Single-Cell Workshop Feedback

Lijiao Ning
UMR 1283 / 8199
2020-10-19

Technologies
&
Experimental Design

Technologies

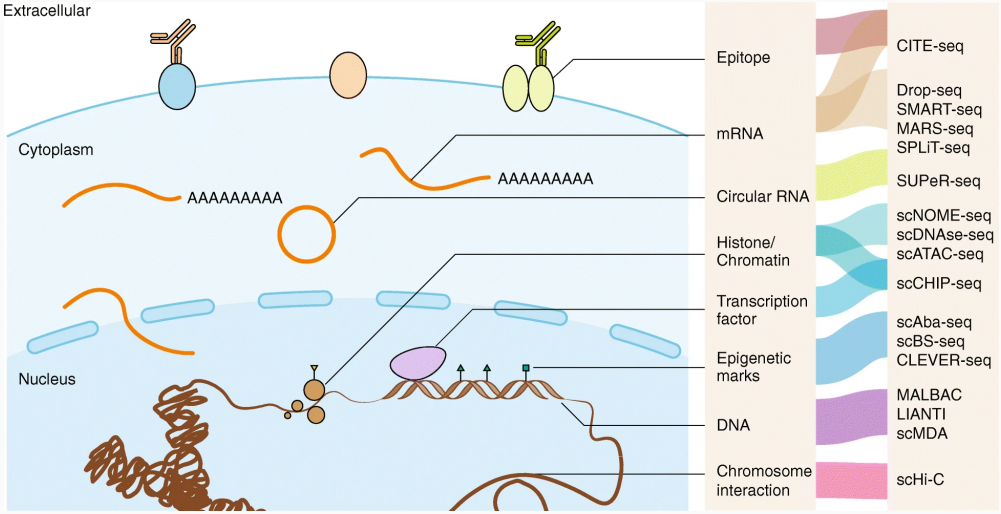
Covered Domains

- Genomics
- Epigenomics
- Transcriptomics
- Proteomics

Why Single Cell?

- Heterogeneity: cell type, cell state (spatial and temporal)
- Rare cells

State-of-the-art



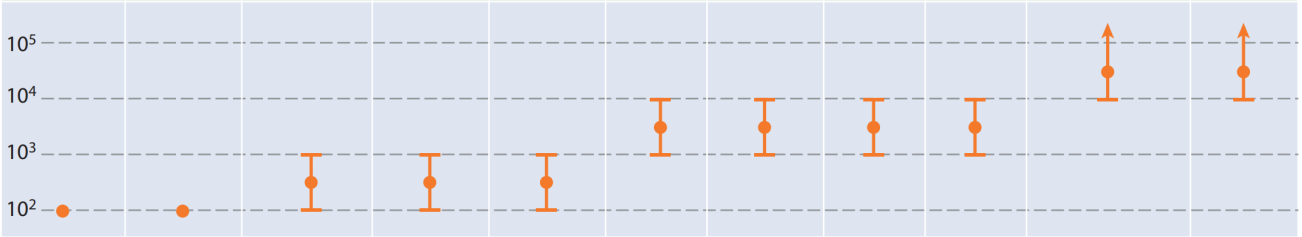
(Ren et *al.*, Genome Biology, 2018)

Experimental Design

- Biological Question?
- Which Technology?
- Sequencing Strategy?
  - Technical variation control: UMI (Unique Molecular Identifier), Spike-ins
  - Sequencing depth
  - Number of cells
- Sample Preparation?
  - Collection time
  - Type, *e.g.*: fresh, frozen, FFPE
  - Cells dissection, *e.g.*: manual, laser, high throughput methods

Experimental Design

Protocols

	SMART-seq2	CEL-seq2	STRT-seq	Quartz-seq2	MARS-seq	Drop-seq	inDrop	Chromium	Seq-Well	sci-RNA-seq	SPLiT-seq
Single-cell isolation	FACS, microfluidics	FACS, microfluidics	FACS, microfluidics, nanowells	FACS	FACS	Droplet	Droplet	Droplet	Nanowells	Not needed	Not needed
Second strand synthesis	TSO	RNase H and DNA pol I	TSO	PolyA tailing and primer ligation	RNase H and DNA pol I	TSO	RNase H and DNA pol I	TSO	TSO	RNase H and DNA pol I	TSO
Full-length cDNA synthesis?	Yes	No	Yes	Yes	No	Yes	No	Yes	Yes	No	Yes
Barcode addition	Library PCR with barcoded primers	Barcoded RT primers	Barcoded TSOs	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers	Barcoded RT primers and library PCR with barcoded primers	Ligation of barcoded RT primers
Pooling before library?	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Library amplification	PCR	In vitro transcription	PCR	PCR	In vitro transcription	PCR	In vitro transcription	PCR	PCR	PCR	PCR
Gene coverage	Full-length	3′	5′	3′	3′	3′	3′	3′	3′	3′	3′
Number of cells per assay											

(Source: lecture 1, K.Lebrigand)

From
Raw Base Call (BCL)
To
Count Matrix

Raw Data Processing

Get FASTQ From BCL Files

- Illumina's bcl2fastq (need to format sample index if applied on 10X data)
- 10X's cellranger mkfastq

```
├── SINCELLTE  ## Project dir
│   └── SCsample1  ## Sample dir
│       ├── SCsample1_S1_L001_I1_001.fastq.gz  ## Sample index
│       ├── SCsample1_S1_L001_R1_001.fastq.gz  ## R1 = BC + UMI
│       └── SCsample1_S1_L001_R2_001.fastq.gz  ## R2 = RNA
├── Reports  ## QC and operation report
│   └── html
│       └── SINCELLTE
├── Stats  ## Stats on processing
│   ├── AdapterTrimming.txt
│   ├── ConversionStats.xml
│   ├── DemultiplexingStats.xml
│   ├── DemuxSummaryF1L1.txt
│   ├── FastqSummaryF1L1.txt
│   └── Stats.json
├── Undetermined_S0_L001_I1_001.fastq.gz  ## failed to get attributed
├── Undetermined_S0_L001_R1_001.fastq.gz  ## Same for R1
└── Undetermined_S0_L001_R2_001.fastq.gz  ## Same for R2
```

Raw Data Processing

Check Quality Of Reads

- FastQC: basic QC on reads
- FastQC Screen: identify cross-species contamination by performing quick mapping
- NGSCheckMate: check sample relatedness (*e.g.*: family, trio, *etc.*), based on samples' variant allele fraction correlation calculated from a known list of SNPs
- Xenome: extract reads from the expected genome (attention: should be only applied on reads in **R2** files, and need to resync R1 with the remaining reads)
- fastp: trim bad quality reads (again, only applied on **R2** files and resync the R1 files)

Raw Data Processing

Get Count Matrices

- cellranger count, takes FASTQ files to perform alignment, with built-in reference packages (*e.g.*: hg19, hg38)

```
outs ## truncated list of files
├── pbmc_1k_protein_v3_web_summary.html
├── raw_matrix
│   ├── barcodes.tsv.gz
│   ├── features.tsv.gz
│   └── matrix.mtx.gz
├── filtered_matrix
│   ├── barcodes.tsv.gz
│   ├── features.tsv.gz
│   └── matrix.mtx.gz
├── pbmc_1k_protein_v3_GTF_rdx.bam
├── pbmc_1k_protein_v3_GTF_rdx.bam.bai
├── pbmc_1k_protein_v3_feature_ref.csv
├── pbmc_1k_protein_v3_feature_ref.fa
└── analysis
    ├── clustering
    ├── pca
    └── tsne
```

Quality Control
&
Normalization
&
Batch Effect

Quality Control

- Number of cells/sample detected
- Filter poor quality cells
    - Number of genes/cell detected
    - Number of UMIs/cell detected
    - % of mitochondrial genes
- Remove doublets: doubletFinder, Scrublet (Python)
- Remove background noise due to ambient mRNAs: SoupX

Quality Control

```r
library(Seurat)
exp_mat ← Read10X(data.dir = "/path/to/filtered_matrix")

## Create a Seurat object from a feature expression matrix
exp_mat ← CreateSeuratObject(
  counts = exp_mat,
  project = "hs_100",
  assay = "RNA",
  min.cells = 3,
  min.features = 100,
  names.field = 1,
  names.delim = "_",
  meta.data = NULL
)

## To access raw counts
exp_mat@assays$RNA@counts ## or directly by
GetAssayData(exp_mat, slot = "counts")[1:3, 1:10]
## 3 x 10 sparse Matrix of class "dgCMatrix"
##
## AP006222.2  1 . . . . . . . 1 .
## FAM41C      . . . . . . . . . .
## NOC2L       3 . . 2 . 3 5 . 8 .
```

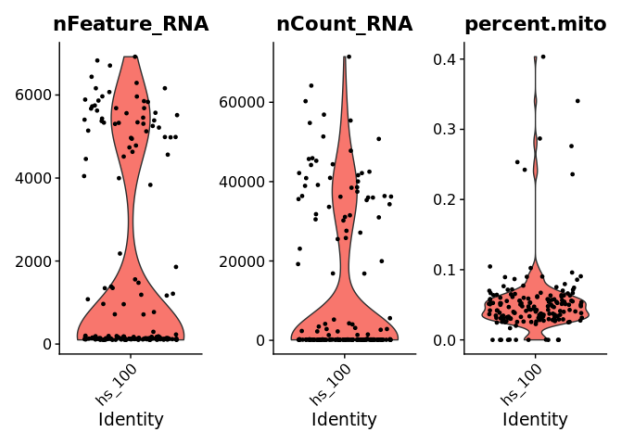
Quality Control

```
## QC metrics are stored in metadata
## Add % mito to metadata
exp_mat[["percent.mt"]] ← PercentageFeatureSet(
  object = exp_mat, pattern = "^MT-"
)

head(exp_mat@meta.data)
##                     orig.ident nCount_RNA nFeature_RNA  percent.mt
## AAAGATGAGAAACGAG-1     hs_100      27144         5125 0.022184300
## AACCATGGTCAGAAGC-1     hs_100       4288         1558 0.018880000
## AACTCCCTCCCAGGTG-1     hs_100        125          108 0.014776925
## AACTCTTGTTCTGAAC-1     hs_100      37486         5307 0.000742153
## AAGCCGCTCCCAGGTG-1     hs_100        227          183 0.036745407
## AAGTCTGAGGACGAAA-1     hs_100       4018         1477 0.020592667
```

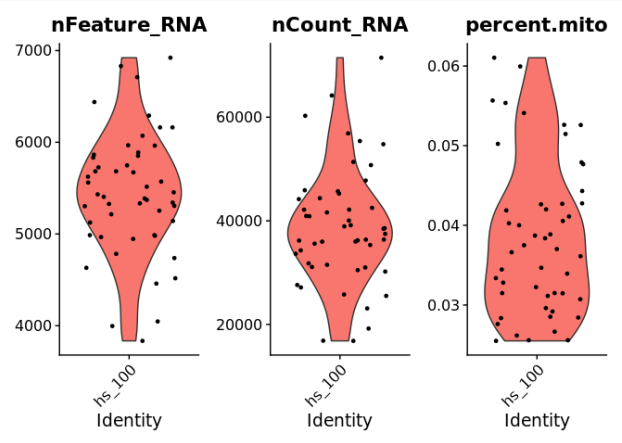
Quality Control

```r
## Visualize QC metrics
VlnPlot(
  object = exp_mat,
  features = c(
    "nFeature_RNA",
    "nCount_RNA",
    "percent.mt"
  )
)

## Filter poor quality cells
exp_mat ← subset(
  x = exp_mat,
  subset = (nFeature_RNA > 3000) &
    (nCount_RNA > 10000) &
    (percent.mito > 0.01) &
    (percent.mito < 0.2)
)
```

Before



After



Normalization

Global Scaling

Hypothesis: cells are homogeneous in RNA content (not always verified)

=> Scaling factor: median UMI count or 10,000 in Seurat and Cell Ranger

scRNA-seq Specfic Scaling

Zero-inflation in single-cell data while a lots of genes expressions are low, methods based on per-gene statistics (upper quartile, edgeR TMM, DESeq2 size factors) will be off

=> Estimation of cell-specific size factors using deconvolution: implemented in scater, scran

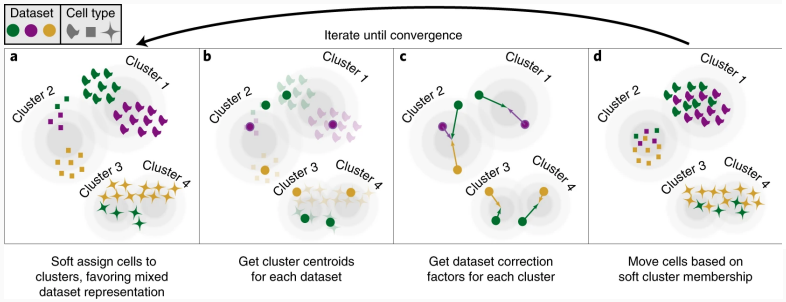
Other Methods

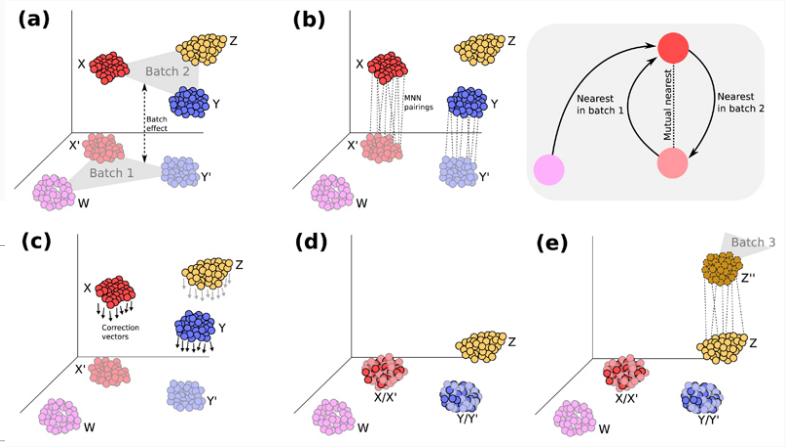
- Variance Stabilization
- Normalization based on spike-ins or housekeeping genes
- *etc.*

Batch Effect

Data integration methods[1], *e.g.*:

- ComBat: sva
- Canonical correlation analysis: Seurat
- Projects cells into a shared embedding: Harmony
- Mutual Nearest Neighbors (MNN): batchelor





[1] A benchmark of batch-effect correction methods for single-cell RNA sequencing data, Tran *et al.*, 2020

Dimension Reduction
&
Clustering

Dimension Reduction

Feature Extraction

- Filter low expressed genes, *e.g.*: < x% of cells, average < threshold
- Highly variable genes (coefficient of variation $C_V = \sigma/\mu$)
- Highly correlated genes (Spearman's rho)

Dimension Reduction

- PCA: linear method
- tSNE (t-Distributed Stochastic Neighbor Embedding)
- UMAP (Uniform Manifold Approximation and Projection)
- *etc.*

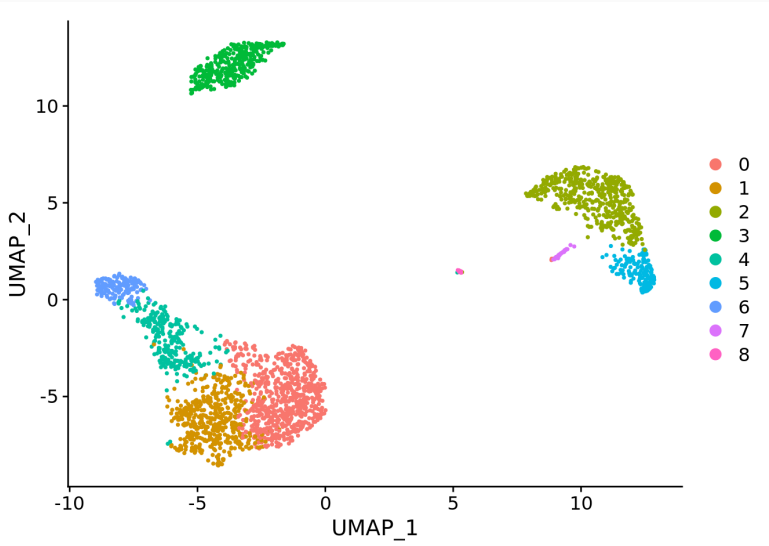
```
exp_mat ← FindVariableFeatures(
  object = exp_mat,
  selection.method = "vst",
  nfeatures = 2000
)
exp_mat ← ScaleData(exp_mat)
exp_mat ← RunPCA(exp_mat)
print(exp_mat[["pca"]], dims = 1:2, nfea
## PC_ 1
## Positive:  CST3, TYROBP, LST1, AIF1,
## Negative:  MALAT1, LTB, IL32, IL7R, C
## PC_ 2
## Positive:  CD79A, MS4A1, TCL1A, HLA-D
## Negative:  NKG7, PRF1, CST7, GZMB, GZ

ElbowPlot(exp_mat) ## determine how many

exp_mat ← RunTSNE(object = exp_mat, dim
exp_mat ← RunUMAP(object = exp_mat, dim
```

Clustering

E.g.: K-means based, hierarchical clustering, KNN-based, graph-based clustering

```
## Example of KNN-based
exp_mat ← FindNeighbors(
  object = exp_mat,
  reduction = "pca",
  dims = 1:10
)
exp_mat ← FindClusters(
  object = exp_mat,
  resolution = 0.8
  ## higher resolution = more clusters
)

## To access clusters
head(Idents(exp_mat), 3)
## AAACATACAACCAC AAACATTGAGCTAC AAACATT
##              1              3
## Levels: 0 1 2 3 4 5 6 7 8
```

```
DimPlot(exp_mat, reduction = "umap")
```



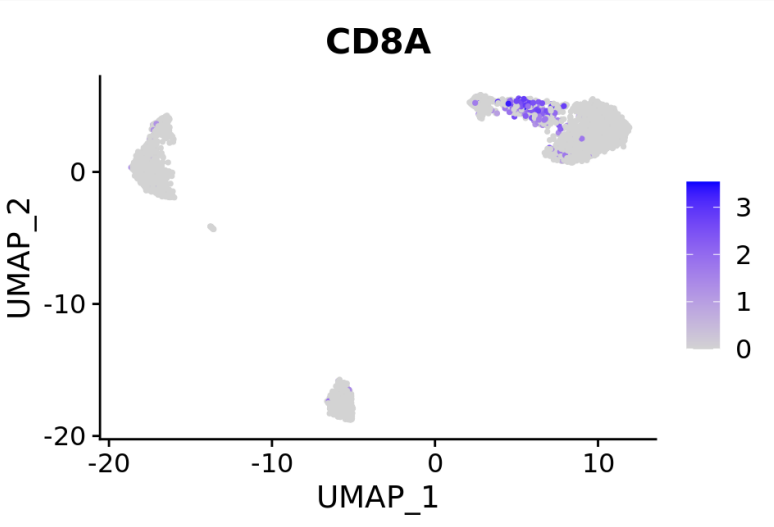
(Source: Seurat Guided tutorial - 2,700 PBMCs)

Cell Type Annotation

Cell Type Annotation

- Manual ways:
  - prior knowledge in literature
  - public database like panglaoDB (human and mouse) or scRNAseq (Bioconductor)
  - ask your biologist ;)
- Label transferring: predict unknown samples using reference by checking the similarity (in Seurat: `FindTransferAnchor` + `TransferData`, other R packages: CHETAH, SingleR, *etc.*)

```
# find all markers of cluster 1
cluster1.markers ← FindMarkers(
  object = exp_mat, ident.1 = 1
)
```



(Source: lecture 7, A.Cortal)

Differential Expression Analysis

Differential Expression Analysis

- SCDE: Bayesian method to compare two groups of single cells
- MAST: adapted generalized linear model for bimodal and/or zero-inflated single cell data
- Wilcoxon Rank Sum test
- Student's t-test
- *etc.*

```
# Find DE markers for each found cluster
up_markers ← FindAllMarkers(
  object = exp_mat,
  test.use = "wilcox",
  only.pos = TRUE, ## chekc only up-regulated genes
  min.pct = 0.1, ## filter genes expressed in few cells
  logfc.threshold = 0.5
)
```

Example Workflow

Example Workflow

Some complete analysis pipelines: Seurat, Scater (Bioconductor), Scran (Bioconductor), SCANPY(Python)

Seurat Standard Workflow:

```
library(Seurat)
exp_mat ← Read10X(data.dir = "path/to/dir_matrices")
exp_mat ← CreateSeuratObject(counts = exp_mat)
exp_mat ← NormalizeData(object = exp_mat)
exp_mat ← FindVariableFeatures(object = exp_mat)
exp_mat ← ScaleData(object = exp_mat)
exp_mat ← RunPCA(object = exp_mat)
exp_mat ← FindNeighbors(object = exp_mat)
exp_mat ← FindClusters(object = exp_mat)
exp_mat ← RunTSNE(object = exp_mat)
DimPlot(object = exp_mat, reduction = "tsne")
```

Further Analyses

Further Analyses

Functionnal Analysis

- Based on MCA (multiple correspondence analysis): CellID

Trajectory Analysis

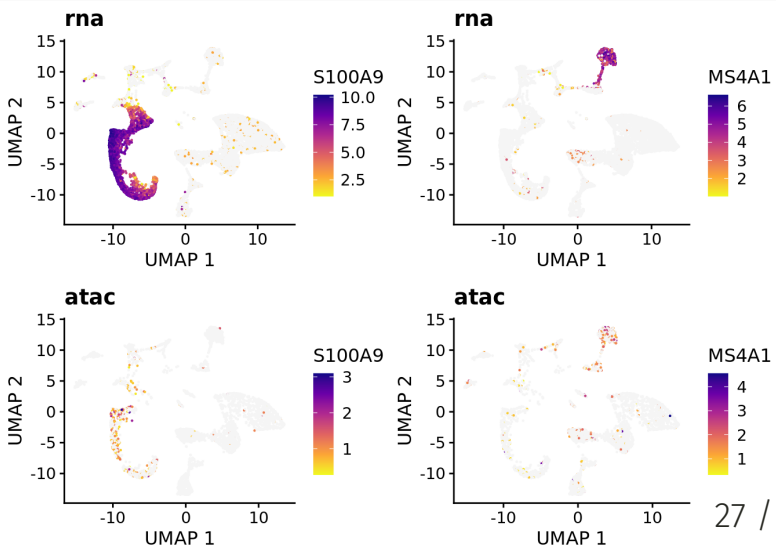
- ElPiGraph
- STREAM (Single-cell Trajectories Reconstruction, Exploration And Mapping)

Integration multi-omics

Compensate for missing or unreliable information present in each omics.

Integration methods:

- Transfer learning: Signac
- Matrix factorization: LIGER



Some Readings

- Workshop slides: https://cloud.biologie.ens.fr/index.php/s/RXkNGwtR1MLf5mo
- AlJanahi, Aisha A et al. “An Introduction to the Analysis of Single-Cell RNA-Sequencing Data.” Molecular therapy. Methods & clinical development vol. 10 189-196. 2 Aug. 2018, doi:10.1016/j.omtm.2018.07.003
- Introduction to Single-cell RNA-seq: https://hbctraining.github.io/scRNA-seq/
- Seurat guided analyses: https://satijalab.org/seurat/vignettes.html
- Orchestrating Single-Cell Analysis with Bioconductor: https://osca.bioconductor.org/index.html


Thank You