis: Visium

- Cell Type Deconvolution
  - *RCTD*
  - *SpatialDWLS (Giotto)*
  - *BayesSpace*

- Alignment of scRNA data with SST Data
  - *CytoSpace*
  - *Tangram*
  - *Cell2location*

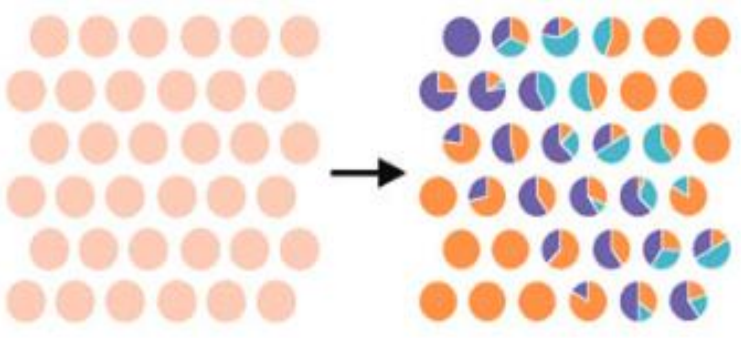


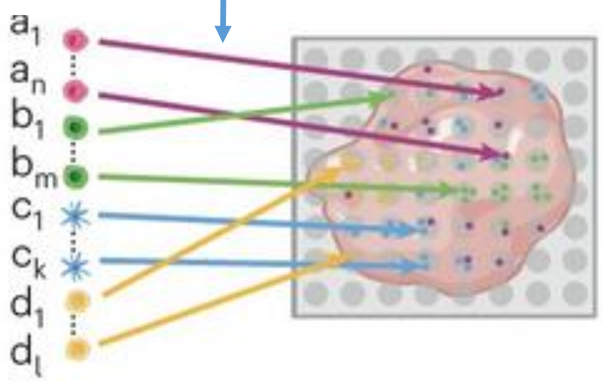
Gene expression matrix

	Gene 1	Gene 2	Gene *N*
Spot 1	10	5	6
Spot 2	5	21	20
...	...	...	...
Spot *N*	8	1	2

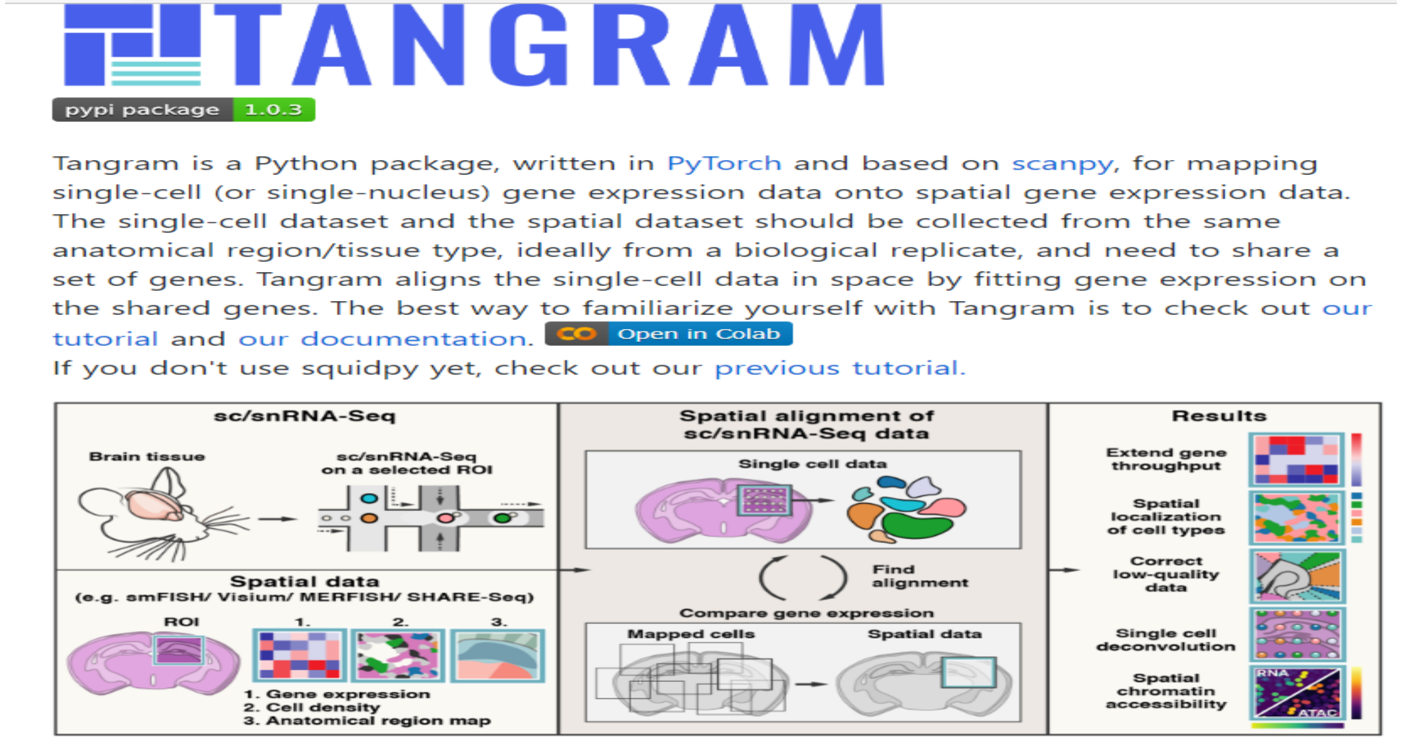
Spatial information

	x	*y*
Spot 1	141.2	511.4
Spot 2	514.9	219.9
...	...	...
Spot *N*	8	127.4





Integrating scRNA Data with ST Data



TANGRAM

pypi package 1.0.3

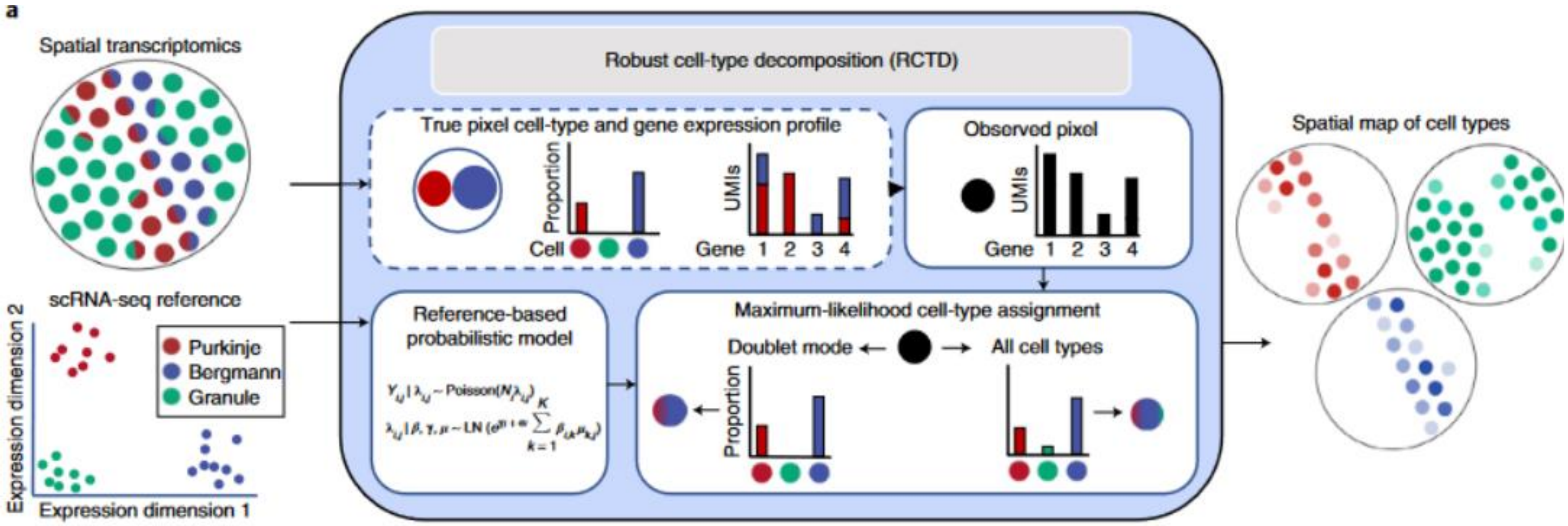
Tangram is a Python package, written in PyTorch and based on scanpy, for mapping single-cell (or single-nucleus) gene expression data onto spatial gene expression data. The single-cell dataset and the spatial dataset should be collected from the same anatomical region/tissue type, ideally from a biological replicate, and need to share a set of genes. Tangram aligns the single-cell data in space by fitting gene expression on the shared genes. The best way to familiarize yourself with Tangram is to check out our tutorial and our documentation. Open in Colab

If you don't use squidpy yet, check out our previous tutorial.

CytoSPACE

scvi-tools

RCTD : Robust Cell Type Decomposition

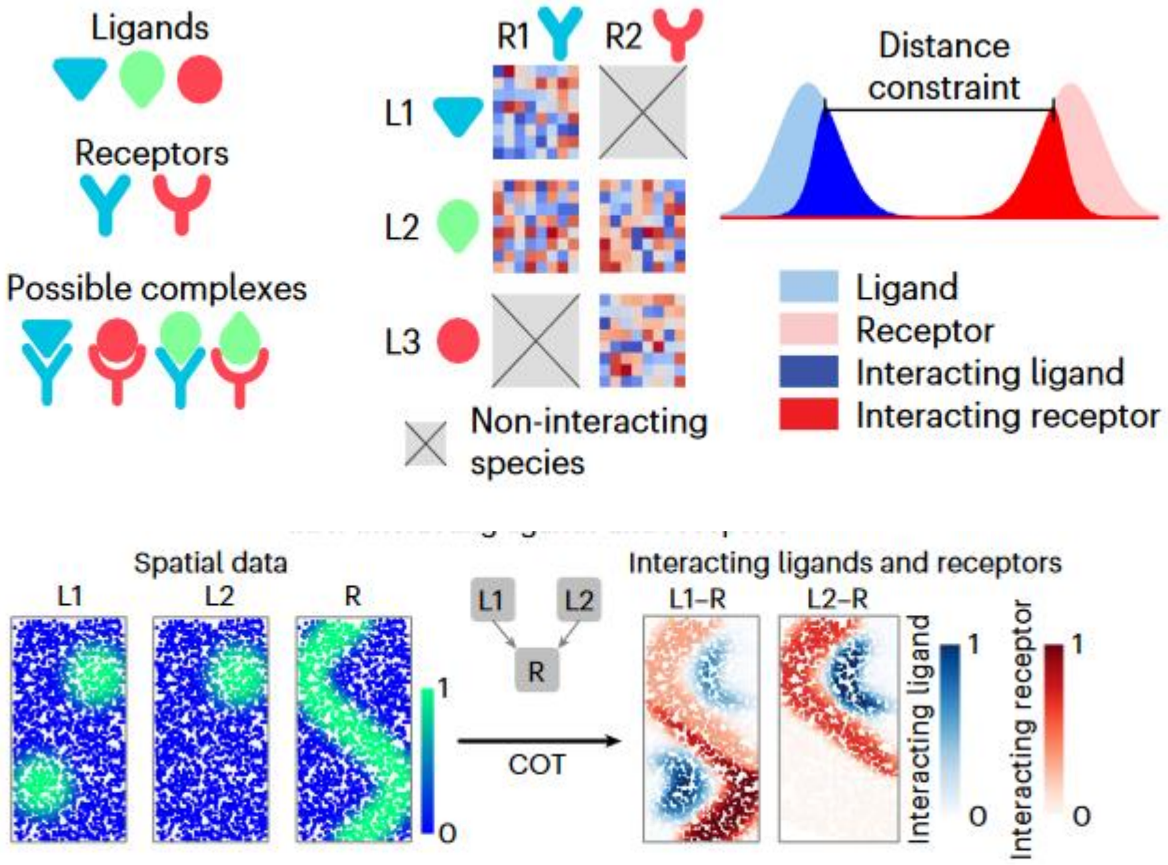

a
Spatial transcriptomics
scRNA-seq reference
Purkinje
Bergmann
Granule
Expression dimension 2
Expression dimension 1
Robust cell-type decomposition (RCTD)
True pixel cell-type and gene expression profile
Proportion
Cell
UMIs
Gene 1 2 3 4
Observed pixel
UMIs
Gene 1 2 3 4
Reference-based probabilistic model
$Y_{i,j} | \lambda_{i,j} \sim \text{Poisson}(N_i \lambda_{i,j})$
$\lambda_{i,j} | \beta, \gamma, \mu \sim \text{LN}(e^{\gamma_j + \varepsilon} \sum_{k=1}^{K} \beta_{i,k} \mu_{k,j})$
Maximum-likelihood cell-type assignment
Doublet mode
All cell types
Proportion
Spatial map of cell types

```
# run RCTD with many cores
RCTD <- create.RCTD(query, reference, max_cores = 8)
RCTD <- run.RCTD(RCTD, doublet_mode = "doublet")
```

Cable *et al,* 2022

Downstream Analysis: Cell-Cell Communication

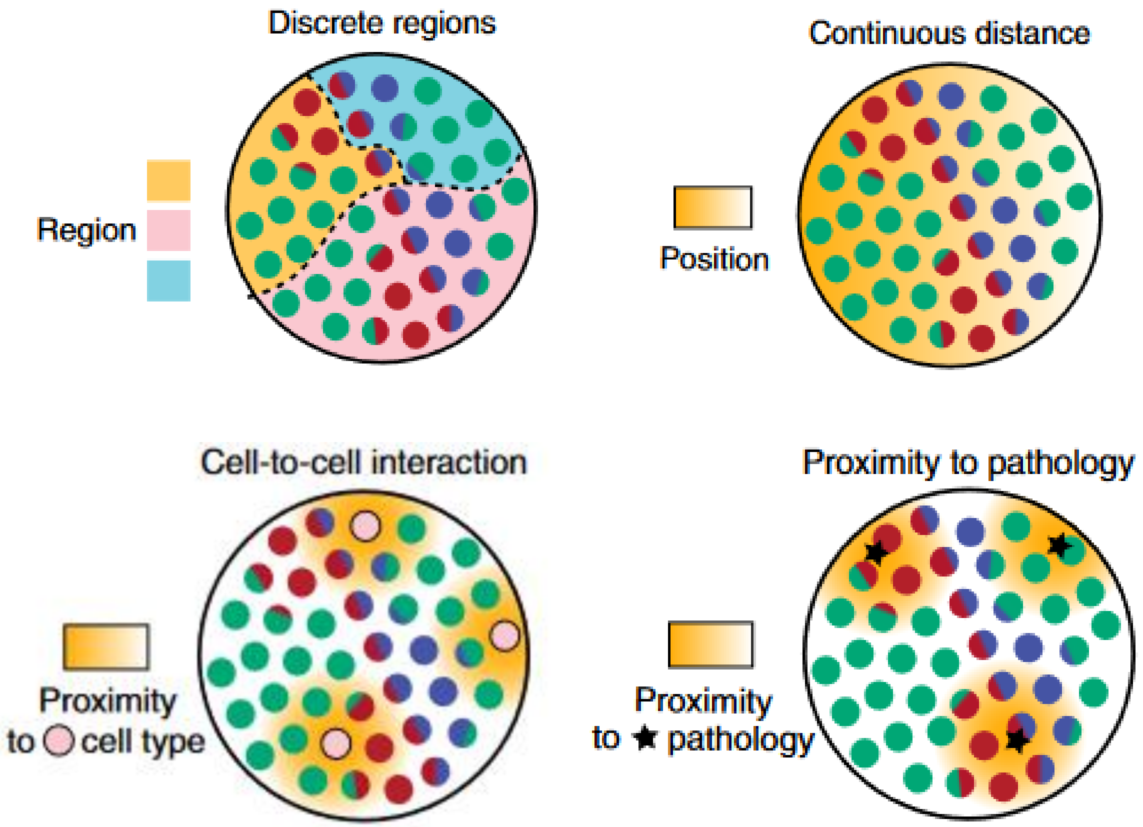
- Spatial CellChat (v2)
- COMMOT: collective optimal transport based
- SpaTalk : knowledge-graph-based cell-cell communication inference



Cang *et al*, 2023

Downstream Analysis: Spatial DE

- Cell type specific differential gene expression (*C-SIDE*)
  - Takes into account cell types
  - Accounting for localization of other cell types
  - Can incorporate covariates and replicates
  - Can detect DE due to pathology, anatomical regions, cell-cell interactions etc



Implemented in spacexr package

Cable *et al*, 2022

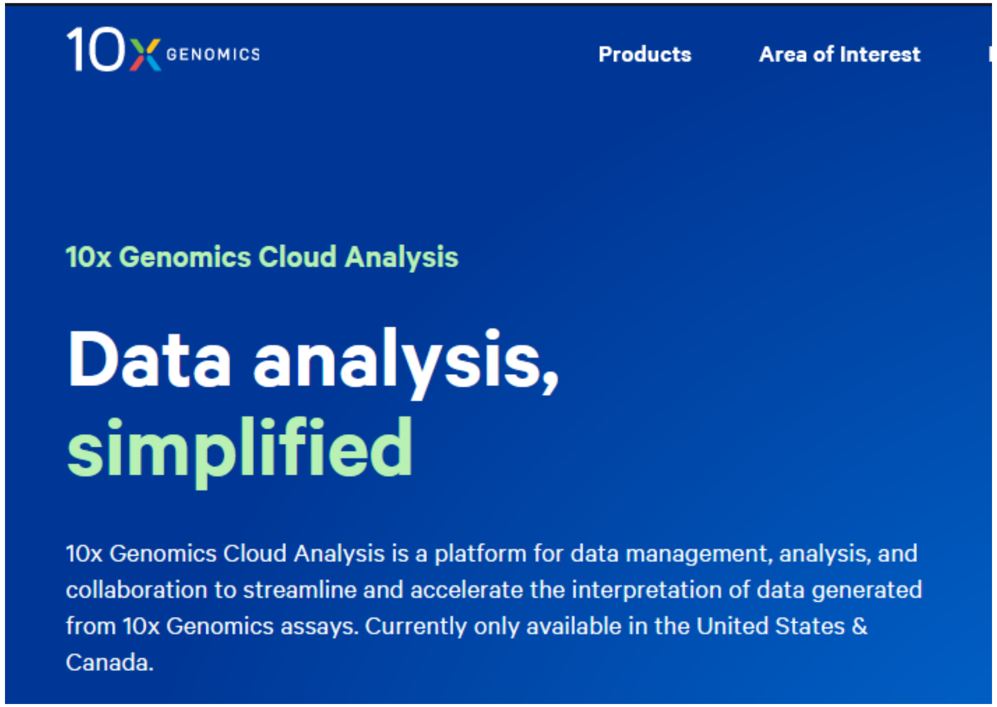
Challenges in ST Data Analysis

- Wide range of protocols and data processing pipelines
- A larger variety of file formats and data structures due to heterogeneity of methodologies
- No standardized tissue and quality control measurements or benchmarks
- Increased data volume, run time and memory usage demands more hardware and hands on time
- Computation tools are evolving rapidly

Data Analysis Choices

- Tools choices: proprietary pipeline or open source?

- Local desktop, HPC or cloud computing? Cost, flexibility, capacity and ease of use.



10x Genomics Cloud Analysis

Data analysis, simplified

10x Genomics Cloud Analysis is a platform for data management, analysis, and collaboration to streamline and accelerate the interpretation of data generated from 10x Genomics assays. Currently only available in the United States & Canada.

Hands on Session

- Either Rstudio or Jupyter notebook environment
- Path to jupyter notebook for commands and expected output /dfs6/pub/ucightf/workshop/Seurat4_GRTHVisiumworkshop_Sept23.ipynb
- Use cp command to copy the notebook to your own directory
- Feel free to use your own data. Just point to the correct input directory