Day1 Single cell RNA-seq

NIEHS Workshop--Analyzing NGS data

https://github.com/zhuy16/NIEHS_NGS_Workshop

July 13, 2021

9am-2pm

Yunhua Zhu, PhD

zzhonghua@gmail.com

Computational Genomics Specialist, Transcriptomics

Medical Science & Computing

Instructor's background

- Bachelor @ National University of Singapore(NUS) in Biochemistry 2000-03
- Ph.D @ NUS in stem cell biology | 2006 -10
- Postdoc @ Hopkins with wet lab & dry lab | 2014 - 2019
  - Neurogenesis w/t single-cell RNA-seq
  - Neurodegeneration w/t single-nucleus RNA-seq
  - Learning R, Bash script statistics through Google and Youtube
- Computational Genomics Specialist @ BCBB | 2019
  - Single-cell RNA-seq, CITE-seq
    - SMARTseq2 pipeline with FASTQC, RSEM, SENIC
    - 10X genomics pipeline with Cellranger, Seurat, Scanpy and RNA Velocity.
  - Bulk RNA-seq
    - Deconvolution Bulk RNA-seq using ABIS and CybersortX.
    - Functional annotation with clusterProfiler.
  - Epigenetics
    - ATAC-seq and Chip-seq

Aims of the workshop

- Target audience: NIEHS researchers who want to learn NGS data analysis.
- Aims: to provide an overview of NGS analysis workflow for scRNA-seq, bulk RNA-seq and ChIP-seq.
- Prerequisite: comfortable with shell scripts, R and python codes.
- We will use biowulf HPC as the computation environment, and use Jupyter lab as an interface
- The github repository is for illustration only. Data files are included in Biowulf folders for use in the class.
- The README.md contains information to set up the workplace.
- This is 2-day (4hr each day) workshop on NGS data analysis.
  - On the 1st day, we will focus on single cell RNA-seq. I will start from NGS_Workshop_Day1.pdf from the resources folder, do a ~30mins presentation.
  - This will help setting up the working environment and get background knowledge on single-cell RNA-seq.
  - The presentation will be followed by practical exercises in the scRNA-seq/notebooks folder.

  - On the 2nd day, we will discuss Bulk RNA-seq and ChIP-seq. I will start from the NGS_Workshop_Day2.pdf file to do a presentation, which will be followed by exercises in the bulk-RNA and ChIP-seq folder.
- **We will use jupyter lab interface on Biowulf (an NIH HPC system), Please refer to Getting_started.md to setup the working environment.**

Keep in mind

- It is very limited time for a broad topic
- I will not try to cover many details.
- It is designed to provide just an overview on computation and a concrete set of starting materials.
- Most of these are selected online materials (with references) that is tailored to run on Biowulf.
- Questions and feedback are welcome in the workshop and in the google sheet.
  - **Questions & Feedback**

Agenda -- day 1, single cell RNA-seq

- 9-10AM, setting up the system
  - Get to know each other
  - Set up pseudolink (NIEHS_NGS) to the /data/studentxx folder
  - Copy workshop files to NIEHS_NGS
  - Initiate jupyter lab
- Questions and break (10 minutes)
- 10:30-11AM, Introducing single cell RNA-seq 30 minutes.
- 11-12AM, From bcl, fastq to expression matrix
  - Convert bcl to fastq files using the tiny-bcl folder 15 minutes
  - Using cellranger count to get expression matrix
- 12-12:30PM, Questions and break
- 12PM-1PM, Run basic pipelines
  - Using Seurat pipeline to import alignment result do QC, and normalization.
  - (optional) Using Scanpy pipeline to import alignment results to do QC, and normalization.
- 1PM-1:30PM, use RNA velocity to infer trajectory.
  - Run Velocyto CLI to get the reads aligned to non-spliced region.
  - Import into scVelo (python) to do trajectory inference.
- 1:30PM—2PM, Prepare for day2
  - Download and install IGV, IGB

Agenda -- day 2, bulk RNA-seq and ChIP-seq

Part II, Bulk RNA-seq

- 9:00-9:15AM, connect to biowulf
- 9:15-9:30AM, introduction to bulk RNA-seq
- 9:30-10AM, From Fastq files, alignment with RSEM, Tophat
- 10AM-10:30AM, Visualize the output using IGV.
- 10:30-11:00AM, Normalization and differential analysis with DESeq2
- Questions and break

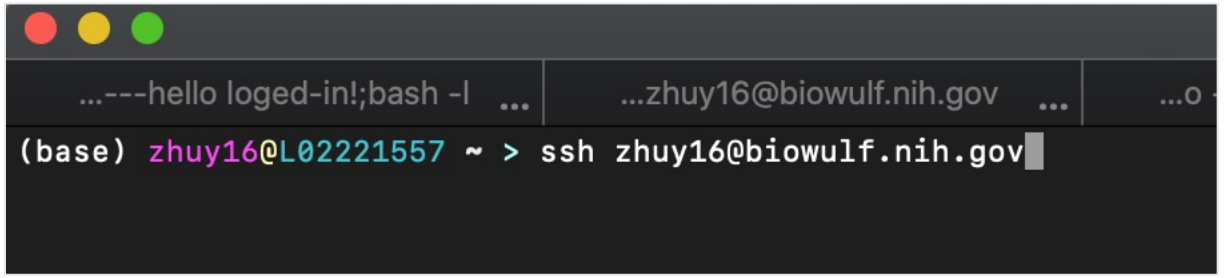
Part III, Chip-seq

- 11:30AM-12PM, Introduction to ChIP-seq analysis
- 12:00-12:15AM, Alignment to genome using Bowtie2
- 12:15-12:25AM, Use MACS2 to call peaks, and use Homer to call differential peaks
- 12:25-12:45AM, Use IGB to visualize peaks
- 12:45-1:15AM, Use CEAS to annotate peaks and summarize statistics.
- 1:15am -1:45AM, Use Homer to study TF binding site analysis. And use GREAT for gene ontology.

Review & Summary

(code can be copied from Getting_started.md)

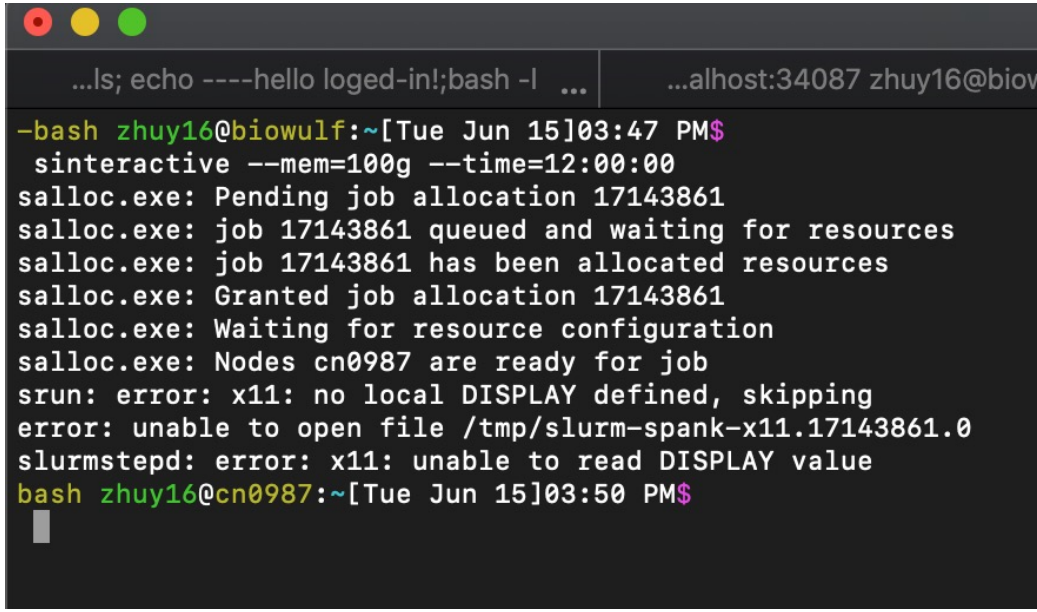
Log in to biowulf and go to the folder



- ssh user_id@biowulf.nih.gov
- # set up sudo links to your storage folder:
- module load tmux
- tmux new # get a detachable terminal
- sinteractive # get a basic compute node
- cd
- ln -s /data/user_id NIEHS_NGS

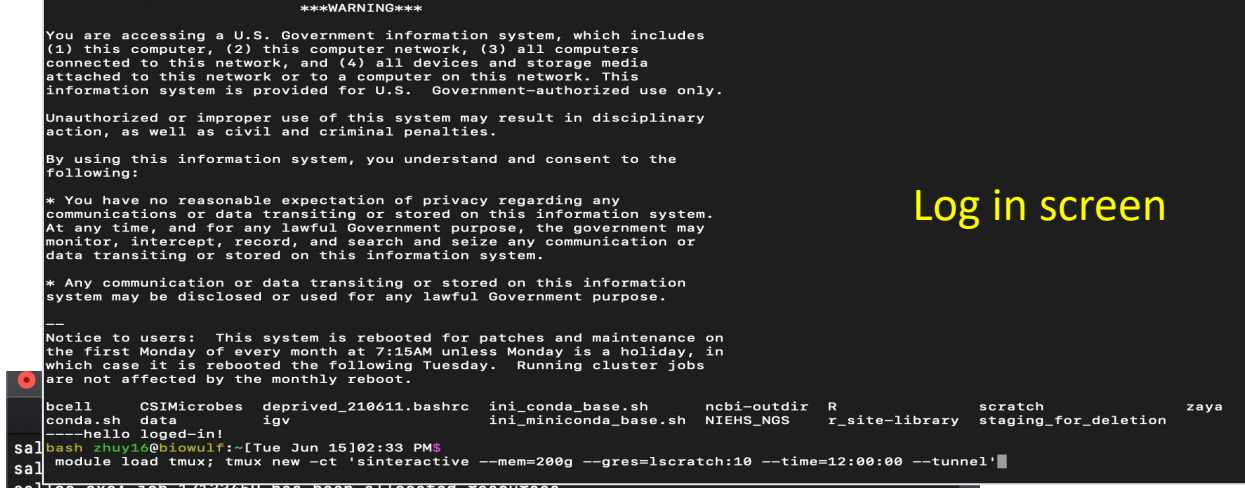
copy the workshop material to your scratch folder:

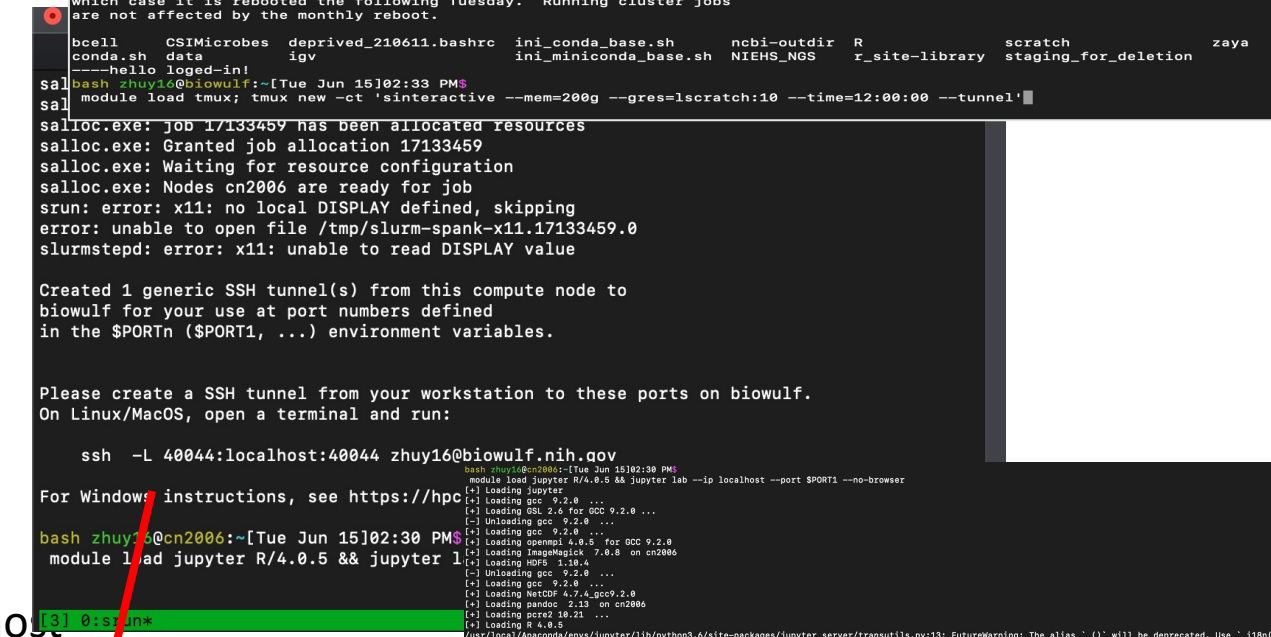
- cp -r /spin1/users/classes/NIEHS_NGS/workshop ~/NIEHS_NGS/

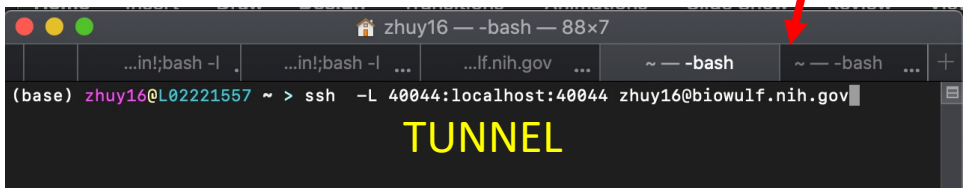


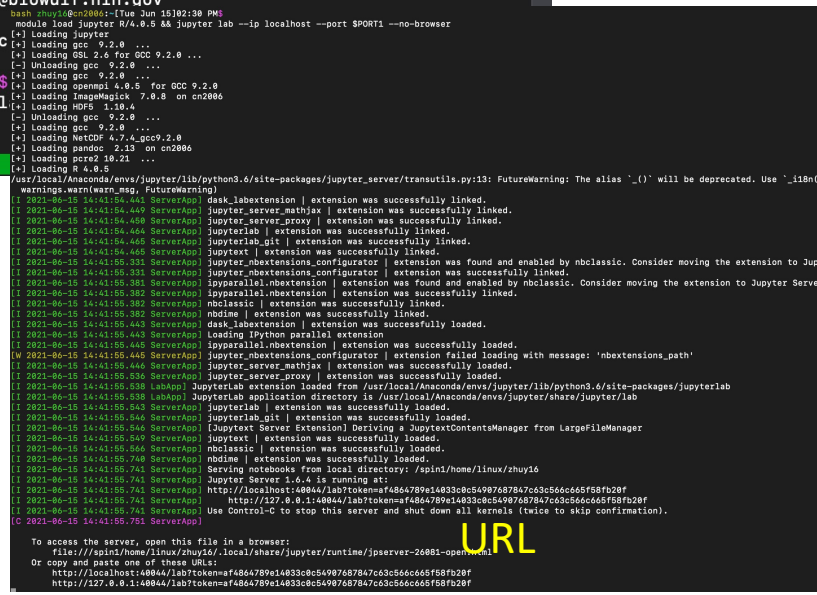
Start up the jupyter lab, --our working interface

- # Open another terminal
- ssh studentxx@biowulf.nih.gov
- Enter password
- module load tmux
- module load tmux; tmux new -ct 'sinteractive --mem=50g --time=12:00:00 --tunnel'
- # Copy the tunnel script to another (3rd) terminal, execute and enter password, to establish a ssh tunnel between local computer and the work node.
- module load jupyter R/4.0.5 && jupyter lab --ip localhost --port $PORT1 --no-browser
- # wait until an URL link appears, copy it to your web browser, to get connected to biowulf through the jupyter lab interface.



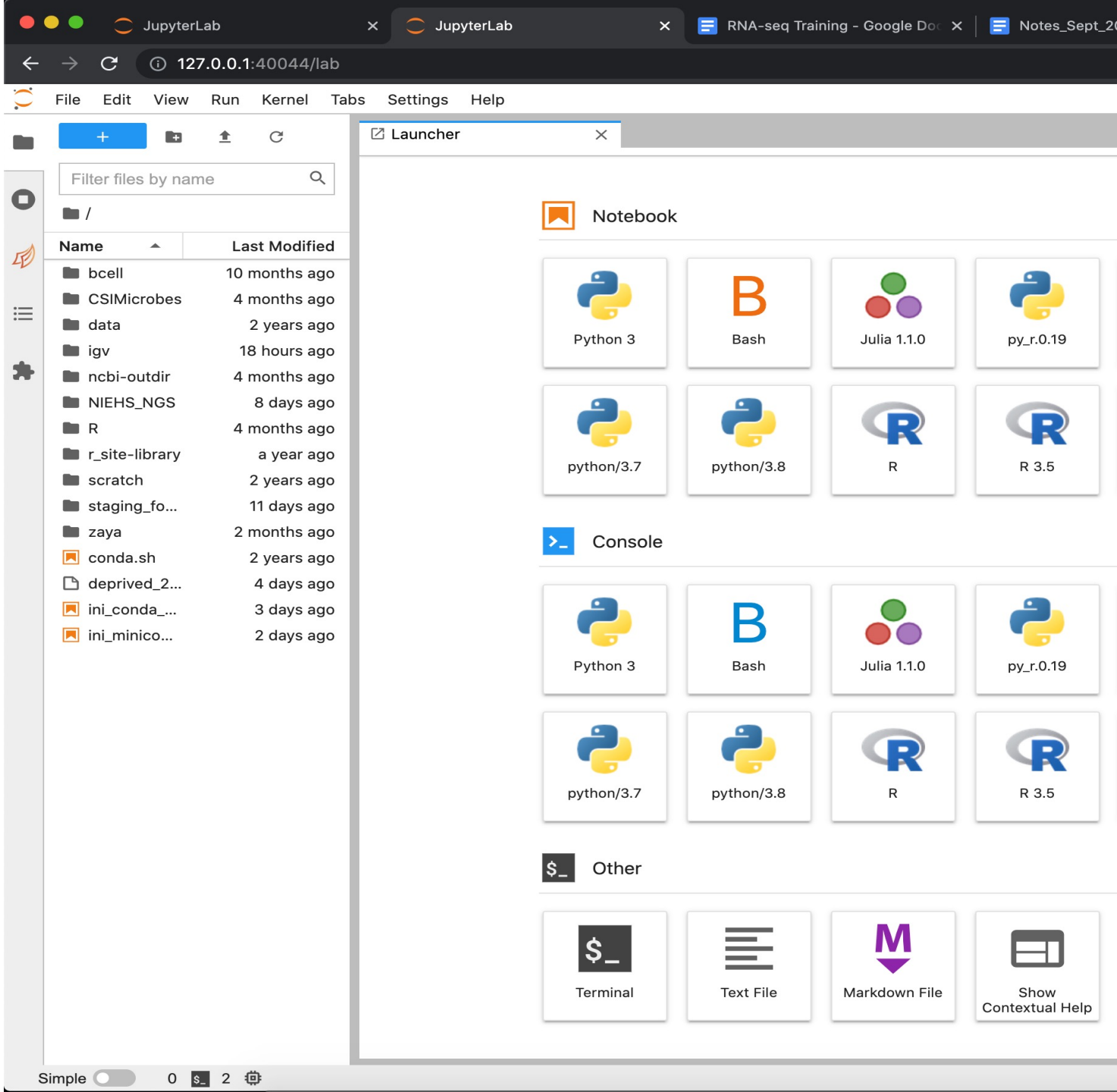






Jupyter lab

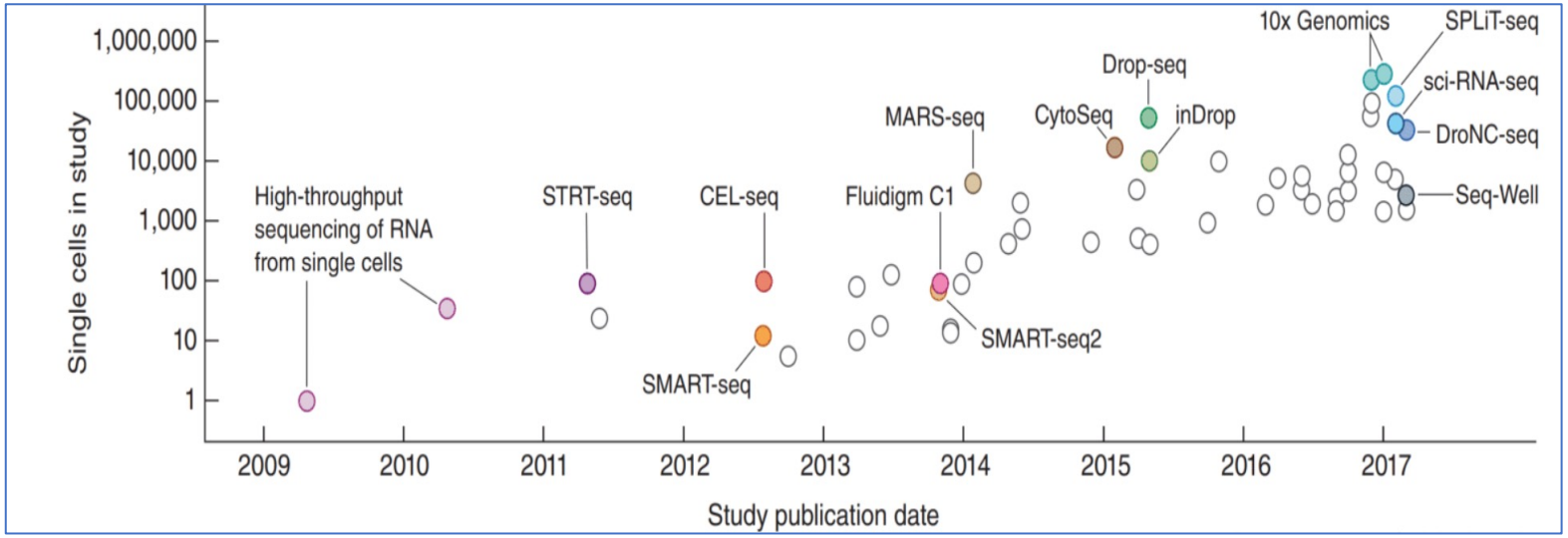
- Navigate files
- Bash terminal
- Control kernels
- Jupyter notebooks for
  - Kernals
    - Bash
    - R
    - Python
    - Julia
  - Document analysis
  - Git version control
  - Report results



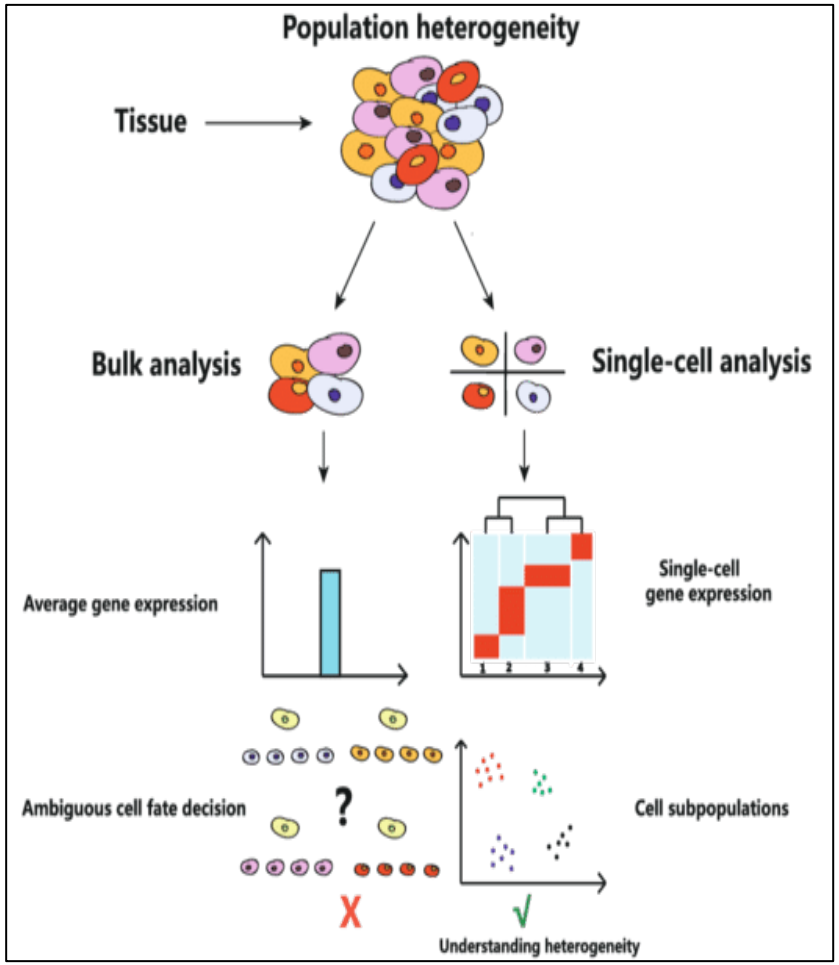
Single cell RNA-seq

- Evolution of single cell technologies
- Introduction to 10X genomics
- Bioinformatic workflow

Evolution of single cell technologies

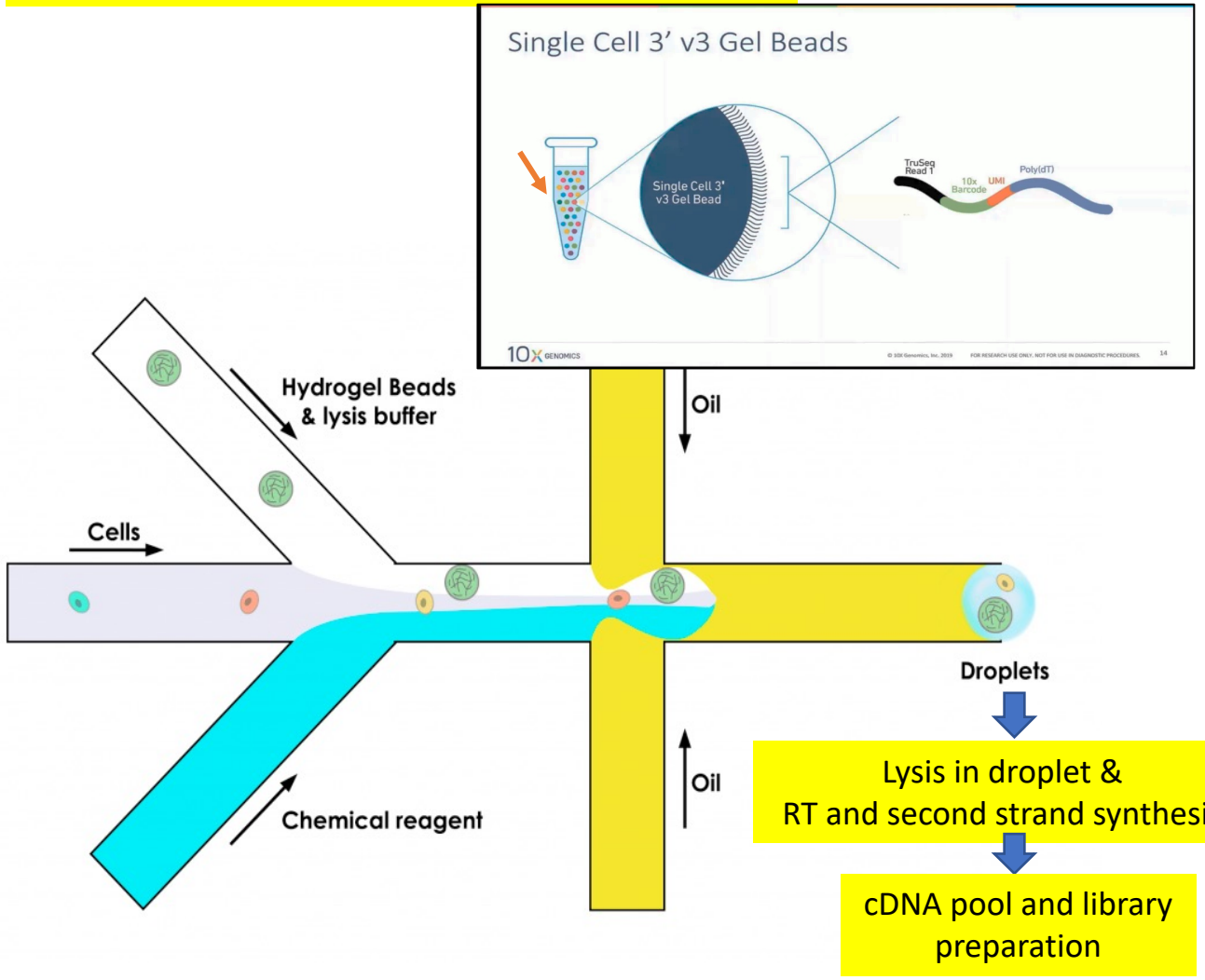


Exponential scaling of single-cell RNA-seq in the past decade. Nature protocols 13;4;599-604)



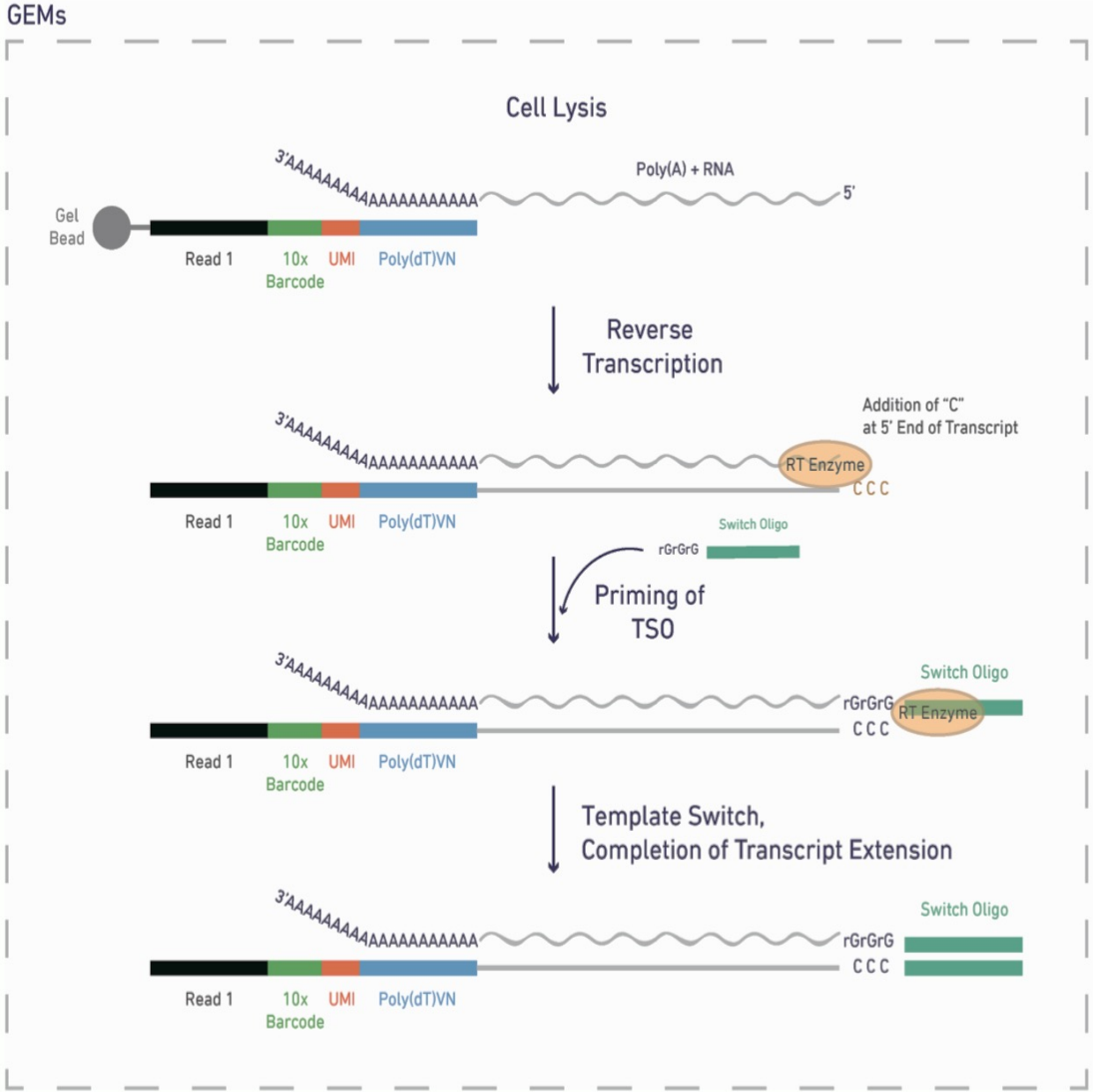
10X Genomics– commercial solution that facilitates automatic generation of Gel Bead-in-Emulsion (GEM)

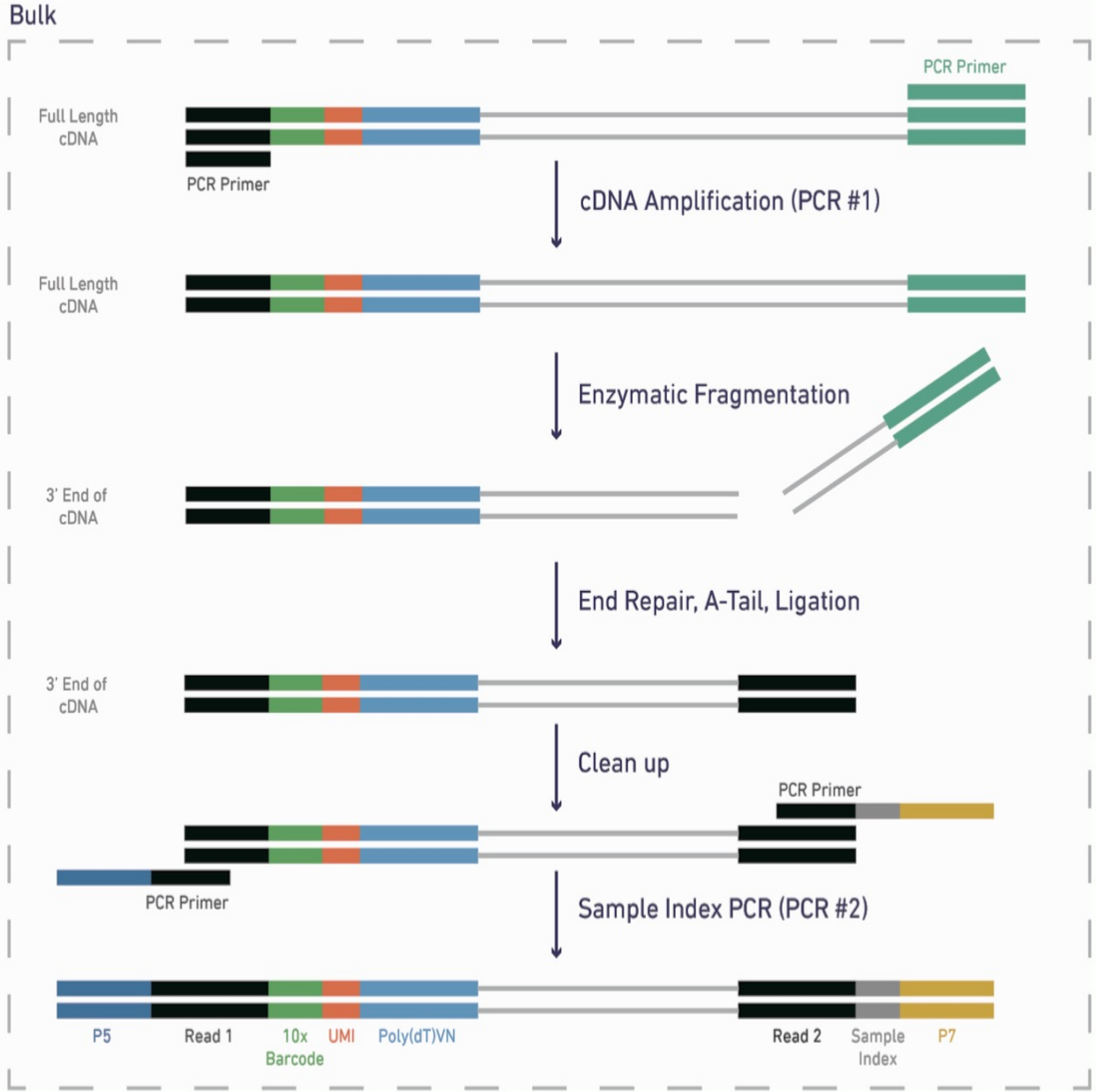
- In a GEM droplet, one hydrogel bead and one cell were captured
- One hydrogel bead is attached with millions of poly-T primers with an identical unique barcode.
- cDNA and the second-strand synthesis in the droplet
- Droplets (~1nL) are disrupted to collect all the barcoded samples for highly multiplexed library preparation and sequencing
- Standardized automation and reagent has made sequencing library preparation very efficient



Inside individual GEMs (Gel Bead-in-Emulsion)

Pooled cDNA processed in bulk

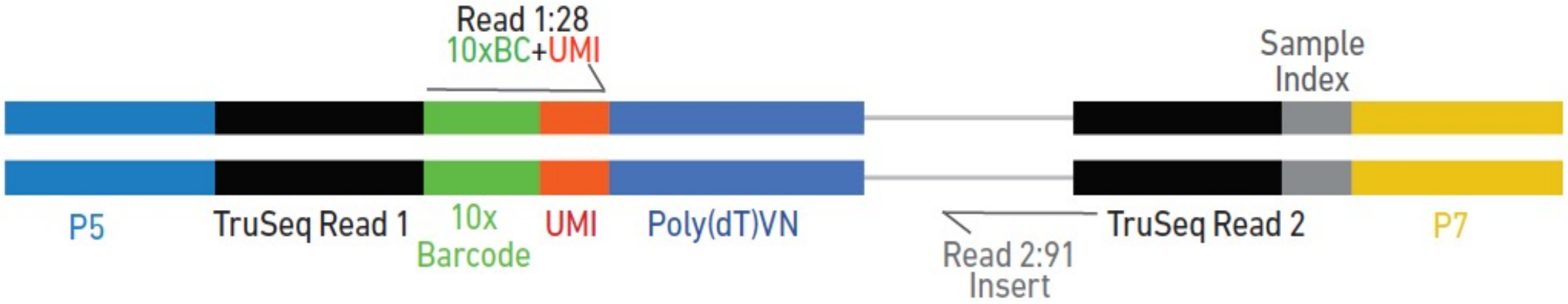




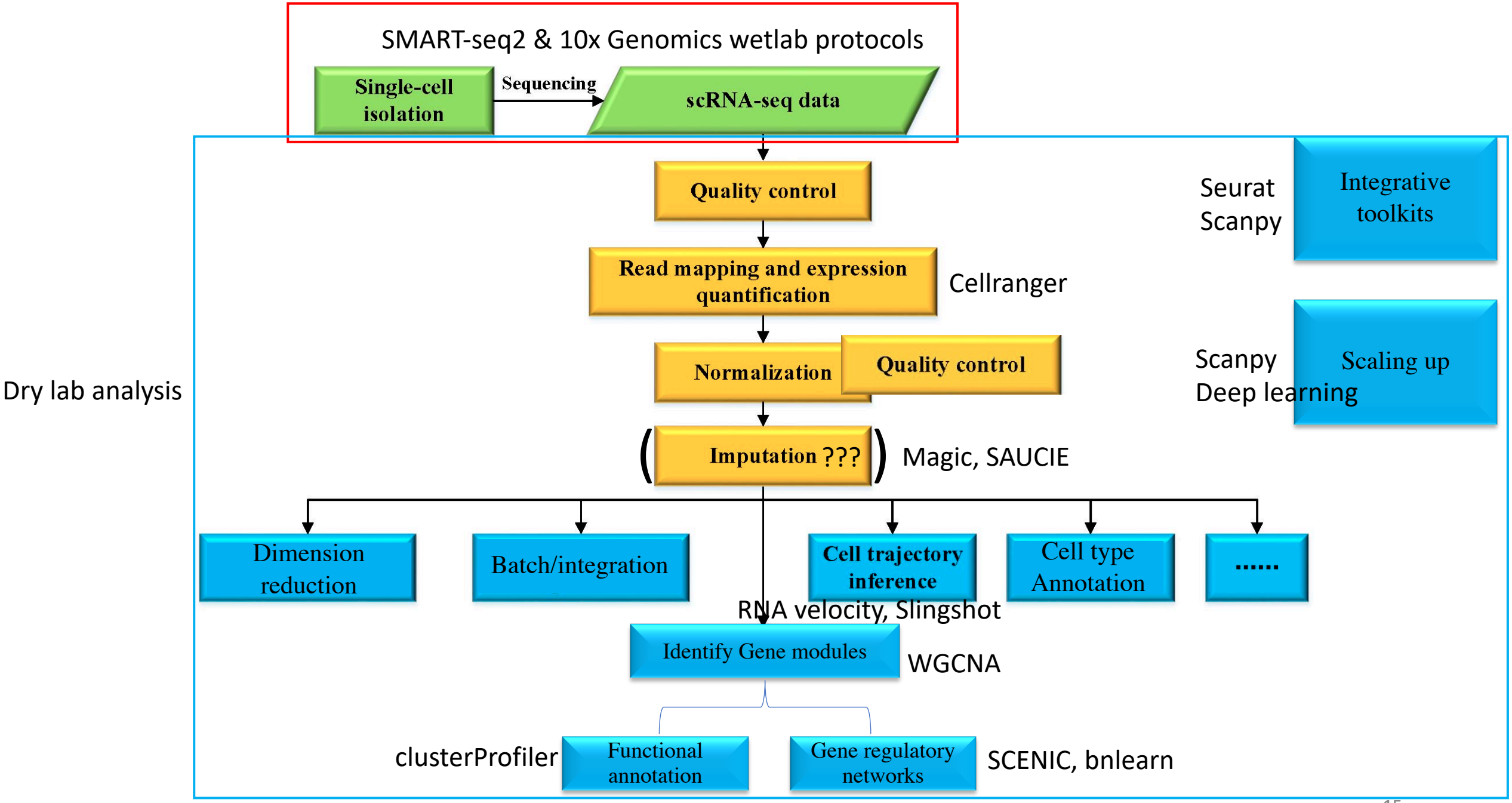
https://assets.ctfassets.net/an68im79xiti/4fIy9tr6qQuCWamIii0iEa/40658acce7a6756e38537584897840e3/CG000108_AssayConfiguration_SC3v2.pd

Next-seq reading the paired ends

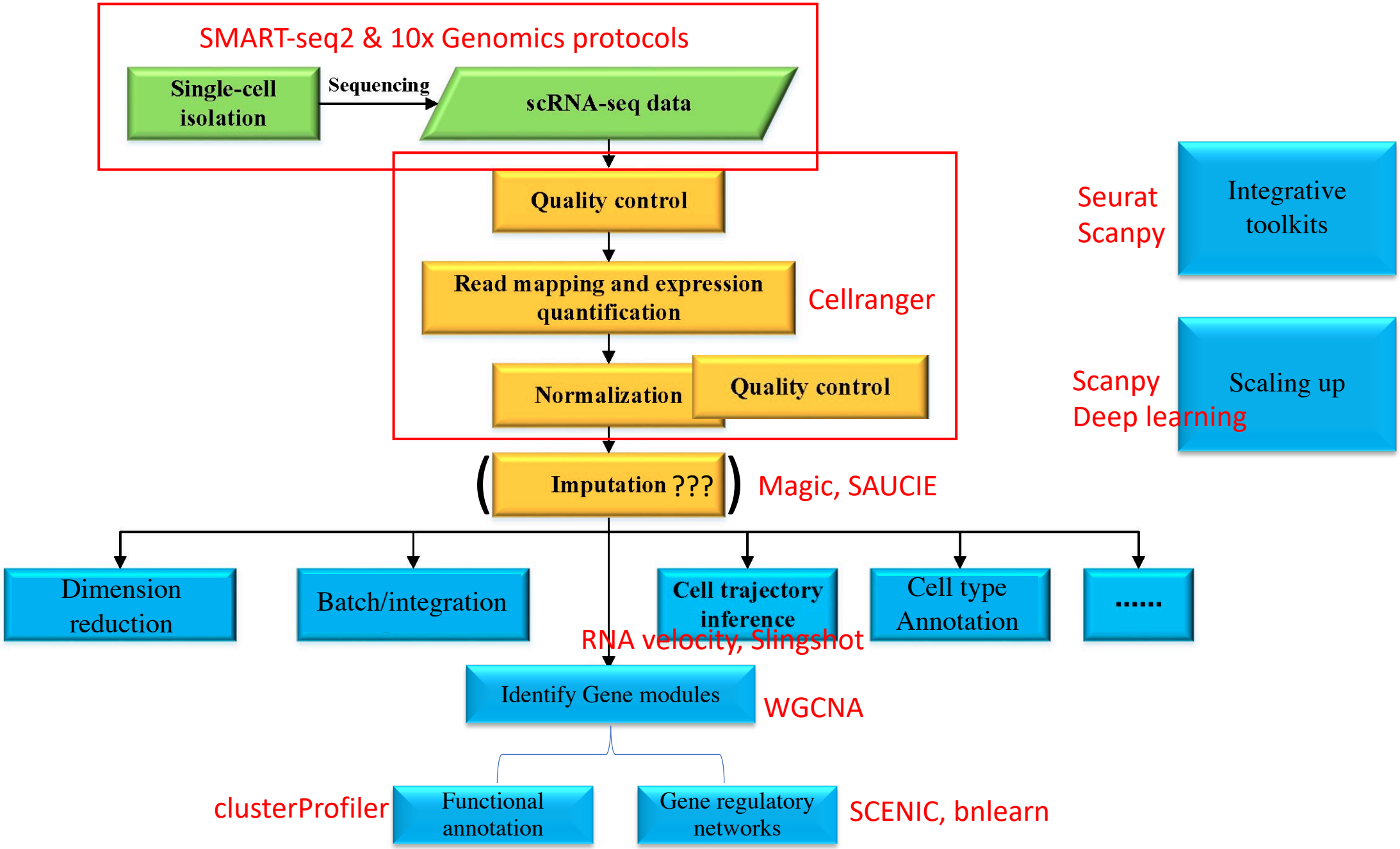
- 10XBC: 16 bp barcodes 2^16=65536 possible unique cells
- UMI, 2^12=4096 unique copies of mRNA can be distinguished for each gene
- Read2 will read into cDNA to identify the identity of the gene
- Sample barcode, identify the batch of your library sample
- All information will be summarized by the Cellranger software



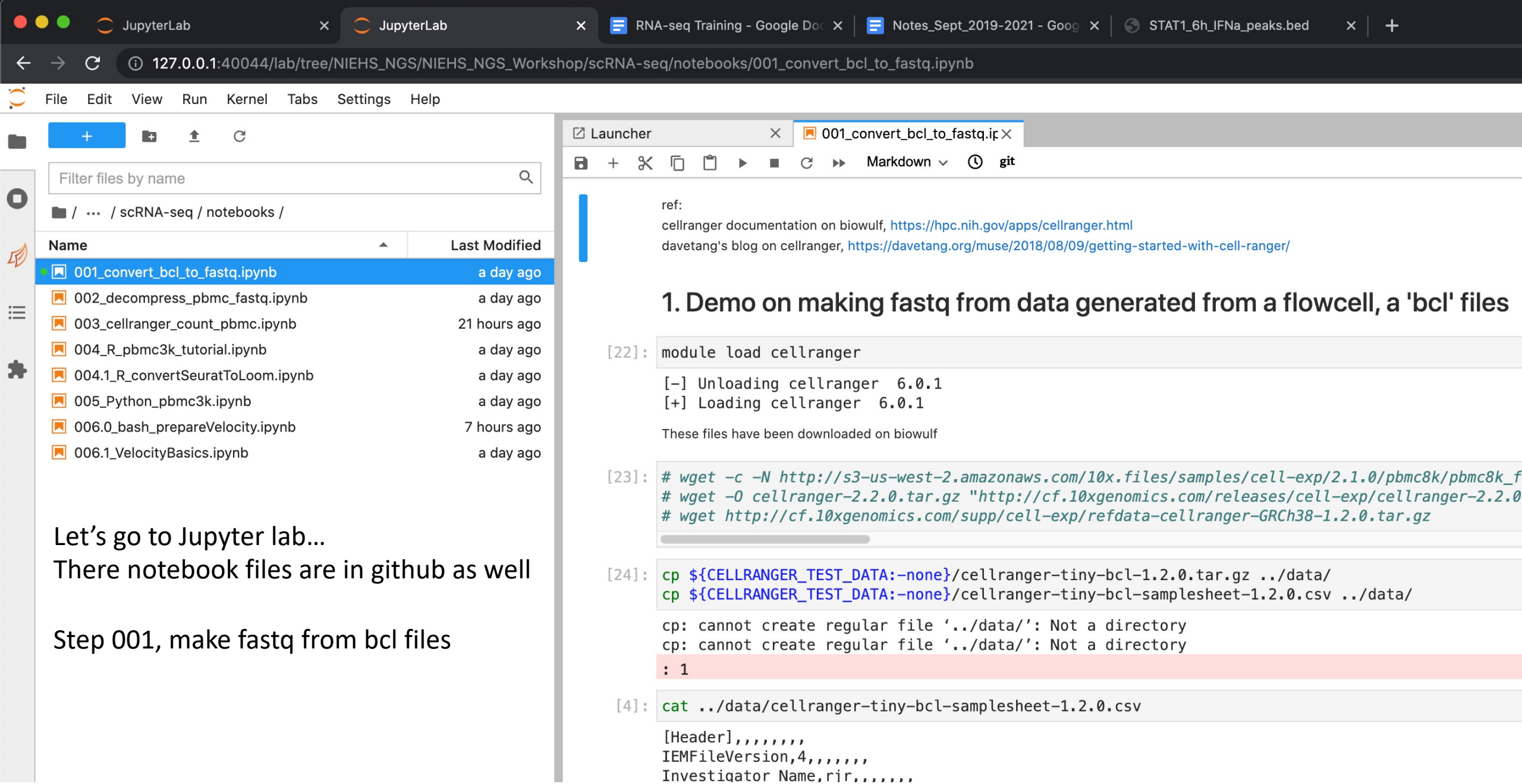
https://github.com/niaid/NGS_Intro/blob/master/notes/sequence_qc_class.md



https://www.frontiersin.org/files/Articles/441123/fgene-10-00317-HTML/image_m/fgene-10-00317-g001.jpg



https://www.frontiersin.org/files/Articles/441123/fgene-10-00317-HTML/image_m/fgene-10-00317-g001.jpg

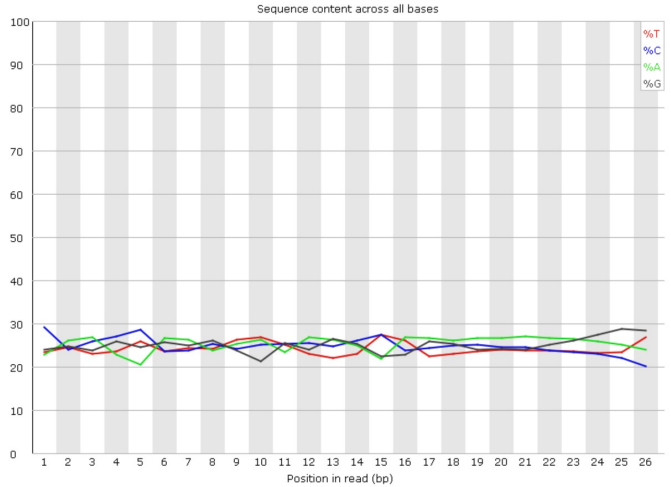


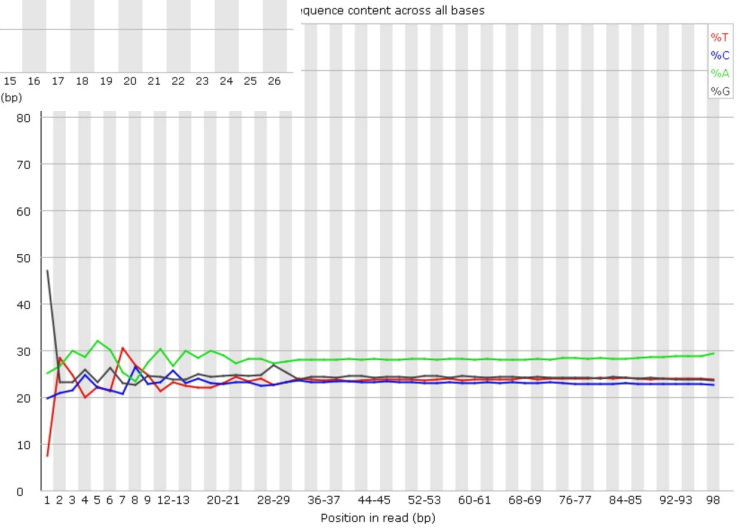
Let's go to Jupyter lab…
There notebook files are in github as well

Step 001, make fastq from bcl files

QC on sequencing results

```
scrALI001_S1_L001_I1_001.fastq.gz
scrALI001_S1_L001_R1_001.fastq.gz
scrALI001_S1_L001_R2_001.fastq.gz
```

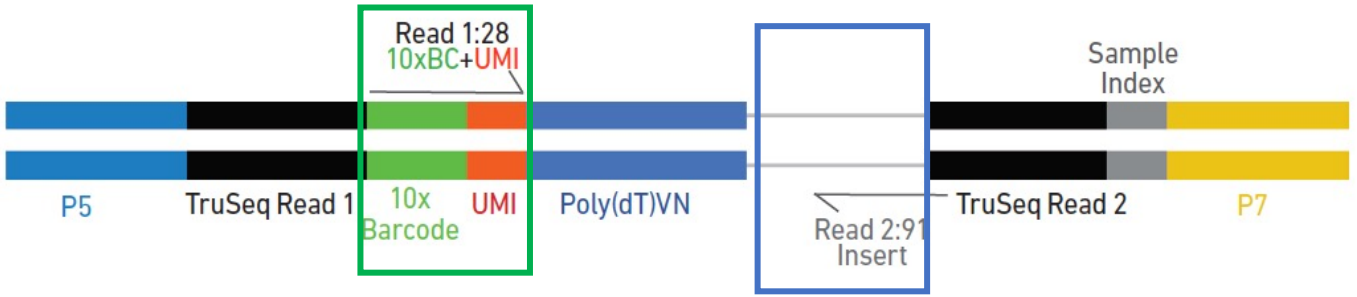




- I1
  - Index file. All identical (or one of 4) at Babraham
- R1
  - Barcode reads
    - 16bp cell level barcode
    - 10bp UMI

R2

- 3’ RNA-seq read



https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/using/count

Alignment results and Quality Controls

- Cellranger
  - cellranger mkfastq
  - cellranger count
  - cellranger aggr for multiple batches
- Quality controls
  - FASTQC on fastq files
  - Number of cells per experiment
  - Number of UMI per cell
  - Number of genes per cell
  - Percentage of mitochondrial reads
  - Removal of doublets/aggregates

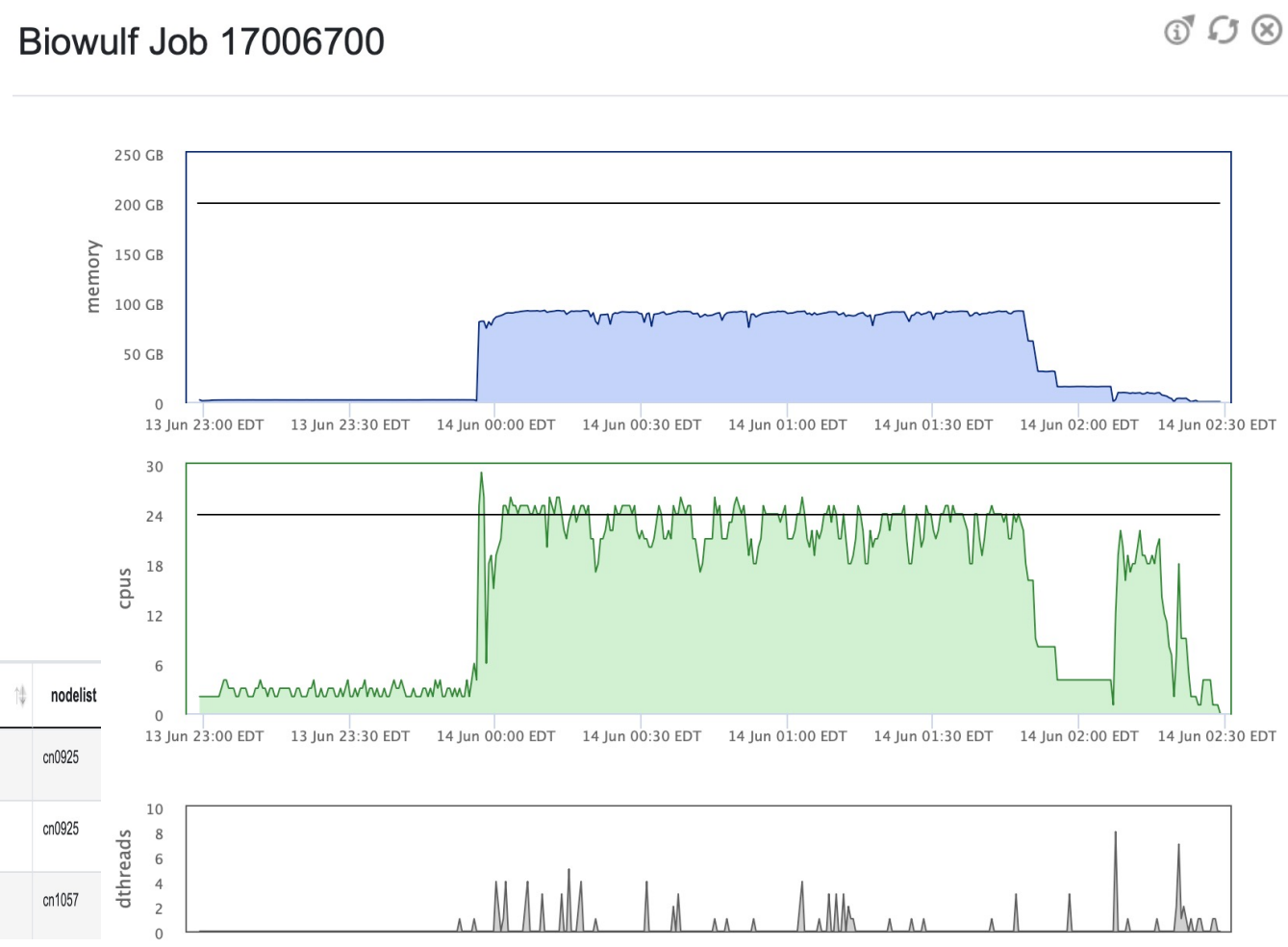
Monitoring jobs status on biowulf

Login to the user dashboard to monitor
https://hpc.nih.gov/dashboard/

jobid	jobname	state	statetime	nodelist
17006700	cellranger_pbmc3.sh	RUNNING	2021-06-13 22:58:42 EDT	cn0925
17006693	cellranger_pbmc.sh	FAILED	2021-06-13 22:57:30 EDT	cn0925
17006574	sinteractive	RUNNING	2021-06-13 22:45:00 EDT	cn1057

Sbatch jobs can be listed on terminal using
sjobs -u user_id

```
 sjobs -u zhuy16
User    JobId     JobName     Part         St  Reason  Runtime     Walltime     Nodes  CPUs   Memory  Dependency  Nodelist
=================================================================================================================
zhuy16  17006003  cellranger  norm         R             1:39:28      2:00:00      1     16  200 GB              cn3422
zhuy16  17006574  sinteracti  interactive  R               14:30      8:00:00      1      2  100 GB              cn1057
zhuy16  17006700  cellranger  norm         R                0:48     12:00:00      1     24  200 GB              cn0925
=================================================================================================================
```



jobname	cellranger_pbmc3.sh
user	zhuy16
submitted	2021-06-13 22:58:41 EDT
state	COMPLETED
submission script	/data/classes/NIEHS_NGS/RNA-workshop-2021/scRNA-seq/data/cellranger_pbmc3.sh
work directory	/data/classes/NIEHS_NGS/RNA-workshop-2021/scRNA-seq/data

Cellranger count report

FATSTQ

I1.FASTQ
R1.FASTQ
R2.FASTQ
} Cellranger →

report summary …

Estimated Number of Cells

15,894

Mean Reads per Cell	Median Genes per Cell
11,380	2,174

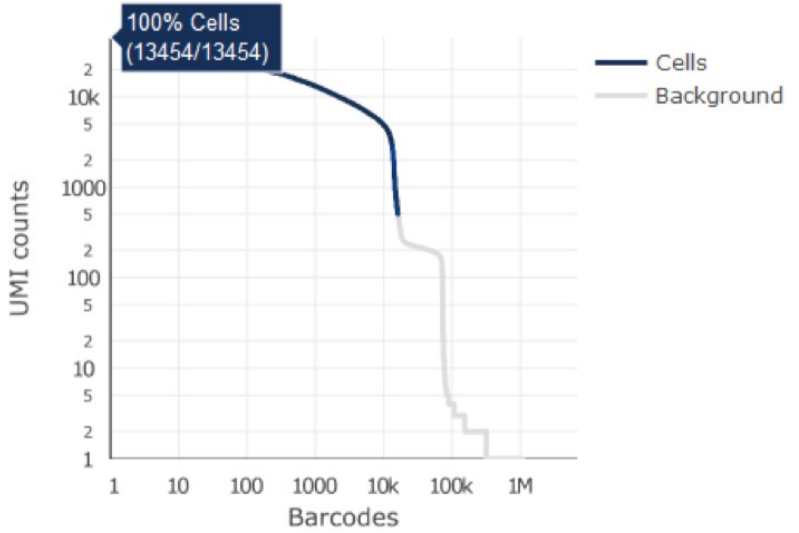
Sequencing

Number of Reads	180,878,636
Valid Barcodes	98.1%
Sequencing Saturation	10.3%
Q30 Bases in Barcode	98.4%
Q30 Bases in RNA Read	82.7%
Q30 Bases in UMI	98.7%

Mapping

Reads Mapped to Genome	95.4%
Reads Mapped Confidently to Genome	90.2%
Reads Mapped Confidently to Intergenic Regions	3.0%
Reads Mapped Confidently to Intronic Regions	12.8%
Reads Mapped Confidently to Exonic Regions	74.4%
Reads Mapped Confidently to Transcriptome	71.9%
Reads Mapped Antisense to Gene	0.9%

Cells



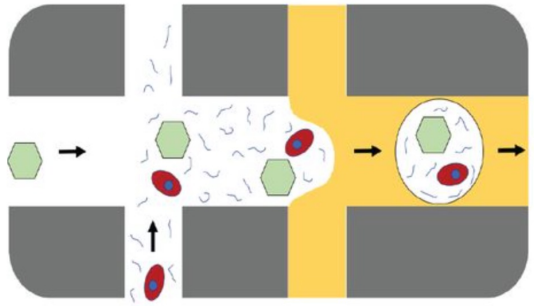
Estimated Number of Cells	15,894
Fraction Reads in Cells	88.1%
Mean Reads per Cell	11,380
Median Genes per Cell	2,174
Total Genes Detected	20,185
Median UMI Counts per Cell	5,742

Sample

Name	embryoid_d4
Description	
Transcriptome	mm10
Chemistry	Single Cell 3' v3
Cell Ranger Version	3.0.2

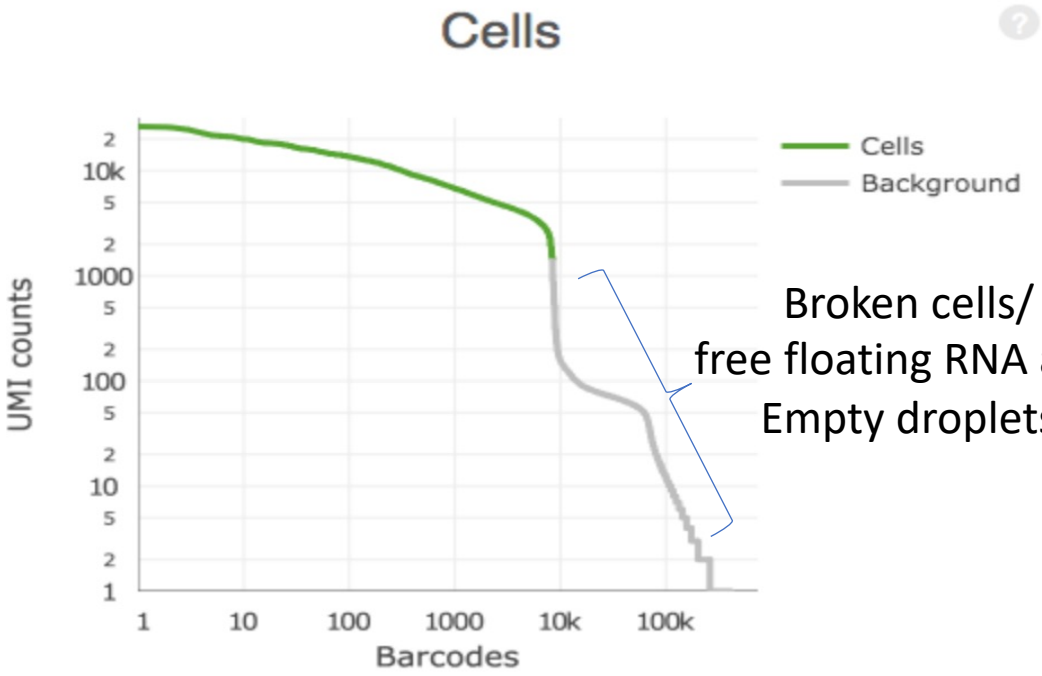
cellranger count output

- Reads level
  - Cellranger
    - cellranger mkfastq
      - Generate fastq files from image ".bcl" files
    - **cellranger count → sparse matrix**
      - umi, unique molecule identifier
    - cellranger aggr
      - Combine count data from multiple batches
  - (For CITE-seq and HASH-tag)
    - Cite-seq-count



Output:

```
$ cd /home/jdoe/runs/sample345/outs
$ tree filtered_feature_bc_matrix
filtered_feature_bc_matrix
├── barcodes.tsv.gz     --cells
├── features.tsv.gz     --genes
└── matrix.mtx.gz       --sparse matrix
0 directories, 3 files
```



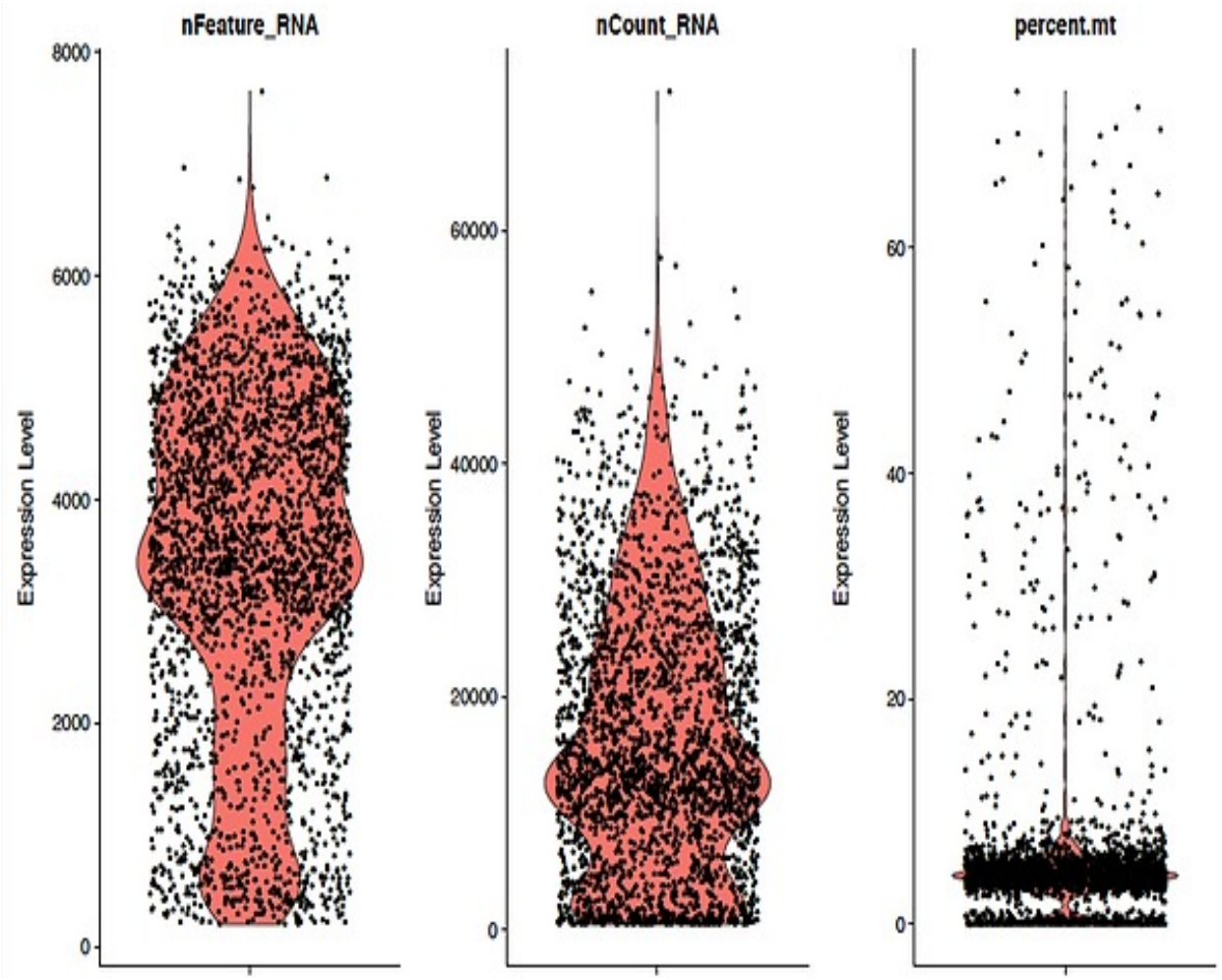
Broken cells/
free floating RNA and
Empty droplets

Ranked by number of associated UMIs

https://davetang.org/muse/2018/08/09/getting-started-with-cell-ranger/

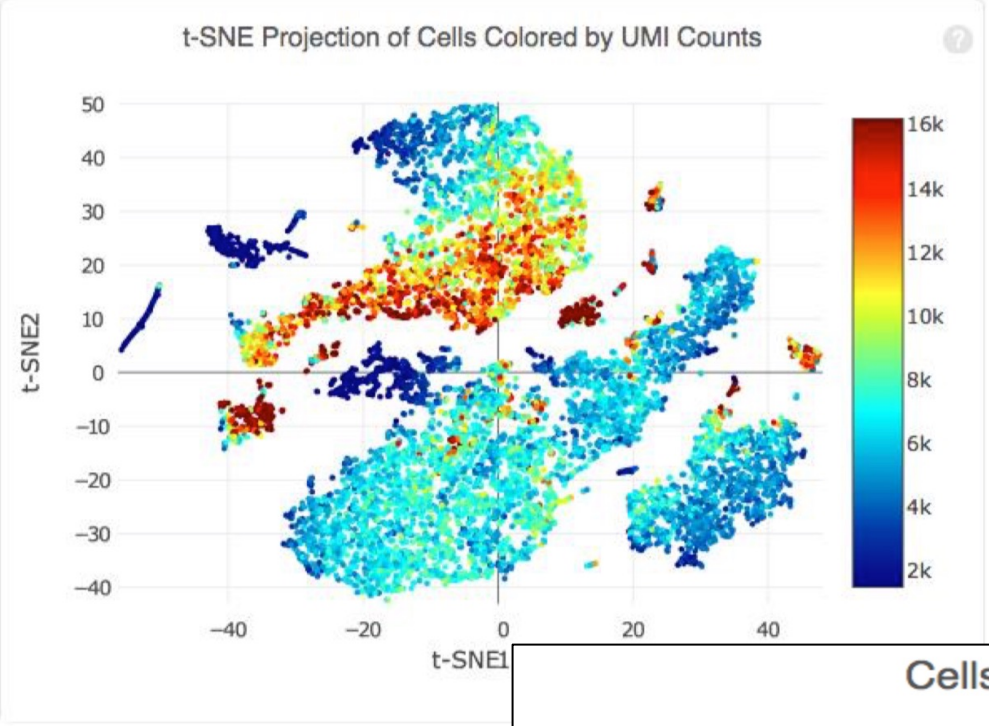
Seurat -- Quality Controls

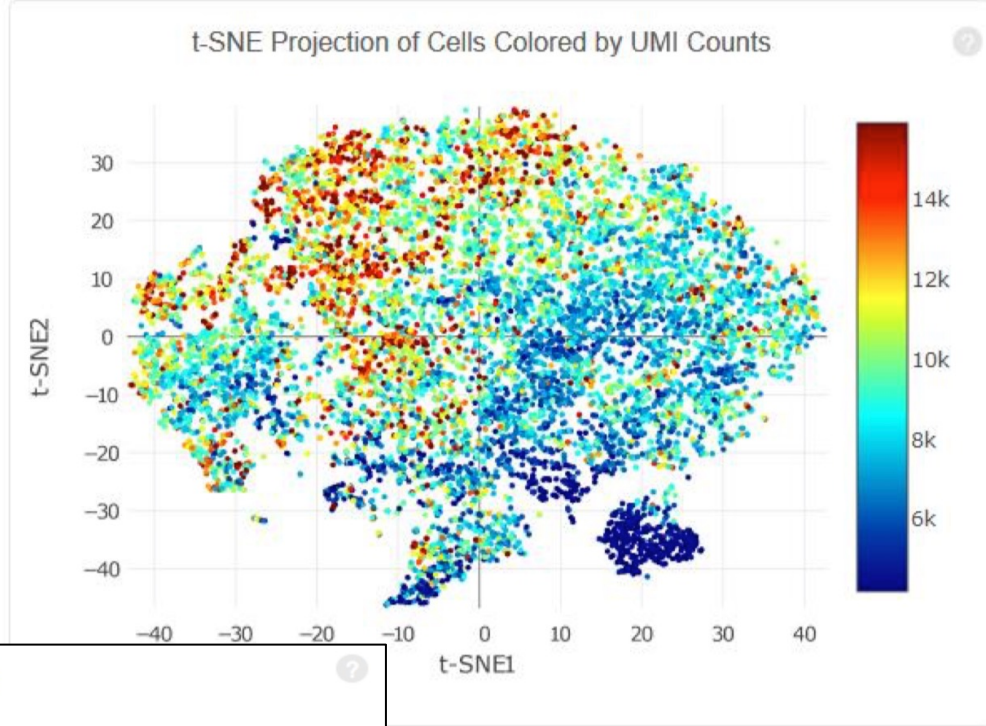
- Quality of reads – FASTQC
- Percentage of mitochondrial reads
  - Too many mitochondria reads may indicate that cells are dying/dead/broken
  - Case-by-case, the range may vary with method of library prep methods/cell type
- How many cells are you capturing?
  - Typically few thousands in each 10X run
- The sequencing depth
  - Are they acceptable in the field (minimal 2,000 reads /cell?) determined by the Cellranger
- Alignment to the genome and exons
  - Should be 90-100% to the genome
  - A reasonably narrow range 70-80% to the exons
  - (Could be 30% to exons if you use nucleus, which contain lots of introns)
- Expected markers expressed?
  - Highly expressed genes, cell type markers, automatic detection such as scMCA etc
- Be prepared to see differences between RNA (because of the depth and dropouts) and proteins.
- Confounding factors?
  - Batch effect?
  - Is your dimension reduction capturing biological or technical variations?
  - Can be evaluated by WGCNA and visualization in PCA or tSNE

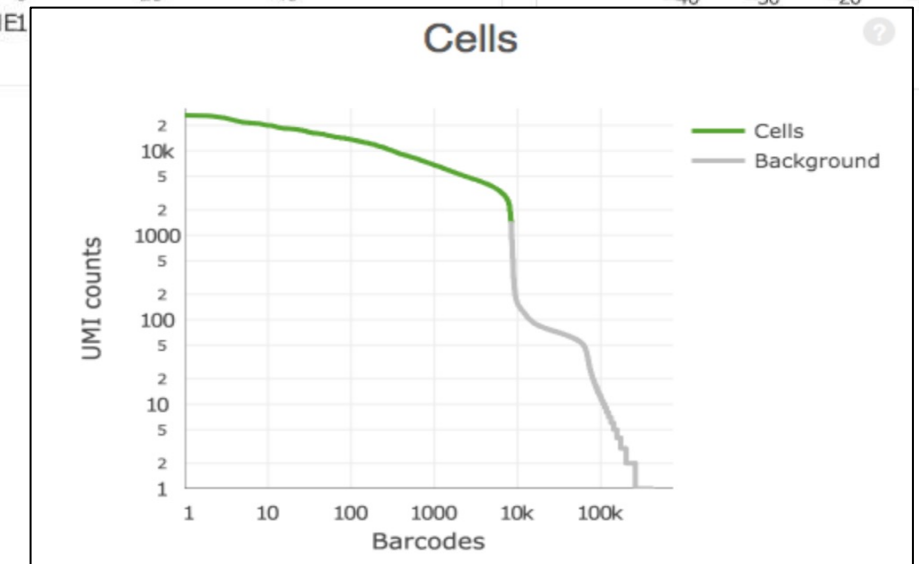


https://www.biostars.org/p/377422/

Is coverage variation affecting your data?



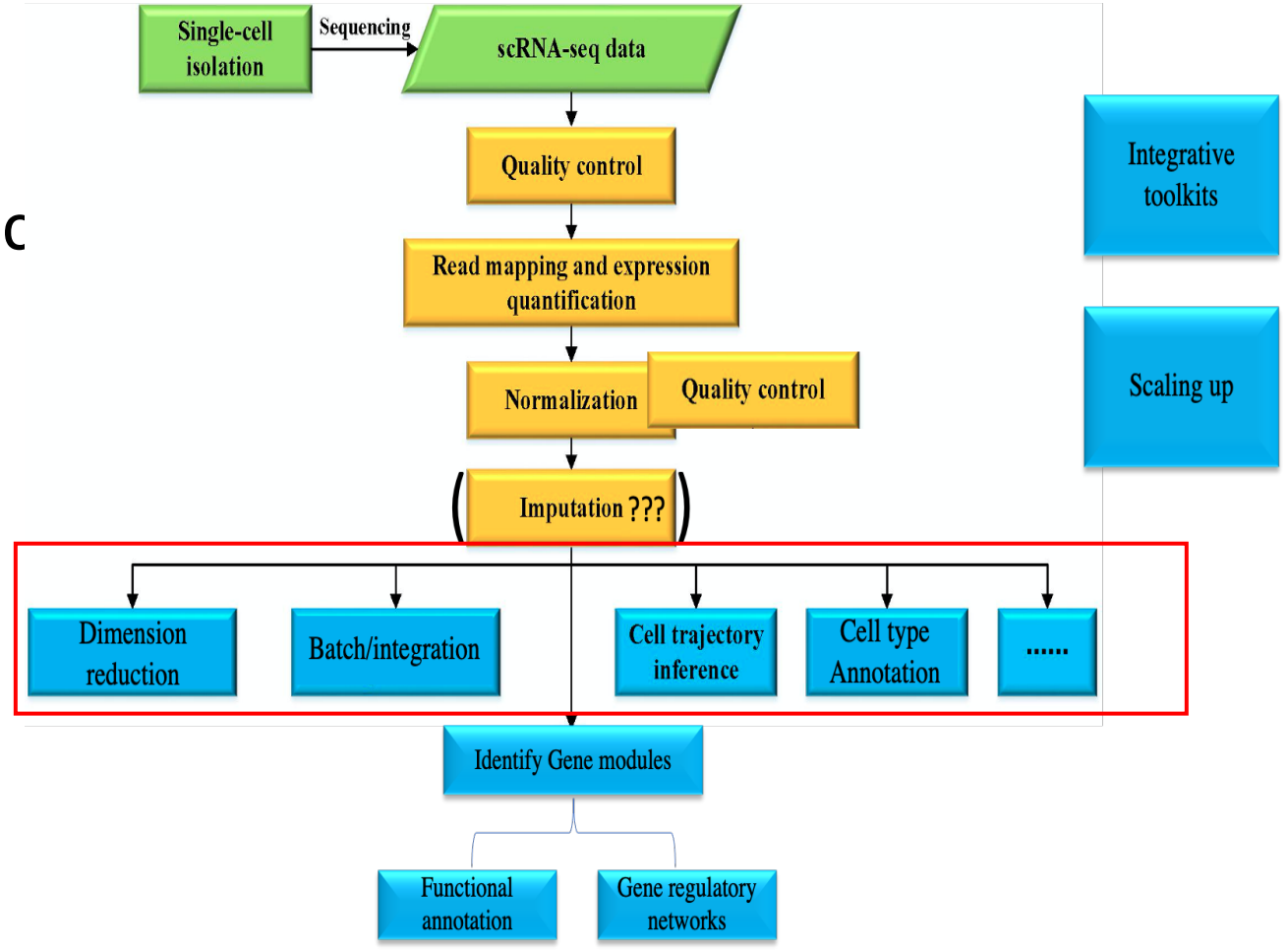




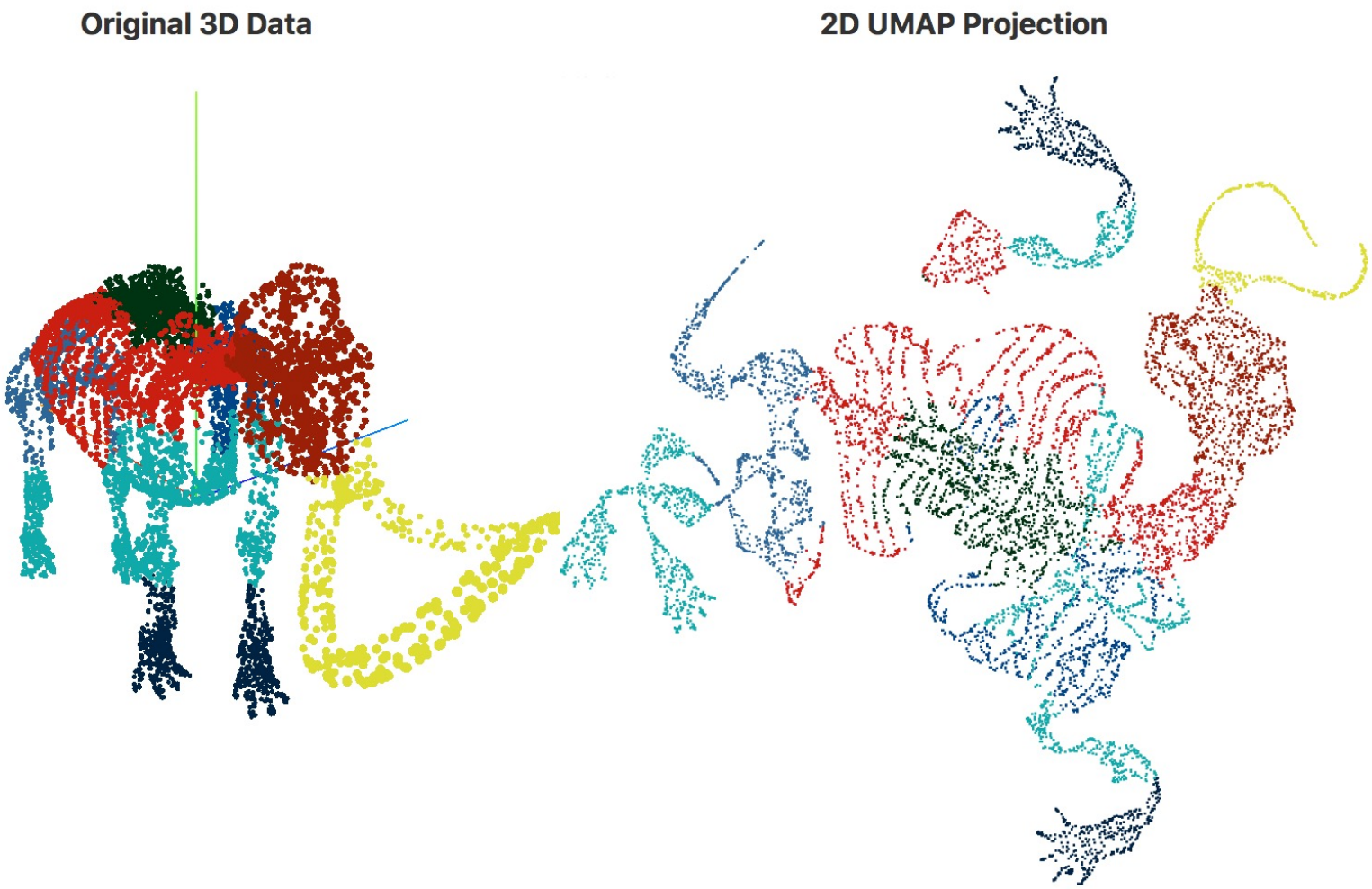
UMI dominate the variance?
Need to do batch correction?
Is the level of separation enough/expecte
(to much background RNA?

Cell-based analysis

- Clustering and annotate the biologic identities of clusters
- Inference of trajectory
- Batch correction/data integration.
- comprehensive workflow/toolkits



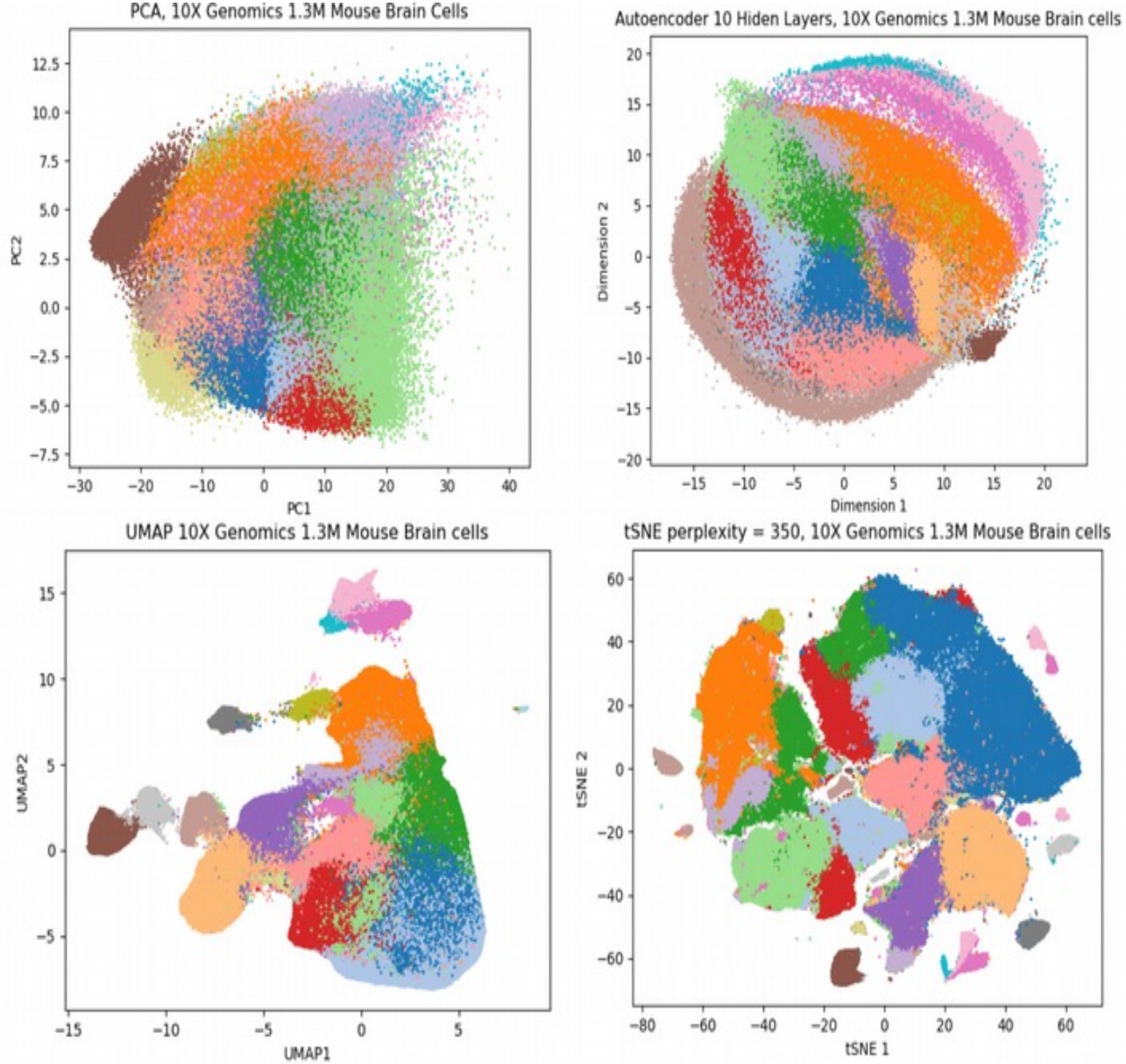
Dimension reduction--an intuitive illustration



Screenshot by Mariam from
https://pair-code.github.io/understanding-umap/

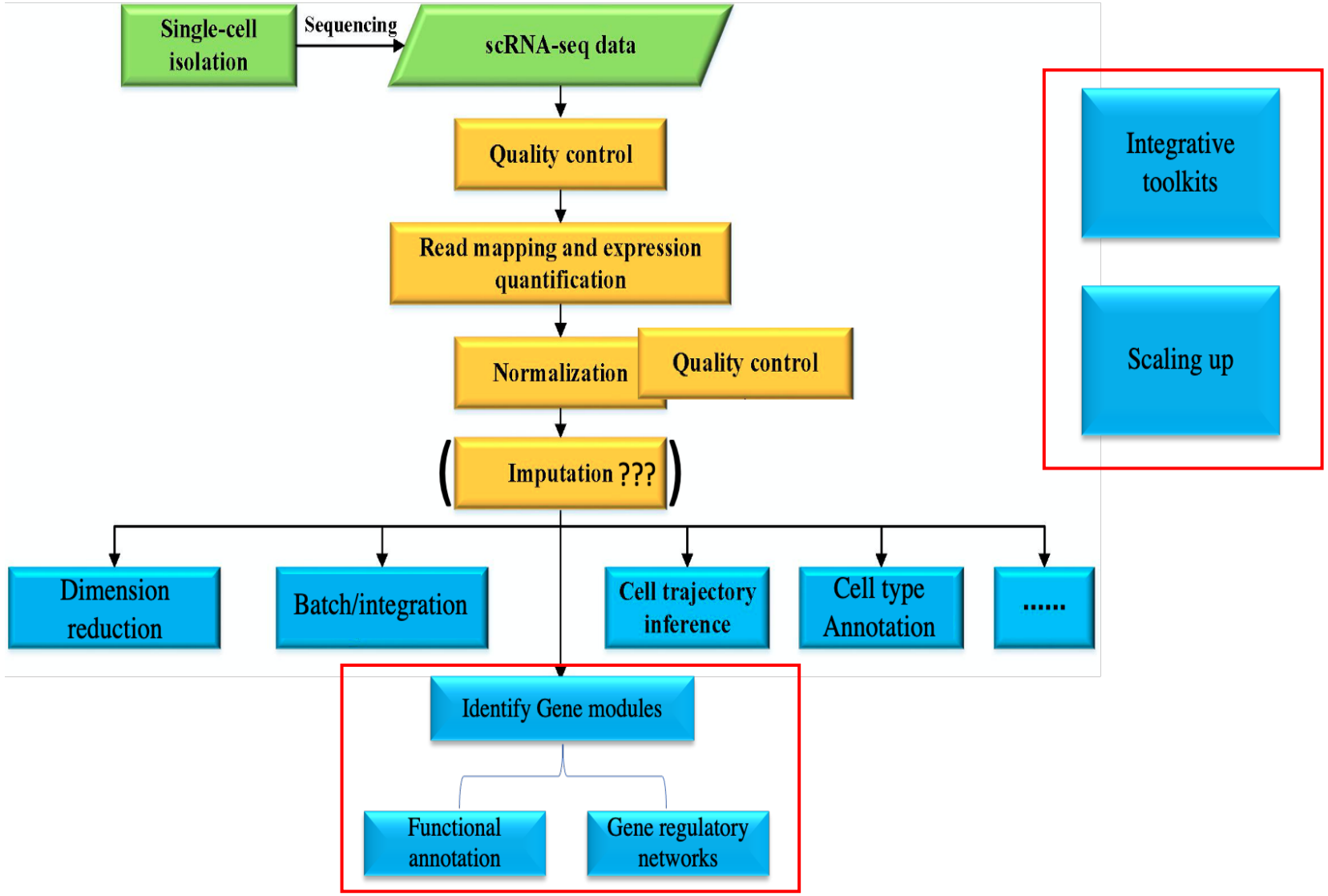
Dimension reduction

- Dimension reduction
  - PCA
    - Linear reduction
    - Based on distances
    - 2D structure in PCA depends on certain observed dominant variations
    - Often not sufficient for large number of cells
  - tSNE
    - Non-linear reduction
    - Attention to local similarity
    - Global shape is less meaningful
    - Add new data changes the whole pattern
  - UMAP
    - Consider both global and local structure
    - Learnt embeddings can be saved for new batch of data
  - AutoEncoder
    - Fast algorithm to handle up to millions of cells



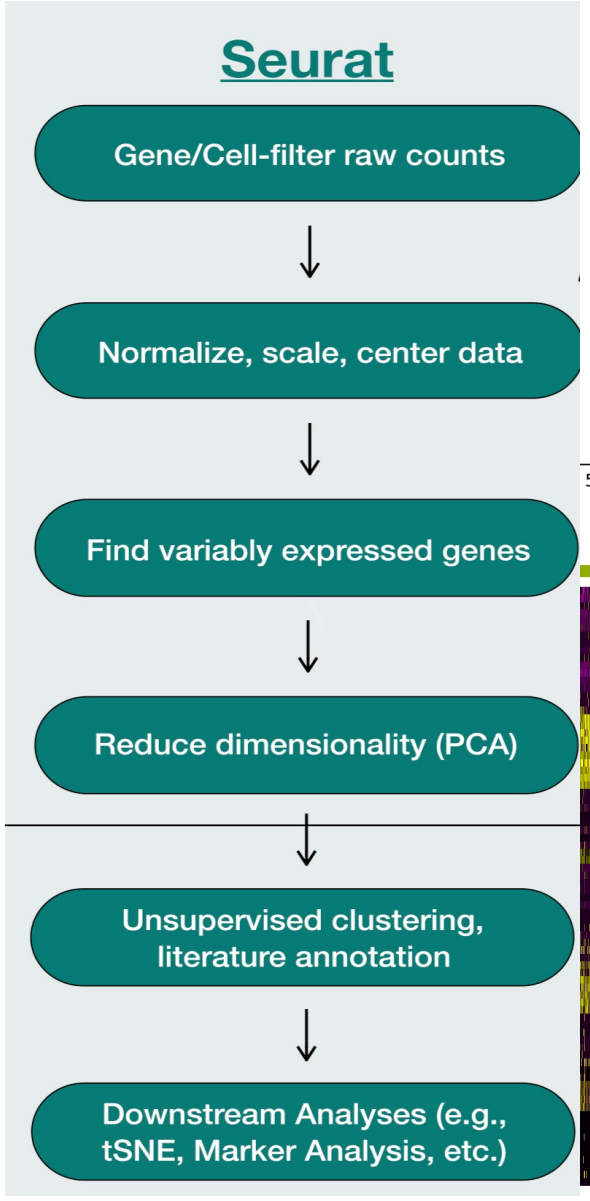
https://towardsdatascience.com/deep-learning-for-single-cell-biology-935d45064438

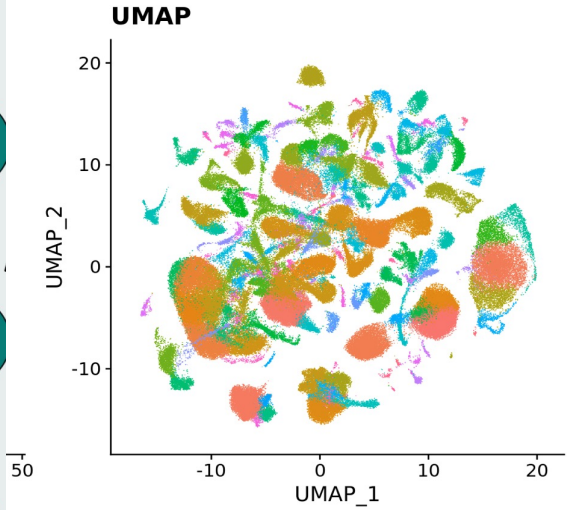
Comprehensive toolkits

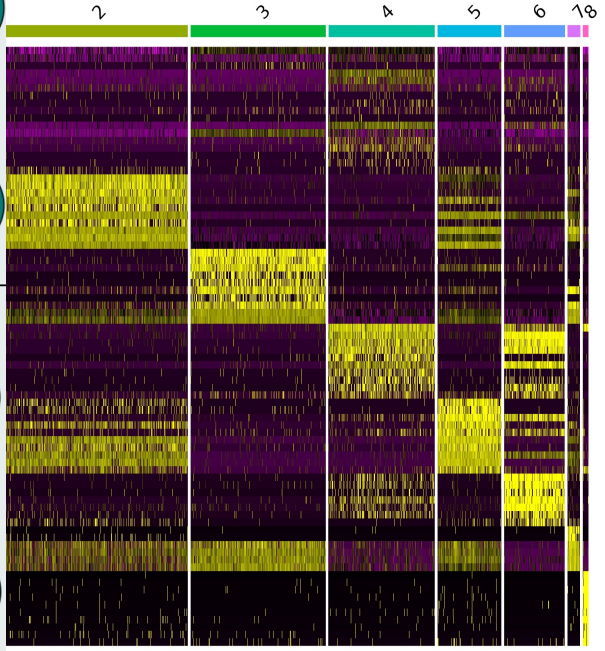


Comprehensive pipeline tools for explorative analysis -- Seurat

- Seurat pipeline
  - General QC assessment
  - Cell type annotation
  - Batch correction and meta analysis
  - Multimodal analysis (for CITE-seq, Hash-tagging, ATAC-seq)
  - Comparative analysis across different conditions







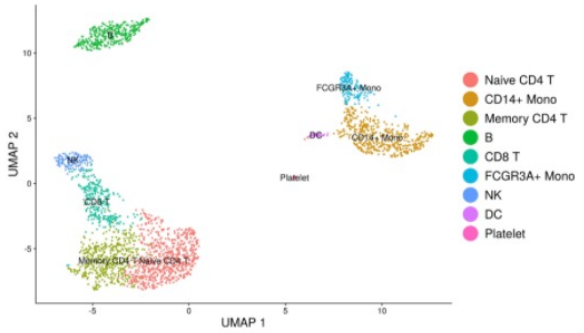
https://satijalab.org/seurat/vignettes.html

Multiple vignettes for different tasks

Basic pipeline: QC,
Dimension reduction
Clustering
Marker identifications

Integrating with CITE-seq, HASH-seq

Guided tutorial — 2,700 PBMCs

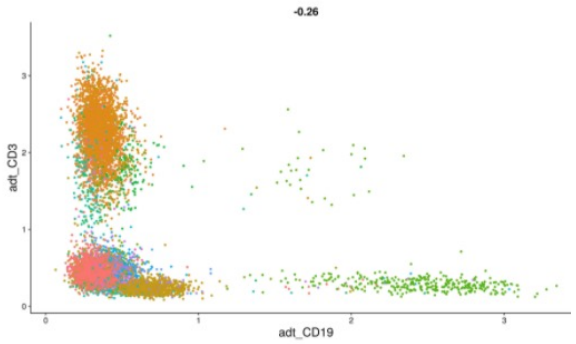


A basic overview of Seurat that includes an introduction to common analytical workflows.

Start from here

GO

Multimodal analysis

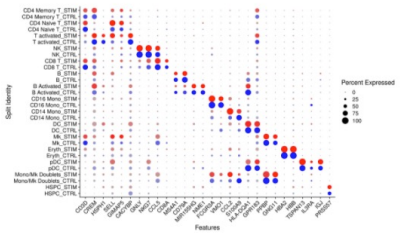


An introduction to working with multi-modal datasets in Seurat.

GO

Multiple pipelines for integrating data

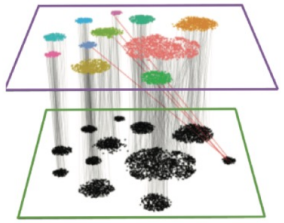
Introduction to scRNA-seq integration



An introduction to integrating scRNA-seq datasets in order to identify and compare shared cell types across experiments

GO

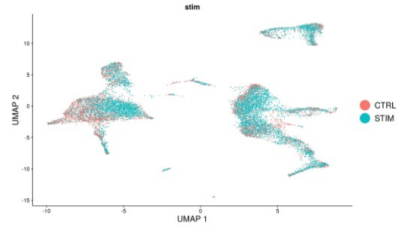
Mapping and annotating query datasets



Learn how to map a query scRNA-seq dataset onto a reference in order to automate the annotation and visualization of query cells

GO

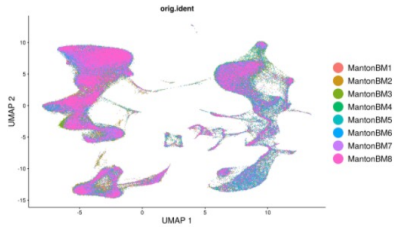
Fast integration using reciprocal PCA (RPCA)



Identify anchors using the reciprocal PCA (rPCA) workflow, which performs a faster and more conservative integration

GO

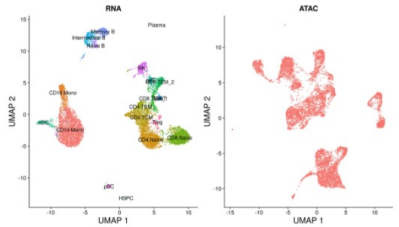
Tips for integrating large datasets



Tips and examples for integrating very large scRNA-seq datasets (including >200,000 cells)

GO

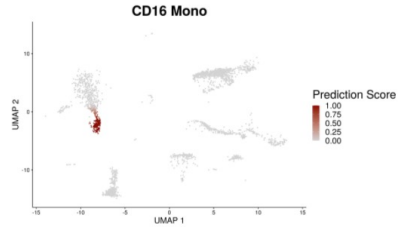
Integrating scRNA-seq and scATAC-seq data



Annotate, visualize, and interpret an scATAC-seq experiment using scRNA-seq data from the same biological system

GO

Multimodal Reference Mapping

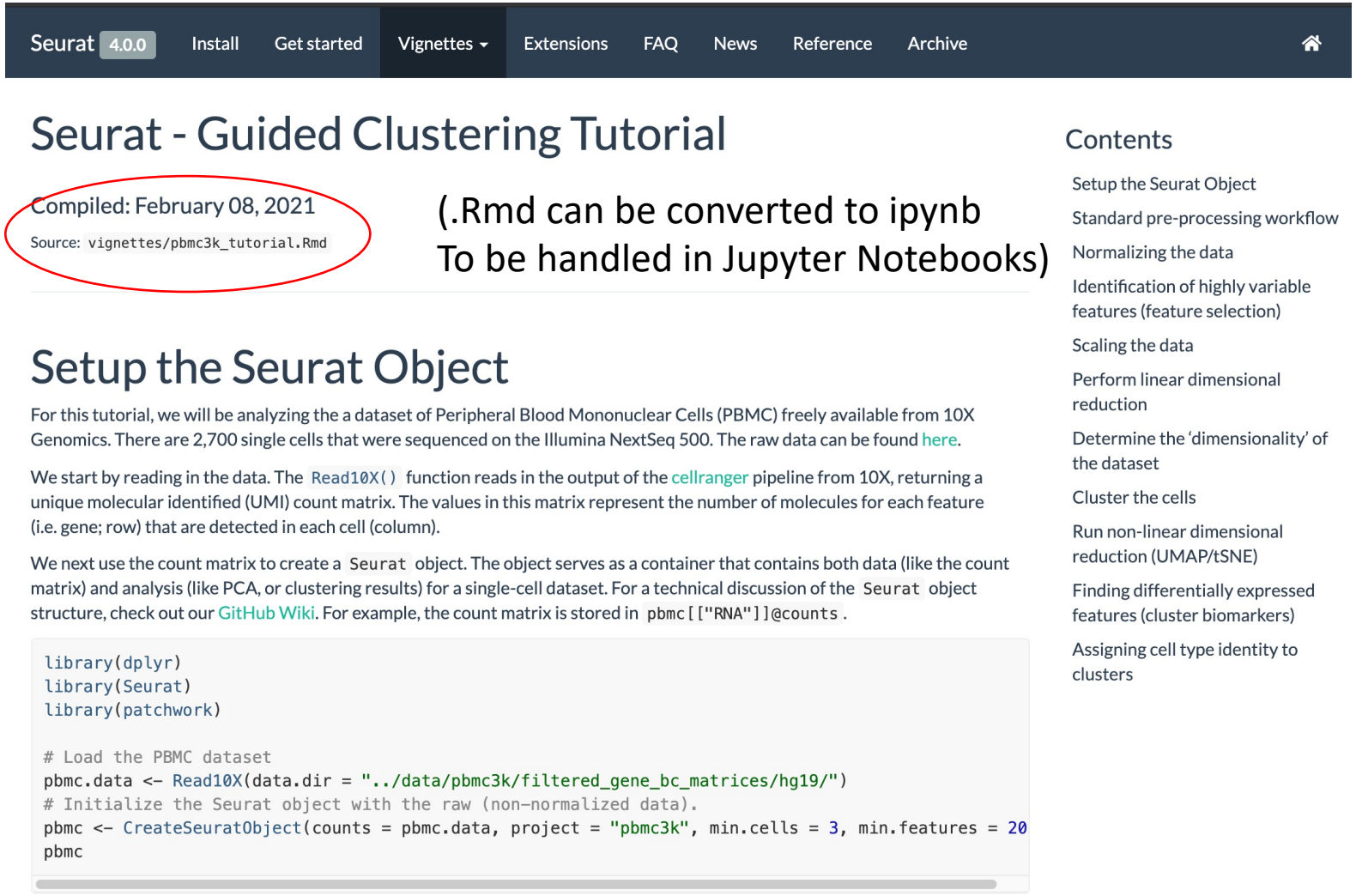


Analyze query data in the context of multimodal reference atlases.

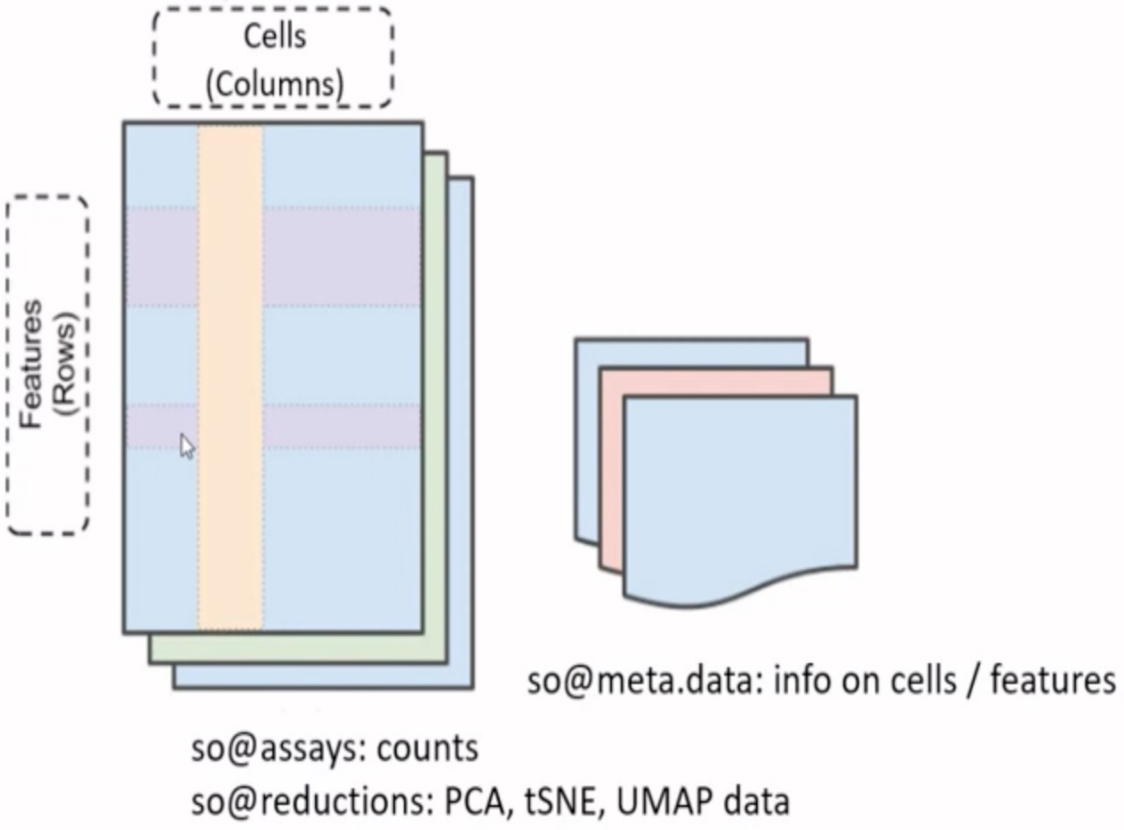
GO

Download vignette/tutorial to use on your data

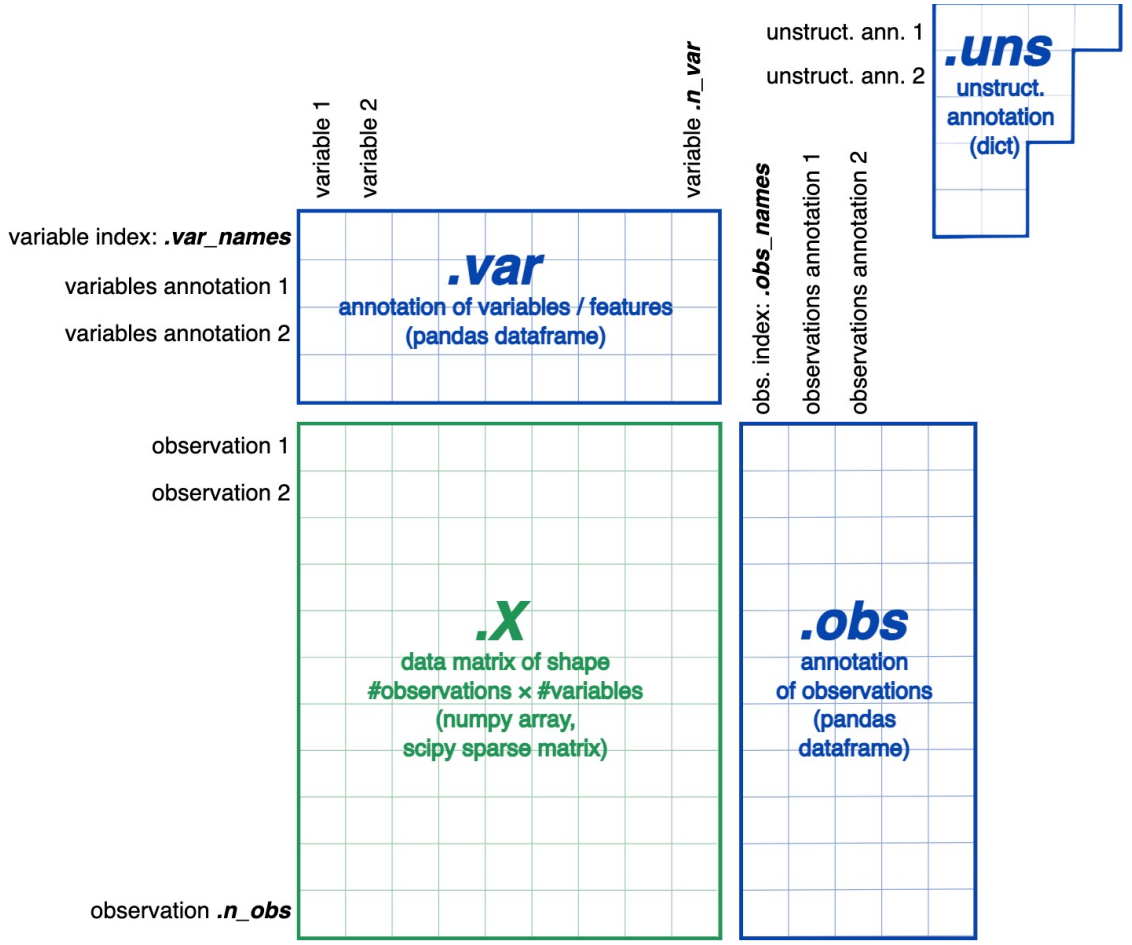
- Linked to github for you to download the code.
- Follow through the tutorial using sample data included in the package.
- Change the input file to your our data to use the analytic pipelines.



R vs Python, Object-Oriented Programming (OOP)



Seurat Object in Seurat



Anndata Object in Scanpy

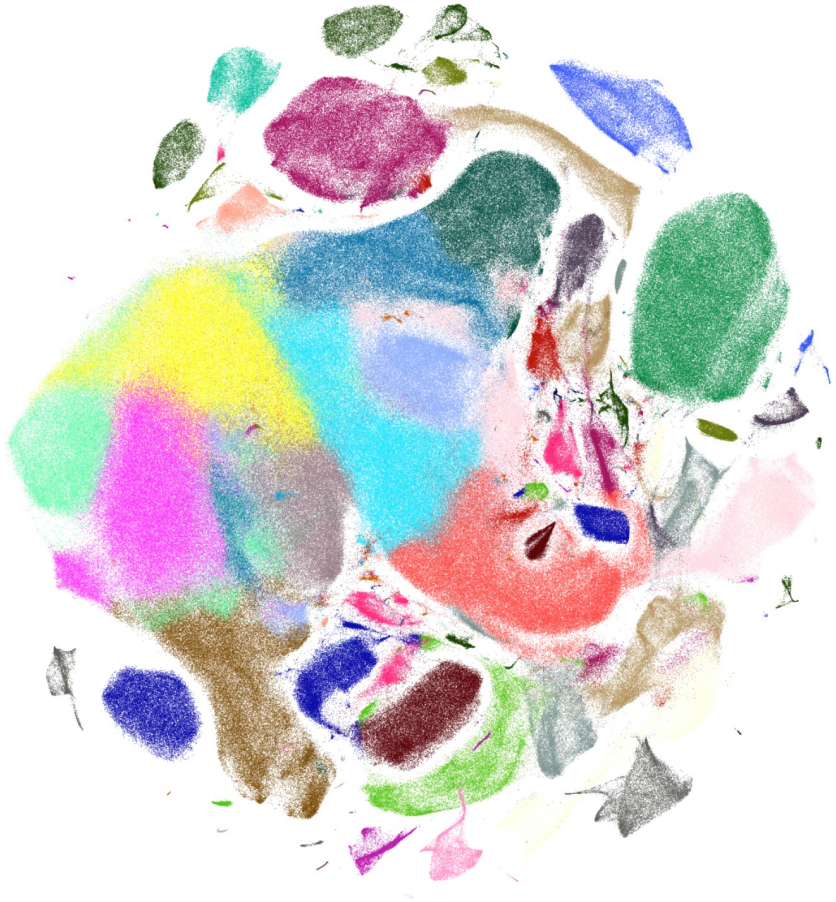
Scale up with python implementations

- Python packages/toolkits are increasingly popular
  - scanpy pipeline
  - scVelo pipeline
- Some has a R rapper.

scanpy in R

- Use python in R through Reticulate
- Use R in python through rpy2

tSNE plot of
1300000 neurons by scanpy



Scanpy vs. Seurat

Satija et al., Nat. Biotechn. (2015)

Scanpy is benchmarked with Seurat.

- preprocessing: <1 s vs. 14 s
- regressing out unwanted sources of variation: 6 s vs. 129 s
- PCA: <1 s vs. 45 s
- clustering: 1.3 s vs. 65 s
- tSNE: 6 s vs. 96 s
- marker genes (approximation): 0.8 s vs. 96 s

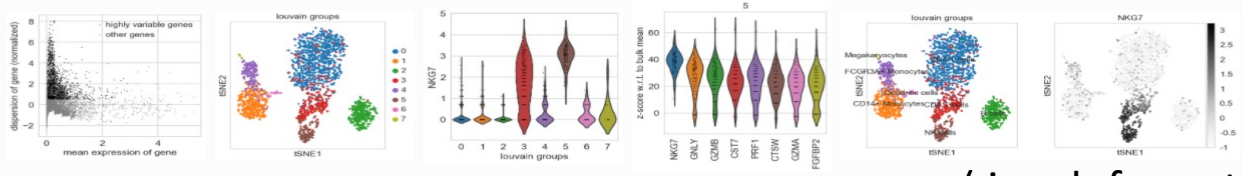
https://scanpy.readthedocs.io/en/stable/tutorials.html

- Same data set, similar analysis with Seurat.
- But implemented in python.
- Using jupyter notebook as a interface.

Tutorials

Clustering

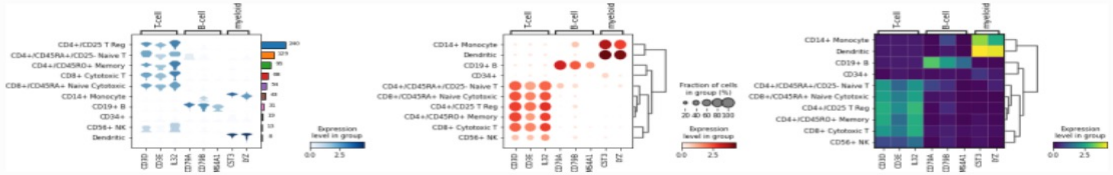
For getting started, we recommend Scanpy's reimplementation → tutorial: pbmc3k of Seurat's [Satija15] clustering tutorial for 3k PBMCs containing preprocessing, clustering and the identification of cell types via known marker genes.



(.ipynb format can be converted to .RMD to be run in Rstudio)

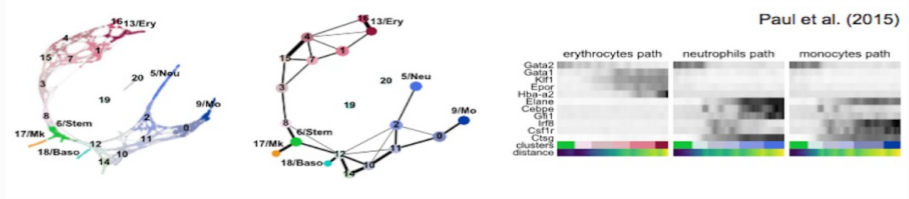
Visualization

This tutorial shows how to visually explore genes using scanpy. → tutorial: plotting/core



Trajectory inference

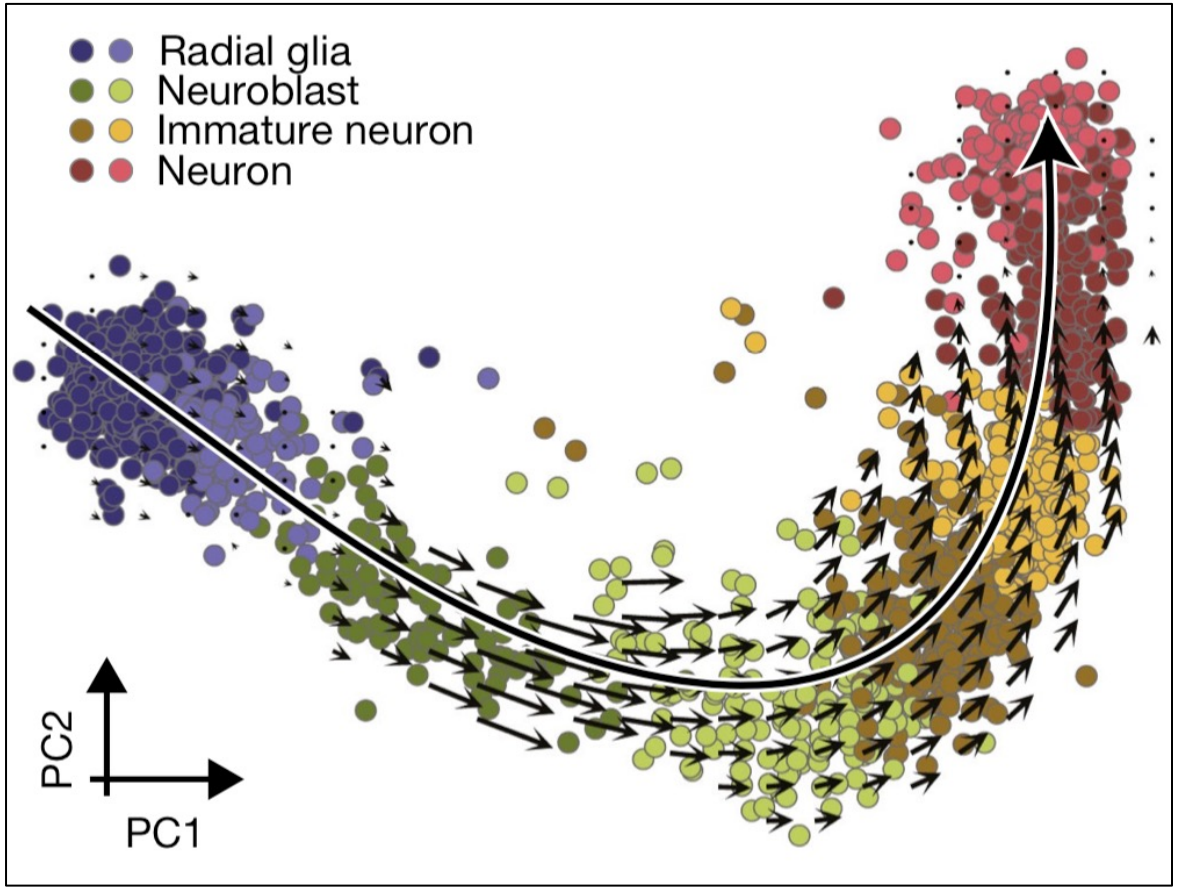
Get started with the following example for hematopoiesis for data of [Paul15]: → tutorial: paga-paul15



More examples for trajectory inference on complex datasets can be found in the PAGA repository [Wolf19], for instance, multi-resolut

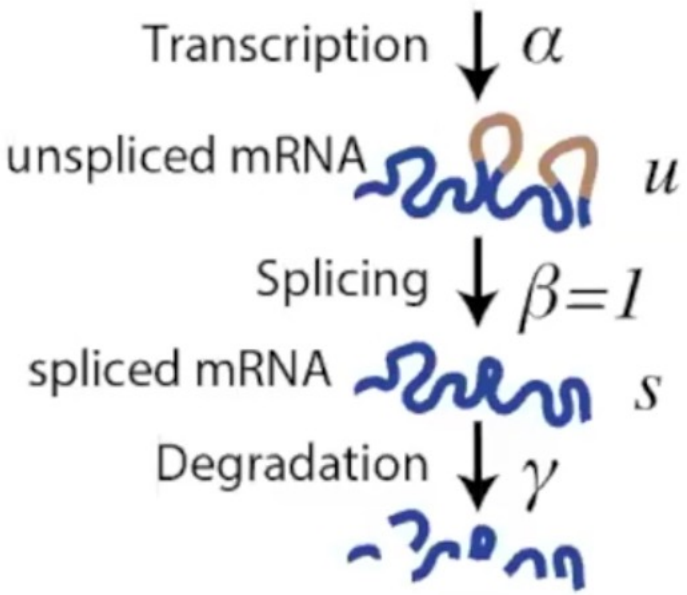
Trajectory analysis by RNA velocity

- When cells differentiate, **new genes** will start to be expressed
- Transcripts have introns and will be spliced off given time
- Through assessing the present percentage of reads in introns, increase or decrease of expression can be modeled

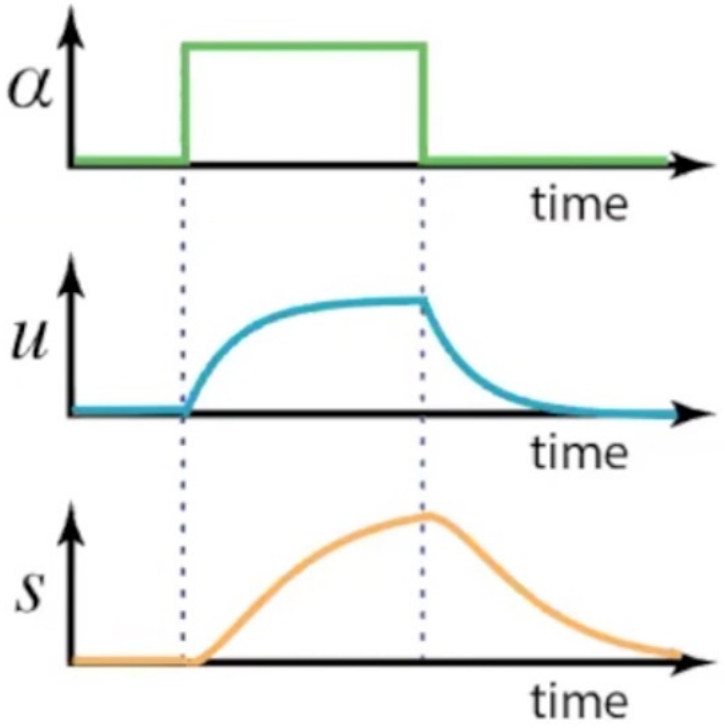


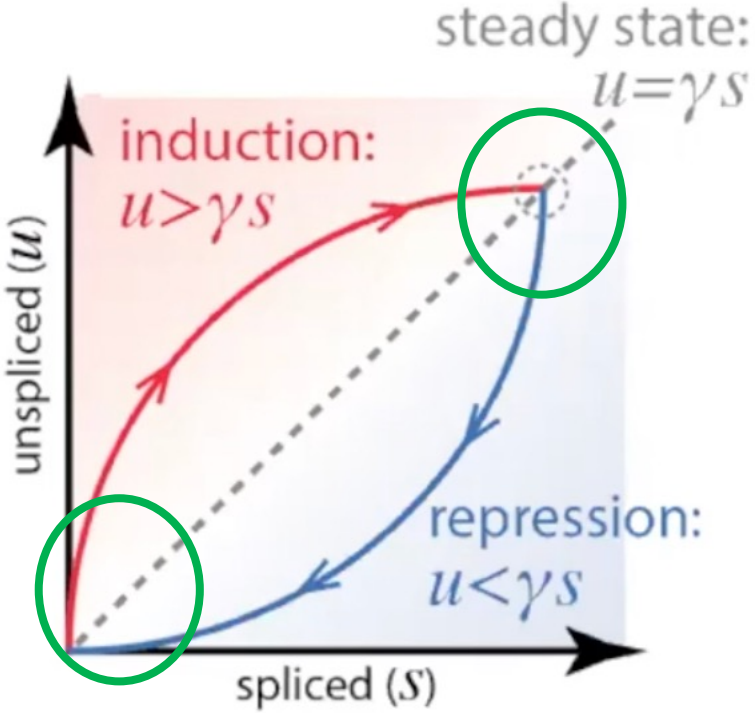
https://liorpachter.wordpress.com/tag/velocyto/
http://pklab.med.harvard.edu/software.html

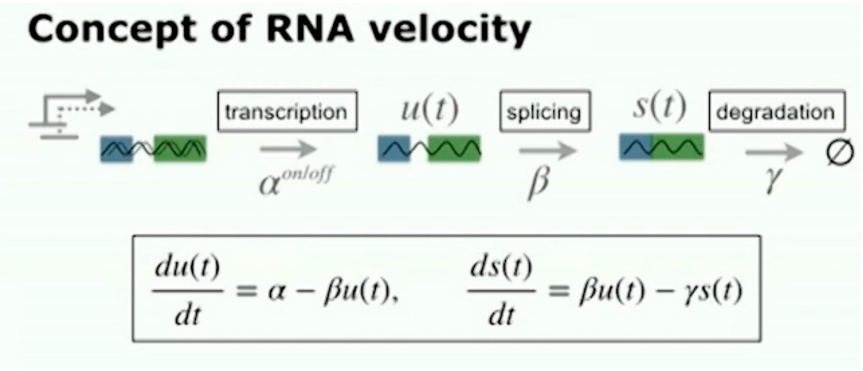
Modeling RNA dynamics

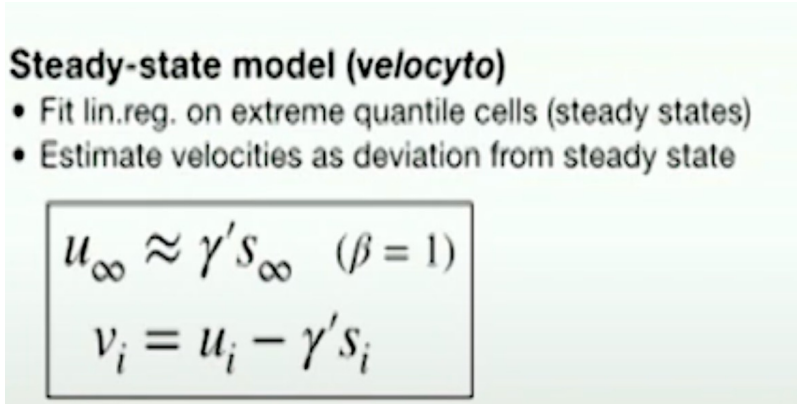


$$s'(t)=u-\gamma s$$



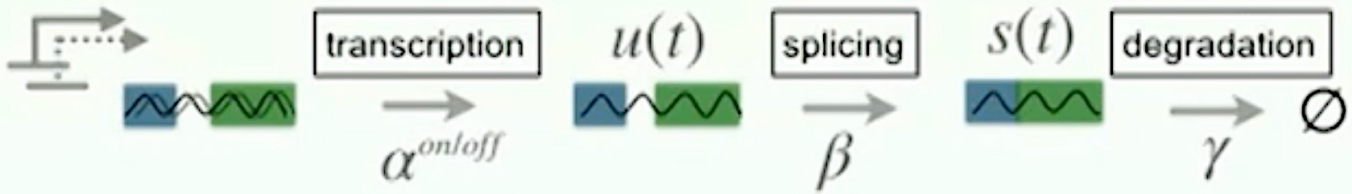






2 assumptions:
steady states has been observed
a constant splicing rate β across all RNA

Generalizing RNA velocity to dynamical populations



$$u(t) = u_0 e^{-\beta\tau} + \frac{\alpha}{\beta}(1 - e^{-\beta\tau})$$

$$s(t) = s_0 e^{-\gamma\tau} + \frac{\alpha}{\gamma}(1 - e^{-\gamma\tau}) + \frac{\alpha - \beta u_0}{\gamma - \beta}(e^{-\gamma\tau} - e^{-\beta\tau}) \qquad \tau = t - t_0$$

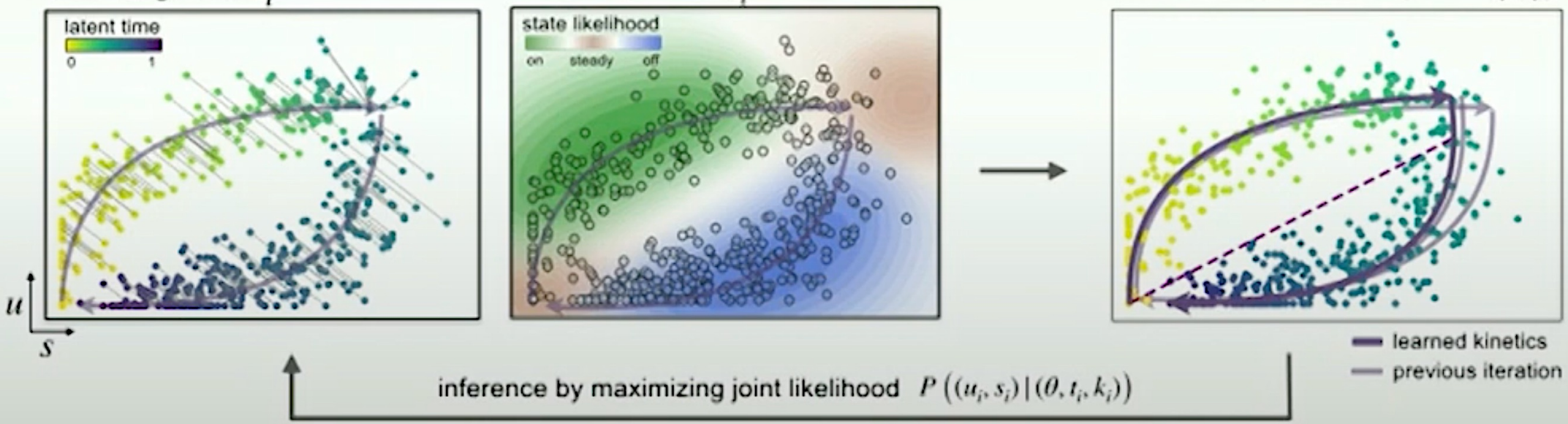
parameters of **reaction rates** $\theta = (\alpha^{off}, \alpha^{on}, \beta, \gamma)$

cell-specific **latent variables** (switch, time, state) $\eta_i = (t_0^{(i)}, t_i, k_i)$

time assignment t_i

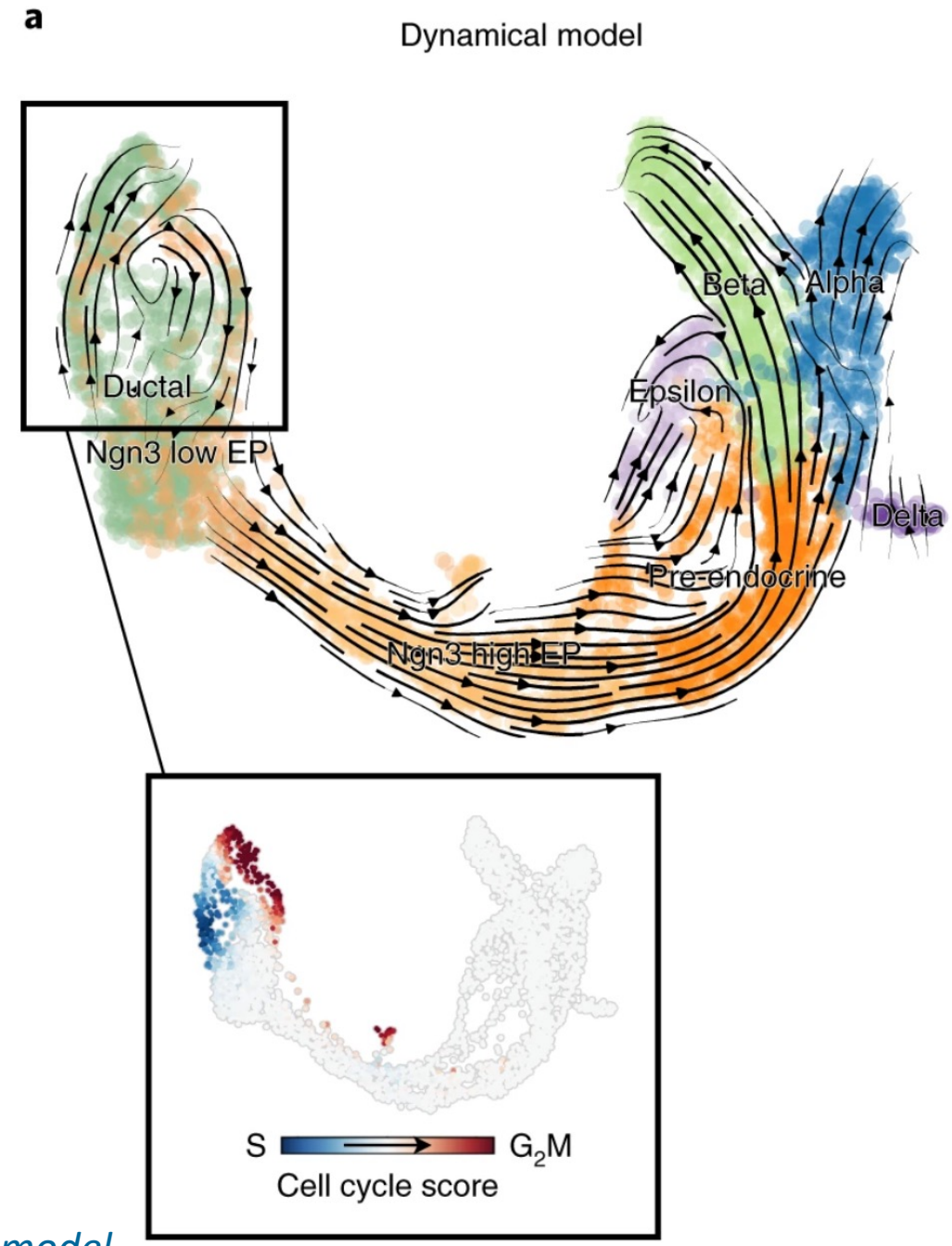
state assignment k_i

parameter update $\theta = (\alpha^{on/off}, \beta, \gamma)$



Steps of RNA-velocity

- From cellranger produced bam files
  - Sort bam files
  - Using Velocyto CLI to identify exonal/intronal reads as loom files.
  - Use scVelo to model the data and recover RNA dynamics
  - Through Markov process to predict which neighbor is the most probabal destiny for he cell.
  - Backtracking and forward tracking to get the destiny and ancestor of the cell.



Nature, 2018. invented the concept
Nature Biotechnology, 2020. a generalized model

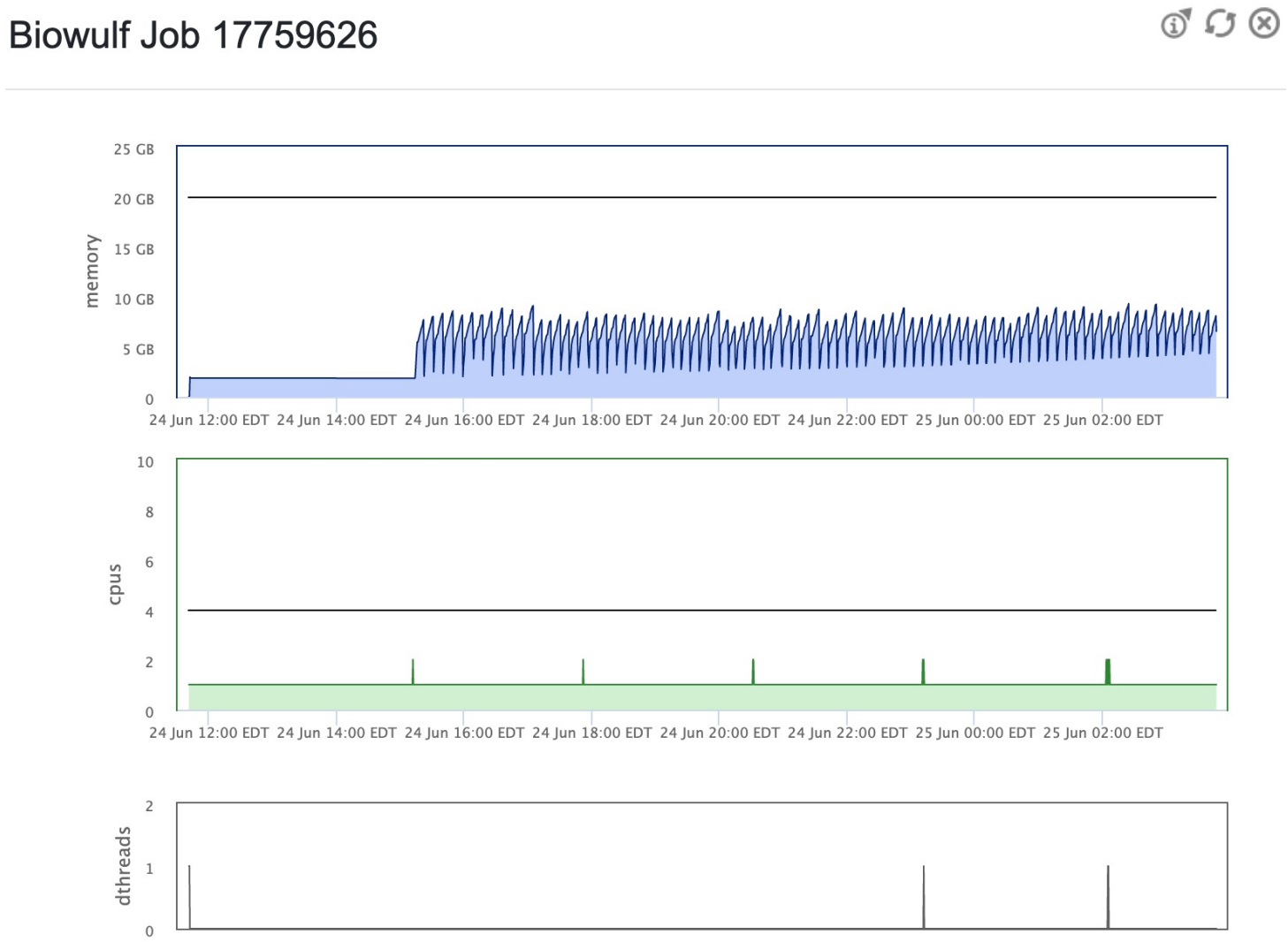
Get information about spliced/unspliced transcripts from bam files

```
#Build pipe line and shared scripts and instruction to researchers
https://github.com/zhuy16/single-cell-RNA-seq/tree/master/scvel_notebooks


Step 1, finding reads mapped to un-spliced regions.

#Sort BAM files with Barcodes
samtools sort -t CB -O BAM -
o cellsorted_possorted_genome_bam.bam
possorted_genome_bam.bam

#Analyze reads from spliced/non-spliced
transcripts,
module load python/3.7.3-foss-2016b
velocyto run10x  cellranger_output_folder  genes.gtf
```

```
cellsorted_possorted_genome_bam.bam.tmp.0000.bam
cellsorted_possorted_genome_bam.bam.tmp.0001.bam
cellsorted_possorted_genome_bam.bam.tmp.0002.bam
cellsorted_possorted_genome_bam.bam.tmp.0003.bam
cellsorted_possorted_genome_bam.bam.tmp.0004.bam
cellsorted_possorted_genome_bam.bam.tmp.0005.bam
cellsorted_possorted_genome_bam.bam.tmp.0006.bam
cellsorted_possorted_genome_bam.bam.tmp.0007.bam
cellsorted_possorted_genome_bam.bam.tmp.0008.bam
cellsorted_possorted_genome_bam.bam.tmp.0009.bam
cellsorted_possorted_genome_bam.bam.tmp.0010.bam
cellsorted_possorted_genome_bam.bam.tmp.0011.bam
cellsorted_possorted_genome_bam.bam.tmp.0012.bam
```

```
cellsorted_possorted_genome_bam.bam.tmp.0509.bam
cellsorted_possorted_genome_bam.bam.tmp.0510.bam
cellsorted_possorted_genome_bam.bam.tmp.0511.bam
cellsorted_possorted_genome_bam.bam.tmp.0512.bam
cellsorted_possorted_genome_bam.bam.tmp.0513.bam
cellsorted_possorted_genome_bam.bam.tmp.0514.bam
cellsorted_possorted_genome_bam.bam.tmp.0515.bam
```

A example for velocyto CLI job – half a day



Demo on RNA velocity

Summary of day 1, scRNA-seq

- Set up the working environment
  - Login and pseudolinks to data
  - Copy files to scratch drive
  - Evoke jupyter lab IDE
- Introduction to scRNA-seq
- Make fastq from bcl files
- Use cellranger count to do alignment
- Use Seurat/scanpy pipeline to do QC and dimension reduction
- Use RNA velocity to infer trajectory
- For tomorrow: please install IGV and IGB for visualization of alignment, on your local machine.

Reference

- Make fastq and cellranger count
  - https://davetang.org/muse/2018/08/09/getting-started-with-cell-ranger/
- Seurat pipeline on pbmc
  - https://satijalab.org/seurat/articles/pbmc3k_tutorial.html
- Scanpy pipeline
  - https://scanpy-tutorials.readthedocs.io/en/latest/pbmc3k.html
- RNA velocity
  - velocyto CLI
    - https://velocyto.org/velocyto.py/tutorial/cli.html
  - scVelo
    - Scvelo: https://scvelo.readthedocs.io/VelocityBasics/