Introduction to Sequencing-Based Spatial Transcriptomic Data Analysis

Jenny Wu

Director of Bioinformatics

Genomics Research and Technology Hub

Chao Family Comprehensive Cancer Center

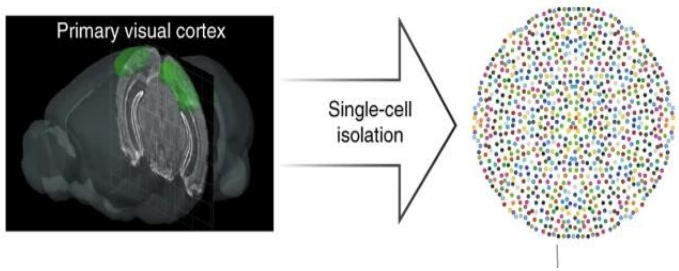
UC Irvine

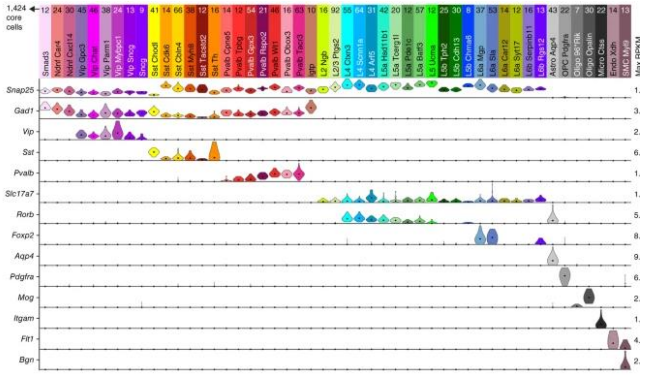
Outline

- 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 of 10x Visium data with Space Ranger
- Downstream analysis pipeline for Visium - Seurat Workflow
  - ✓ Data import
  - ✓ QC, filtering and feature selection
  - ✓ Dimension reduction and clustering
  - ✓ Data visualization and integration
- Advanced topics: cell type deconvolution, integration with scRNA data, and inter cellular communication

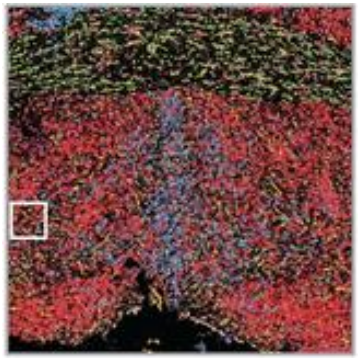
Why Spatial Biology

- Single cell sequencing reveals cellular heterogeneity





- Spatial information is important

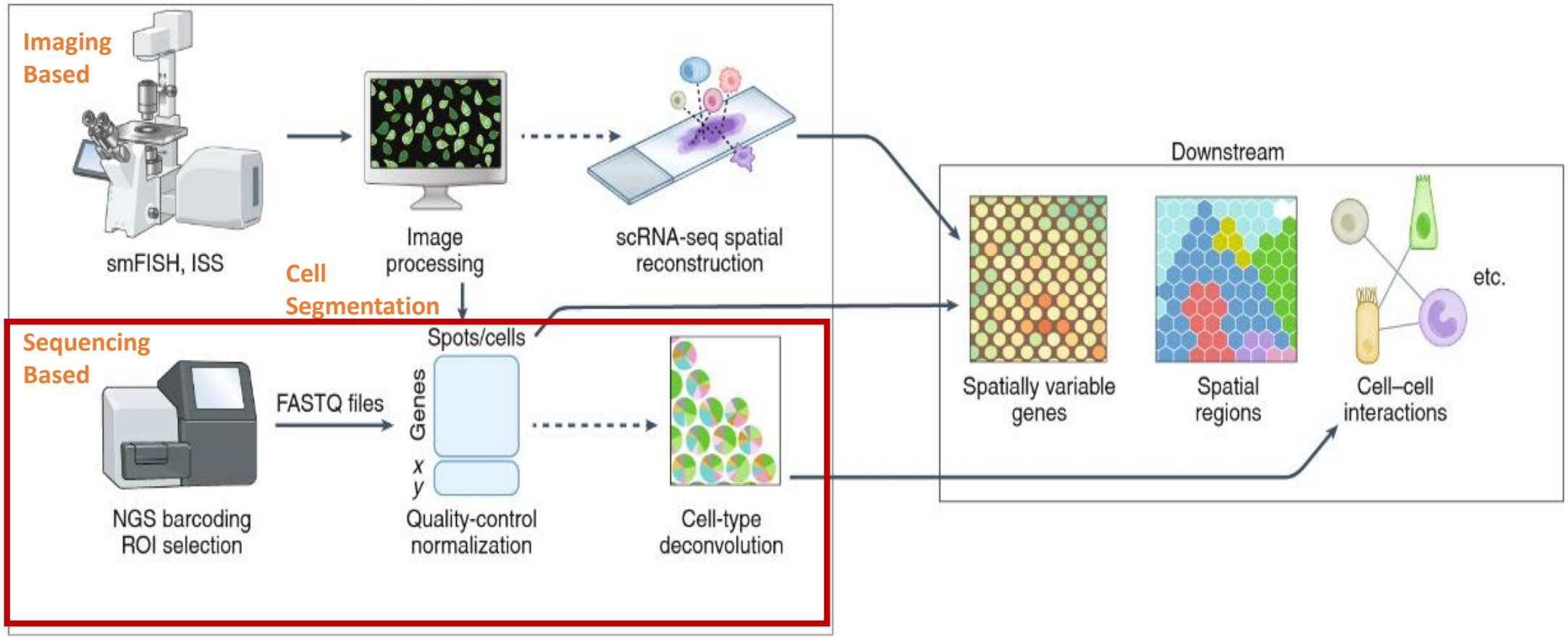


Spatial technologies tested

Technology	RNA / Protein	Readout	Input tissue	Tissue area to evaluate	Panel size	Resolution	QC Metrics
CODEX	Protein (Antibody-based)	Imaging	FFPE sections (5µm onto coated CODEX coverslips)	up to 1.5 x 1.5 cm (but imaging time increases with area; realistically ~0.3-0.5cm^2)	Up to 43 antibodies	Single-cell / subcellular	
Visium	RNA	Sequencing	FFPE sections (5µm onto Visium slide; stained with H&E or IF) or fresh-frozen	6.5 x 6.5 mm square per section (0.43cm^2) ; 4 sections per slide	18,000 genes	55 µm per spot	~ 50,000 mean reads/spot ~ 4,000-5,000 genes per spot ~ 99% reads mapped to probe set
Slide-Seq	RNA	Sequencing	Fresh-frozen sections	up to 3 mm diameter circular area (~7mm^2)	Whole transcriptome	10 µm per bead	~ 150 UMIs/bead
MERSCOPE	RNA	Imaging	Fresh- or fixed-frozen sections (FFPE soon possible?)	up to 1 cm^2 within circular region w/ 20 mm diameter	300 genes	Single-cell / subcellular	Experiments failed
CosMx	RNA	Imaging	FFPE sections (5 µm onto the back of regular glass slides) or fresh-frozen sections	up to 25 FOV boxes with 0.9 x 0.7 mm (0.63mm^2) within tissue area of max 15 x 25mm (3.75cm^2)	960 genes	Single-cell / subcellular	- 162 mean transcripts/cell - 73% of transcripts assigned - ~12.8 Mio transcripts detected in 0.116 mm^2 tissue area - ~58,000 cells profiled
Xenium	RNA	Imaging	FFPE sections (5µm onto Xenium slides)	up to 9 x 31 mm (2.79cm^2) per slide, 2 slides per run	280 genes	Single-cell / subcellular	- ~ 110 median transcripts/cell - 40 Mio transcripts detected in 0.89 cm^2 tissue area - ~300,000 cells profiled

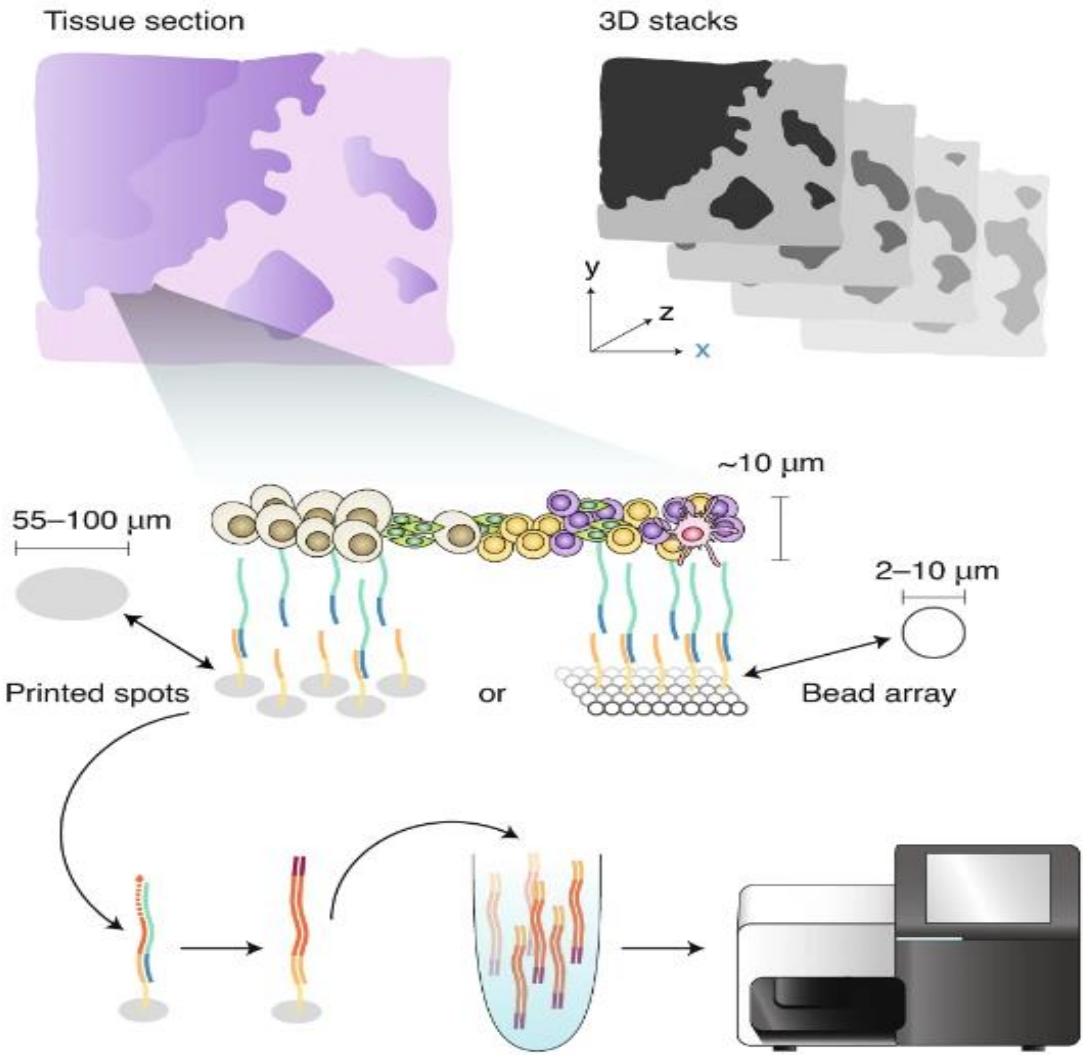
Quy Nguyen

Imaging Based vs Sequencing Based



Moses *et al*, 2022

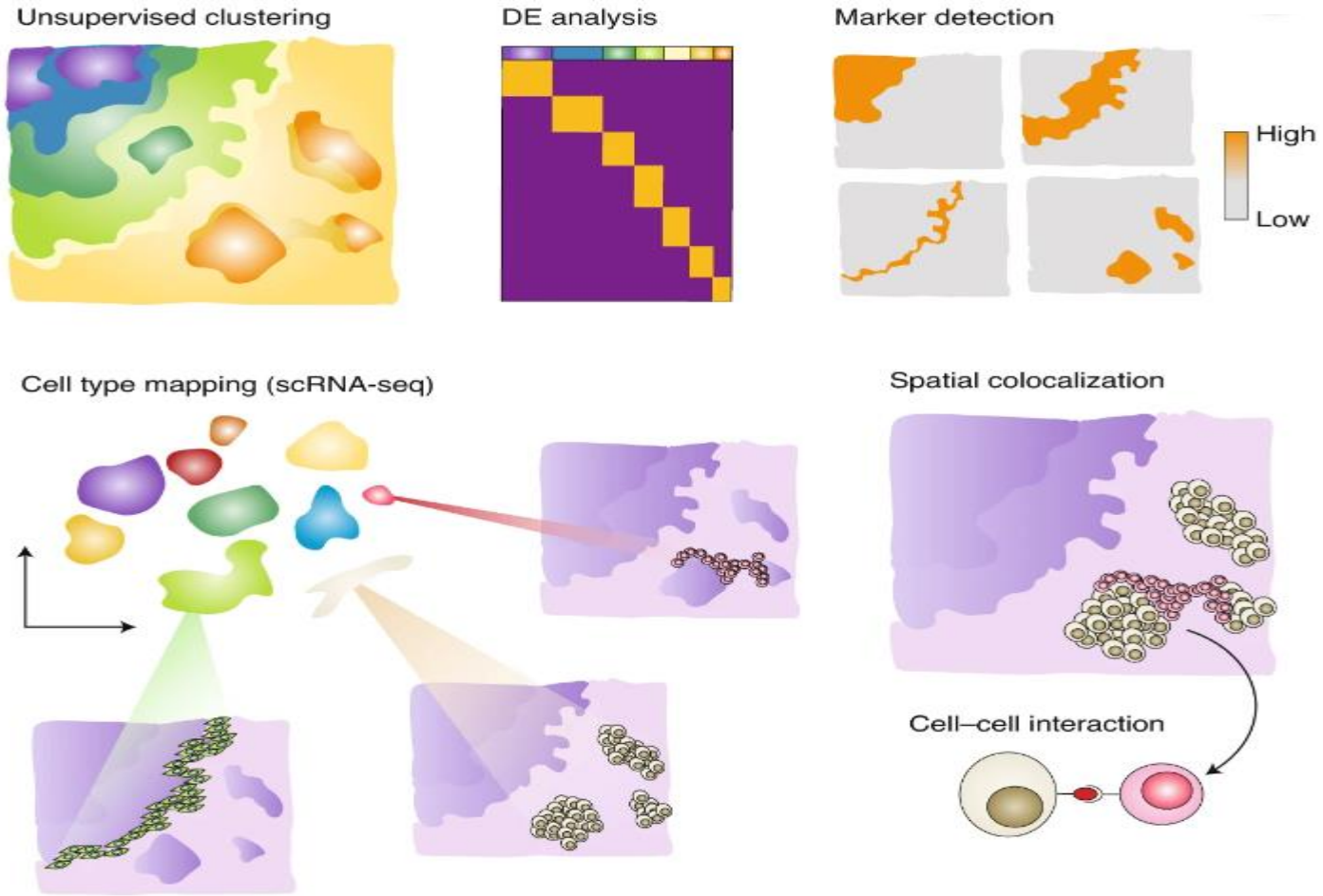
Sequencing based Spatial Transcriptomics



- ✓ 10x Visium or ST
- ✓ Slide-seq, Slide-seq v2

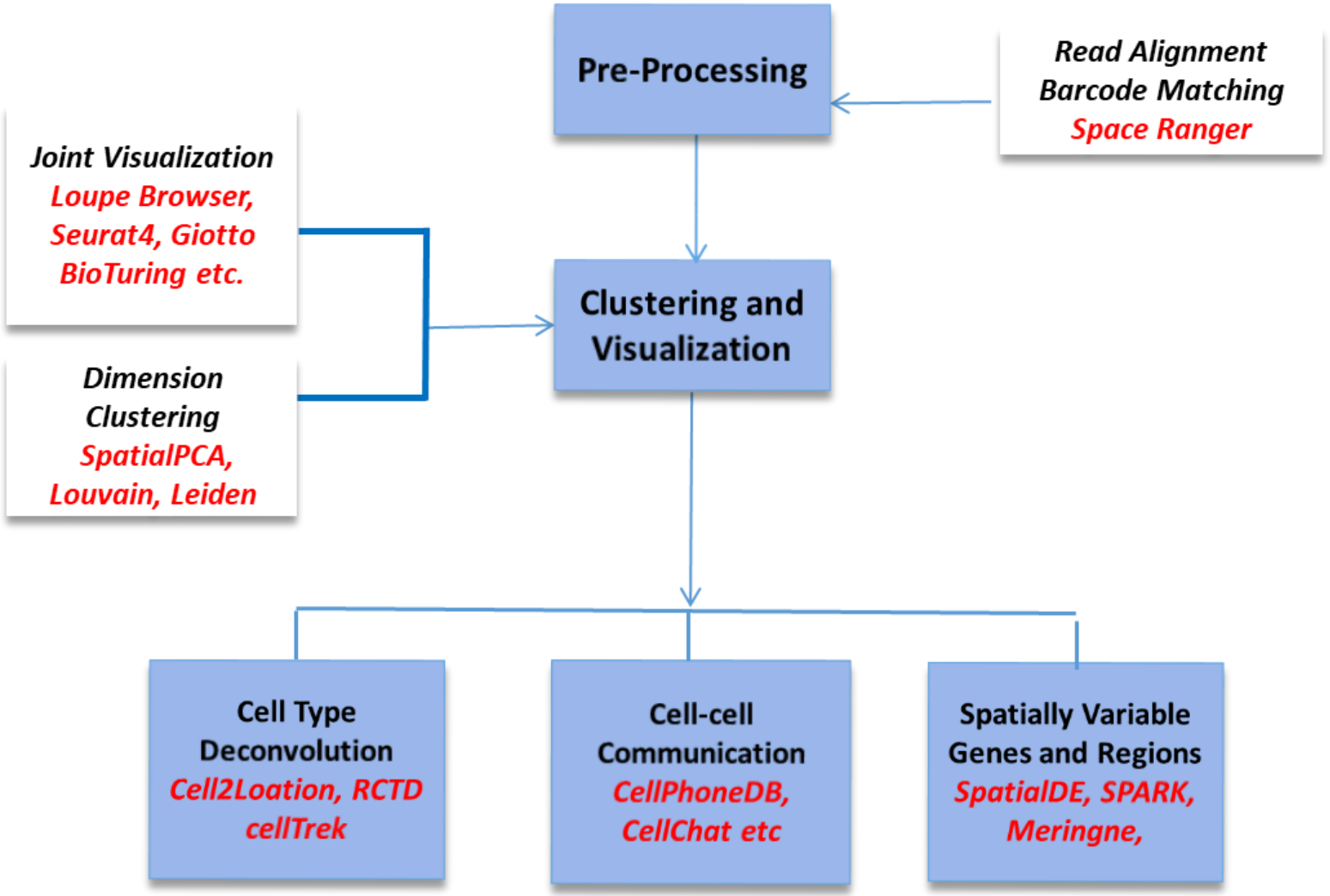
Larsson *et al*, 2021

Downstream Analysis

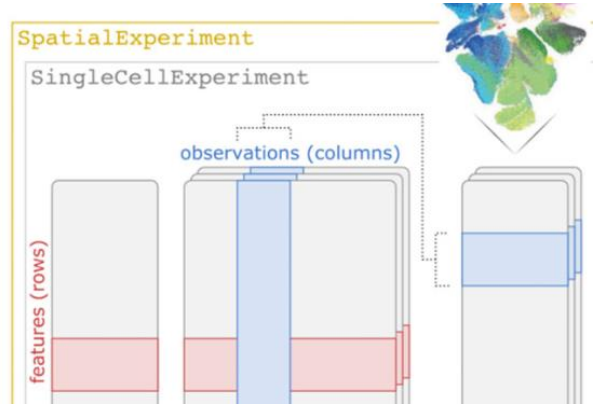


Larsson *et al*, 2021

Visium Data Analysis Work Flow



Analytical tools for ST Downstream Analysis



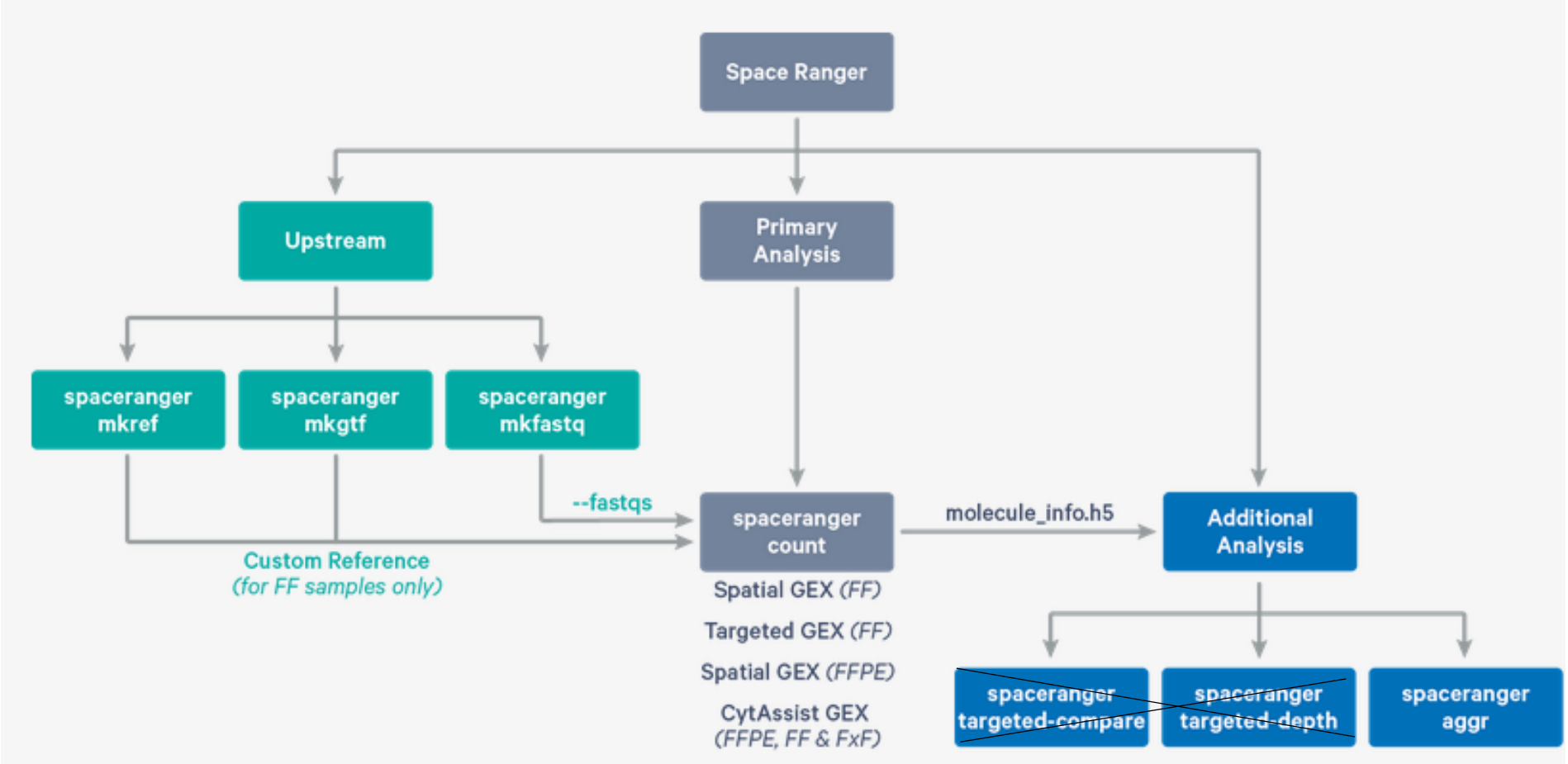
Giotto

squidpy

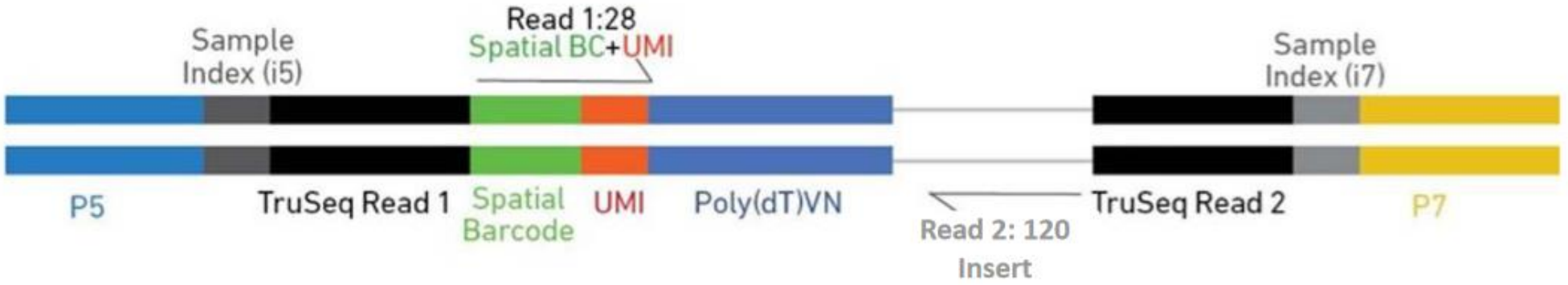
BioTuring

QuPath

Primary Analysis with 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

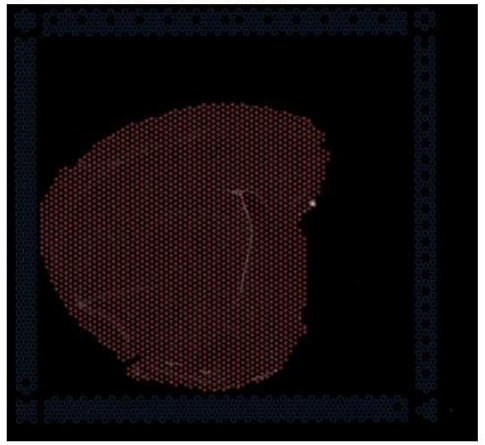
Number of Reads	145,580,962
Valid Barcodes	97.9%
Valid UMIs	100.0%
Sequencing Saturation	69.5%
Q30 Bases in Barcode	96.4%
Q30 Bases in RNA Read	94.4%
Q30 Bases in UMI	95.9%

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

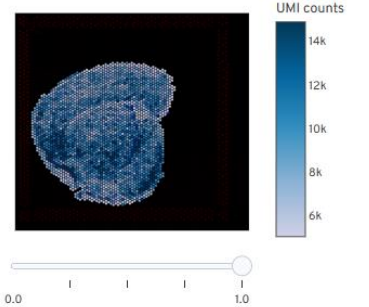
A1_1-1

Summary | Gene Expression

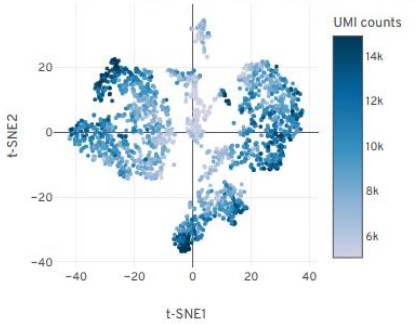
UMIs Detected

Color Scale: Default

Tissue Plot with Spots Colored by UMI Count



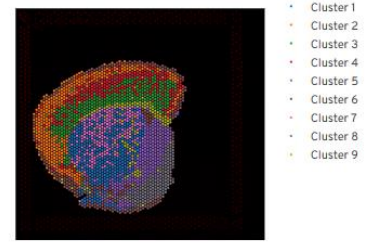
t-SNE Projection of Spots Colored by UMI Counts



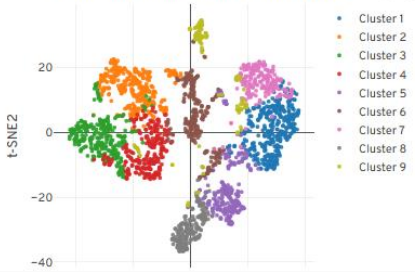
Clustering

Clustering Type: Graph-based

Tissue Plot with Spots Colored by Clustering



t-SNE Projection of Spots by Clustering



MatrixMarket Format for Feature 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
```

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

Seurat: Data Import

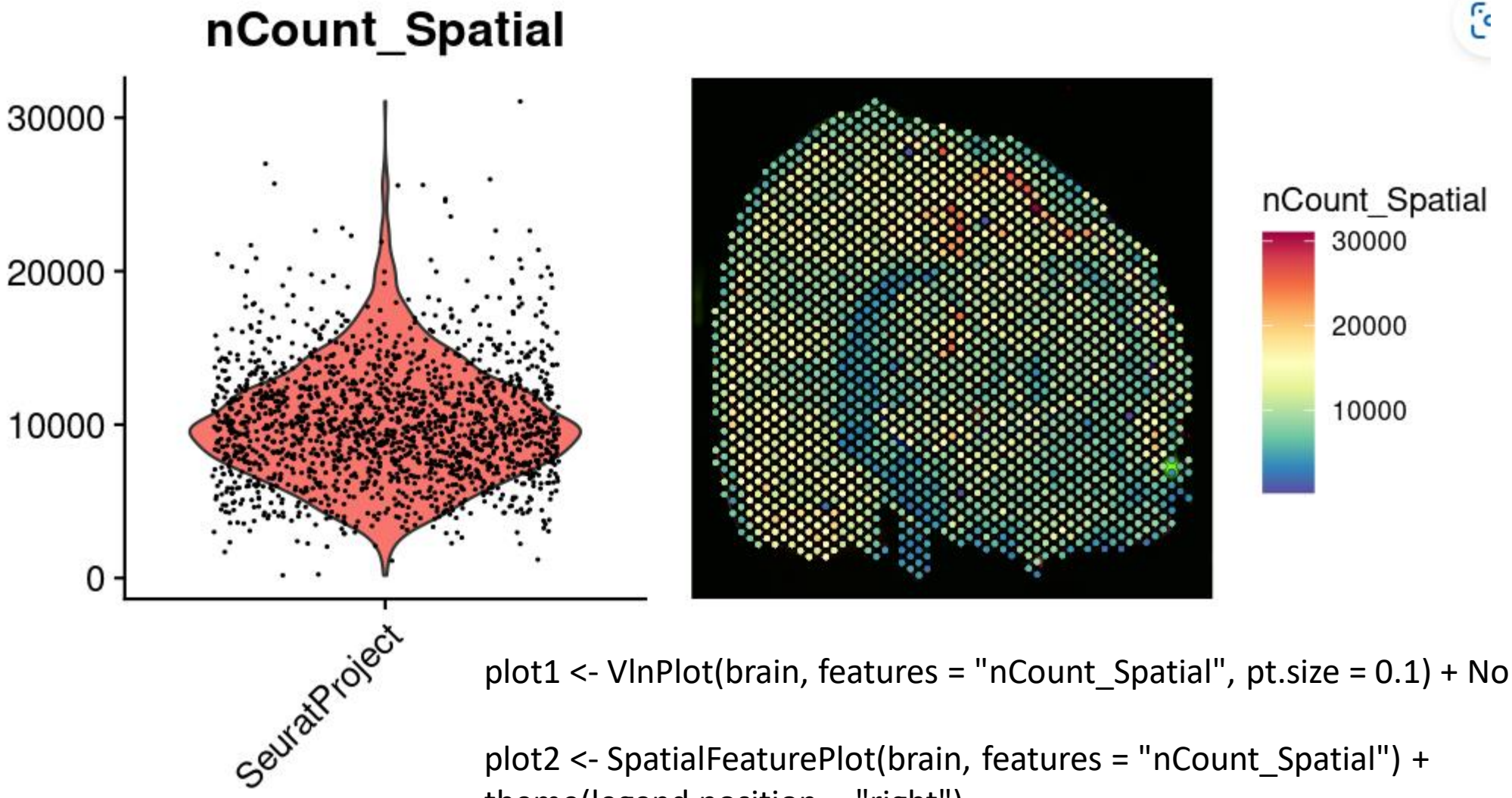
Load a 10x Genomics Visium Spatial Experiment into a `Seurat` object

Source: `R/preprocessing.R`

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

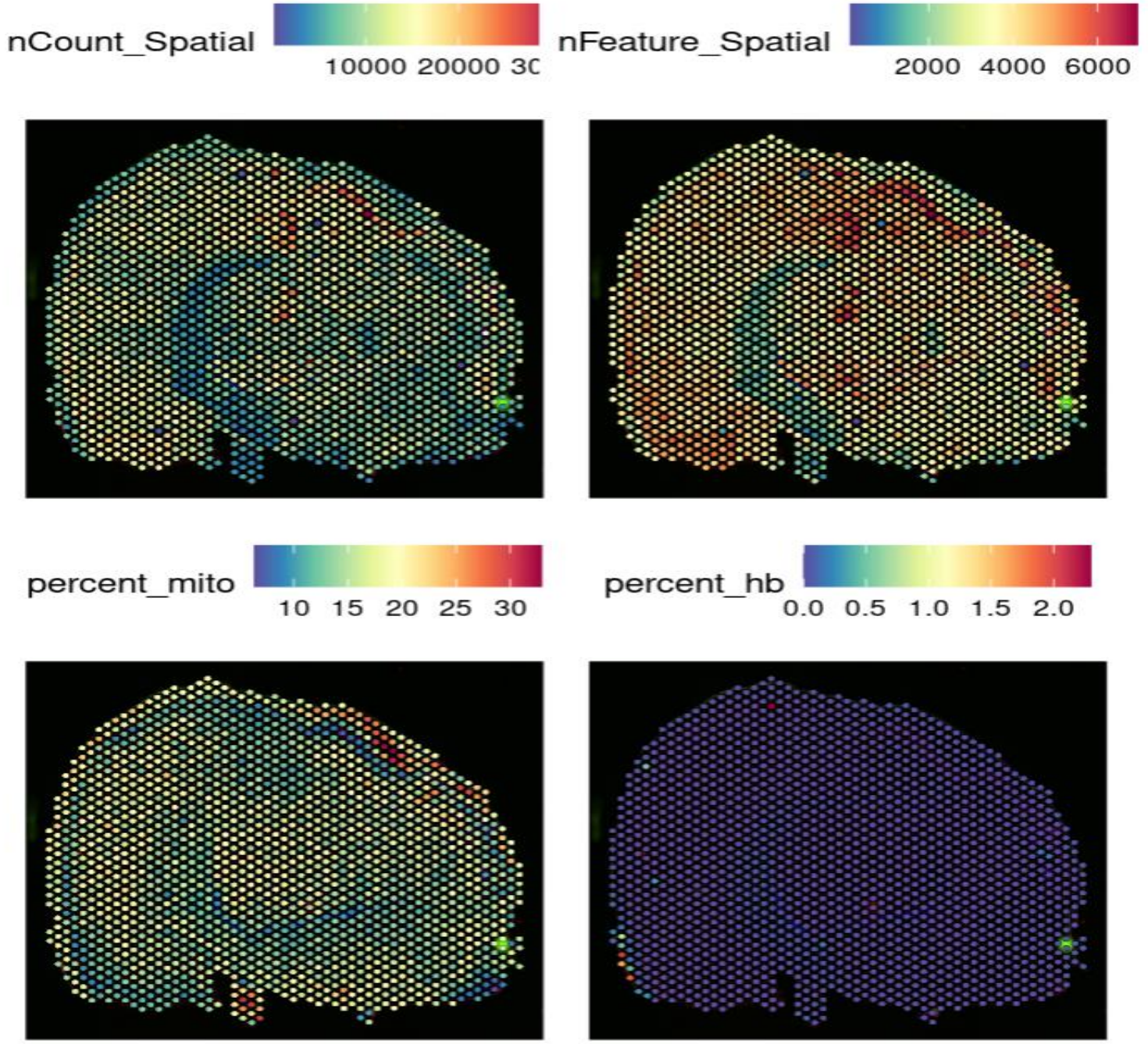
```
Load10X_Spatial(
  data.dir,
  filename = "filtered_feature_bc_matrix.h5",
  assay = "Spatial",
  slice = "slice1",
  filter.matrix = TRUE,
  to.upper = FALSE,
  image = NULL,
  ...
)
```

Seurat: QC and Filtering



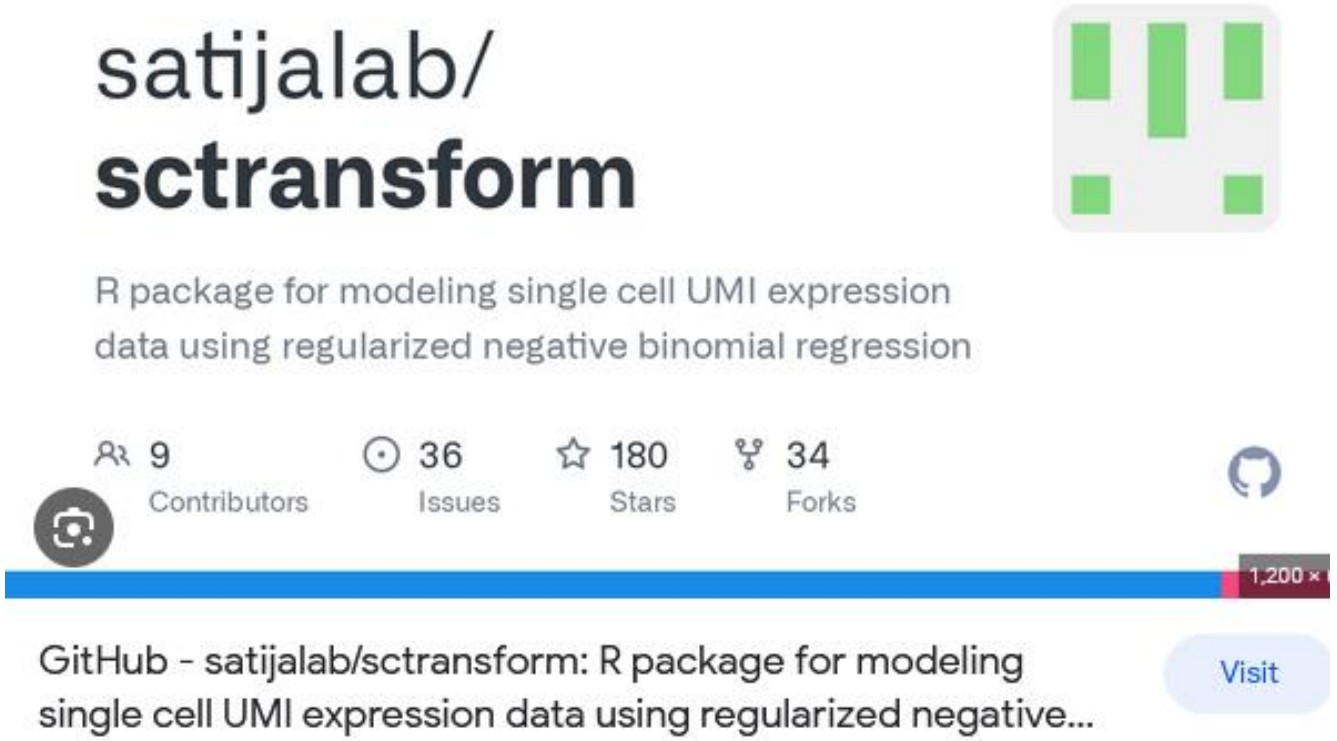

```
plot1 <- VlnPlot(brain, features = "nCount_Spatial", pt.size = 0.1) + NoLegend()

plot2 <- SpatialFeaturePlot(brain, features = "nCount_Spatial") +
theme(legend.position = "right")
```

QC




```
SpatialFeaturePlot(brain,
features =
c("nCount_Spatial",
"nFeature_Spatial",
"percent_mito","percent_h
b"))
```

Normalization, Feature Selection: SCTransform

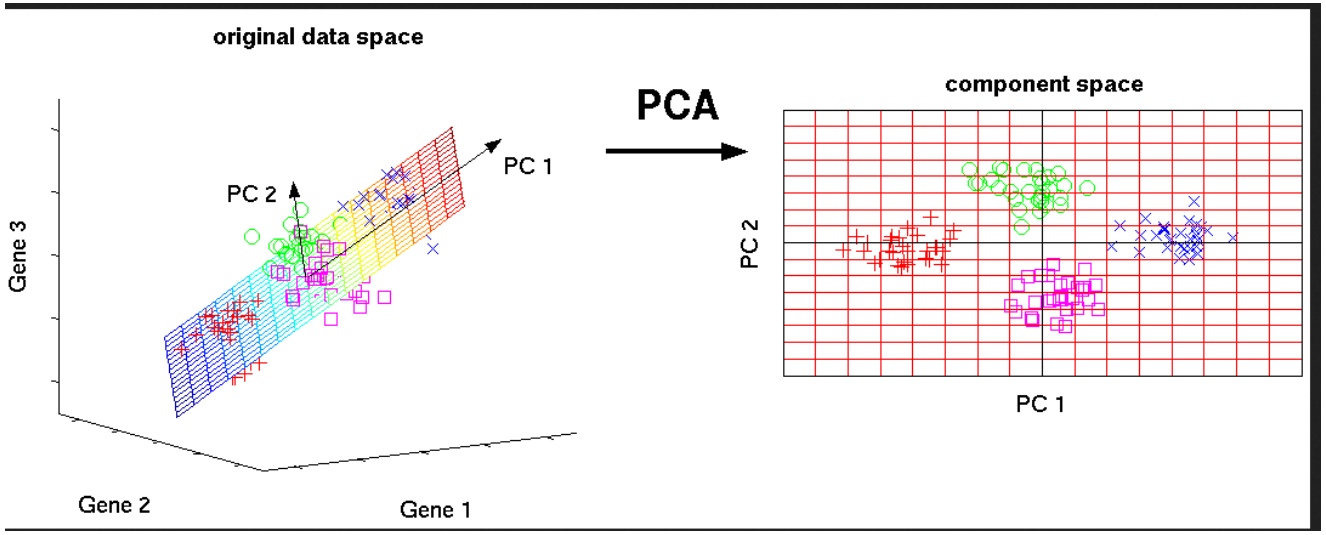


brain <- SCTransform(brain, assay = "Spatial", verbose = FALSE")

Default top 3000 HVGs

Linear Dimension Reduction

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



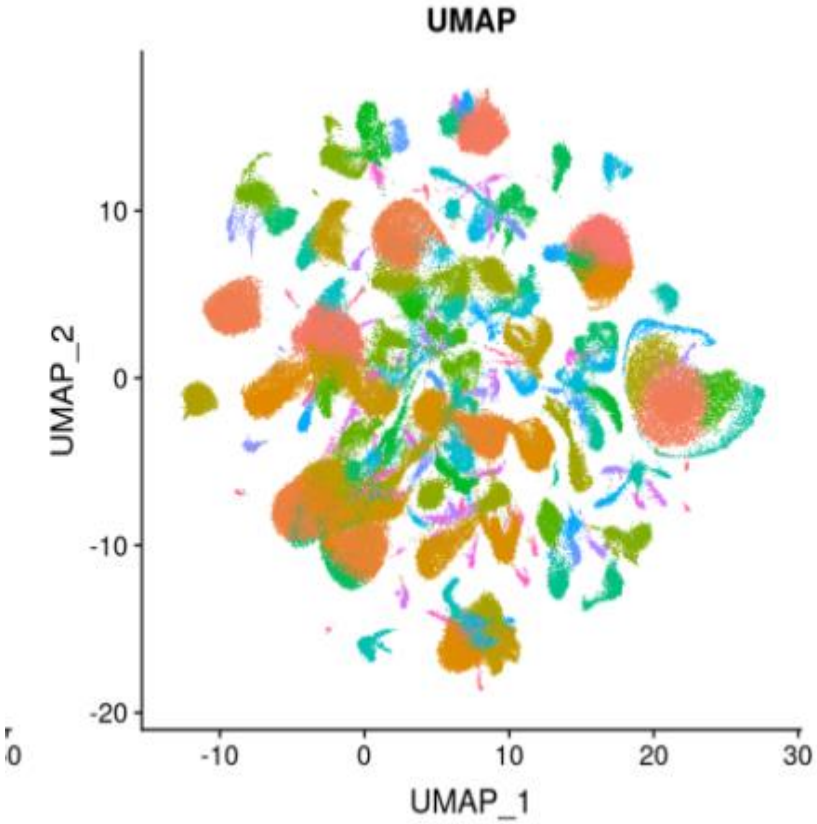
Matthias Scholz, 2015

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

HVGs from SCTransform go into PCA

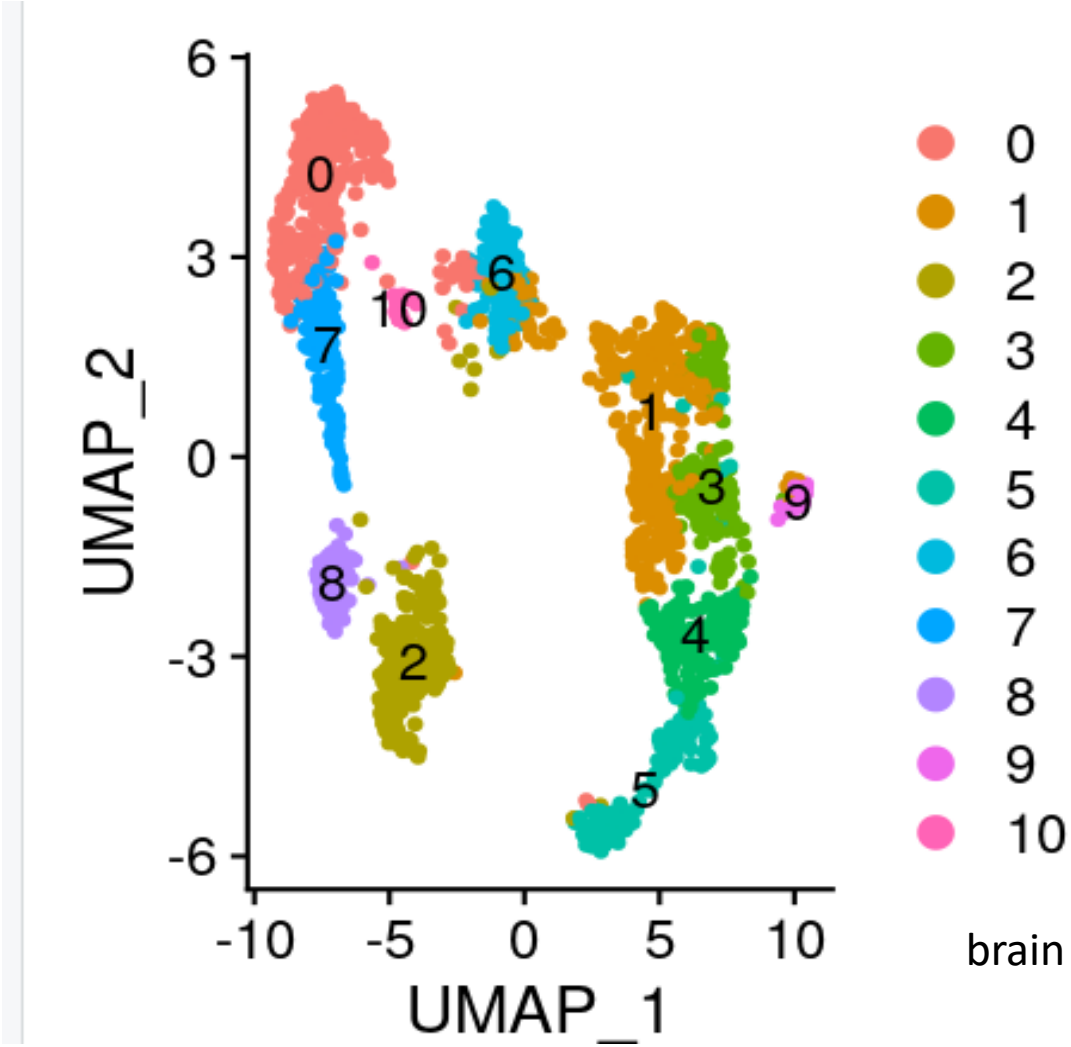
Clustering

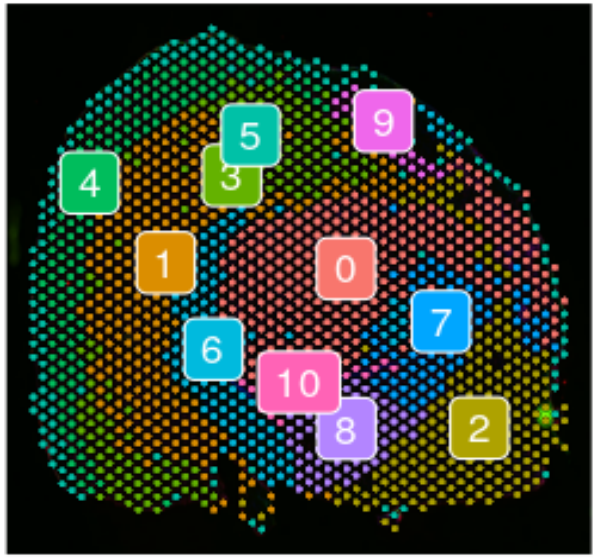
- Distance/similarity metrics
- Algorithm choices
  - K means
  - Hierarchical
  - Density based
  - Graph partition
  - …




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

Non Linear Dimension Reduction : UMAP

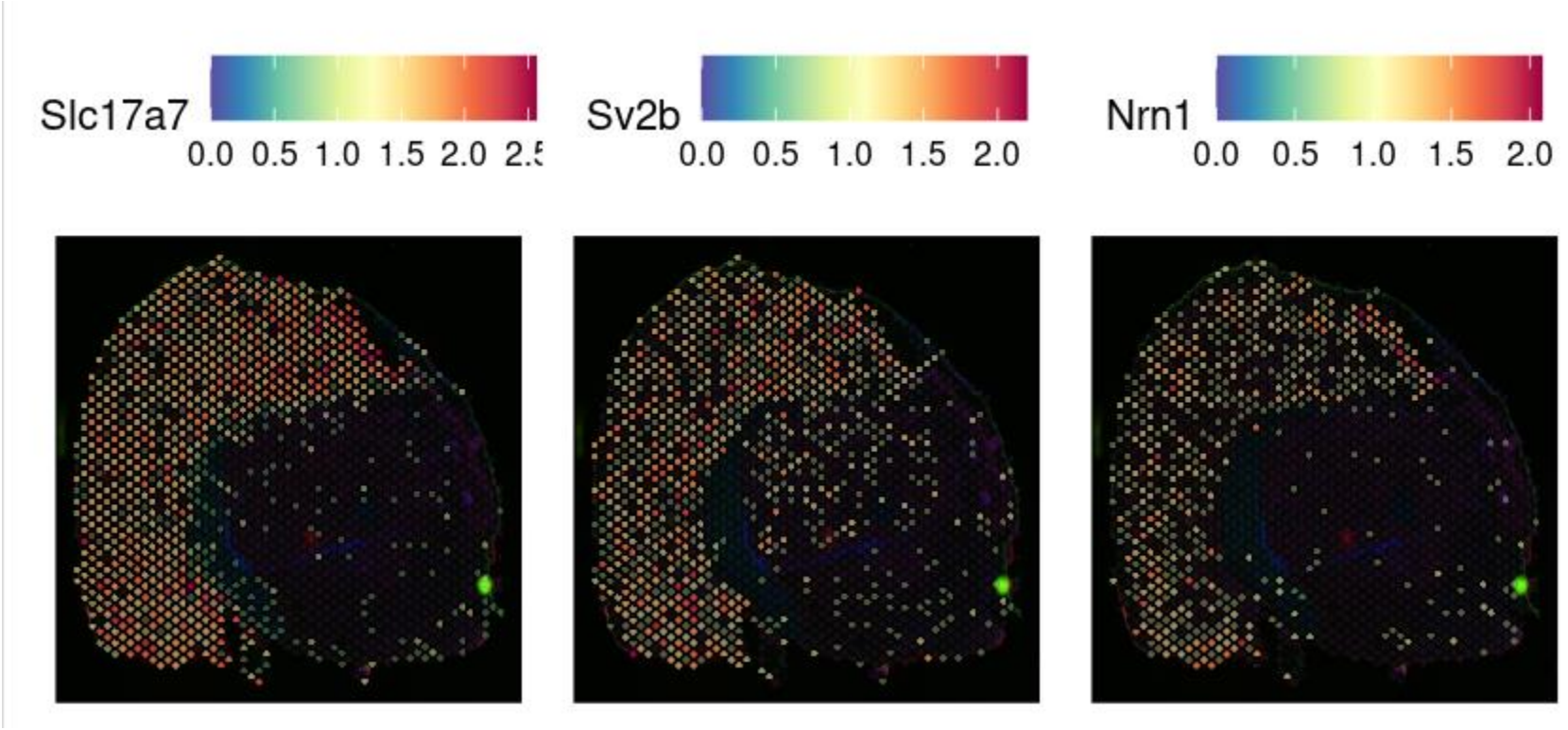




brain <- RunUMAP(brain, reduction = "pca", dims = 1:30)

UMAP is not clustering!

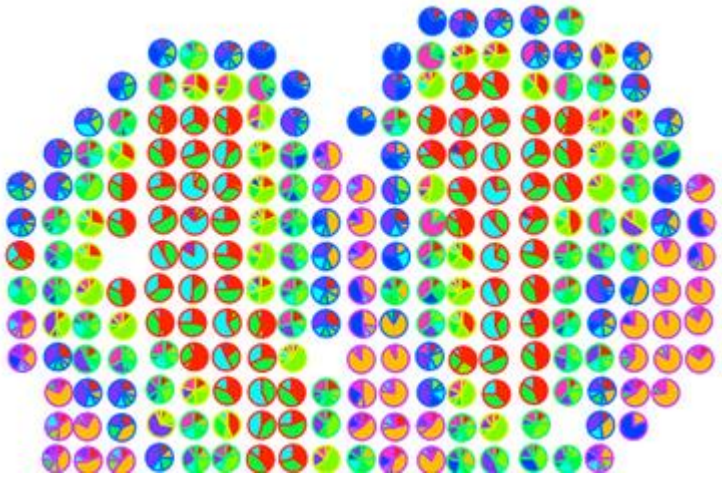
Seurat: Visualization of Marker Genes



SpatialFeaturePlot(brain, features = top.features, ncol = 3, alpha = c(0.1,1))

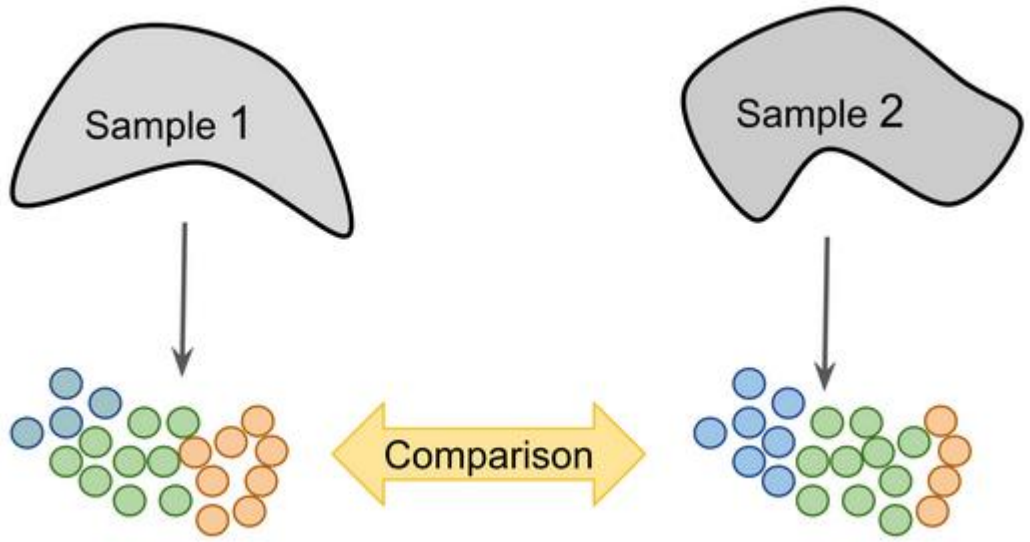
Cell Type Deconvolution

- Bulk RNA-seq deconvolution
  - ✓ MusiC
  - ✓ SCDC
  - ✓ CIBERSORTx
- Sequencing based ST deconvolution
  - ✓RCTD: robust cell type decomposition
  - ✓cell2location
  - ✓SpatialDWLS
  - ✓STdeconvolve



Data Integration

- Allows for analyzing single cell samples from different technologies and conditions at the same time
  - ✓ Harmony
  - ✓ LIGER
  - ✓ Seurat



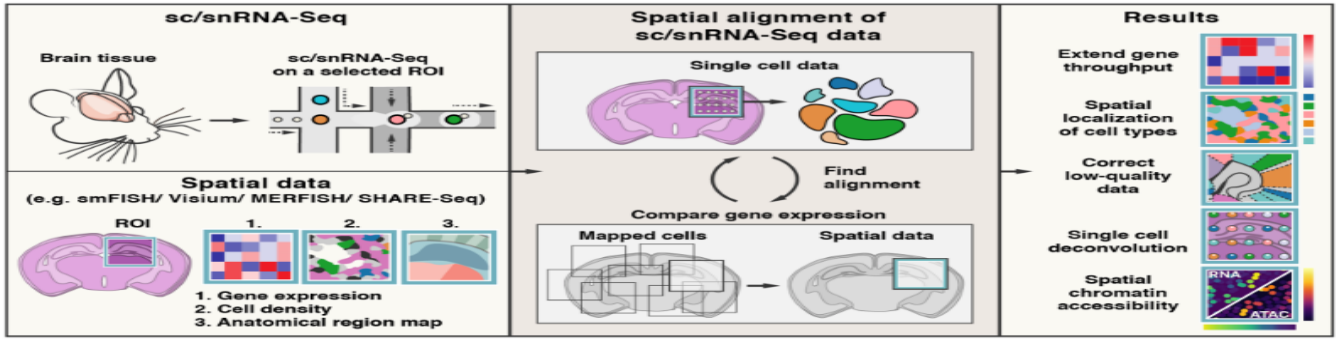
Integrating scRNA Data with ST Data

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

Cell-Cell Communication Network

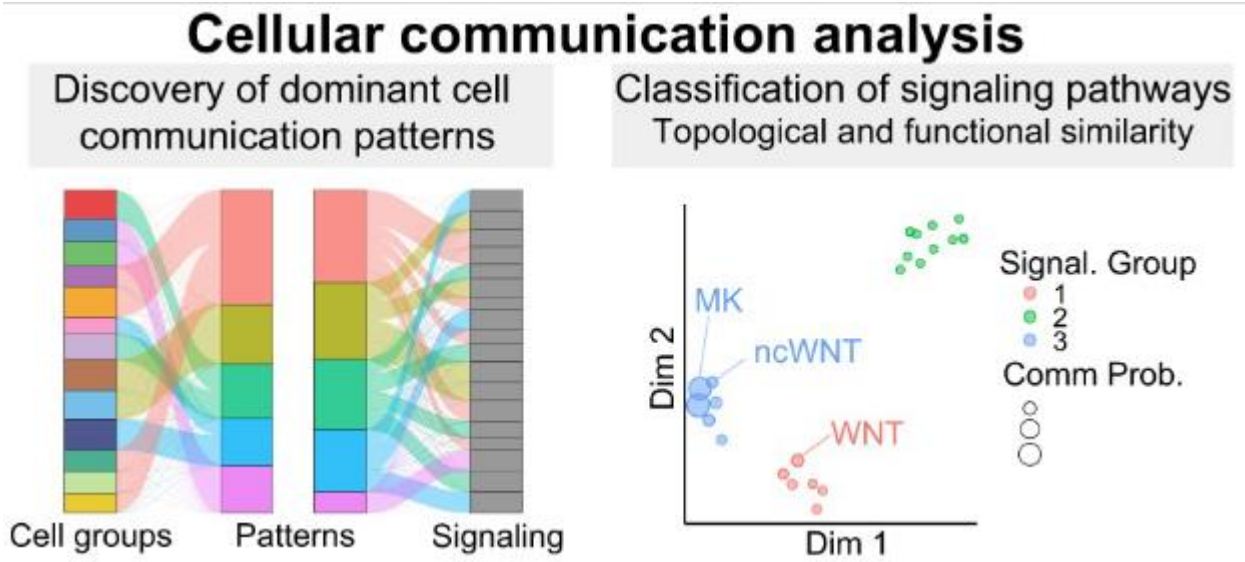
- Use network-based analysis and ligand-receptor biology to predict signaling between groups of cells

CellChat
Integrated Web-based CellChat Explorer

✓CellChat

✓SpaOTSc, CytoTalk

✓SpaTalk



Spatial Omics Data Analysis

Challenges:

- Wide range of protocols and data processing pipelines
- Increased data volume, run time and memory usage demands more hardware and staff time
- Lack of standardized metrics or benchmarks

Data Analysis Choices

- Tools choices: proprietary pipeline or open source?
- Local desktop, HPC or cloud computing? Cost, flexibility, capacity and ease of use.

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