Material for this workshop

- This presentation
- ArchR_demo_1_create_arrow_files.Rmd
- ArchR_demo_2_analysis.Rmd
- Input data

Please follow the pre-workshop instructions on the workshop webpage

Introduction to scATAC-seq Data Analysis

November 14-15 2024

Reuben Thomas
Michela Traglia
Ayushi Agrawal

GLADSTONE INSTITUTES
SCIENCE OVERCOMING DISEASE

Introductions

Reuben Thomas

Associate Core Director

Michela Traglia

Senior Statistician

Ayushi Agrawal

Bioinformatician III

Natalie Elphick

Bioinformatician I

Aim of the workshop

- To give an overview of the biological insights using scATAC-seq analysis

- Highlight assumptions and limitations underlying the methods

- Understand the relevance and impact of difference between scRNAseq and scATAC-seq workflow

- Experience how to analyze scATAC-seq data in ArchR

Workshop organization

- **Session 1 (Thursday, 1pm-4pm)**

1. Cell regulome and ATAC-seq
2. Technology
3. From sequencer to fragments file
4. Pre-processing and QC
   - Break
5. Normalization, Dimensionality reduction, embedding
6. Clustering and cell type annotation based on feature markers
   - Break
7. Advance analysis: Calling Peaks and Motif enrichment

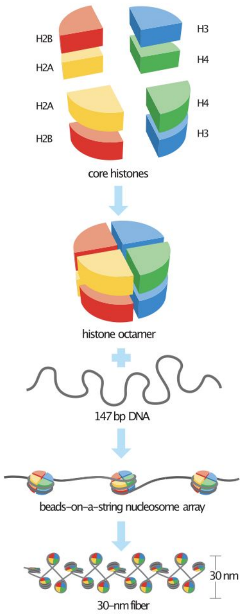
- **Session 2 (Friday, 1-4 pm)**

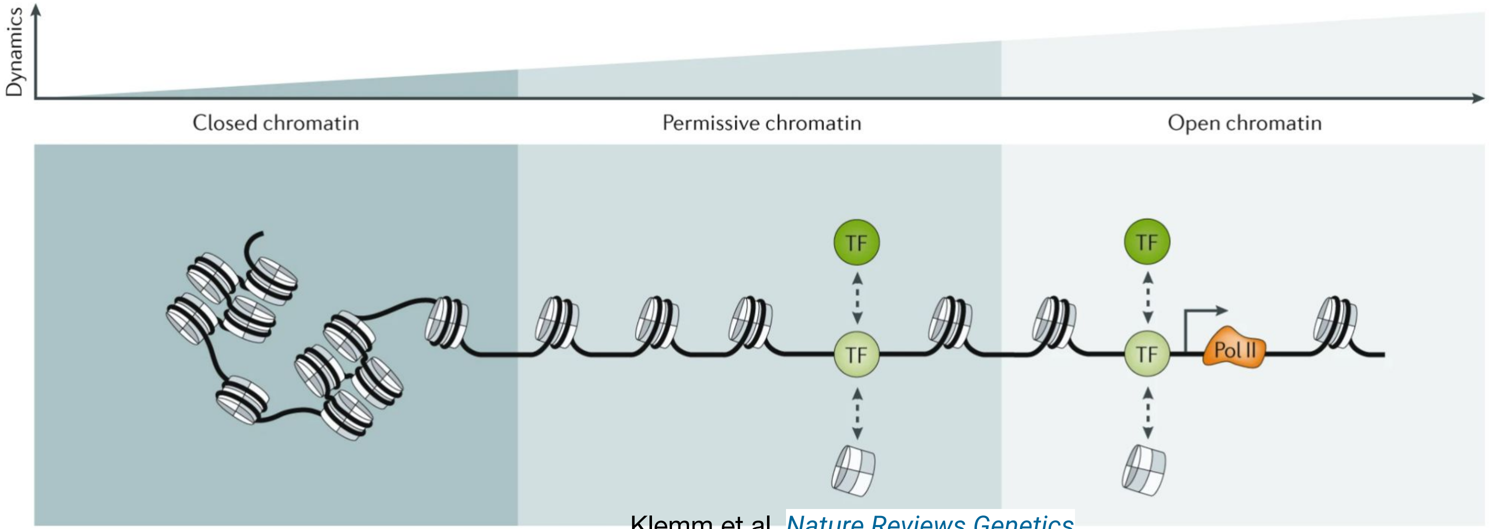
8. Intro to ArchR
   - Demo

scATAC-seq and scRNA-seq Data Integration workshop - November 22 at 1pm
Register here

1. Chromatin architecture of the regulatory regions

Chromatin accessibility



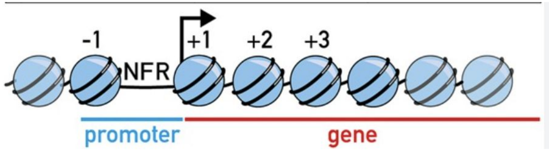


Klemm et al, *Nature Reviews Genetics*

- Nucleosome is an octamer of histone proteins wrapped by ~147 bp of DNA.
- Organized into chromatin with limited accessibility to external factors.
- Accessibility is dynamic and depends by the occupancy and topological organization of nucleosomes.

Non uniform organization of nucleosomes

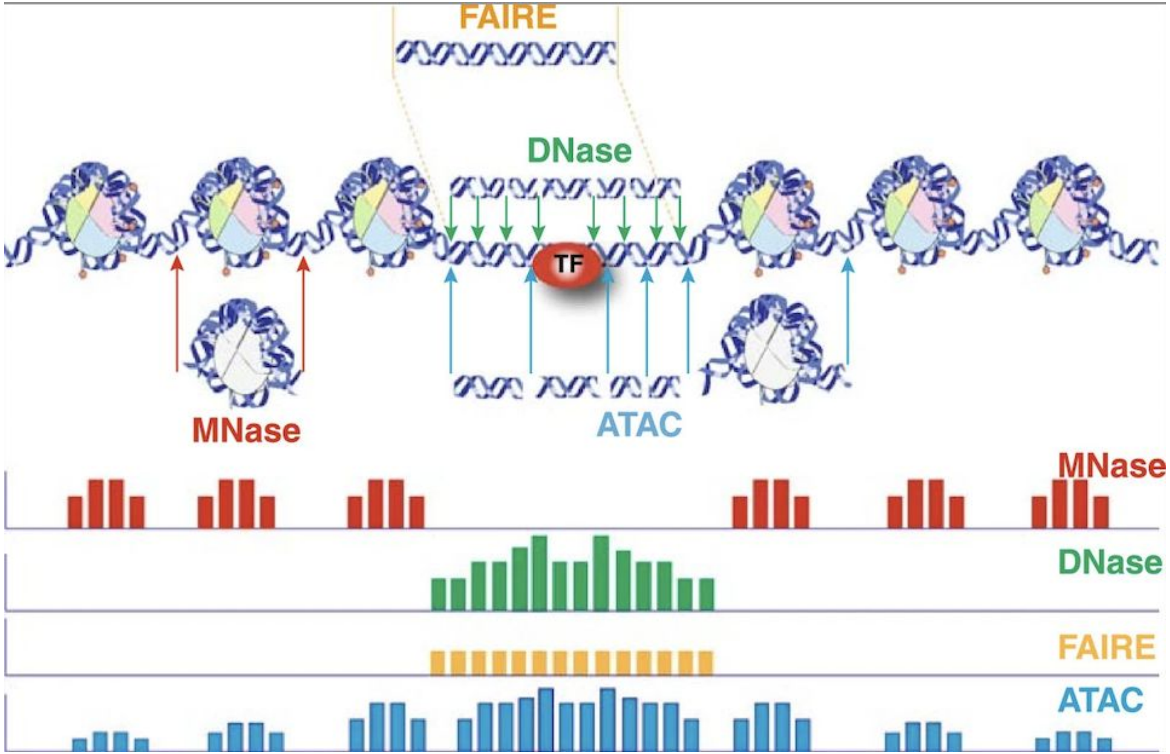
- Nucleosome depletion at regulatory loci (promoters, enhancers, etc.)
- Nucleosome-free regions (NFRs) are localized just upstream of the transcription start site (TSS).



- First nucleosome downstream of the TSS (the + 1 nucleosome) is strongly localized at the same location across cells.
- Accessible genome comprises ~2–3% of total DNA sequence
- > 90% of regions bound by Transcription Factors (TF).

Genome-wide profiling of chromatin accessibility to identify **candidate regulatory genomic regions** in a tissue or cell type

Which regulatory region are you interested in?



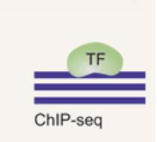
MNase: indirect study of accessibility - nucleosome occupancy

DNase: DNase I hypersensitive site - nucleosome depleted regions

FAIRE: easy but background

ATAC:

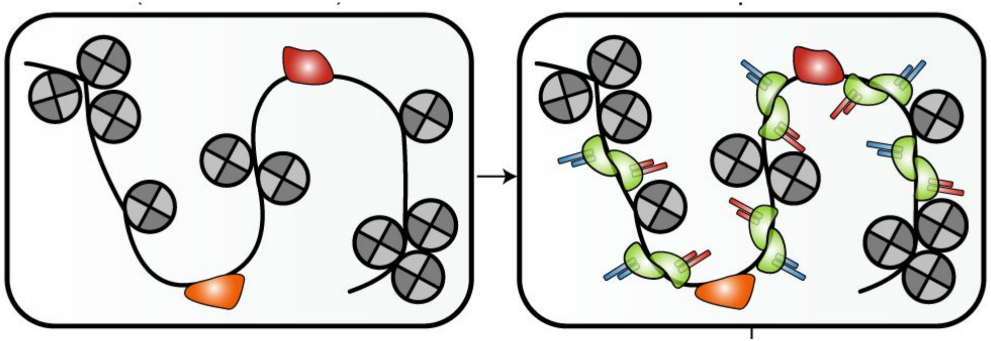
- simple and fast two-step protocol
- high sensitivity
- low starting cell number (500 to 50,000 cells)
- multiple aspects of chromatin architecture simultaneously at high resolution.



ChIP-seq : binding sites of DNA-associated proteins and can be used to map global binding sites for a given protein.

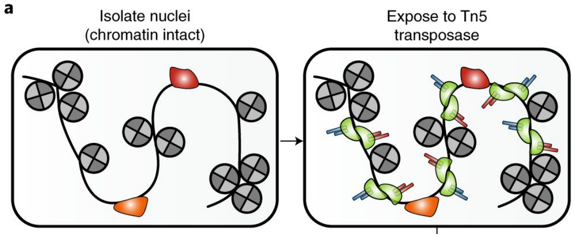
ATAC is based on the transposition system

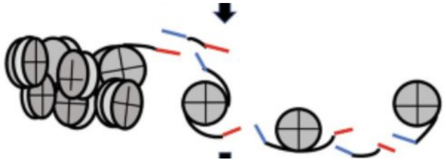
Assay for **T**ransposase-**A**ccessible **C**hromatin with high-throughput sequencing



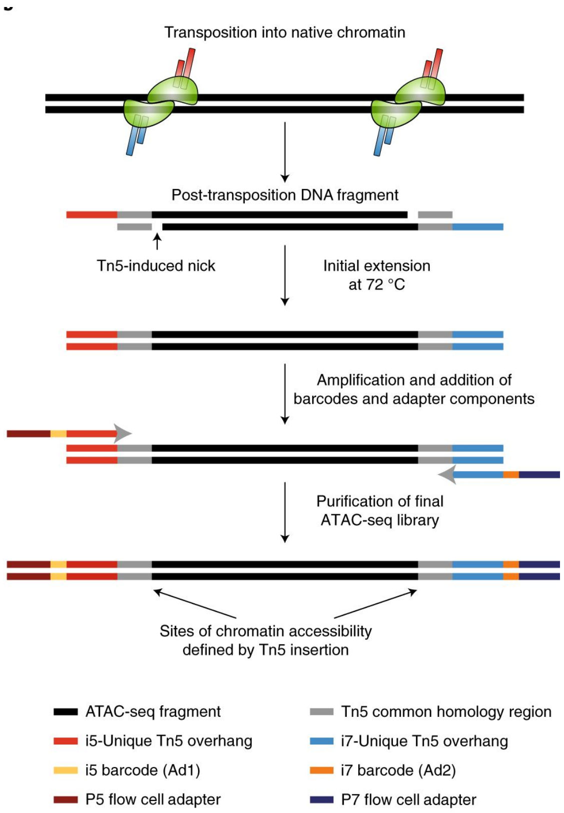
- Transposons are genetic elements that can “jump” to different locations within a genome.
- Bacterial Tn5 - encode a hyperactive Tn5 transposase (Tnp) that can simultaneously <u>cut accessible DNA fragments </u>and ligate sequencing adapters to both strands.
- Fast “cut and paste” and “copy and paste” functions.

Transposition events create fragments





- Binds as a homodimer with **9-bp of DNA between the two Tn5 molecules.**
- Two transposition events = FRAGMENT



- The central point of the “accessible” site is in the very **center of the Tn5 dimer**, not the location of each Tn5 insertion.
- **Adjusting** plus-stranded insertion events by **+4 bp** and minus-stranded insertion events by **-5 bp.**
- Paired-end sequencing.

Tn5 application

2D structure:

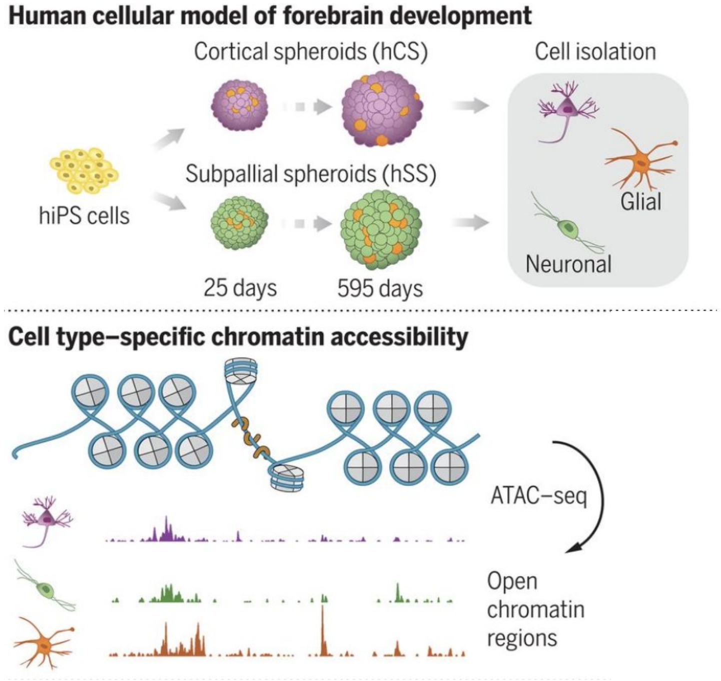
- Bulk ATAC-seq
- scATAC-seq to study the dynamically regulated chromatin in a cell type-specific manner

Spatial proximity within the nucleus by analysing contacts between genomic regions:

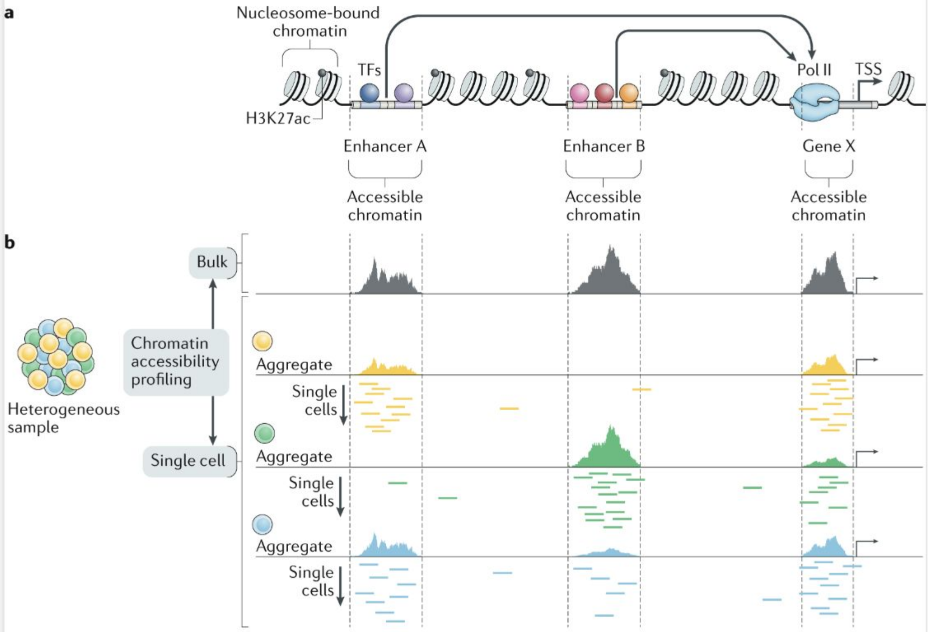
- High-throughput sequencing for the study of chromatin 3D structure - chromatin conformation capture assays 3C (one vs one) and Hi-C (all vs all).

Experimental design is fundamental (I)

- Which epigenetic mechanism do you need to study?
- Is single cell suitable for my purpose?

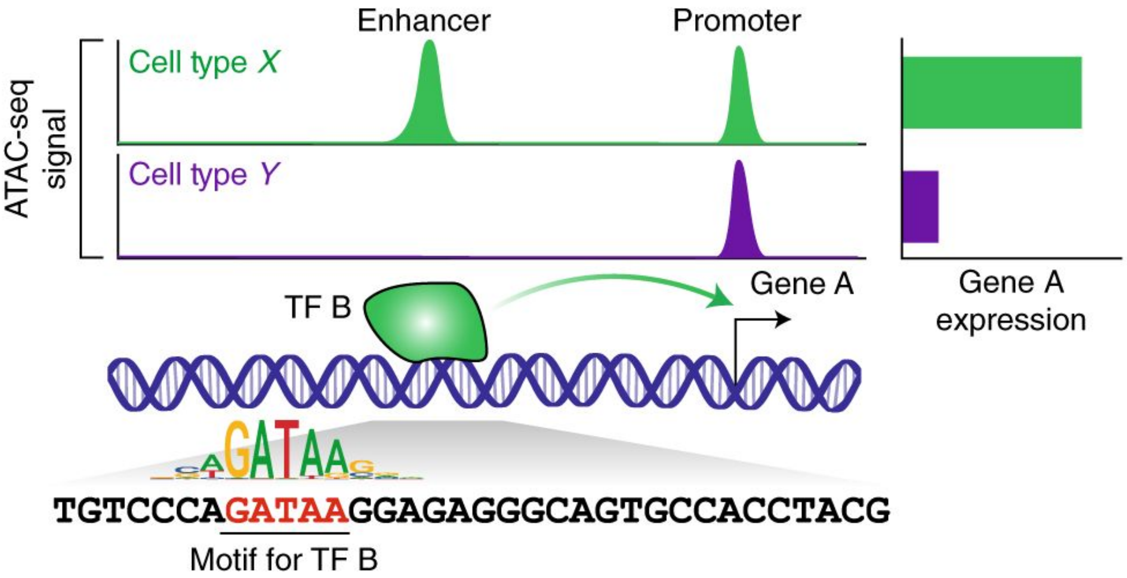

Human cellular model of forebrain development
Cortical spheroids (hCS)
Cell isolation
hiPS cells
Subpallial spheroids (hSS)
Glial
Neuronal
25 days
595 days
Cell type–specific chromatin accessibility
ATAC–seq
Open chromatin regions

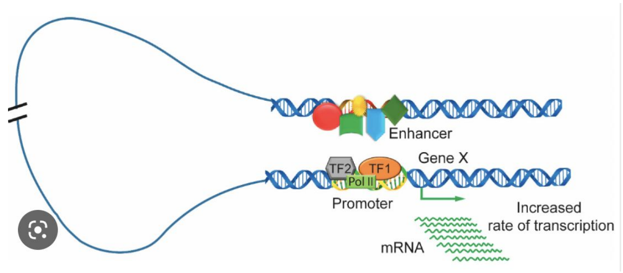
Trevino et al, Science 2020

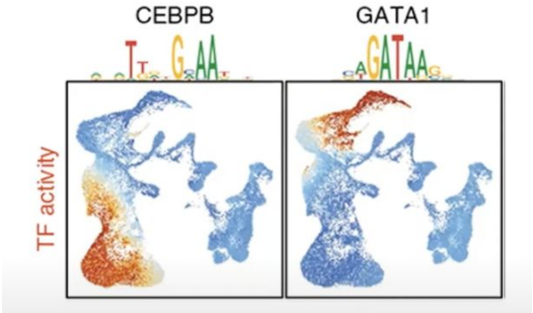


Nature Reviews Methods Primers, 2021

What biological insight we can get from scATAC-seq







- One step further than scRNA-seq:
    - In trajectory analysis, identify variable TF and enhancers
    - Gene regulatory networks between transcription factor gene and accessible target genes

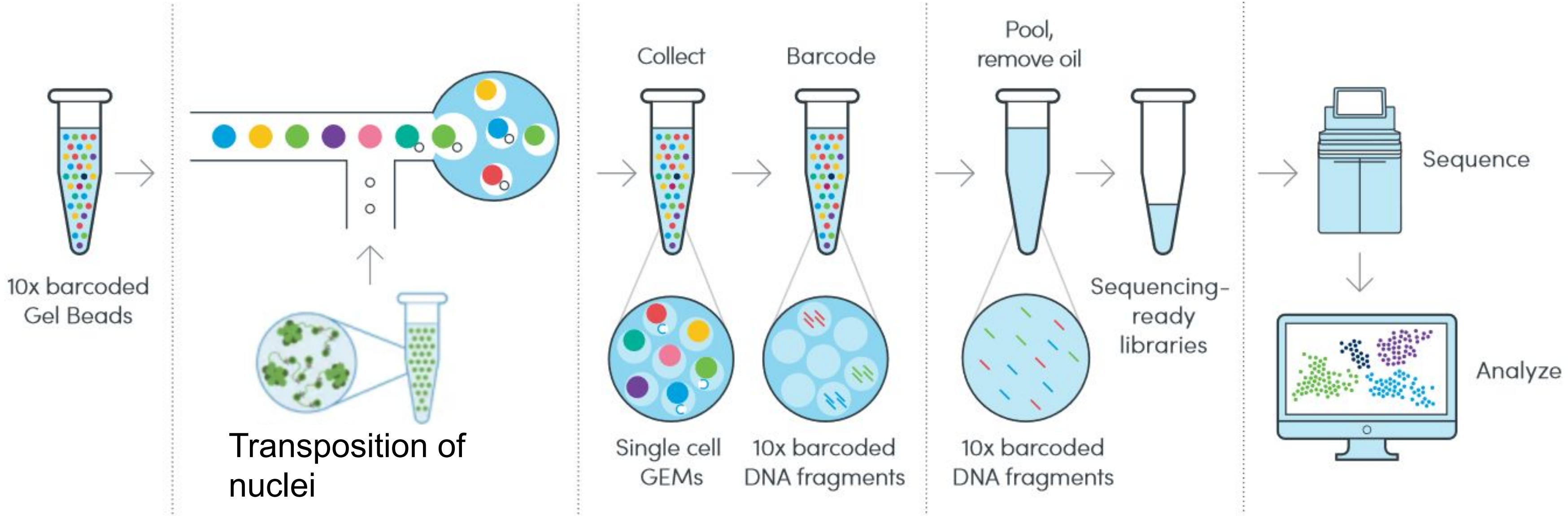
scATAC-seq workflow

From bench	**Through the bioinformatic analysis**	**To the biological interpretation**
Your cell’s regulome question	Fragments count	Embedding
Library preparation & sequencing	QC filtering low quality cells	Clustering
Demultiplex, read alignment and quantification	Layered dimensionality reduction	Find Markers - cell annotation

For more biological insight in cell’s regulome		
Peak calling	Motif Enrichment	Motif Footprinting

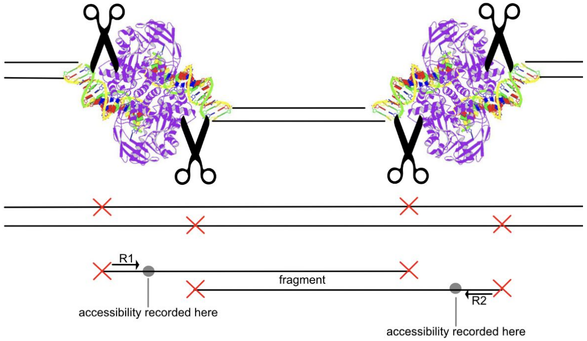
2. scATACseq technology

Single cell ATAC-seq protocol



Transposition of nuclei

10X Genomics scATAC-seq website

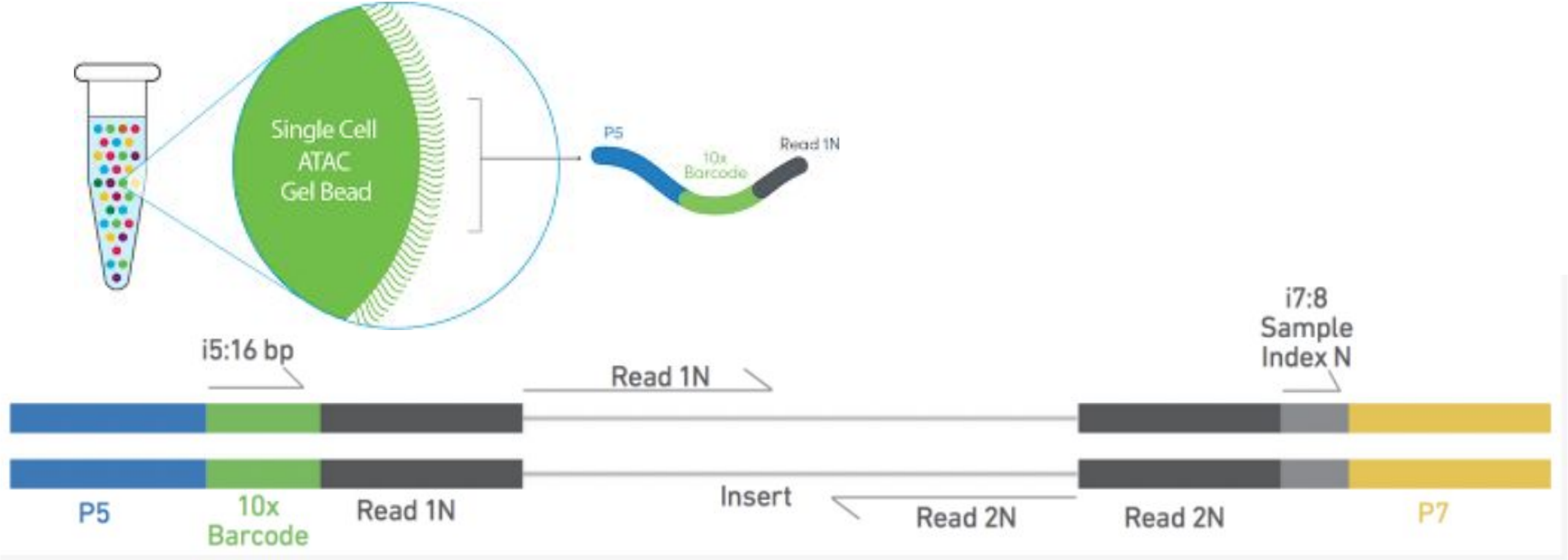


Transposase image accessed from the Protein Data Bank, https://rcsb.org/structure/1MUH

doublets

Tn5 enters the nuclei, fragments DNA in open chromatin regions, and adds the indexing adapters

Single cell ATAC-seq protocol



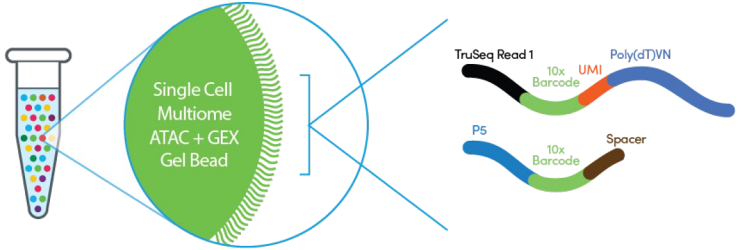
Identify nucleus of origin

Identify sample of origin (i.e. pooling)

10X Genomics scATAC-seq website

Experimental design is fundamental (II)

- Do I need to perform scRNA-seq and scATAC-seq separately or Multiome analysis (scATAC+scRNAseq)?



Multiome:

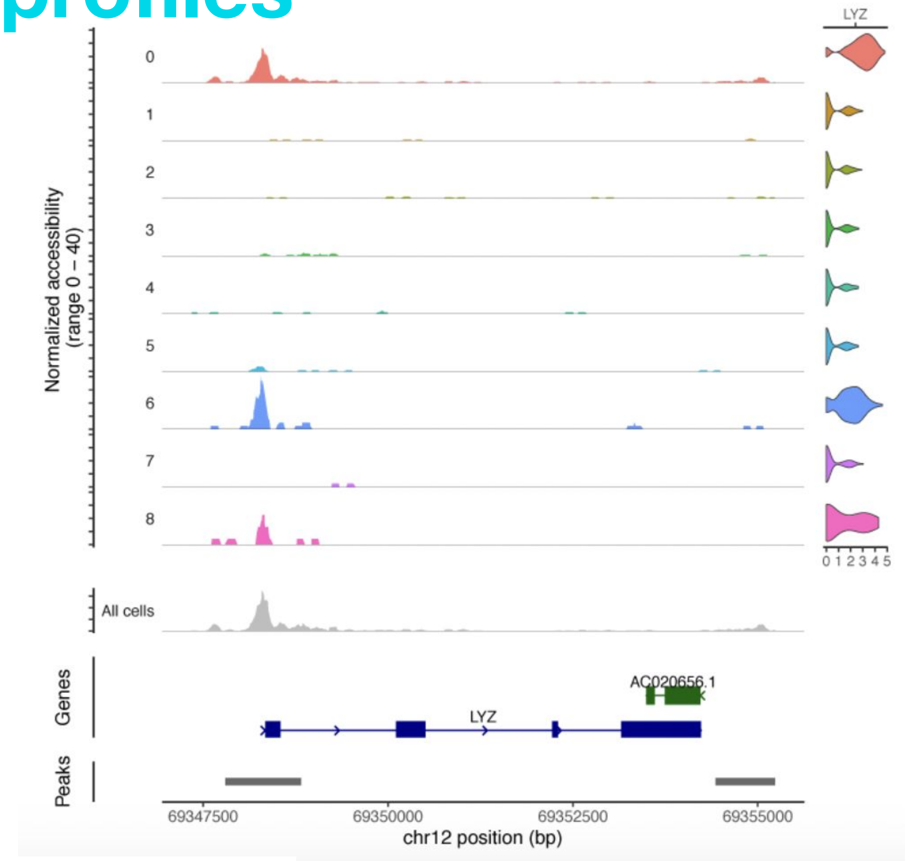
+ Simultaneously profiling gene expression and open chromatin from the same cell.

+ More straightforward downstream analysis.

+ No need to integrate scRNA and scATAC seq data.

- Costs

Simultaneously expression and accessibility profiles



Clusters of gene expression for LYZ

Recommendation: whether the budget allows, use Multiome

Data provided by 10X Genomics.

Caveats of scATAC-seq

A typical human scATAC-seq dataset contains 100 - 10,000 cells and 1,000 - 100,000 sequence reads per cell.

- **Drops with no cell/doublets**

- **Low detection efficiency** - limited # of accessible regions from each cell (5–15% of peaks detected)
- In a cell, **most cis-REs don’t have mapped read** - cis-REs in the genome >> 100,000.
- Activities of all cis-regulatory elements is a continuous steady-state activity in a cell.

  -> scATAC-seq give a **picture of a specific timepoint.**

Knowledge check 1

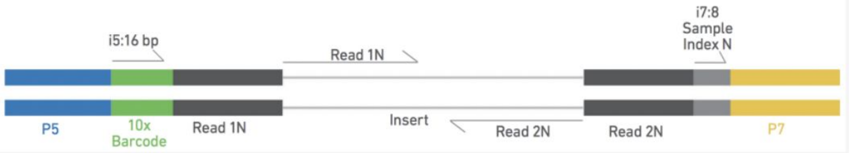
Using scATAC-seq, it is possible to identify:

1. An enhancer active in different cell types.
2. A gene poorly expressed in a particular cell type.
3. Cell-specific peaks of accessible chromatin
4. All of the above.

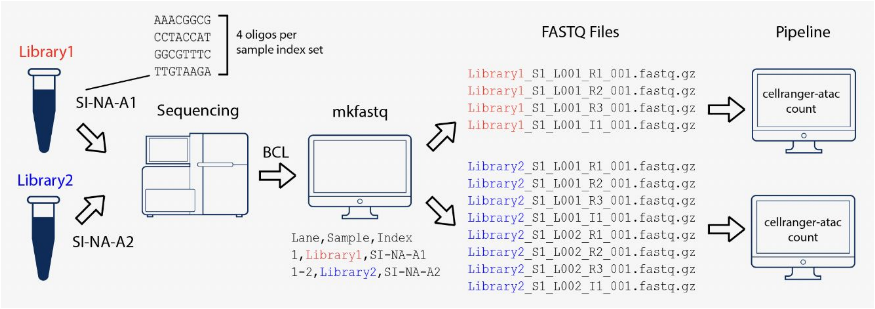
3. From raw sequencing data to the fragments file

Tool: Cellranger-ATAC

Cellranger ATAC



Two 10x Genomics libraries - multiplexed on a single flow cell

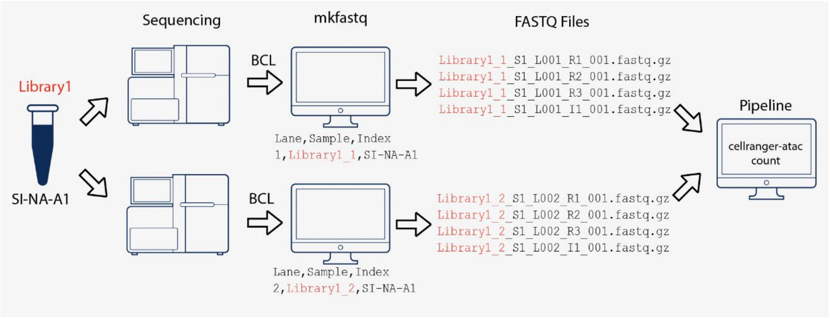


```
Lane,Sample,Index
1,Library1,SI-NA-A1
1-2,Library2,SI-NA-A2
```

```
Library1_S1_L001_R1_001.fastq.gz
Library1_S1_L001_R2_001.fastq.gz
Library1_S1_L001_R3_001.fastq.gz
Library1_S1_L001_I1_001.fastq.gz

Library2_S1_L001_R1_001.fastq.gz
Library2_S1_L001_R2_001.fastq.gz
Library2_S1_L001_R3_001.fastq.gz
Library2_S1_L001_I1_001.fastq.gz
Library2_S1_L002_R1_001.fastq.gz
Library2_S1_L002_R2_001.fastq.gz
Library2_S1_L002_R3_001.fastq.gz
Library2_S1_L002_I1_001.fastq.gz
```

One 10x Genomics library sequenced on two flow cells



```
Lane,Sample,Index
1,Library1_1,SI-NA-A1
```

```
Library1_1_S1_L001_R1_001.fastq.gz
Library1_1_S1_L001_R2_001.fastq.gz
Library1_1_S1_L001_R3_001.fastq.gz
Library1_1_S1_L001_I1_001.fastq.gz
```

```
Lane,Sample,Index
2,Library1_2,SI-NA-A1
```

```
Library1_2_S1_L002_R1_001.fastq.gz
Library1_2_S1_L002_R2_001.fastq.gz
Library1_2_S1_L002_R3_001.fastq.gz
Library1_2_S1_L002_I1_001.fastq.gz
```

1. Demultiplex (BCL to FASTQ)
   - ***cellranger-atac mkfastq***

2. Alignment, filtering, barcode counting, and additional steps
   - ***cellranger-atac count***

3. Aggregate counts from multiple runs
   - ***cellranger-atac aggr***

Reads mapped to chrM, specified as non-nuclear contigs, are filtered out

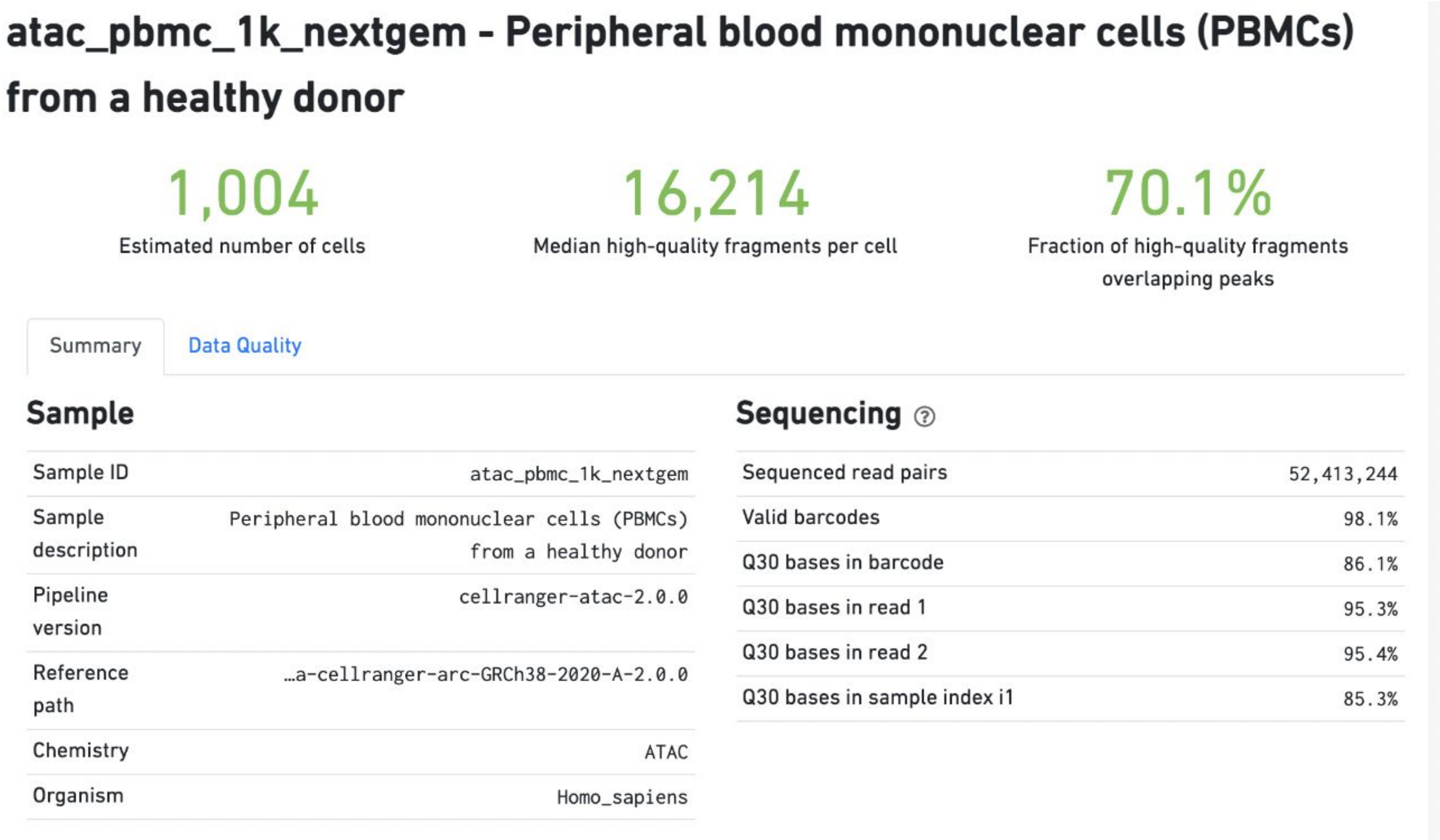
(Images from 10x Genomics website)

Cell Ranger-ATAC outputs

- BAM files with aligned reads
- **web_summary.html**
- **Fragments and bulk peak files.**
- Secondary analysis (e.g., dimensionality reduction, clustering, etc.) -> discard
- Loupe file to interactively view secondary analysis results with Loupe Browser from 10x Genomics
- Molecule info

Cell Ranger outputs

(more detail on HTML output)



atac_pbmc_1k_nextgem - Peripheral blood mononuclear cells (PBMCs) from a healthy donor

1,004
Estimated number of cells

16,214
Median high-quality fragments per cell

70.1%
Fraction of high-quality fragments overlapping peaks

Sample

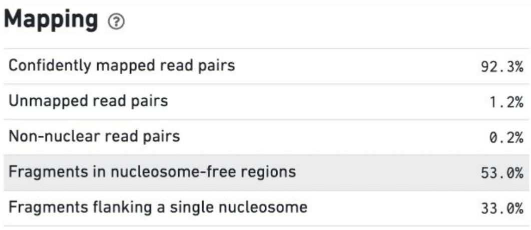
Sample ID	atac_pbmc_1k_nextgem
Sample description	Peripheral blood mononuclear cells (PBMCs) from a healthy donor
Pipeline version	cellranger-atac-2.0.0
Reference path	…a-cellranger-arc-GRCh38-2020-A-2.0.0
Chemistry	ATAC
Organism	Homo_sapiens

Sequencing ?

Sequenced read pairs	52,413,244
Valid barcodes	98.1%
Q30 bases in barcode	86.1%
Q30 bases in read 1	95.3%
Q30 bases in read 2	95.4%
Q30 bases in sample index i1	85.3%

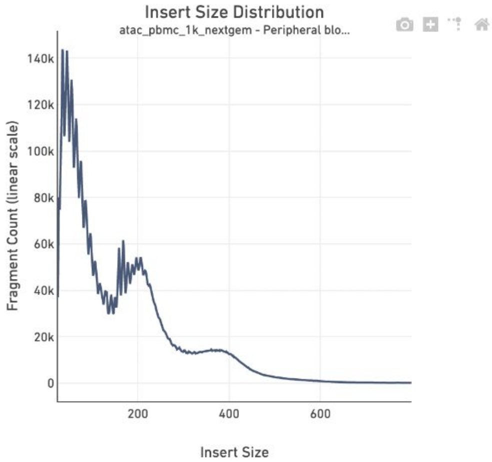
Cell Ranger outputs

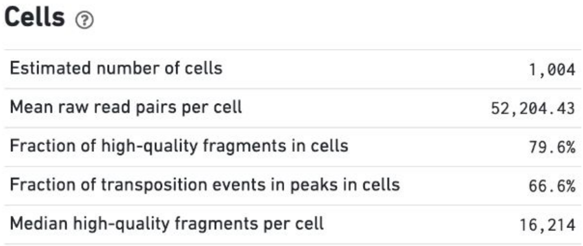
(more detail on HTML output)



Mapping ?

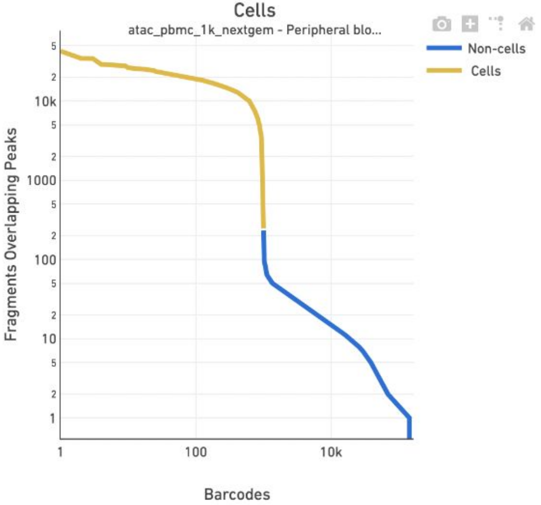
Confidently mapped read pairs	92.3%
Unmapped read pairs	1.2%
Non-nuclear read pairs	0.2%
Fragments in nucleosome-free regions	53.0%
Fragments flanking a single nucleosome	33.0%

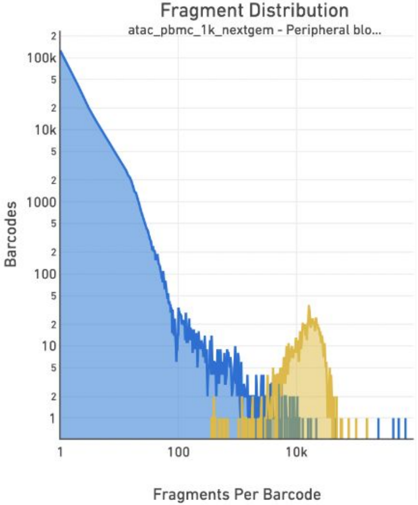


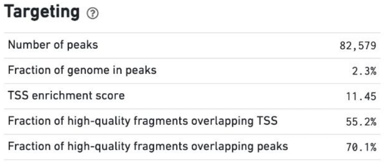


Cells ?

Estimated number of cells	1,004
Mean raw read pairs per cell	52,204.43
Fraction of high-quality fragments in cells	79.6%
Fraction of transposition events in peaks in cells	66.6%
Median high-quality fragments per cell	16,214

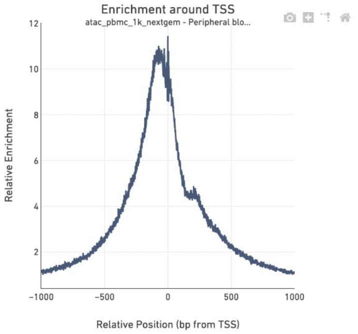


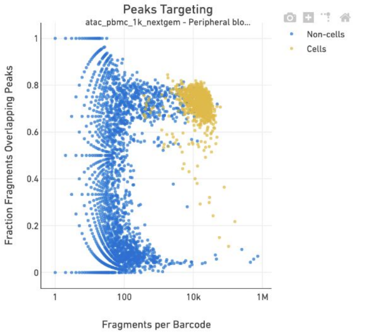




Targeting ?

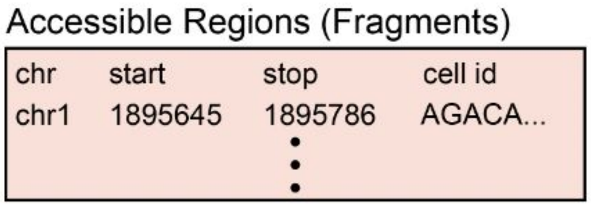
Number of peaks	82,579
Fraction of genome in peaks	2.3%
TSS enrichment score	11.45
Fraction of high-quality fragments overlapping TSS	55.2%
Fraction of high-quality fragments overlapping peaks	70.1%





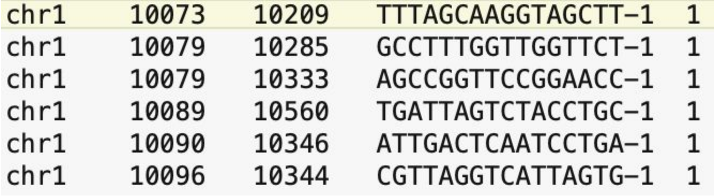
Output cellranger -> fragments linked to the cell ID

`atac_fragments.tsv.gz`



Accessible Regions (Fragments)

chr	start	stop	cell id
chr1	1895645	1895786	AGACA...
		⋮	



```
chr1    10073   10209   TTTAGCAAGGTAGCTT-1  1
chr1    10079   10285   GCCTTTGGTTGGTTCT-1  1
chr1    10079   10333   AGCCGGTTCCGGAACC-1  1
chr1    10089   10560   TGATTAGTCTACCTGC-1  1
chr1    10090   10346   ATTGACTCAATCCTGA-1  1
chr1    10096   10344   CGTTAGGTCATTAGTG-1  1
```

- Header lines: # record information about the sample, the reference used and primary contigs in the reference.
- Each scATAC-seq fragment and the corresponding cell ID, one fragment per line.

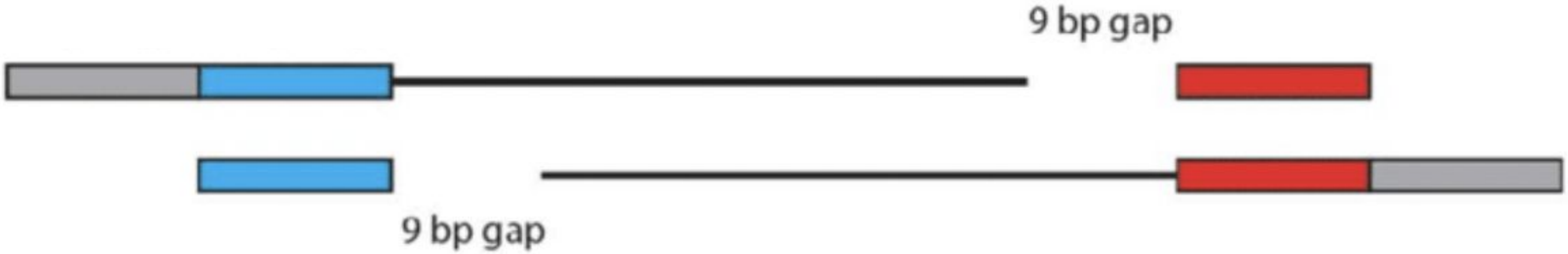
Different frameworks to process the fragments and perform the analyses downstream.

- SnapATAC - Nature Comm 2021 and SnapATAC2 - Nature Methods 2024
- Signac - extension of Seurat for the analysis of single-cell chromatin data
- ArchR - shown during this workshop [paper]

4. Pre-processing and cell QC

ArchR: `ArchRProject(); createArrowFiles() (` `addDoubletScores();filterDoublets()` `plotGroups(); plotFragmentSizes()`

Fragments position need to be adjusted for 9bp -> insertions



In ArchR for rapid access, **fragment files** chunked and converted to a compressed temporary HDF5-formatted file including:

- Chromosome
- offset-adjusted chromosome start position
- offset-adjusted chromosome end position
- Cellular barcode ID.

insertions-> the offset-adjusted single-base position at the very center of an accessible site.

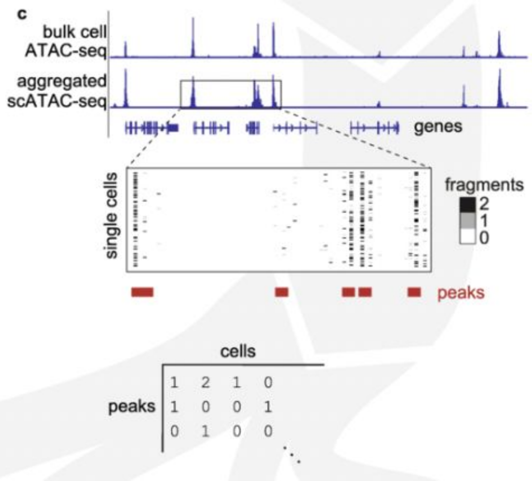
Sparsity of data issue

- Transposition is rare.
- Majority of accessible regions are not transposed -> many loci having 0 accessible alleles.
- 1= accessible; 0=???
- 0 in scATAC-seq could mean “not-accessible” **or** “not sampled”
- Binarized scATAC-seq data matrix mostly 0s.

Input for analyses downstream

- **Count matrix on a set of peaks** (Signac - Stuart lab)
  - Bulking cells to call peak -> no rare cells peaks

- **Genome-wide tiles** (SnapATAC, SnapATAC2 and **ArchR**
  - 5kb - SnapATAC
  - Accessibility regions are long bp << 5kb
  - ArchR + SnapATAC2 implement **insertion counts** across **genome-wide 500bp tiles**
    - Better resolution
    - Allows for the identification of clusters prior to calling peaks
    - 3B bp in 500bp tiles: 6 million features to be included in the cell by tile matrix



1	2	1	0
1	0	0	1
0	1	0	0

```
createArrowFiles(..,addTileMat = TRUE)
```

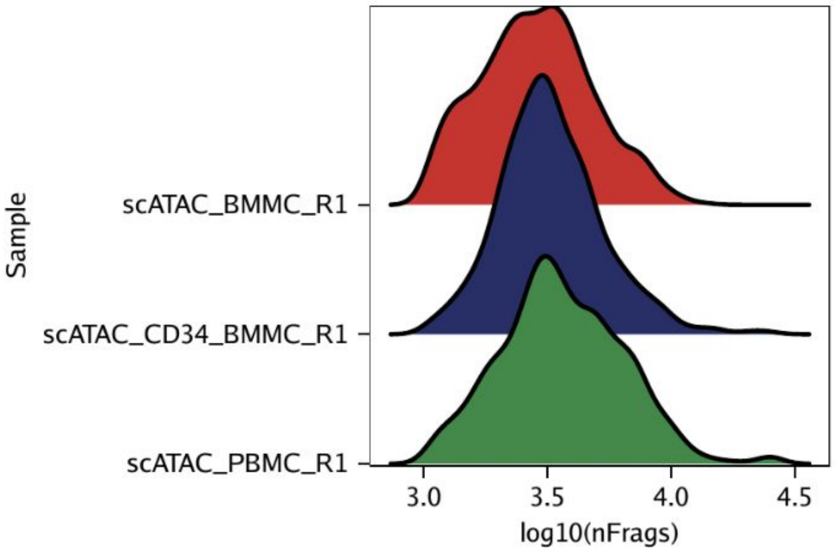
Knowledge check 2

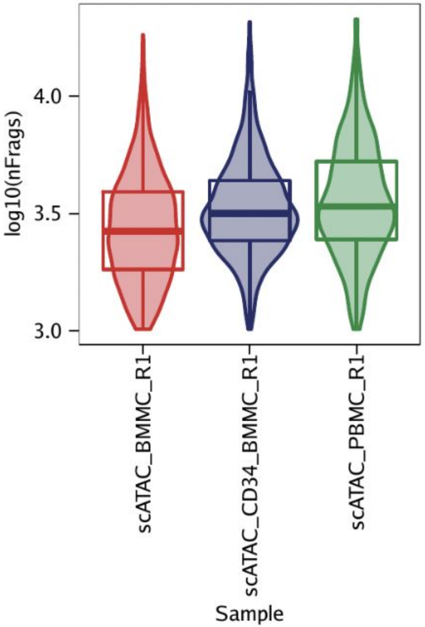
What is the best input matrix for sparse data to preserve cell specificity:

1. Bulk peak matrix (across cells)
2. Insertion counts across genome-wide tiles

Filtering out low quality cells

1. **Number of unique nuclear fragments > 1000**



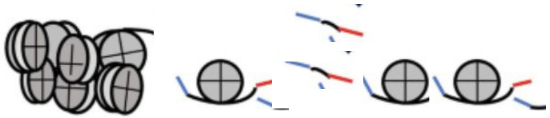


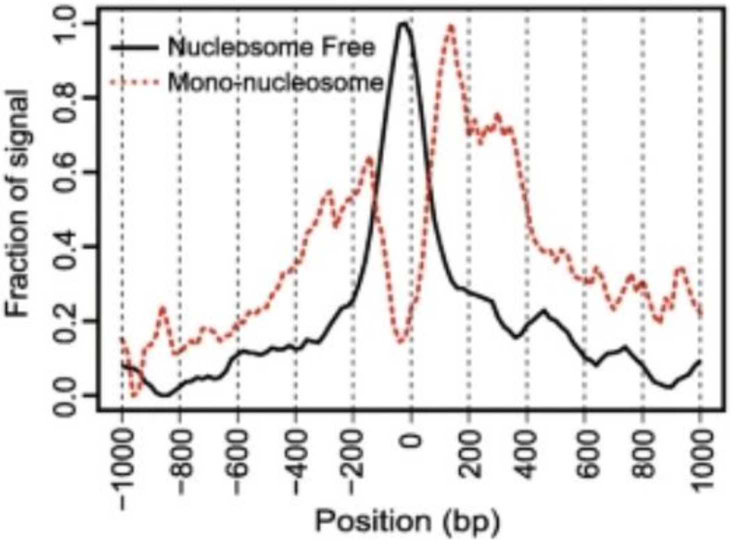
```
plotGroups(..., name = "log10(nFrags)",..)
```

```
plotGroups(..., name = "log10(nFrags)",plotAs = "violin")
```

Filtering out low quality cells

2. Signal-to-background ratio - > TSS enrichment





Nucleosome-free regions (NFR) (< 100 bp)

Mono-, di-, and tri-nucleosomes regions (~ 200, 400, 600 bp, respectively)

Fragments from the NFR enriched around the transcription start site (TSS) of genes.

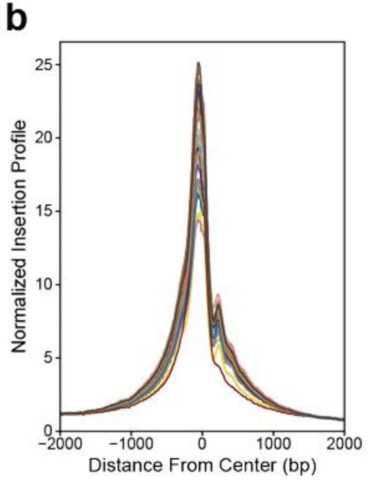
Fragments from nucleosome-bound regions depleted at TSS + enrichment of flanking regions around TSS.

Filtering out low quality cells

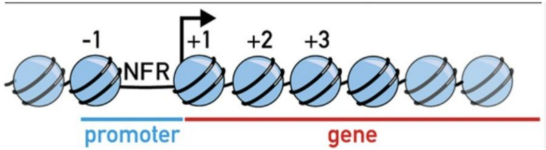
2. Signal-to-background ratio - > TSS enrichment

NFR enriched at gene TSS regions compared to other genomic regions.

1. Aggregated distribution of reads centered on the TSSs - avg accessibility
2. Extension of 2000 bp in either direction.
3. Normalized by the average read depth in the 100 bps at each of the end flanks.
4. Fold change at each position over that average read depth.
5. Increase in signal up to a peak in the middle.
6. **TSS enrichment metric** is the signal value at the center of the distribution after this normalization.



`plotTSSEnrichment()`



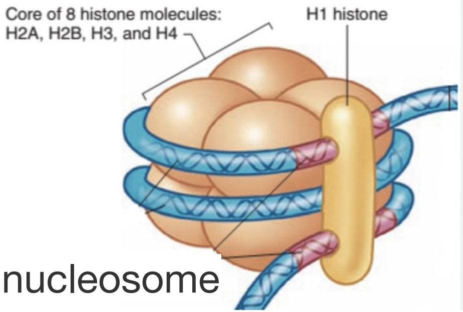
Good quality cells TSS > 4

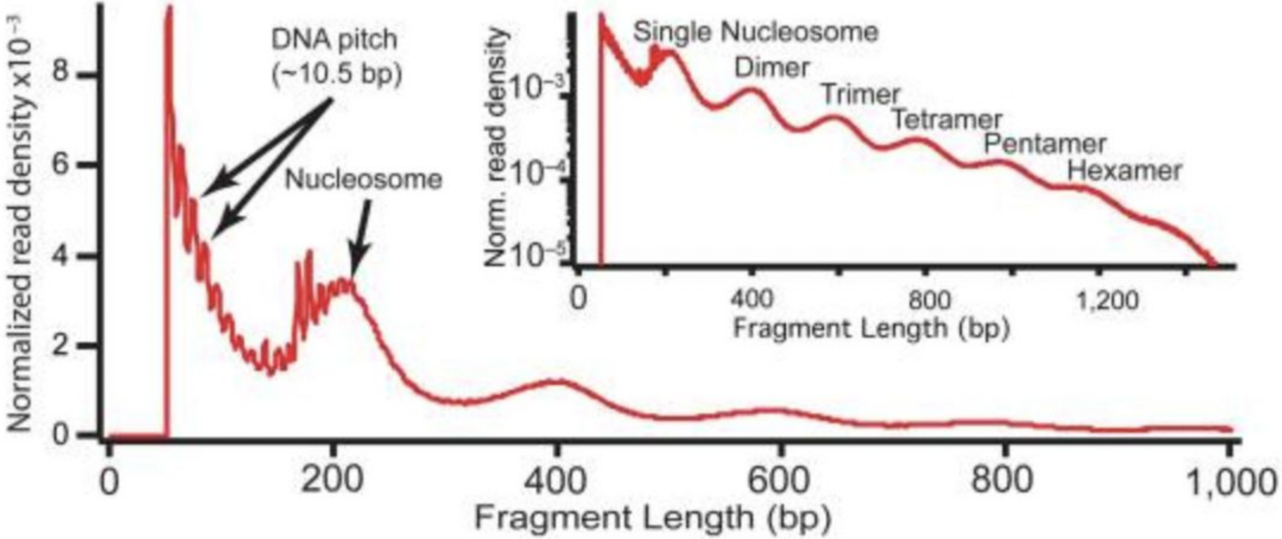
Low signal-to-background ratio -> dead or dying cells (de-chromatinized DNA allows for random transposition genome-wide).

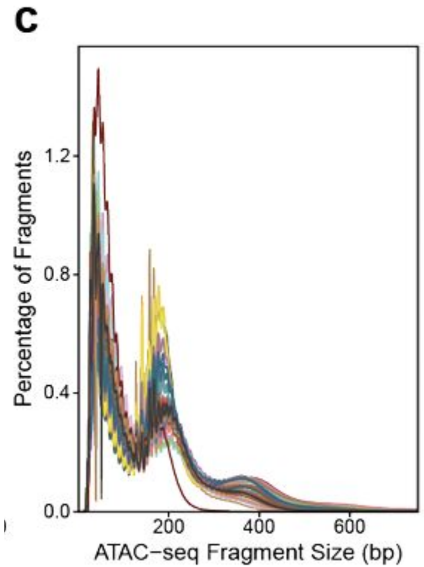
Filtering out low quality cells

3. Fragment size distribution

- Nucleosomal periodicity in the distribution of fragment sizes.
- Depletion of fragments that are the length of DNA wrapped around a nucleosome (approximately 147 bp).

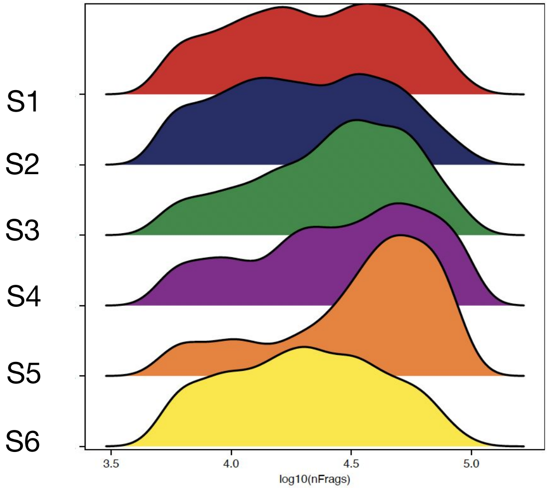




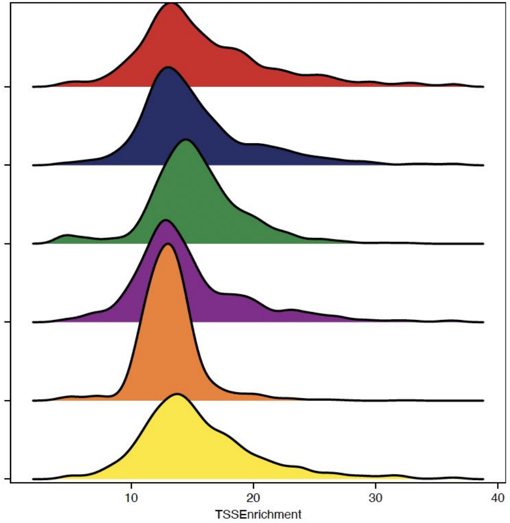


```
plotFragmentSizes(..)
```

Visualize the quality of your data

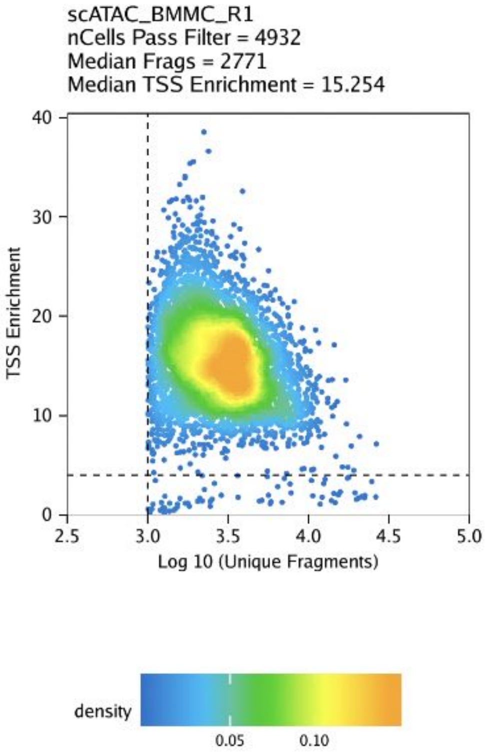


# unique fragments

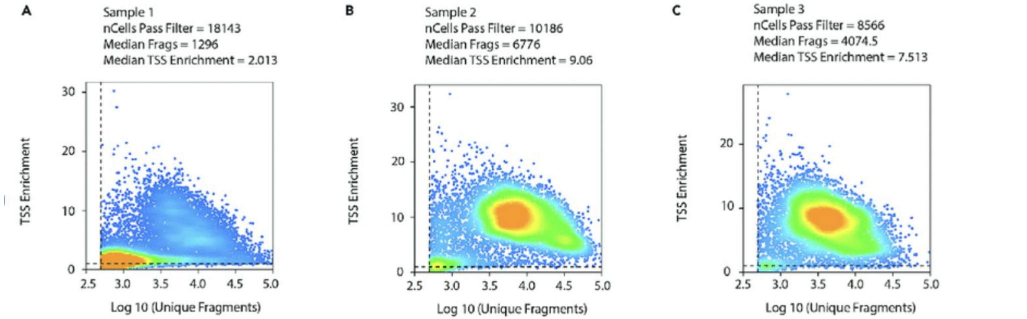


TSS enrichment

TSS enrichment vs unique fragments



scATAC_BMMC_R1
nCells Pass Filter = 4932
Median Frags = 2771
Median TSS Enrichment = 15.254



A
Sample 1
nCells Pass Filter = 18143
Median Frags = 1296
Median TSS Enrichment = 2.013

B
Sample 2
nCells Pass Filter = 10186
Median Frags = 6776
Median TSS Enrichment = 9.06

C
Sample 3
nCells Pass Filter = 8566
Median Frags = 4074.5
Median TSS Enrichment = 7.513

DOI:10.1016/j.xpro.2022.101491

To summarize the pre-processing

- Create a ArchR project using the fragments.tsv file
- In ArchR command specify the thresholds for min fragment size and TSS enrichment
- Specify to add the Tile Matrix (we will add other matrices too)
- Plots to verify the trend across samples

Break ~ 5 minutes

Workflow

From bench	**Through the bioinformatic analysis**	**To the biological interpretation**
Your cell’s regulome question	Fragments count	Embedding
Library preparation & sequencing	QC filtering low quality cells	Clustering
Demultiplex, read alignment and quantification	Layered dimensionality reduction	Find Markers - cell annotation

For more biological insight in cell’s regulome		
Peak calling	Motif Enrichment	Motif Footprinting

5. Normalization and visualization on lower dimensional space

ArchR: `addIterativeLSI(); addUMAP(); addTSNE()`

For dimensionality reduction we need to identify features that can separate cells

- Highly variable genes ??
- Most accessible features ??

Alternative normalization and dimensionality reduction

L = D - A

L: Graph Laplacian

D: Degree of each node

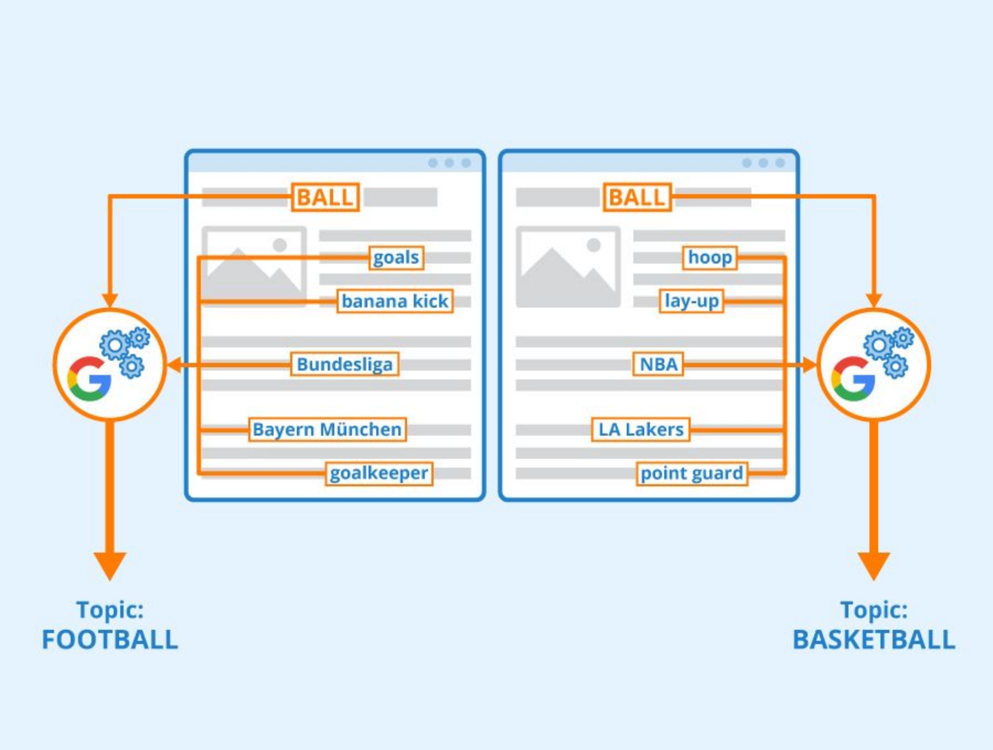
A: Node adjacency matrix

Consideration: cells look very similar.

- Lots of 0s, bias in the PCs -> misleading similarity.
- **Spectral embedding** (using eigenvectors from Graph Laplacian - created from cell-cell similarity matrix)
  - Introduced in 2021 SnapATAC, with recent 2024 updates SnapATAC2
- **Latent Semantic Indexing (LSI)**
  - Introduced in 2015, implemented in different ways by Signac and 2021 by ArchR.

Latent semantic Indexing (LSI)

To find co-occurrences of words in documents to give insights into the **topics** of those documents



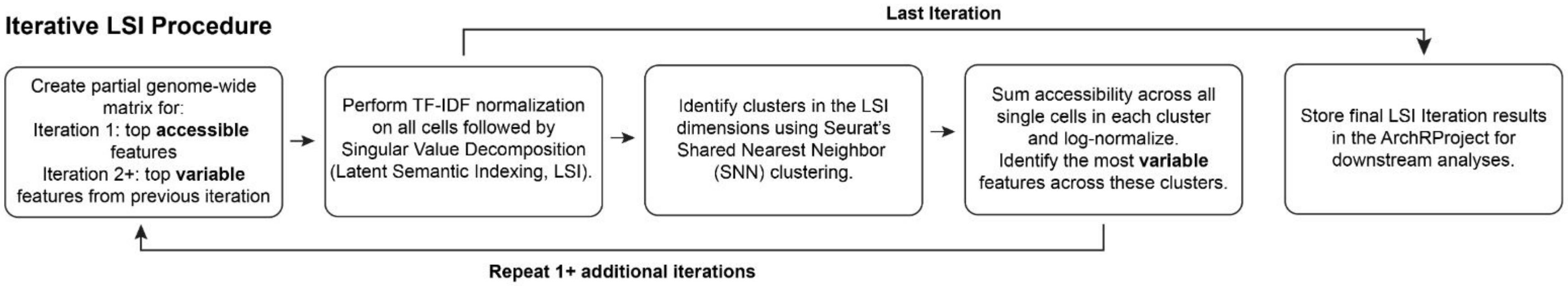
- How many times a word appears in a document?
- Term frequency of a word in a document
- Are common words (this, what, and if) informative? -> low ranking
- Inverse document frequency of the word across a set of documents.
- Use singular value decomposition (SVD) to identify patterns between words and documents.

LSI to normalize scATAC-seq data

- Documents -> cells; Words -> regions/peaks
- Help to identify features that are more **“specific”** rather than **commonly accessible.**
- Normalize for term frequency followed by depth norm to a constant 10,000
- Normalized by the inverse document frequency -> weighted features
- The term frequency-inverse document frequency (TF-IDF) matrix reflects how important a *region/peak* is to a *sample/cell*
- Low dimensional space -> singular value decomposition (SVD) of the most *valuable* information across cells/samples

Excel sheet illustrating LSI computations

LSI to process scATAC-seq data



Iterative LSI Procedure

Create partial genome-wide matrix for:
Iteration 1: top **accessible** features
Iteration 2+: top **variable** features from previous iteration

Perform TF-IDF normalization on all cells followed by Singular Value Decomposition (Latent Semantic Indexing, LSI).

Identify clusters in the LSI dimensions using Seurat's Shared Nearest Neighbor (SNN) clustering.

Sum accessibility across all single cells in each cluster and log-normalize.
Identify the most **variable** features across these clusters.

Store final LSI Iteration results in the ArchRProject for downstream analyses.

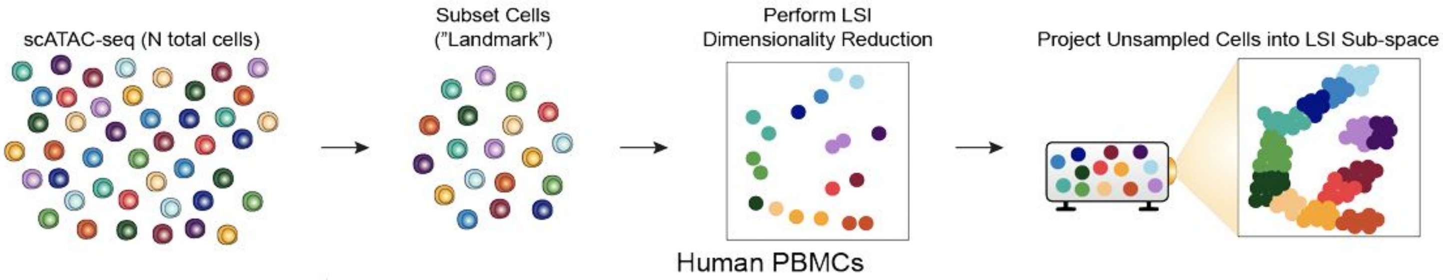
Last Iteration

Repeat 1+ additional iterations

- Identify low specific clusters of cells (set a low resolution) based on the most accessible 500bp tiles using TF-IDF normalization and the Seurat clustering approach
- Derive the most variables features across clusters based on averaged accessibility within a cluster
- Use the most variable feature as in scRNA-seq and iterate N times
- Tune the N iterations if batch effects are present

```
addIterativeLSI()
```

Estimated LSI approach for large dataset



Similar to Iterative LSI but

- Use a subset of cells to define the sub-space
- Normalize the remaining cells and project to the sub-space.

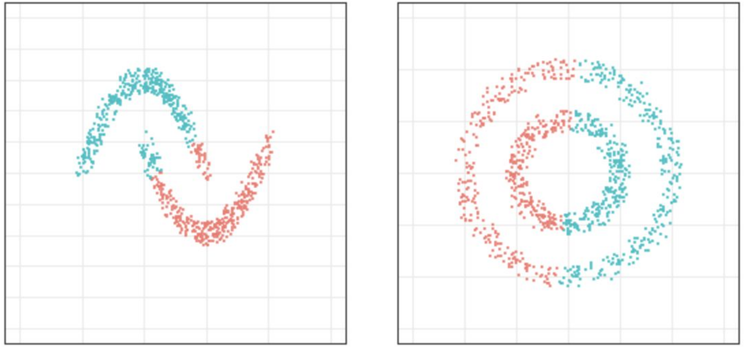
For very large datasets

- Speeds up dimensionality reduction
- Decrease the granularity of the data
- Similar to landmark diffusion maps (LDM) implemented by SnapATAC but required for > 25k cells
- ArchR required for >200k cells otherwise Iterative LSI is good

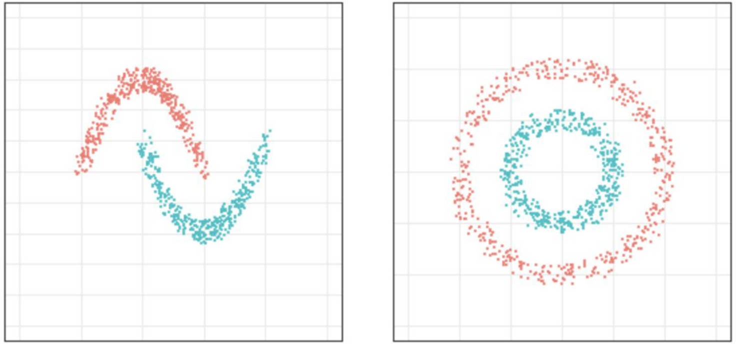
Spectral embedding

- Cells are considered nodes in a graph
- Edges between cells reflect cells that are immediately to each other
- Spectral embedding is a partitioning technique of the resulting graph using the eigenvalue decomposition of the Graph Laplacian
- The embedding has been used for clustering

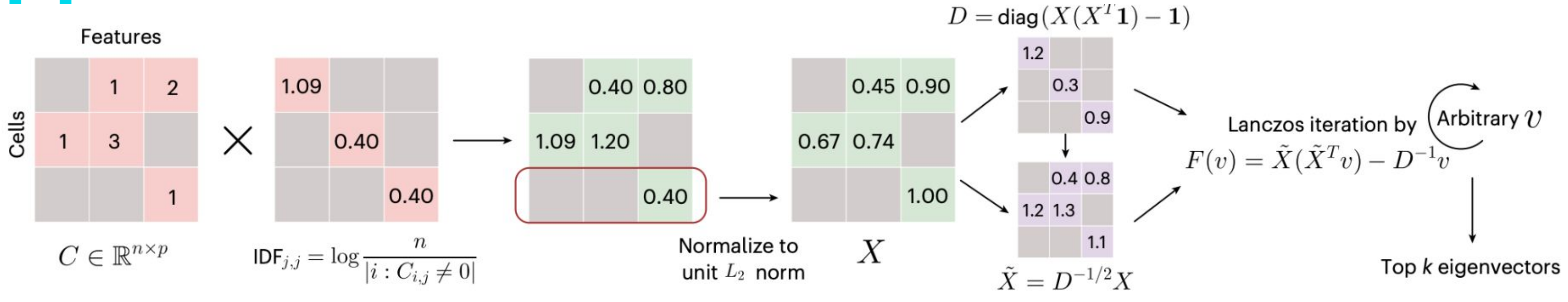
K-Means clustering



Spectral clustering



SnapATAC2 spectral embedding approach


Features
Cells
1 2
1 3
1
×
1.09
0.40
0.40
0.40 0.80
1.09 1.20
0.40
0.45 0.90
0.67 0.74
1.00
$C \in \mathbb{R}^{n \times p}$
$\text{IDF}_{j,j} = \log \frac{n}{|i : C_{i,j} \neq 0|}$
Normalize to unit L_2 norm
X
$D = \text{diag}(X(X^T\mathbf{1}) - \mathbf{1})$
1.2
0.3
0.9
0.4 0.8
1.2 1.3
1.1
$\tilde{X} = D^{-1/2}X$
Lanczos iteration by
Arbitrary v
$F(v) = \tilde{X}(\tilde{X}^T v) - D^{-1}v$
Top k eigenvectors

feature scaling with inverse term frequency (*Downweights features that are accessible in all cells*)

row-wise L2 norm normalization

Identification of eigenvectors of Graph Laplacian created using normalized matrix X

Graph-based spectral embedding leverages eigenvalues and eigenvectors of the pairwise cell similarity matrix to perform dimensionality reduction - > time consuming

Matrix-free spectral embedding -> computes eigenvectors using the Lanczos algorithm wo constructing a full similarity matrix

[A fast, scalable and versatile tool for analysis of single-cell omics data, Nature Methods 2024]()

Knowledge check 3

Which dimensionality reduction method is able to preserve the **local structure of cells (neighborhood)** in the low dimensional space?

1. PCA
2. UMAP
3. Both

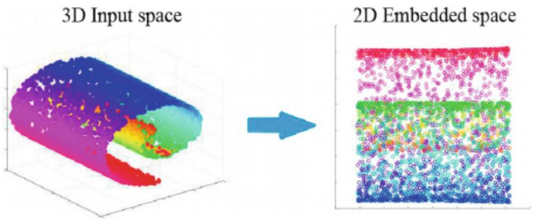
Embeddings methods for visualization: UMAP or t-SNE

- Visualize in low dimensional space preserving the local vs global structure (i.e distance) as much as possible.

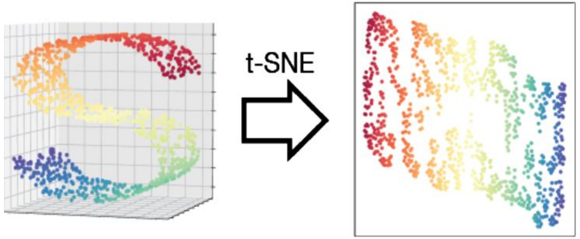
- Linear (PCA, MDS) - Euclidean distance

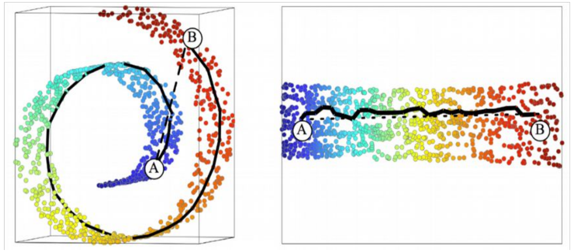
- Non linear combinations methods: t-SNE (t-Distributed Stochastic Neighbor Embedding) and UMAP (Uniform Manifold Approximation and Projection) - transforming Euclidean distances into “probability of being neighbor”

Linear projection

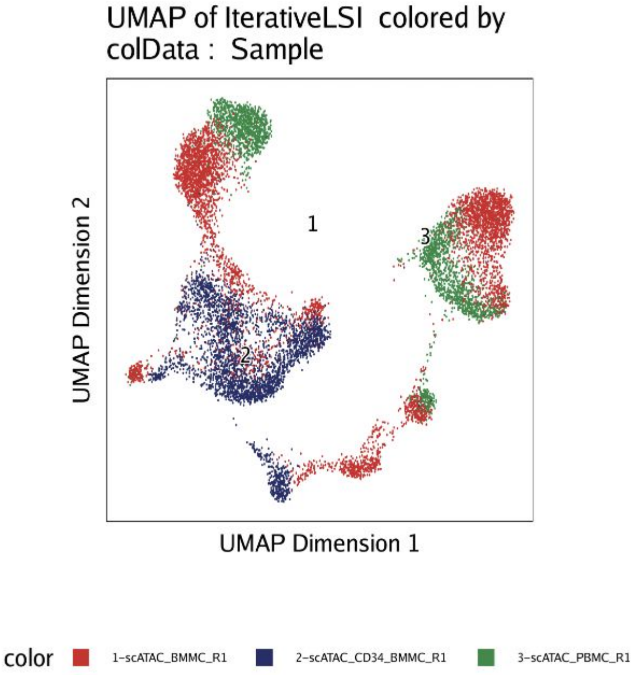


Non linear projection



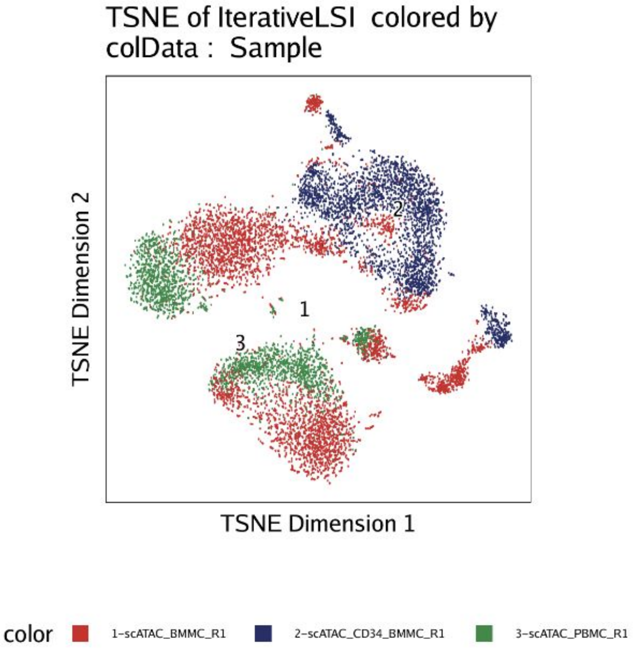


UMAP/tSNE visualization of IterativeLSI data



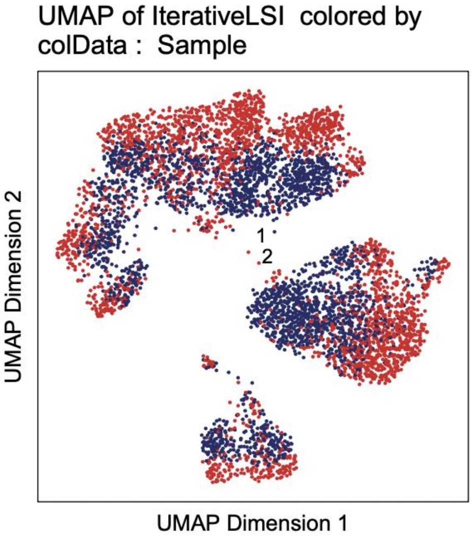
UMAP of IterativeLSI colored by colData : Sample

`addUMAP()`



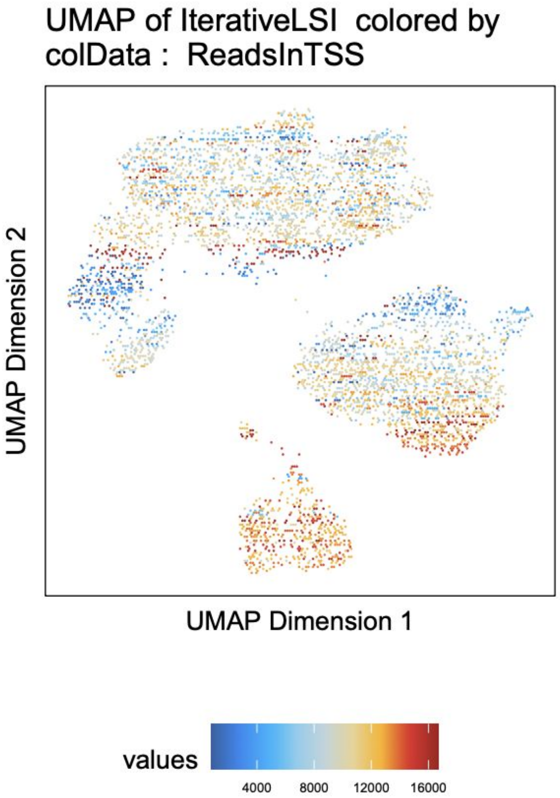
TSNE of IterativeLSI colored by colData : Sample

`addTSNE()`



UMAP of IterativeLSI colored by colData : Sample

ArchR generates metrics plots

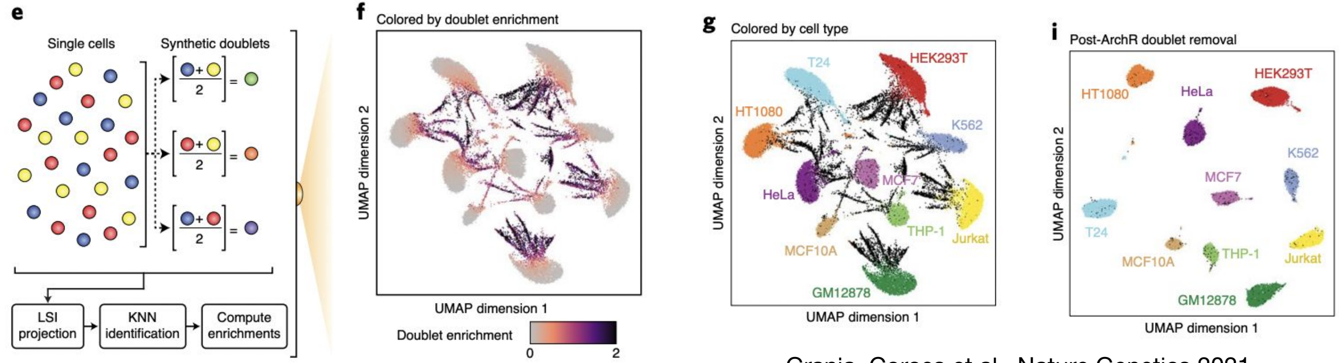


- Reads in Promoter
- Reads in Blacklist
- NucleosomeRatio

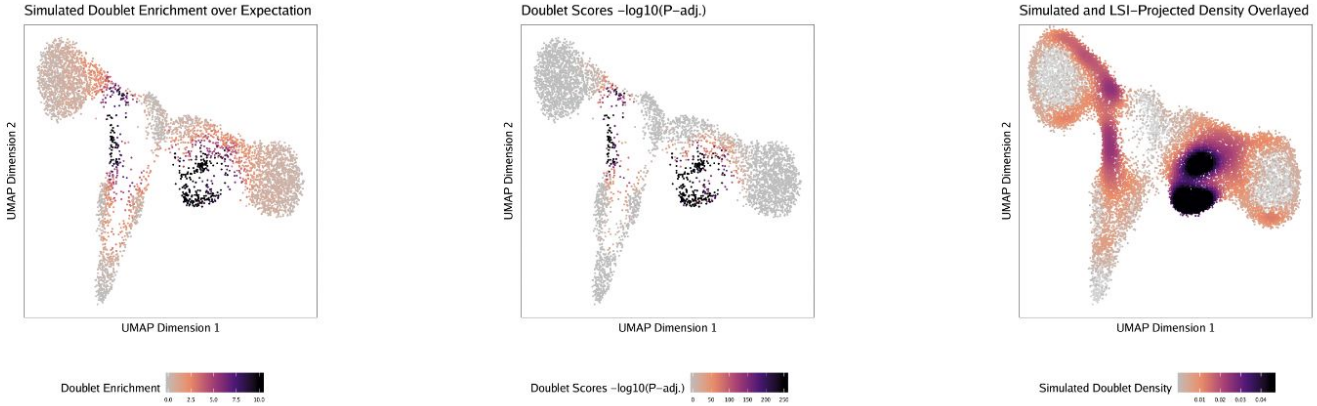
  -> (nDiFrags + nMultiFrags) / nMonoFrags

- PromoterRatio

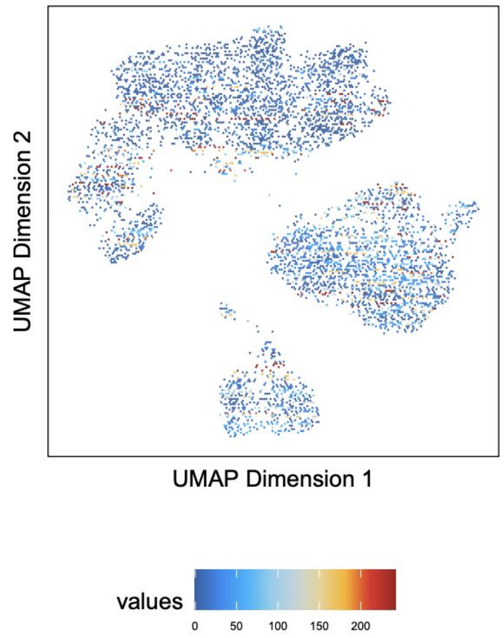
Doublets detection and removal



Granja, Corces et al., Nature Genetics 2021



UMAP of IterativeLSI colored by colData : DoubletScore



Suggestion: check data after and before doublet removal

```
addDoubletScores() ; filterDoublets()
```

Workflow

From bench	Through the bioinformatic analysis	To the biological interpretation
Your cell's regulome question	Fragments count	Embedding
Library preparation & sequencing	QC filtering low quality cells	Clustering
Demultiplex, read alignment and quantification	Layered dimensionality reduction	Find Markers - cell annotation

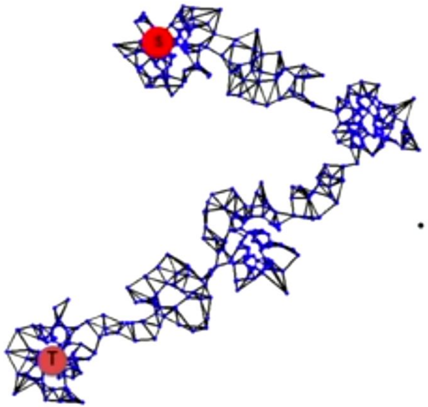
For more biological insight in cell's regulome		
Peak calling	Motif Enrichment	Motif Footprinting

6. Clustering and cell type annotation based on feature markers

ArchR: `addClusters(); getMarkerFeatures()`

ArchR use Seurat approach to find clusters

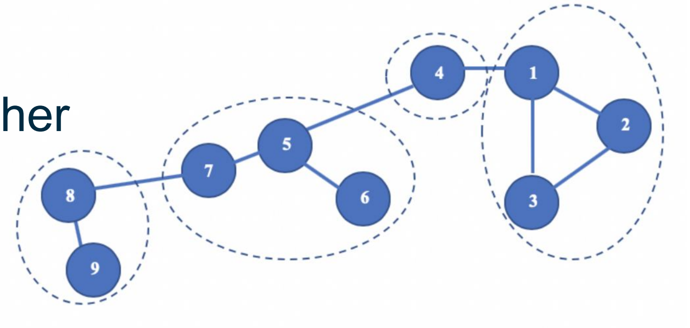
1. Build shared nearest neighbor graph



- Build a k-nearest neighbor graph
- Keep only edges between cells that share a neighbor
- Adjust the edge weights between any two cells based on similarity

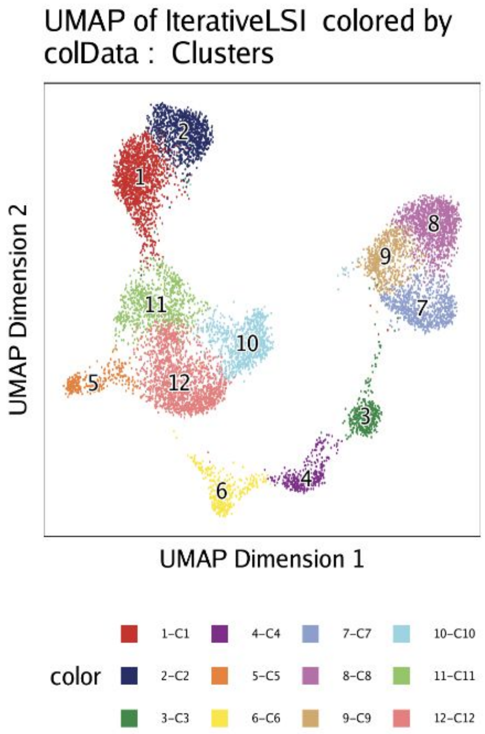
= Shared nearest neighbor graph

2. Community detection



- Cluster of cells more connected than with cells of other communities
- graph-based clustering detects clusters of arbitrary structures
- optimizes modularity - **resolution -** 0.4-1.2 typically returns good results for 3K single-cell datasets

Clustering of IterativeLSI data using Seurat clusters

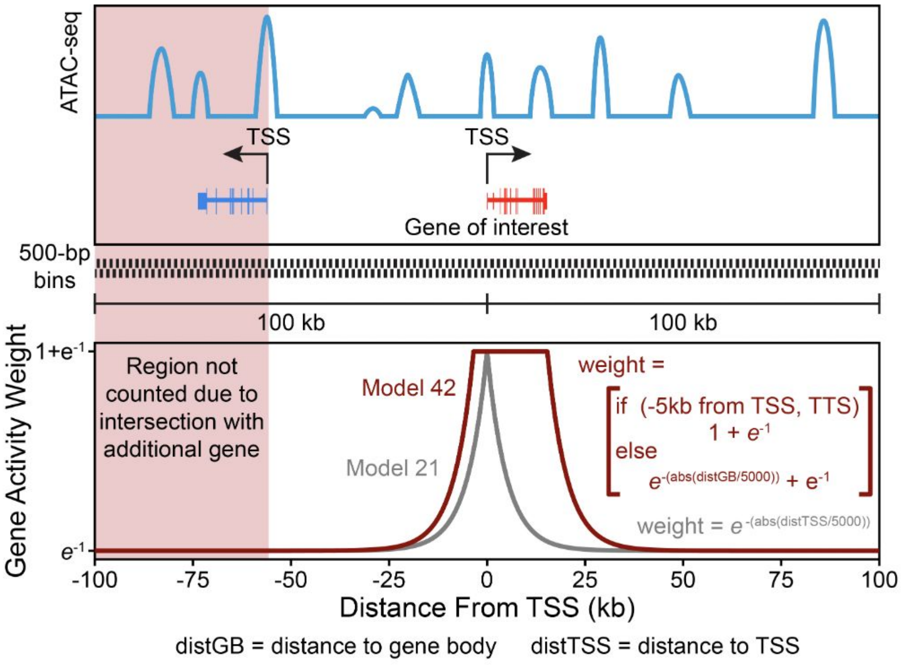


```
addClusters(..., reducedDims = "IterativeLSI", method = 'Seurat",
resolution = N) N[0.4-1.2]
```

How to identify marker genes having accessibility data

- Gene expression is not known.
- **Are the regulatory elements of a gene accessible?**
  - **-> high gene expression, high activity.**
- **Infer the gene expression using the distance between the tile and the gene -> *GENE SCORE MATRIX***
- Validate on scRNAseq and scATACseq datasets

Best model to infer gene score matrix



Granja, Corces et al., Nature Genetics 2021

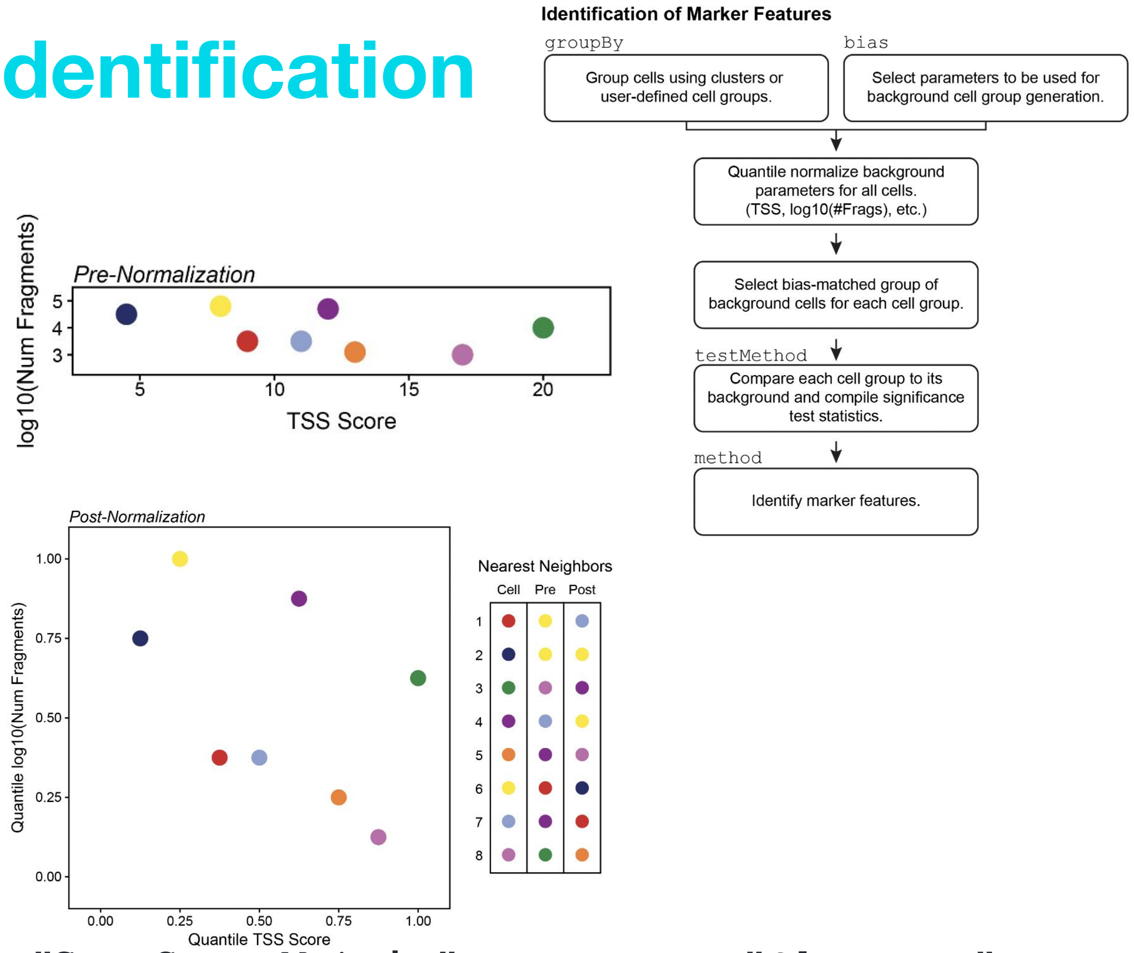
Inferring **gene score** using 56 models.

- Define gene window.
- Overlap tiles.
- Weigh distance and gene size.
- Weights multiplied by #Tn5 insertions within each tile.
- Summed across all tiles within the gene window -> **gene score**
- Depth normalized across genes

```
createArrowFiles(..,addGeneScoreMatrix = TRUE)
```

Relevant features identification

- Getting the marker features based on the cell specific high activity.
- Marker features based on pairwise comparison of two matched cell groups:
  - Select bias for matching
  - Normalization to equal variance
  - Nearest neighbor
- Significant markers one group vs matched group



Identification of Marker Features

groupBy: Group cells using clusters or user-defined cell groups.

bias: Select parameters to be used for background cell group generation.

Quantile normalize background parameters for all cells. (TSS, log10(#Frags), etc.)

Select bias-matched group of background cells for each cell group.

testMethod: Compare each cell group to its background and compile significance test statistics.

method: Identify marker features.

```
getMarkerFeatures(useMatrix = "GeneScoreMatrix",groupBy = "Clusters",
 bias = c("TSSEnrichment", "log10(nFrags)"))
```

To summarize so far

- Create a ArchR project using the fragments.tsv file
- In ArchR command specify the thresholds for min fragment size and TSS enrichment
- Specify to add the Tile Matrix, GeneScore matrix
- Plots to verify the trend across samples
- Add Doublets score and filter for doublets, then check!
- Add IterativeLSI for normalization
- AddUMAP, AddClusters using IterativeLSI (on LSI space)
- Use GeneScoreMatrix to get the marker features

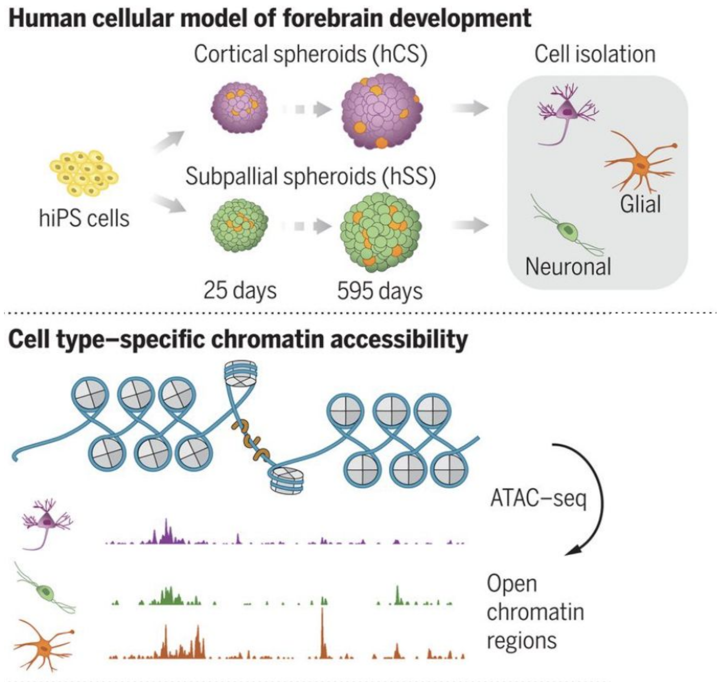
Break ~ 5 minutes

7. Advance analysis: Calling Peaks and Motif enrichment

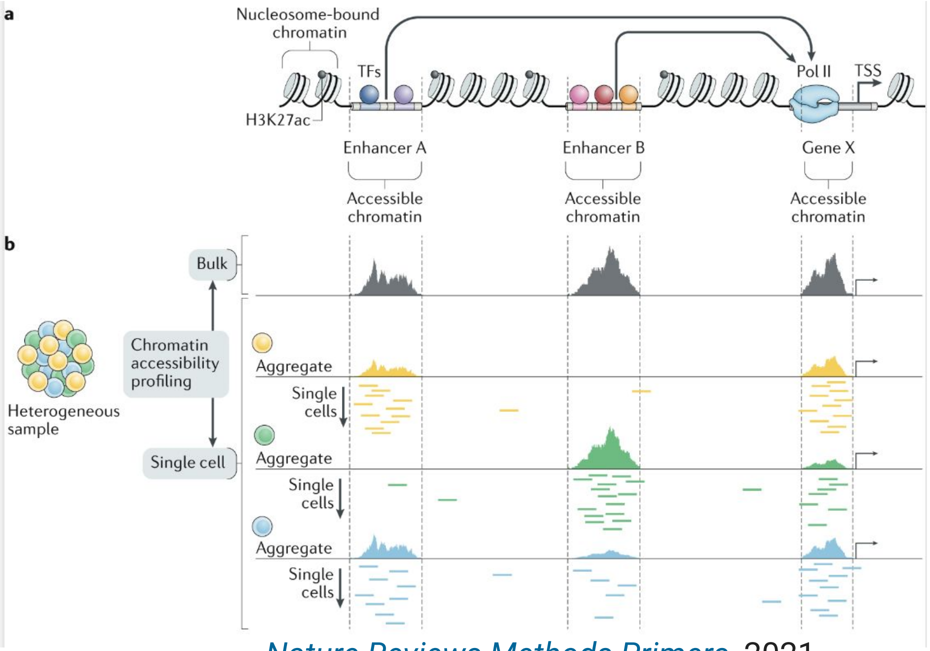
ArchR: `addGroupCoverages(); addReproduciblePeakSet(); peakAnnoEnrichment()`

Where are the cell-type specific enhancers?

- Peak-calling!


Human cellular model of forebrain development
Cortical spheroids (hCS)
Cell isolation
hiPS cells
Subpallial spheroids (hSS)
Glial
Neuronal
25 days
595 days
Cell type–specific chromatin accessibility
ATAC–seq
Open chromatin regions

Trevino et al, Science 2020



Nature Reviews Methods Primers, 2021

Motivating Ideas

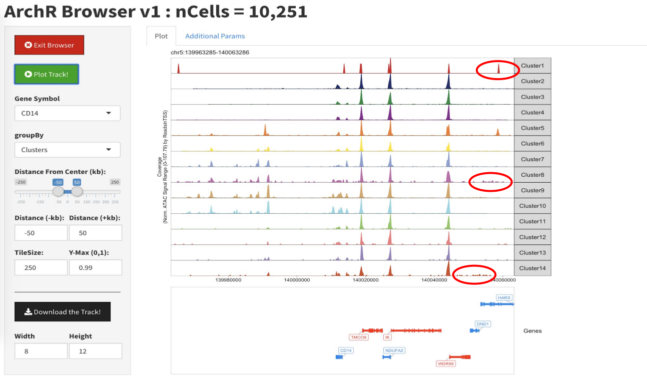
- We cannot identify enhancer regions for each cell due to the sparsity of signal

- **<u>Assumption</u>**: Cells of the same type share enhancers

- Combining information across multiple cells within a cluster will allow us to identify enhancers

- Distinguish signal from noise?

- Enter Peak calling!



ArchR Browser v1 : nCells = 10,251

Gene Symbol: CD14

groupBy: Clusters

Distance From Center (kb): -50 to 50

Distance (-kb): -50

Distance (+kb): 50

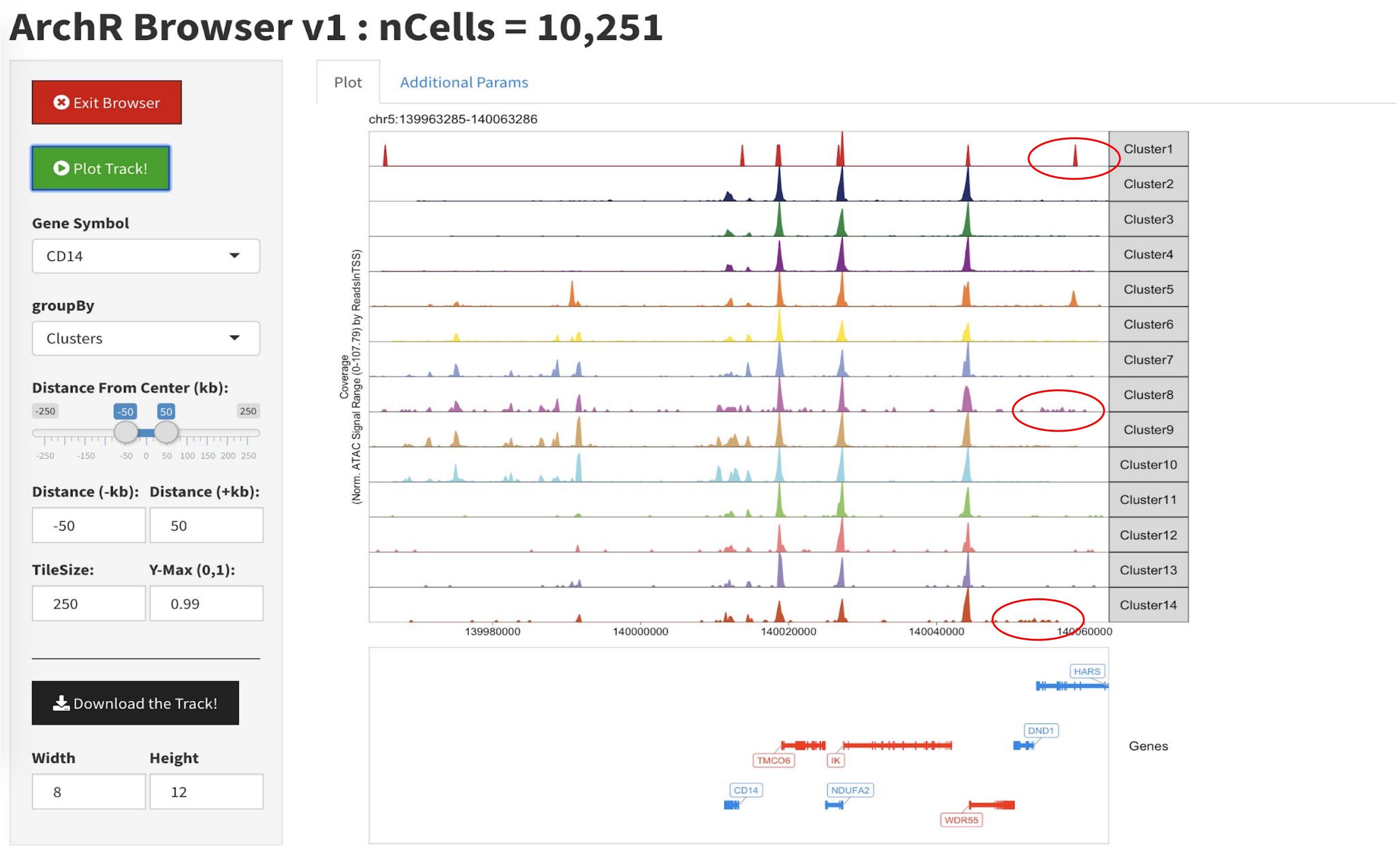
TileSize: 250

Y-Max (0,1): 0.99

Width: 8

Height: 12

What is signal and what is noise?



ArchR Browser v1 : nCells = 10,251

Gene Symbol: CD14

groupBy: Clusters

Distance From Center (kb): -50 to 50

Distance (-kb): -50

Distance (+kb): 50

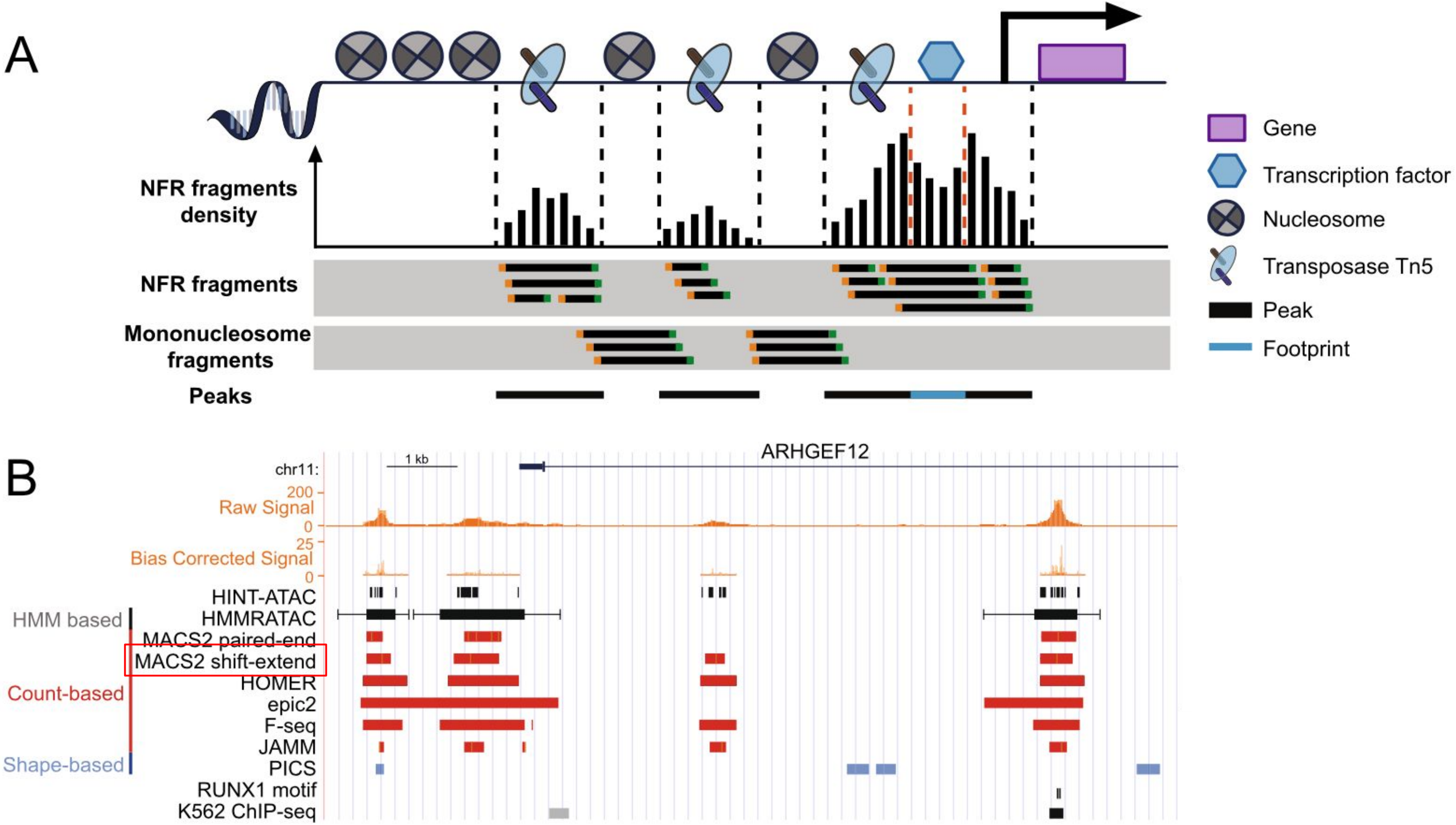
TileSize: 250

Y-Max (0,1): 0.99

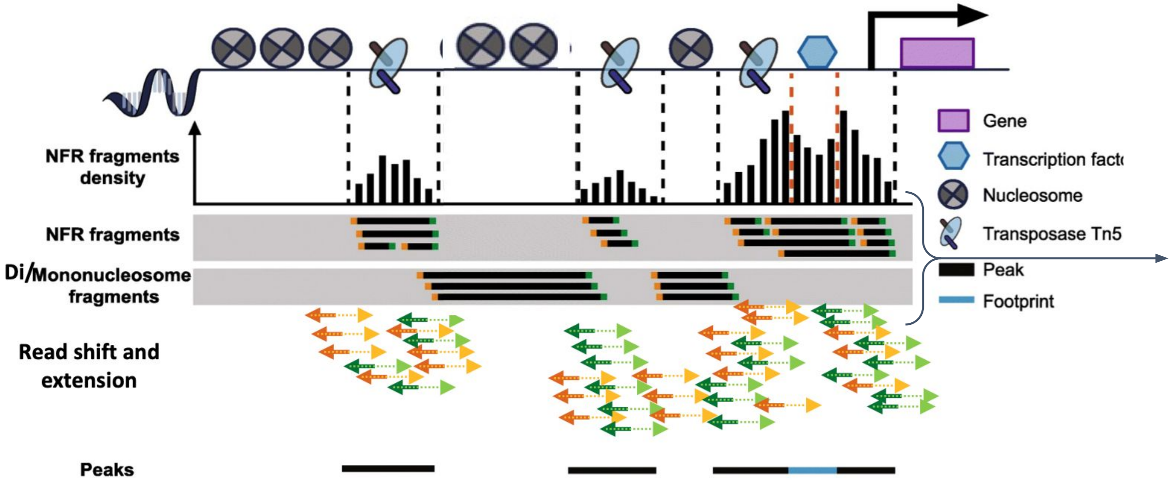
Width: 8

Height: 12

Peak calling

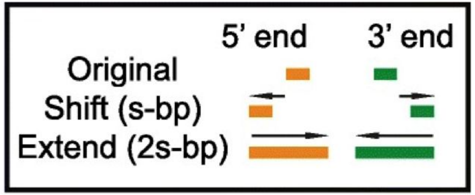


Peak calling: MACS2-shift-extend



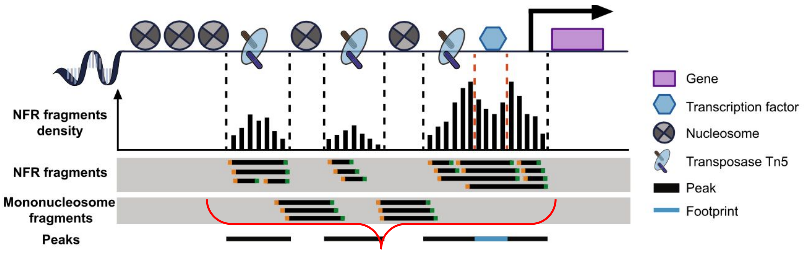
Fragments from individual cells belonging to a given cluster

Modified from Feng Yan et.al Genome Biology 21 (2020)

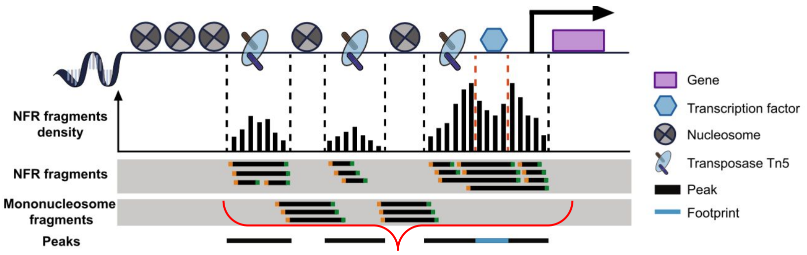


```
addReproduciblePeakSet(ArchRProj = demo_proj,
    groupBy = "Clusters",
    pathToMacs2 = findMacs2(),
    method = "q")
```

Reproducibility assessed by calling peaks separately for each sample



Fragments from individual cells of **sample1** belonging to a given cluster



Fragments from individual cells of **sample2** belonging to a given cluster

```
addGroupCoverages(ArchRProj = demo_proj,
    groupBy = "Clusters")
```

```
addReproduciblePeakSet(ArchRProj = demo_proj,
    groupBy = "Clusters",
    pathToMacs2 = findMacs2(),
    method = "q")
```

(ArchR) steps to obtain a list of all peaks

- For each cell-type/cluster
    - Keep list of non-overlapping 501 bp peaks that are
        - Reproducible across samples
        - Most significant in terms of enrichment over noise

- Will allow us to identify peak regions either unique to or common across cell-types

Peak matrix contains accessibility of each peak for each cell

	Cell_1	Cell_2	…	Cell_m
Peak_1	1	0	…	2
Peak_2	0	3	…	1
…			…	
Peak_n	0	0	…	4

Counts of insertions in each cell mapping to a given cell

Knowledge check 4

Are the counts of the number of insertions in each peak normalized across cells?

1. Yes
2. No

Knowledge check 5

What could be reasons why the same regions equally accessible in two cells have different numbers of sequenced insertions?

1. Differences in total number of mapped fragments
2. Differences in Signal-to-noise/TSS enrichment
3. Both

Use Wilcoxon test to identify differentially accessible peak regions

Cells belonging to cluster C1

Cells belonging to cluster C5

	c1	c2	c3	c4	c5	…	…	c93	c94	c95	c96	c97	c98	c99
p1	1	0	0	0	1	…	…	2	1	2	0	4	1	1
p2	…	…	…	…		…	…							
…						…	…							
pn	…					…	…							…

Cells in clusters C1 and cluster C5 are specifically chosen so they have a similar distribution of log10(nFrags) and TSS

```
getMarkerFeatures(ArchRProj = demo_proj,
useMatrix = "PeakMatrix", groupBy = "Clusters",
  bias = c("TSSEnrichment", "log10(nFrags)"),
  testMethod = "wilcoxon")
```

Use Wilcoxon test to identify differentially accessible peak regions

Cells belonging to cluster C1

Cells belonging to cluster C5

	c1	c2	c3	c4	c5	…	…	c93	c94	c95	c96	c97	c98	c99
p1	1	0	0	0	1	…	…	2	1	2	0	4	1	1
p2	…	…	…	…		…	…							
…						…	…							
pn	…					…	…							…

Peak p1 among cells in cluster C5 appear to be more accessible compared to cells in cluster C1

```
getMarkerFeatures(ArchRProj = demo_proj,
useMatrix = "PeakMatrix", groupBy = "Clusters",
  bias = c("TSSEnrichment", "log10(nFrags)"),
  testMethod = "wilcoxon")
```

Caution 1: Elevated false-positives with this test

- The previous Wilcoxon test treated all cells as being independent of each other
- Not true!
- Cells are grouped based on the sample from which they are derived
- Ignoring this cell-cell correlation results in elevated false positives
- Solution: perform pseudo-bulk of the number of fragments across all cells from each sample
- See details in scRNAseq workshop and this paper

Knowledge check 6

Assume sample-wise batch correction was used to obtain updated clusters and visualizations of the data. Will the fragments counts in the peaks also be corrected?

1. Yes
2. No

Caution 2: Avoid sample-wise batch correction

- You will not be able to distinguish differences in peak accessibility due to the biology you are studying versus those driven by uninteresting technical/batch effects

- See details in our scRNAseq workshop

Workflow

From bench	**Through the bioinformatic analysis**	**To the biological interpretation**
Your cell’s regulome question	Fragments count	Embedding
Library preparation & sequencing	QC filtering low quality cells	Clustering
Demultiplex, read alignment and quantification	Layered dimensionality reduction	Find Markers - cell annotation

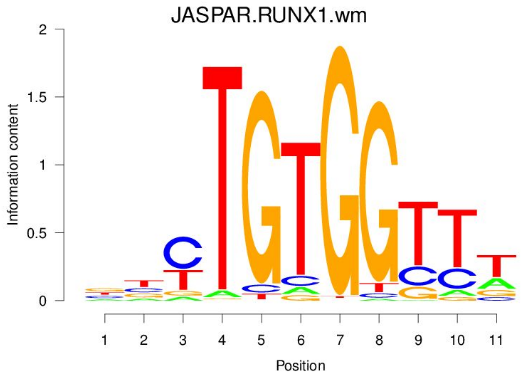
For more biological insight in cell’s regulome

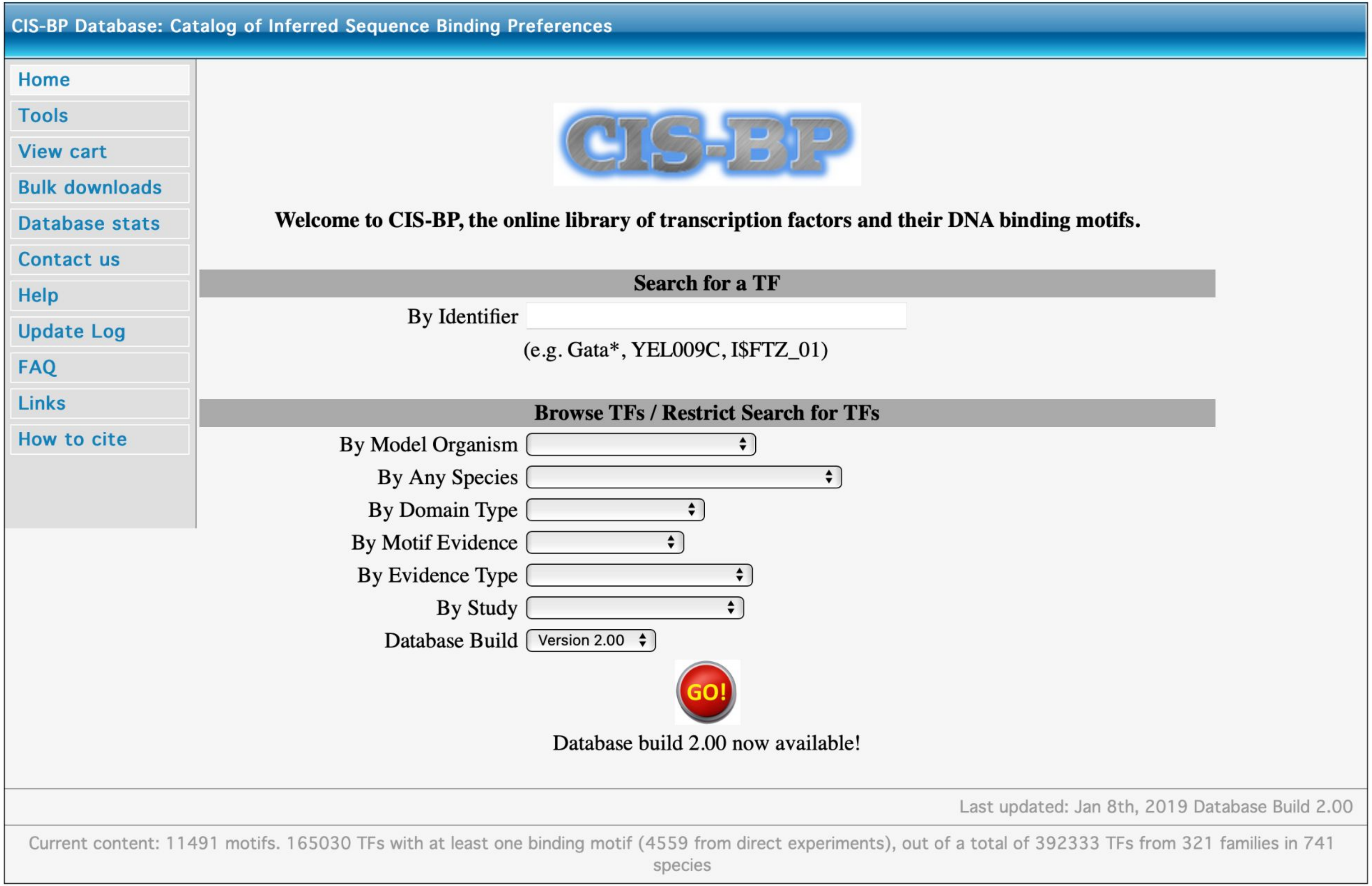
- Peak calling
- Motif Enrichment
- Motif Footprinting

Motif signatures help us identify important transcription factors

- Transcription factors/proteins bind to DNA sequences with particular sequence/motif patterns
- These motif patterns are available in several databases
- We use these patterns to identify peak regions carrying signatures of binding to particular transcription factors

```
addMotifAnnotations(ArchRProj = demo_proj,
    motifSet = "cisbp", name = "Motif")
```





CIS-BP Database: Catalog of Inferred Sequence Binding Preferences

Welcome to CIS-BP, the online library of transcription factors and their DNA binding motifs.

Search for a TF

By Identifier

(e.g. Gata*, YEL009C, I$FTZ_01)

Browse TFs / Restrict Search for TFs

By Model Organism

By Any Species

By Domain Type

By Motif Evidence

By Evidence Type

By Study

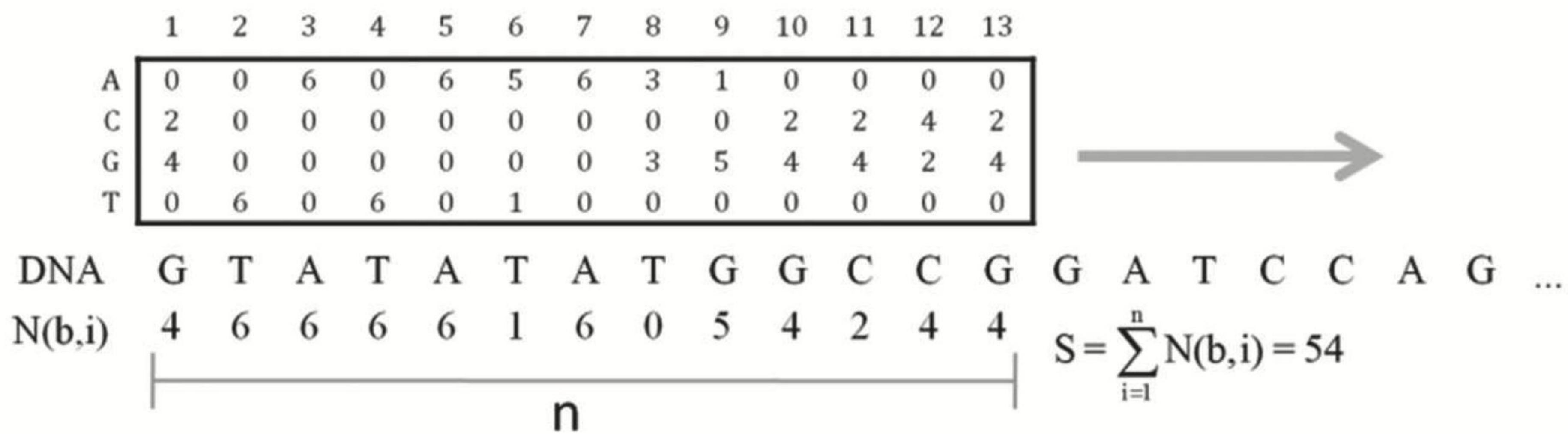
Database Build Version 2.00

Database build 2.00 now available!

Last updated: Jan 8th, 2019 Database Build 2.00

Current content: 11491 motifs. 165030 TFs with at least one binding motif (4559 from direct experiments), out of a total of 392333 TFs from 321 families in 741 species

Identifying motif locations in peak regions by scoring with Position Weight Matrices (PWM)



	1	2	3	4	5	6	7	8	9	10	11	12	13
A	0	0	6	0	6	5	6	3	1	0	0	0	0
C	2	0	0	0	0	0	0	0	0	2	2	4	2
G	4	0	0	0	0	0	0	3	5	4	4	2	4
T	0	6	0	6	0	1	0	0	0	0	0	0	0

DNA G T A T A T A T G G C C G G A T C C A G ...

N(b,i) 4 6 6 6 6 1 6 0 5 4 2 4 4

$$S = \sum_{i=1}^{n} N(b,i) = 54$$

<u>Example:</u> TATA-box motif PWM score calculation for given DNA sequence

<u>Estimate the significance of the score</u> assuming the prior expectations of frequencies of each of the 4 nucleotides (e.g., in all promoter regions of the organism)

Compute Motif presence absence matrix per peak

	Motif_1	Motif_2	…	Motif_k
Peak_1	TRUE	TRUE	…	FALSE
Peak_2	FALSE	FALSE	…	TRUE
…			…	
Peak_n	FALSE	FALSE	…	TRUE

Fisher’s test to identify motifs enriched in one set of peaks versus others

- **Universe of all peaks** identified
- **Subset 1** : peaks identified as differentially present
- **Subset 2** : peaks identified to have a given motif
-
- Is there a significant overlap between these two subsets?
- Answer: Fisher’s exact or the hypergeometric test or ORA

```
peakAnnoEnrichment(ArchRProj = demo_proj,
    seMarker = markersPeaks,
    ArchRProj = demo_proj,
    peakAnnotation = "Motif")
```

Tomorrow, at the end of the workshop (or today if not attending)

please take our survey:

https://www.surveymonkey.com/r/F75J6VZ

We use your suggestions to improve!

Introduction to scATAC-seq Data Analysis

November, 14-15 2024

Reuben Thomas
Michela Traglia
Ayushi Agrawal

GLADSTONE INSTITUTES
SCIENCE OVERCOMING DISEASE

Workshop organization

- **Session 1 (Thursday, 1pm-4pm)**

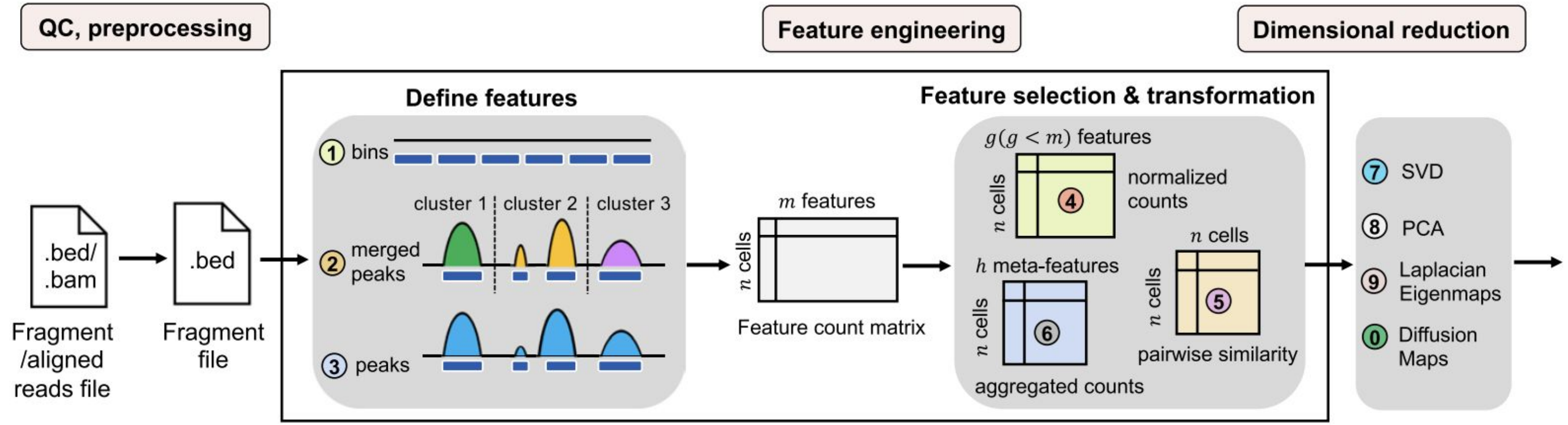
1. Cell regulome and ATAC-seq
2. Technology
3. From sequencer to fragments file
4. Pre-processing and QC
   - Break
5. Normalization, Dimensionality reduction, embedding
6. Clustering and cell type annotation based on feature markers
   - Break
7. Advance analysis: Calling Peaks and Motif enrichment

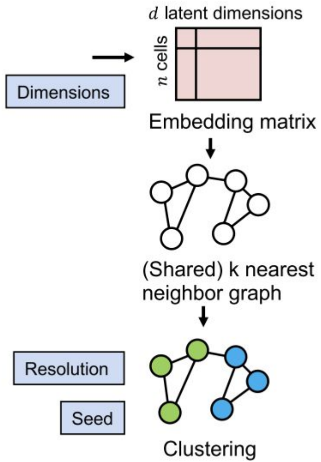
- **Session 2 (Friday, 1-4 pm)**

8. Intro to ArchR
   - Demo

Integration scRNA-seq and scATAC-seq workshop - Nov 22 at 1pm - Register here

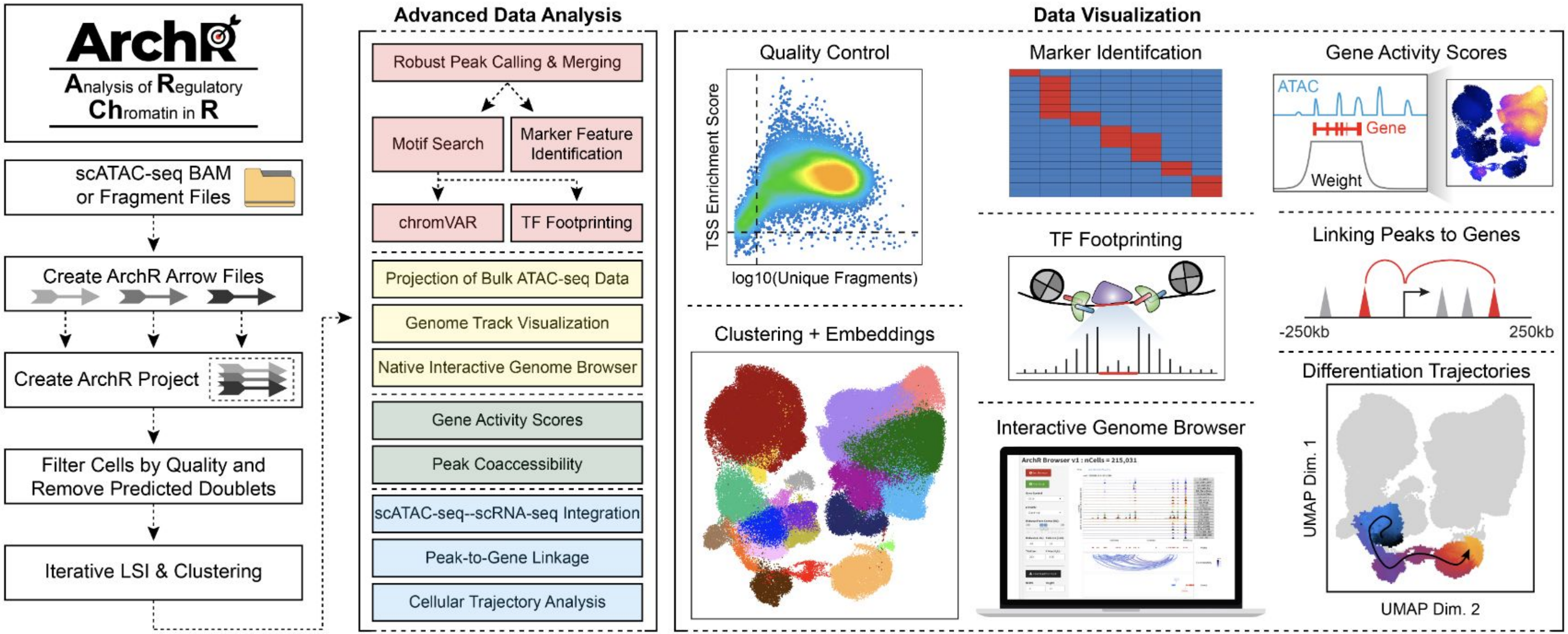
Tools - overview





R lang	A Aggregation: ② + ⑥ + ⑧
R lang	B SnapATAC: ① (5000 bp) +⑤ (Jaccard distance) +⓪
R lang	C ArchR_peaks: ② + ④ + ⑦
R lang	D ArchR_tiles: ①(500 bp) + ④+⑦
R lang	E Signac_all_cell_peaks: ③ + ④ + ⑦
R lang	F Signac_by_cluster_peaks: ② + ④ + ⑦
Python	G SnapATAC2_cosine: ① (500 bp) + ⑤ (cosine distance) + ⑨
Python	H SnapATAC2_jaccard: ① (500 bp) + ⑤ (Jaccard distance) + ⑨

8. ArchR - overview

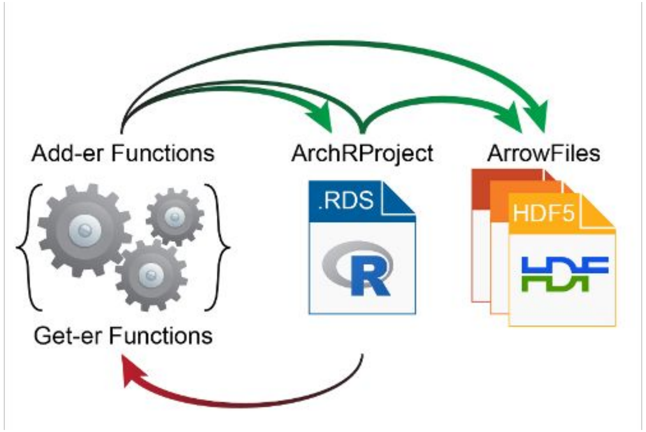

ArchR
Analysis of Regulatory Chromatin in R
scATAC-seq BAM or Fragment Files
Create ArchR Arrow Files
Create ArchR Project
Filter Cells by Quality and Remove Predicted Doublets
Iterative LSI & Clustering
Advanced Data Analysis
Robust Peak Calling & Merging
Motif Search
Marker Feature Identification
chromVAR
TF Footprinting
Projection of Bulk ATAC-seq Data
Genome Track Visualization
Native Interactive Genome Browser
Gene Activity Scores
Peak Coaccessibility
scATAC-seq--scRNA-seq Integration
Peak-to-Gene Linkage
Cellular Trajectory Analysis
Data Visualization
Quality Control
TSS Enrichment Score
log10(Unique Fragments)
Marker Identifcation
Gene Activity Scores
ATAC
Gene
Weight
Clustering + Embeddings
TF Footprinting
Linking Peaks to Genes
-250kb
250kb
Interactive Genome Browser
ArchR Browser v1 : nCells = 215,031
Differentiation Trajectories
UMAP Dim. 1
UMAP Dim. 2

What is an Arrow file / ArchRProj?

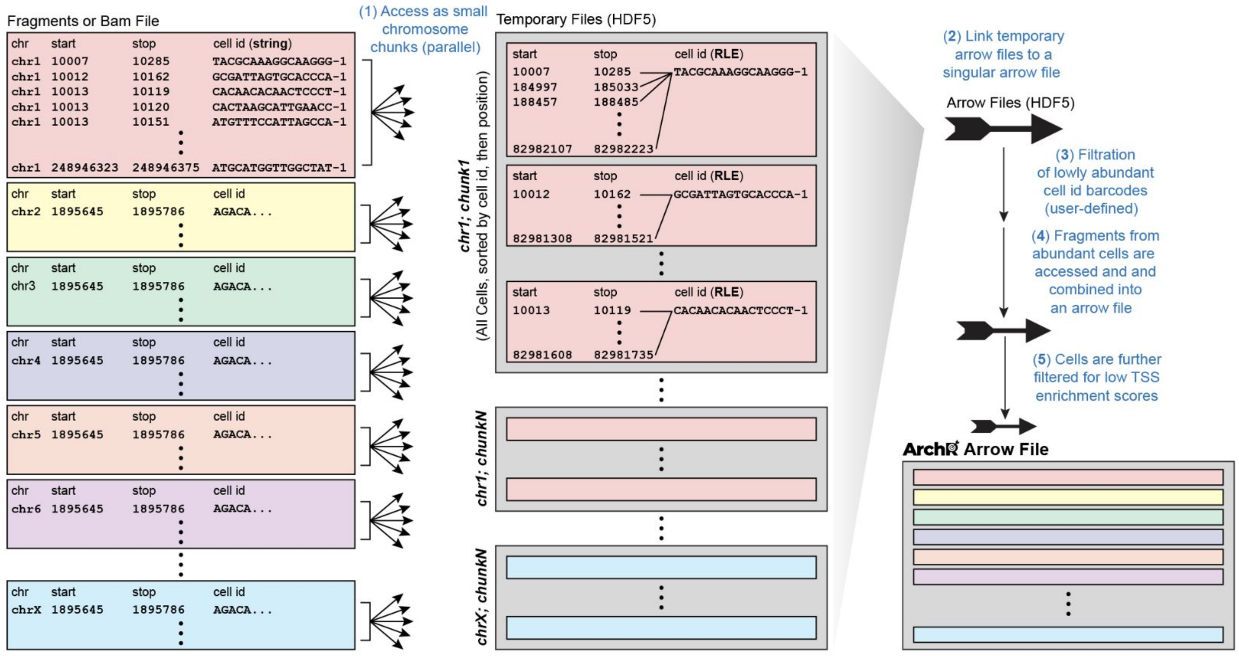
- accessible reads and arrays are organized within
- Used as input for an ArchRProject
- ArchRProject stores the locations of these Arrow files and

extracts their cell-centric metadata

- Arrow file - Stored on disk
- ArchRProj - Stored in memory



ArchR Arrow file creation



Fragments or Bam File

chr	start	stop	cell id (**string**)
chr1	10007	10285	TACGCAAAGGCAAGGG-1
chr1	10012	10162	GCGATTAGTGCACCCA-1
chr1	10013	10119	CACAACACAACTCCCT-1
chr1	10013	10120	CACTAAGCATTGAACC-1
chr1	10013	10151	ATGTTTCCATTAGCCA-1
		⋮	
chr1	248946323	248946375	ATGCATGGTTGGCTAT-1

chr	start	stop	cell id
chr2	1895645	1895786	AGACA...
		⋮	

chr	start	stop	cell id
chr3	1895645	1895786	AGACA...
		⋮	

chr	start	stop	cell id
chr4	1895645	1895786	AGACA...
		⋮	

chr	start	stop	cell id
chr5	1895645	1895786	AGACA...
		⋮	

chr	start	stop	cell id
chr6	1895645	1895786	AGACA...
		⋮	

⋮

chr	start	stop	cell id
chrX	1895645	1895786	AGACA...
		⋮	

(1) Access as small chromosome chunks (parallel)

Temporary Files (HDF5)

chr1; chunk1 (All Cells, sorted by cell id, then position)

start	stop	cell id (**RLE**)
10007	10285	TACGCAAAGGCAAGGG-1
184997	185033	
188457	188485	
	⋮	
82982107	82982223	

start	stop	cell id (**RLE**)
10012	10162	GCGATTAGTGCACCCA-1
	⋮	
82981308	82981521	

⋮

start	stop	cell id (**RLE**)
10013	10119	CACAACACAACTCCCT-1
	⋮	
82981608	82981735	

(**2**) Link temporary arrow files to a singular arrow file

Arrow Files (HDF5)

(**3**) Filtration of lowly abundant cell id barcodes (user-defined)

(**4**) Fragments from abundant cells are accessed and and combined into an arrow file

(**5**) Cells are further filtered for low TSS enrichment scores

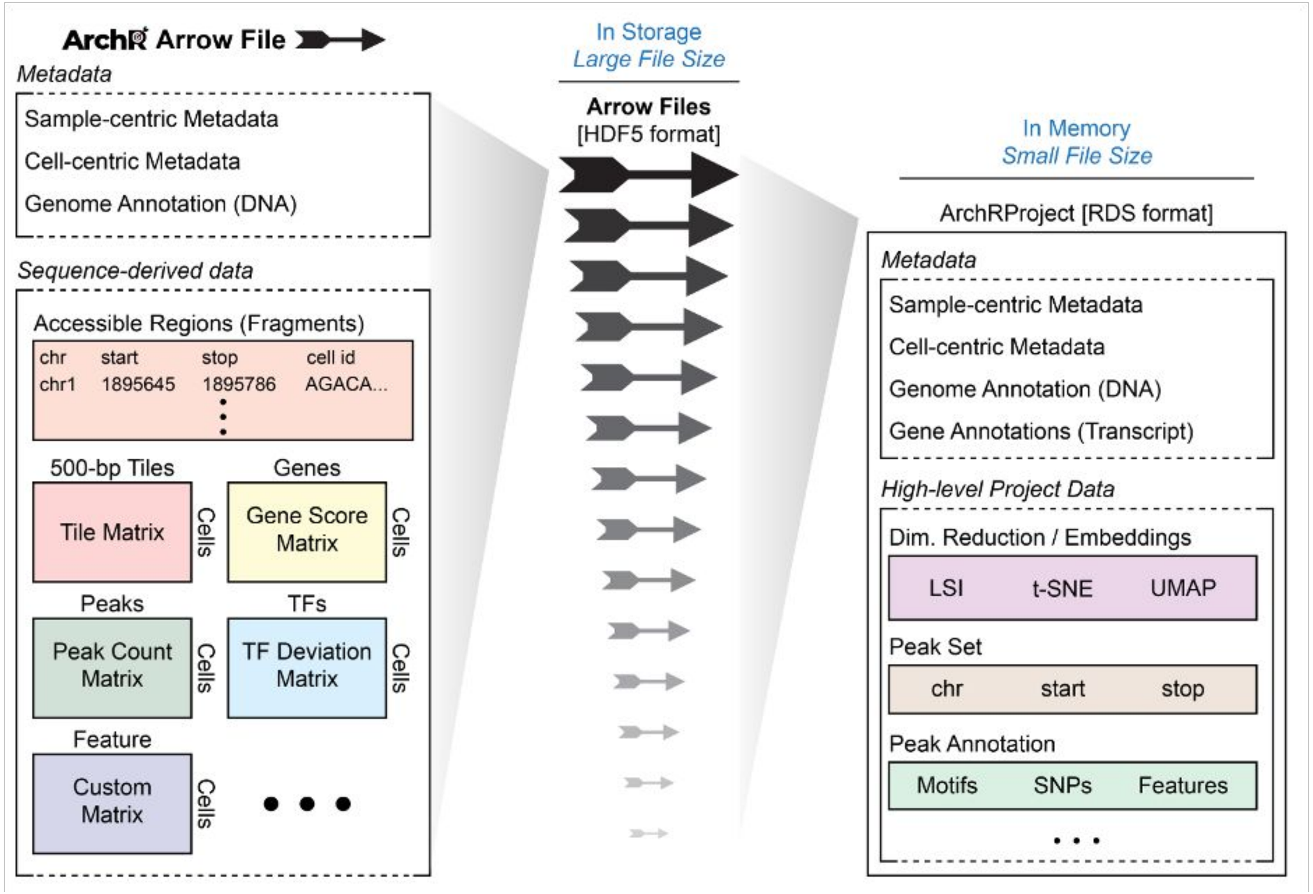
ArchR Arrow File

- All cells represented
- Large tab-delimited text file
- Sorted by chromosome
- Large string size due to cell id

- All cells represented
- Small temporary HDF5 files
- No chromosome string
- Sorted by cell id as run-length encoding

- Only cells passing QC
- Small HDF5 file
- No chromosome string
- Sorted by cell id (RLE)

File infrastructure



ArchR Arrow File

Metadata

Sample-centric Metadata

Cell-centric Metadata

Genome Annotation (DNA)

Sequence-derived data

Accessible Regions (Fragments)

chr	start	stop	cell id
chr1	1895645	1895786	AGACA...
	⋮		

500-bp Tiles: Tile Matrix (Cells)

Genes: Gene Score Matrix (Cells)

Peaks: Peak Count Matrix (Cells)

TFs: TF Deviation Matrix (Cells)

Feature: Custom Matrix (Cells)

● ● ●

In Storage
Large File Size

Arrow Files
[HDF5 format]

In Memory
Small File Size

ArchRProject [RDS format]

Metadata

Sample-centric Metadata

Cell-centric Metadata

Genome Annotation (DNA)

Gene Annotations (Transcript)

High-level Project Data

Dim. Reduction / Embeddings

LSI	t-SNE	UMAP

Peak Set

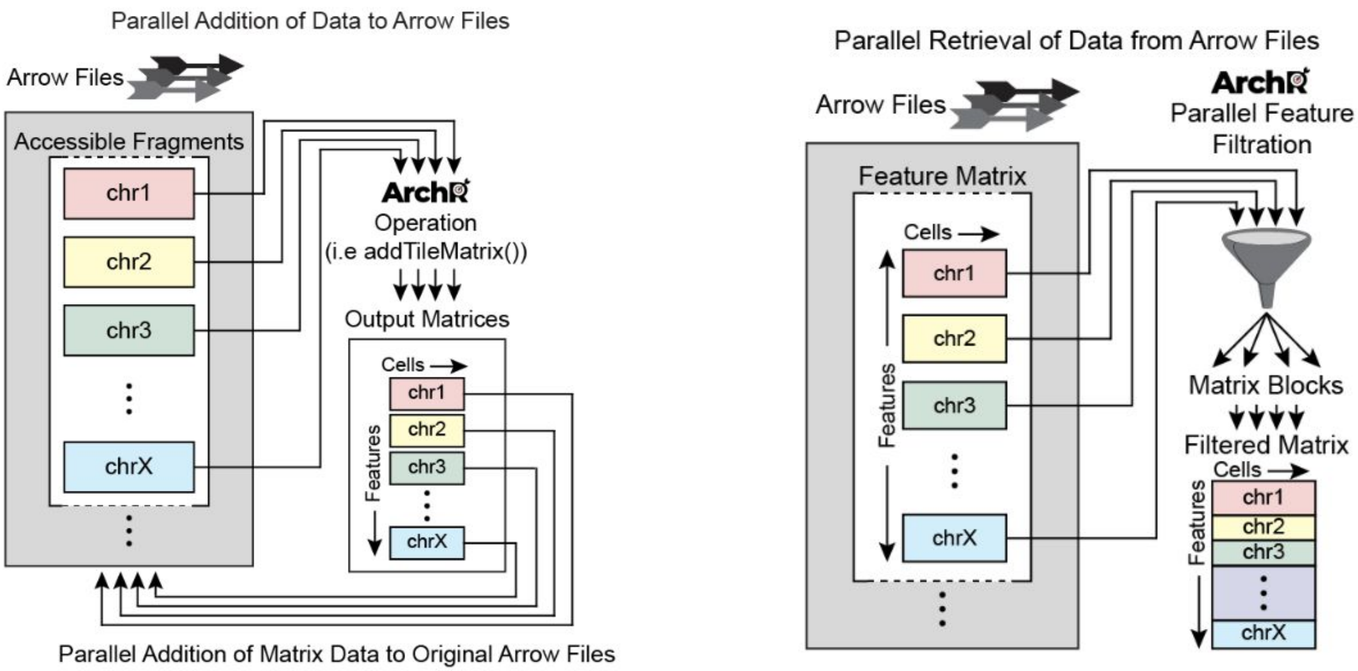
chr	start	stop

Peak Annotation

Motifs	SNPs	Features

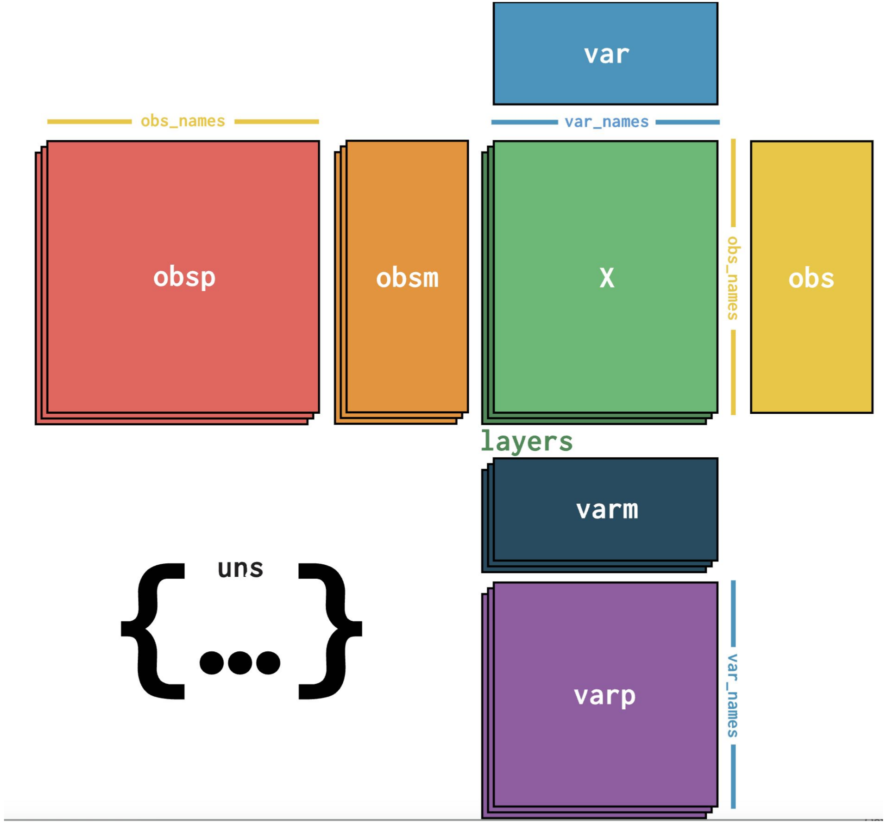
. . .

Information access

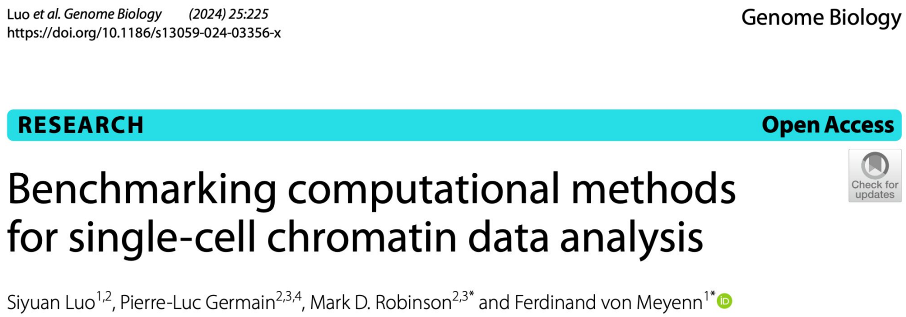


Reading and writing into arrows is done in a parallel fashion

AnnData (used in SnapATAC2) is an alternative to ArrowFiles



Which tool is the best?



Luo *et al. Genome Biology* (2024) 25:225
https://doi.org/10.1186/s13059-024-03356-x

Genome Biology

RESEARCH **Open Access**

Check for updates

Benchmarking computational methods for single-cell chromatin data analysis

Siyuan Luo[1,2], Pierre-Luc Germain[2,3,4], Mark D. Robinson[2,3*] and Ferdinand von Meyenn[1*]

- **SnapATAC** - https://github.com/r3fang/SnapATAC
- **Signac** - https://stuartlab.org/signac/index.html
- **SnapATAC2** - https://github.com/kaizhang/SnapATAC2
- **SnapATAC2** - Fastest, high accuracy; ideal if speed/accuracy are top priorities.
- **ArchR -** Most memory-efficient; suitable for large datasets but has library size bias and struggles with identifying rare cell types and distinguishing similar subtypes.
- **ArchR** requires a large amount of time to create the files needed to run an analysis
- **Signac** requires substantially less time for the object creation step.

- If speed and accuracy are paramount, SnapATAC2 is recommended. If memory efficiency is critical, ArchR might be a better choice, but consider its limitations in cell type identification and potential lower performance in gene activity prediction.

Demo

Input data

- Cryopreserved human peripheral blood mononuclear cells (PBMCs) from a healthy female donor aged 25 were obtained by 10x Genomics.
- Nuclei Isolation for Single Cell Multiome ATAC + Gene Expression Sequencing
- Pbmc_unsorted_3k.fragments.tsv.gz
  - all PBMCs
  - 3,009 cells identified by cellranger
- pbmc_sorted_3k.fragments.tsv.gz
  - Granulocytes were removed by cell sorting
  - 2,711 cells identified by cellranger

Helpful resources

- Wynton Slack channel
  - ucsf-wynton.slack.com

- Gladstone Bioinformatics Core slack channel
  - https://gladstoneinstitutes.slack.com/archives/C0145F1L7QS

- Wynton tutorials
  - https://github.com/ucsf-wynton/tutorials/wiki

For questions:

ayushi.agrawal@gladstone.ucsf.edu
reuben.thomas@gladstone.ucsf.edu
michela.traglia@gladstone.ucsf.edu

Please take our survey to improve our workshops

- https://www.surveymonkey.com/r/F75J6VZ
- ~3 min.

DATA SCIENCE TRAINING PROGRAM: upcoming workshops

Nov 18-19 | **Introduction to Linear Mixed Effects Models**

Nov 22 | **scATAC-seq and scRNA-seq Data Integration**

Dec 5-6 | Working on Wynton

Questions from participant

- Are the insertion counts NOT normalized in peaks? If I remember correctly, the LSI is implemented for normalization.

  Yes, insertion counts in peaks in ArchR are NOT normalized. See sample code in next slide.

- When I analyzed the differential peaks for two groups (control vs mutant) using the default function of getMarkerFeatures() and then filtering by log2FC>0, the result was not the same as the pairwise comparison by defining useGroups and bgdGroups under the function. Theoretically, there should be overlapped peaks between these two methods, but I am confused as to why there were no overlaps.

  Need more information. https://www.archrproject.com/bookdown/pairwise-testing-between-groups.html

- After identifying differential motifs, is it possible to trace back the corresponding peak locations and target genes of the motifs in ArchR? I found that the outputs of diff motifs from ArchR only contain the genomic location of the motifs rather than the peaks. I'm asking this as I'm interested in identifying the specific motif in a different peak of the target gene/DEG in the mutant.

  See sample code in next 3 slides.

Get info on peak count matrix

```
##get the peak matrix
peak_data <- getMatrixFromProject(ArchRProj =
demo_proj, useMatrix = "PeakMatrix")

##view the peak count matrix
peak_data@assays@data$PeakMatrix[1:10, 1:3]

##get info on genomic ranges for peaks - this
will also provide info on closest gene
rowRanges(peak_data)
```

Access/View the LSI reduction

```
View((demo_proj@reducedDims@listData[["IterativeLSI"]]$matSVD))
```

Motif presence in peaks

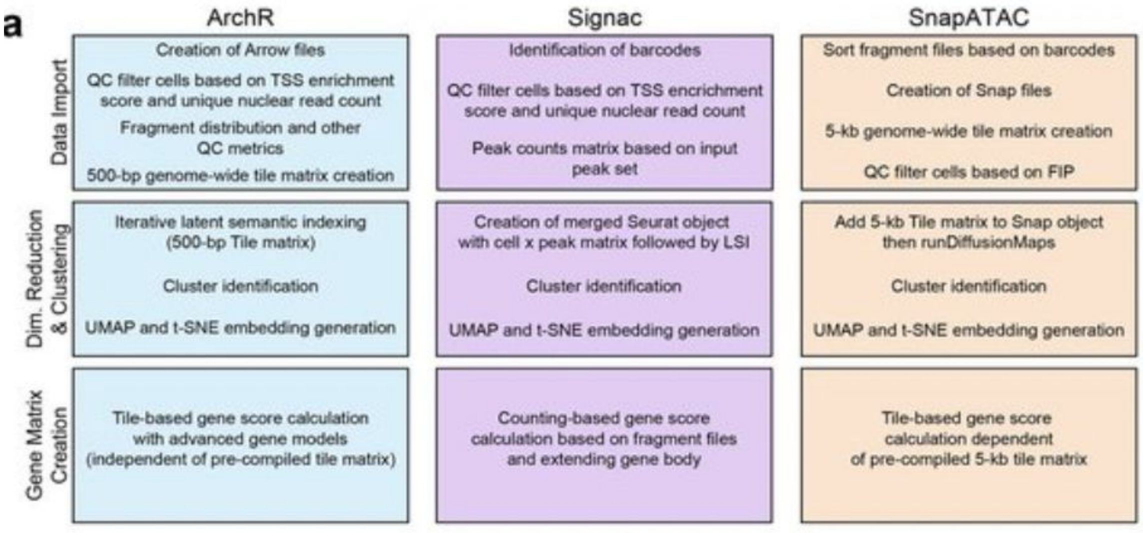
```
motif_presence <- getMatches(ArchRProj =
demo_proj, name = "Motif", annoName = NULL)

motif_presence_matrix <-
motif_presence@assays@data@listData[["matches"]]
%>%
    as.matrix() %>%
     add(0)
```

GLADSTONE INSTITUTES

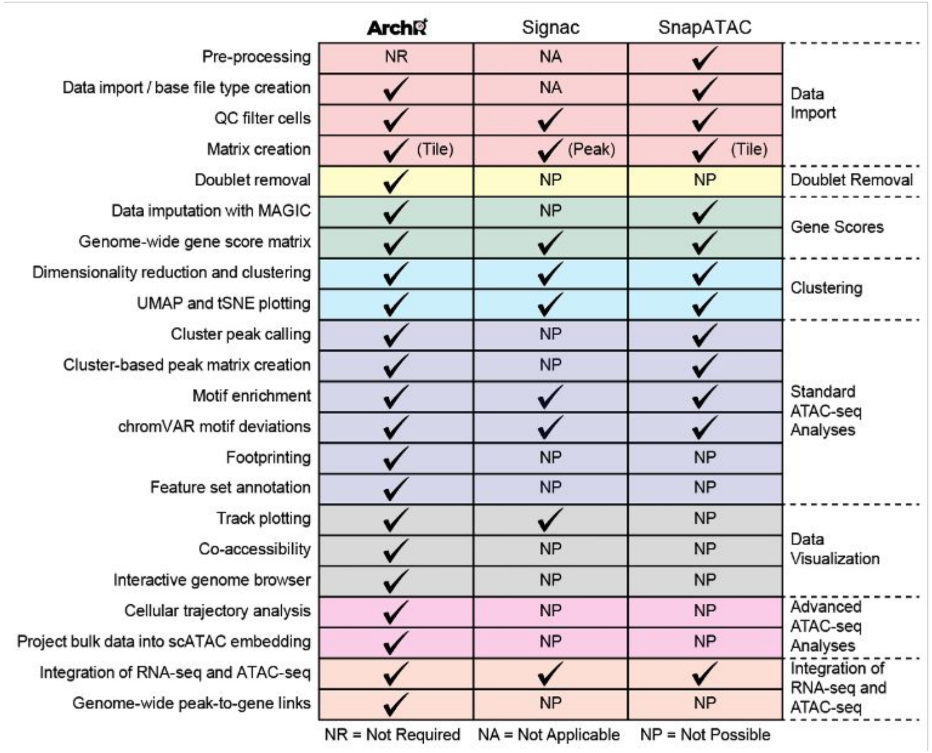
SCIENCE OVERCOMING DISEASE

Which tool is the best?



	ArchR	Signac	SnapATAC	SnapATAC2
Data Import	Creation of Arrow files QC filter cells based on TSS enrichment score and unique nuclear read count Fragment distribution and other QC metrics 500-bp genome-wide tile matrix creation	Identification of barcodes QC filter cells based on TSS encrichment score and unique nuclear read count Peak counts matrix based on input peak set	Sort fragment files based on barcodes Creation of Snap files 5-kb genome-wide tile matrix creation QC filter cells based on FIP	Creation of AnnData files based on 500 bp genome-wide tile matrix QC filtering
Dim. Reduction & Clustering	Iterative latent semantic indexing (500-bp Tile matrix) Cluster identification UMAP and t-SNE embedding generation	Creation of merged Seurat object with cell x peak matrix followed by LSI Cluster identification UMAP and t-SNE embedding generation	Add 5-kb Tile matrix to Snap object then runDiffusionMaps Cluster identification UMAP and t-SNE embedding generation	Add 500bp Tile Matrix to AnnData and then run spectral embedding UMAP and tSNE
Gene Matrix Creation	Tile-based gene score calculation with advanced gene models (independent of pre-compiled tile matrix)	Counting-based gene score calculation based on fragment files and extending gene body	Tile-based gene score calculation dependent of pre-compiled 5-kb tile matrix	Tile-based gene score
	R	R	R	Python

Why use ArchR?



	ArchR	Signac	SnapATAC	
Pre-processing	NR	NA	✓	Data Import
Data import / base file type creation	✓	NA	✓	
QC filter cells	✓	✓	✓	
Matrix creation	✓ (Tile)	✓ (Peak)	✓ (Tile)	
Doublet removal	✓	NP	NP	Doublet Removal
Data imputation with MAGIC	✓	NP	✓	Gene Scores
Genome-wide gene score matrix	✓	✓	✓	
Dimensionality reduction and clustering	✓	✓	✓	Clustering
UMAP and tSNE plotting	✓	✓	✓	
Cluster peak calling	✓	NP	✓	Standard ATAC-seq Analyses
Cluster-based peak matrix creation	✓	NP	✓	
Motif enrichment	✓	✓	✓	
chromVAR motif deviations	✓	✓	✓	
Footprinting	✓	NP	NP	
Feature set annotation	✓	NP	NP	
Track plotting	✓	✓	NP	Data Visualization
Co-accessibility	✓	NP	NP	
Interactive genome browser	✓	NP	NP	
Cellular trajectory analysis	✓	NP	NP	Advanced ATAC-seq Analyses
Project bulk data into scATAC embedding	✓	NP	NP	
Integration of RNA-seq and ATAC-seq	✓	✓	✓	Integration of RNA-seq and ATAC-seq
Genome-wide peak-to-gene links	✓	NP	NP	

NR = Not Required NA = Not Applicable NP = Not Possible

- Comprehensive
- Fast
- Doesn't need to be run on HPC but provides easy export options if desired

Which tool is the best?

- Benchmarking paper: [Genome Biology 2024]
- Evaluated 8 data processing pipelines derived from 5 different scATAC-seq methods.
- SnapATAC2 generally outperforms ArchR in terms of speed, accuracy in cell type identification, and gene activity score prediction.
- **Library Size Bias:** LSI-based methods (Signac, ArchR) are biased by library size; SnapATAC methods mitigate this with linear regression-based normalization.
- **Gene Activity Scores:** SnapATAC and SnapATAC2 outperformed ArchR and Signac for predicting gene activity scores.
- **Strengths & Weaknesses:**
  - **SnapATAC2:** Fastest, high accuracy; ideal if speed/accuracy are top priorities.
  - **ArchR:** Most memory-efficient; suitable for large datasets but has library size bias and struggles with identifying rare cell types and distinguishing similar subtypes.
- **Method performance depends on dataset complexity:**
  - **Simple datasets:** Feature aggregation, SnapATAC and SnapATAC2 performed best.
  - **Complex datasets:** SnapATAC and SnapATAC2 excel.
- **Recommendation:** If speed and accuracy are paramount, SnapATAC2 is recommended. If memory efficiency is critical, ArchR might be a better choice, but consider its limitations in cell type identification and potential lower performance in gene activity prediction.