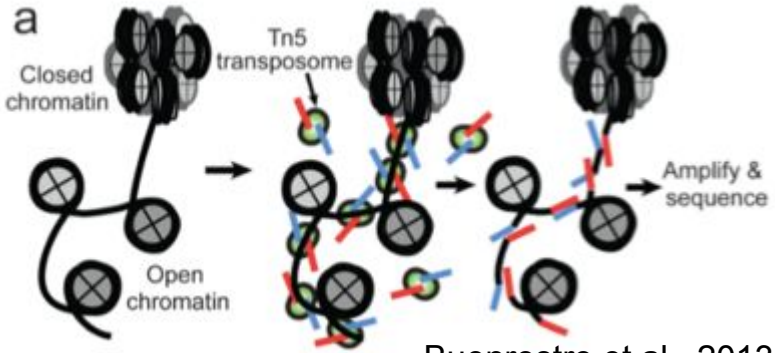
Analysis of scATACseq using the R package ArchR

Wilfrid Richer
Mathias Vandenbogaert
Joshua Waterfall
Eliane Piaggio
Franck Bourdeaut

16/02/2021

Introduction



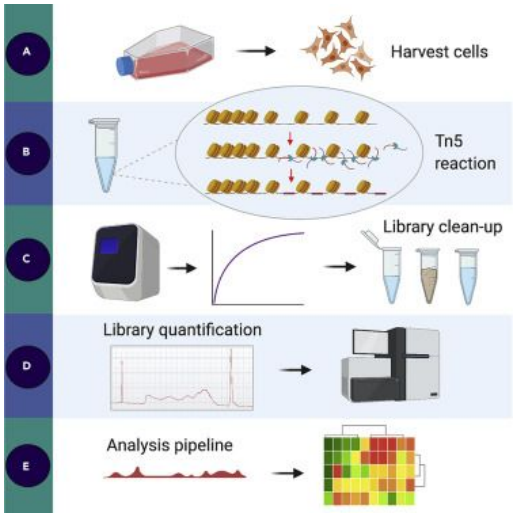
Buenrostro et al., 2013

ATACseq(**A**ssay for **T**ransposase-**A**ccessible **C**hromatin) identifies open chromatin sites where diverse non-histone regulatory factors likely bind.

In 2018 Curie became early test-site partner with 10X Genomics for scATACseq, which launched commercially in 2019.

By providing information about chromatin accessibility, scATAC-seq allows to infer:

-> Gene regulation
-> Epigenetic regulation
-> transcription factor binding site (prediction and footprinting)
-> single-cell clustering
-> mRNA expression level prediction
-> multi-omic integration with scRNA-seq



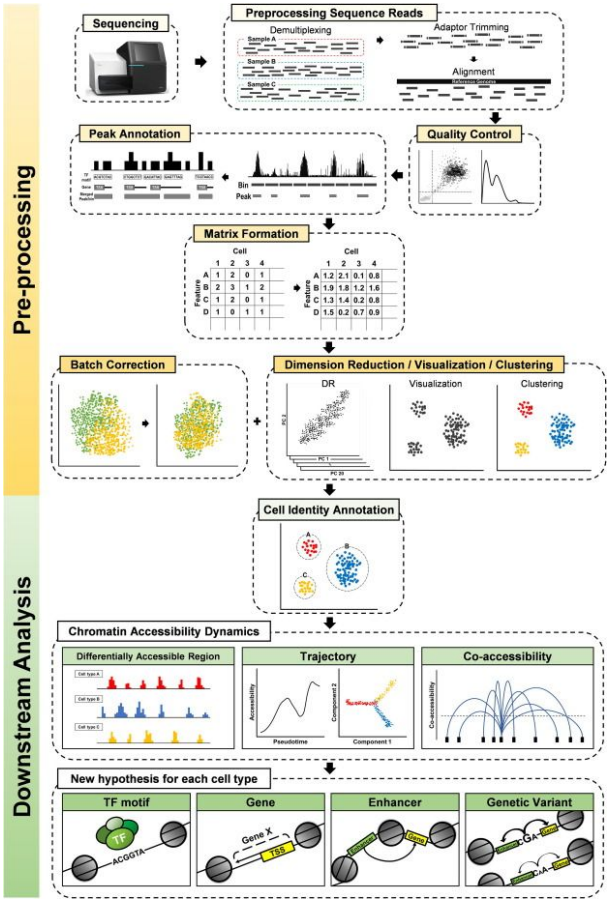
Brunton et al, Cell reports, 2020

Tools to analyse scATACseq

Since the first article on scATACseq in 2015 (Buenrostro et al.), several tools are now available:

- SnapATAC (Fang et al., 2021 Nature comm.)
- Signac (Satija's lab)
- Cicero (Trapnell's and Jay Shendure's labs)
- cisTopic (González-Blas et al., 2019 Nature methods)
- Cusanovich2018 (Cusanovich et al., 2018 Cell)
- **ArchR** (Granja et al., 2021, Nature genetics)

Overview of a typical single-cell ATAC sequencing analysis workflow

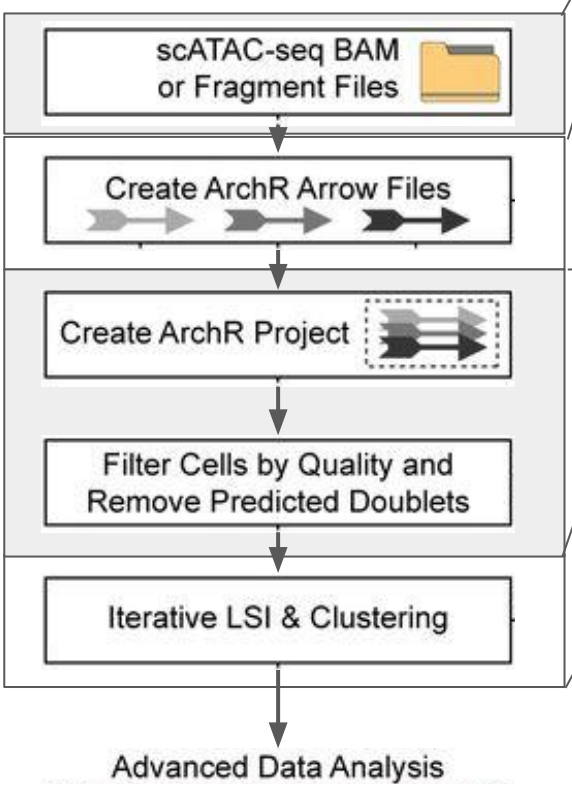


Baek, S. and Lee, I. (2020). Computational and Structural Biotechnology Journal 18, 1429–1439.

Advantages of ArchR:

- Full-featured R package
- Comprehensive suite of scATAC-seq analysis tools
- Extensive software suite from pre-processing data to results
- Exploration of several levels of information
- Fast process and reasonable resource usage (analyze 1 million cells in 8 hours on a MacBook Pro laptop).
- Active development and responsive support (dynamic github support)

ArchR pipeline



From Granja et al. 2020 (bioRxiv)

Import fragments.tsv.gz or BAM files

Create Arrow file

1. Read accessible fragments from the provided input files.
2. Calculate quality control information for each cell (i.e. TSS enrichment scores and nucleosome info).
3. Create a genome-wide TileMatrix using 500-bp bins.
4. Create a GeneScoreMatrix

Inferring doublet

Create ArchRProject and filter the doublets

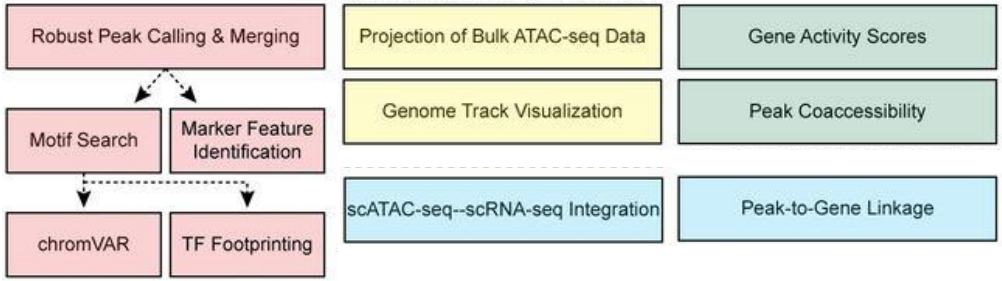
ArchRProject is associated with a set of Arrow files and is the backbone of nearly all ArchR analyses.

Dimensionality reduction and clustering

Iterative Latent Semantic Indexing (iterative LSI)

Batch effect correction

Clustering

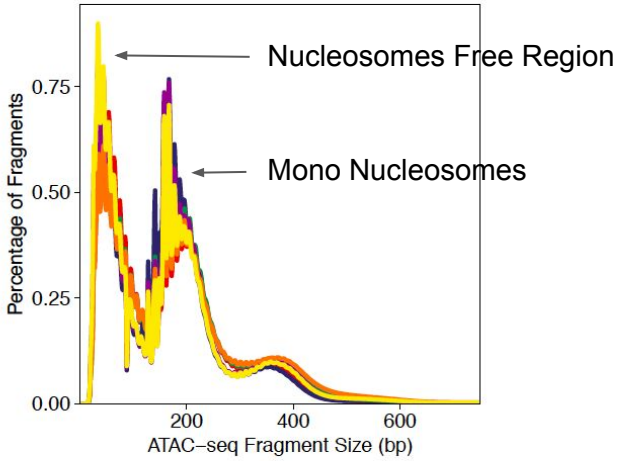


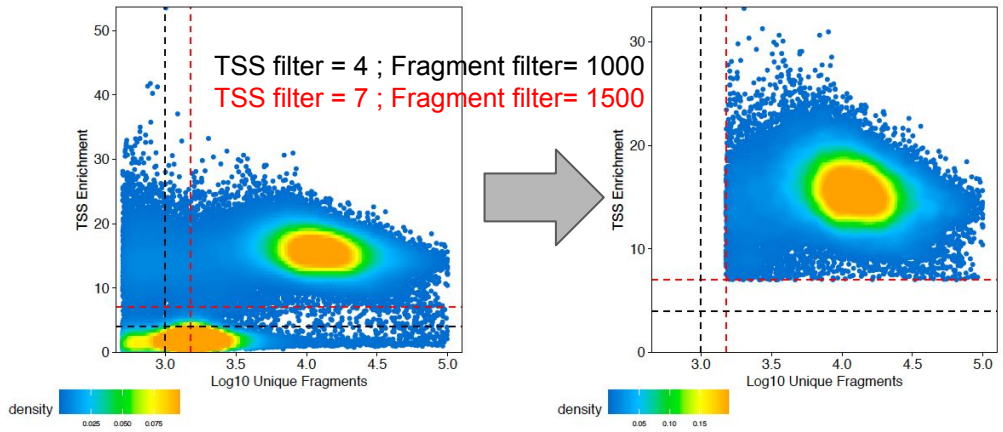
Quality Controls and Filters

ArchR calculates Quality Control information for each cell (i.e. TSS enrichment scores and nucleosome info).

From these QC information, ArchR removes the low-quality cells based on three characteristics:

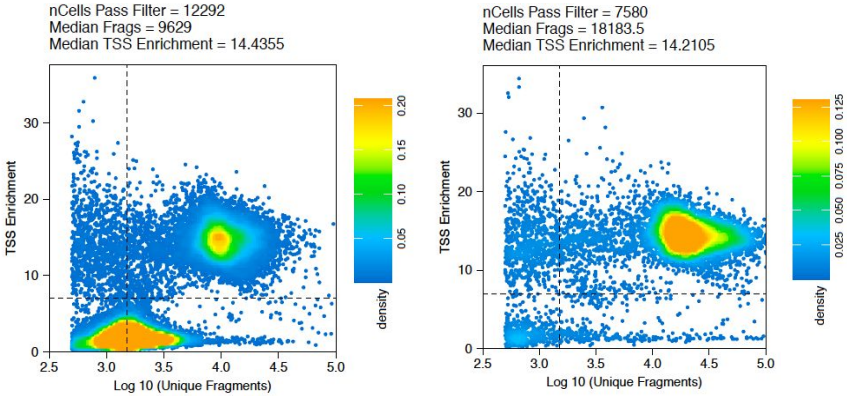
1. T**he fragment size distribution.** Due to nucleosomal periodicity, we expect to see depletion of fragments that are the length of DNA wrapped around a nucleosome (approximately 147 bp).
2. **The TSS enrichment (signal-to-background ratio).** Low signal-to-background ratio is often attributed to dead or dying cells which have de-chromatized DNA which allows for random transposition genome-wide.
3. **The number of unique nuclear fragments** (i.e. not mapping to mitochondrial DNA).

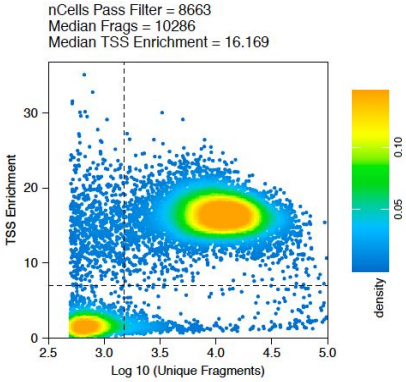


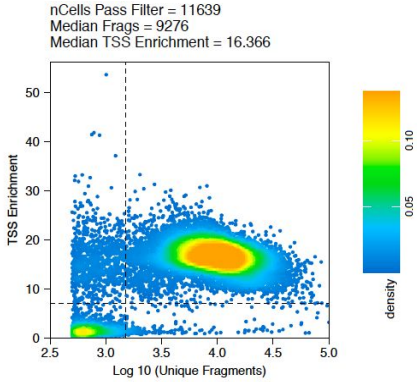


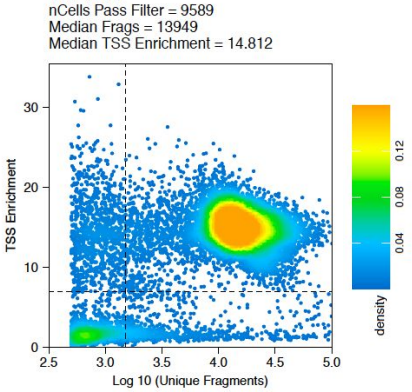
Quality Controls and Filters

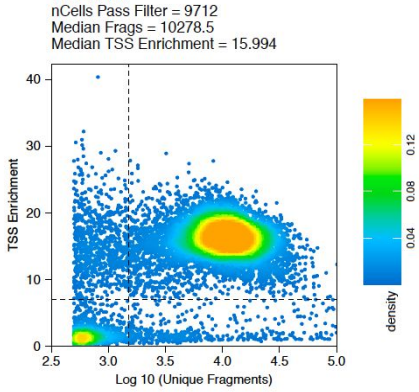
Individual QC samples is available





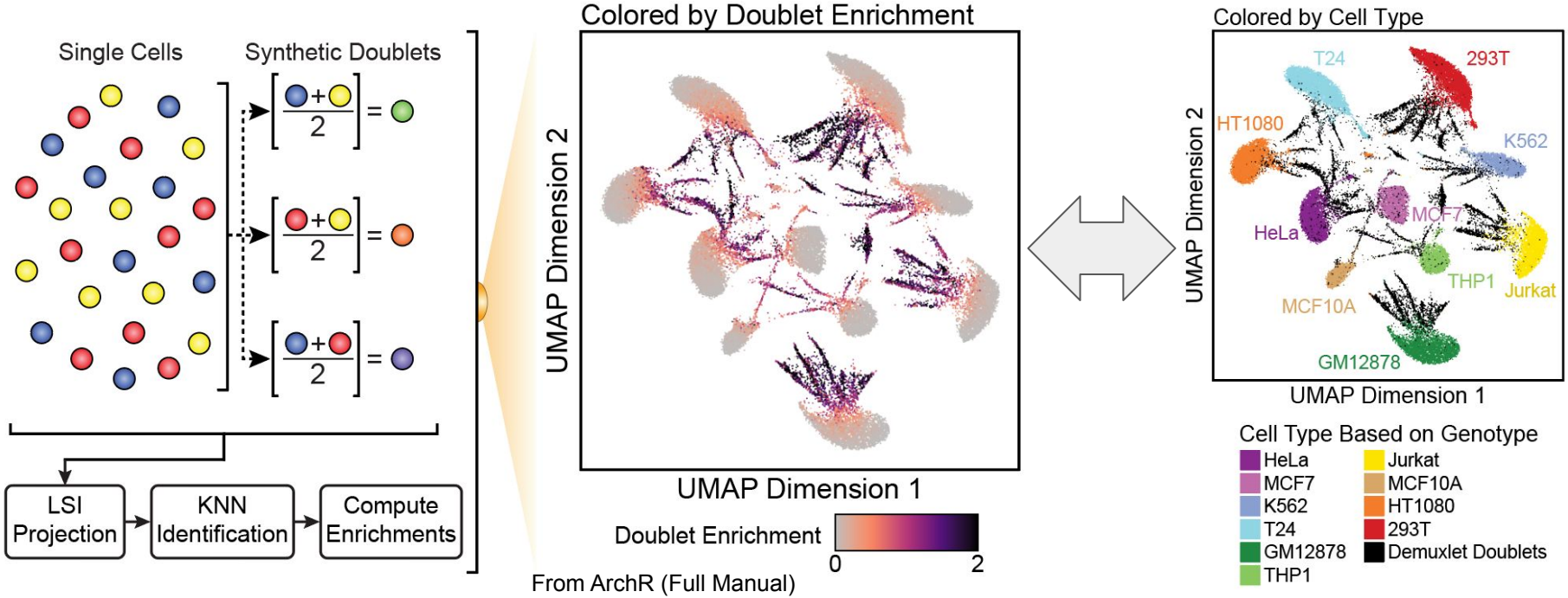






Inferring doublet

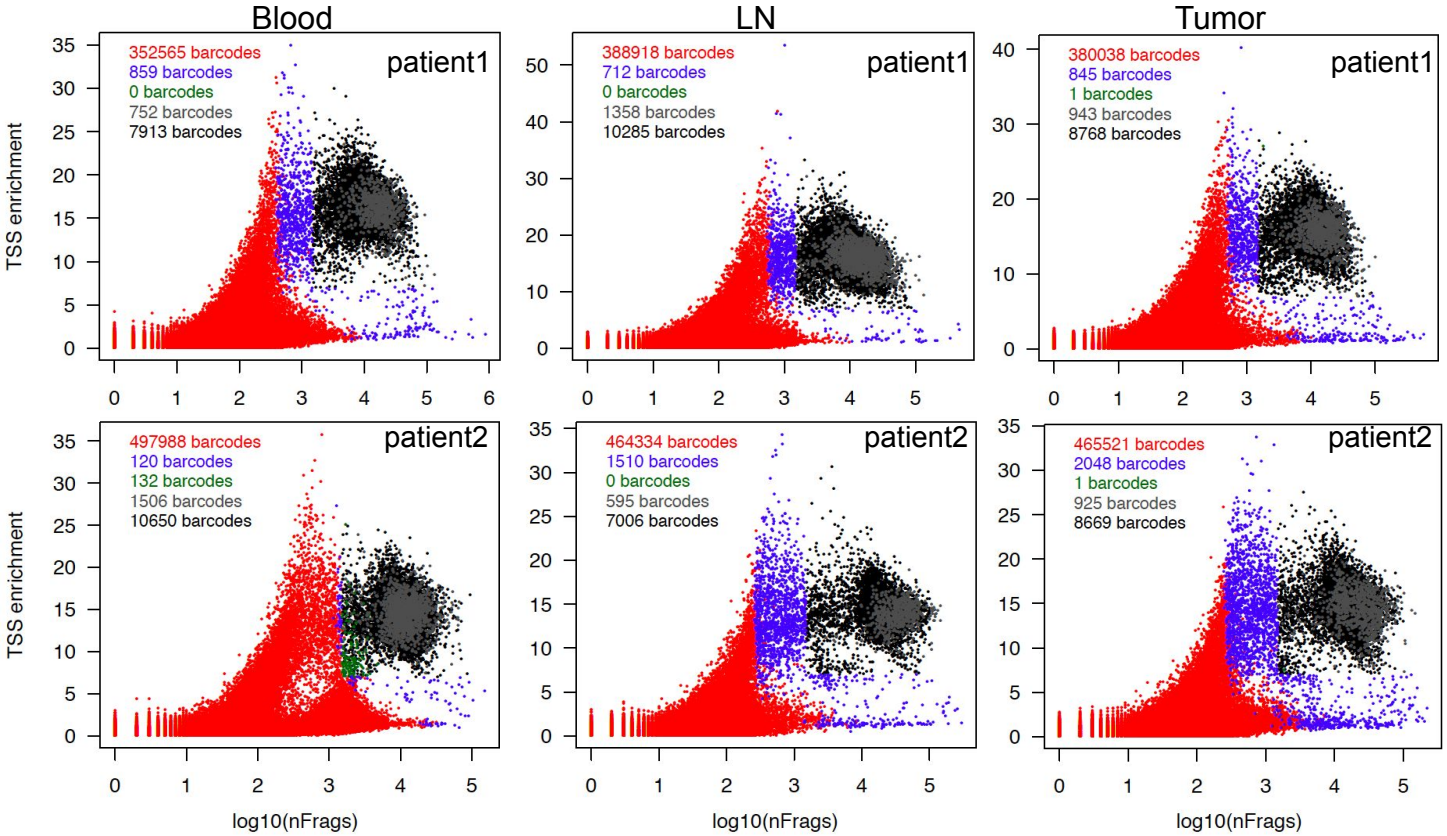
To predict which “cells” are actually doublets, ArchR synthesizes in silico doublets from the data by mixing the reads from thousands of combinations of individual cells.
It projects these synthetic doublets into the UMAP embedding and identify their nearest neighbor. By iterating this procedure thousands of times, it can identify “cells” in the data whose signal looks very similar to synthetic doublets.



In fact, this approach is similar to previous approaches, but differs in that LSI is used for dimensionality reduction and UMAP projection is used for identification.

Comparison between filters applied by Cell ranger and ArchR

Using default filters for Cell ranger
Using suggested filters for ArchR (TSS filter = 7 ; Fragment filter= 1500)



Eliminated by cellranger and ArchR

Eliminated by ArchR / Validated as informative nucleus by Cellranger

Eliminated by Cellranger / Validated as informative nucleus by ArchR

Validated as informative nucleus by Cellranger and ArchR / Detected as doublets by ArchR

Validated as informative nucleus by Cellranger and ArchR

Dimensionality reduction and clustering

scATAC-seq generates a sparse insertion counts matrix (500-bp tiles; binary data of ~6 million of features) making it impossible to identify variable peaks for standard dimensionality reduction. To get around this issue, ArchR use LSI (Latent Semantic Indexing), a layered dimensionality reduction approach for sparse and noisy data.

Rather than identifying the most variable peaks, ArchR tries using the most accessible features as input to LSI.

However, when running **multiple samples the results could shown high degrees of noise and low reproducibility**.

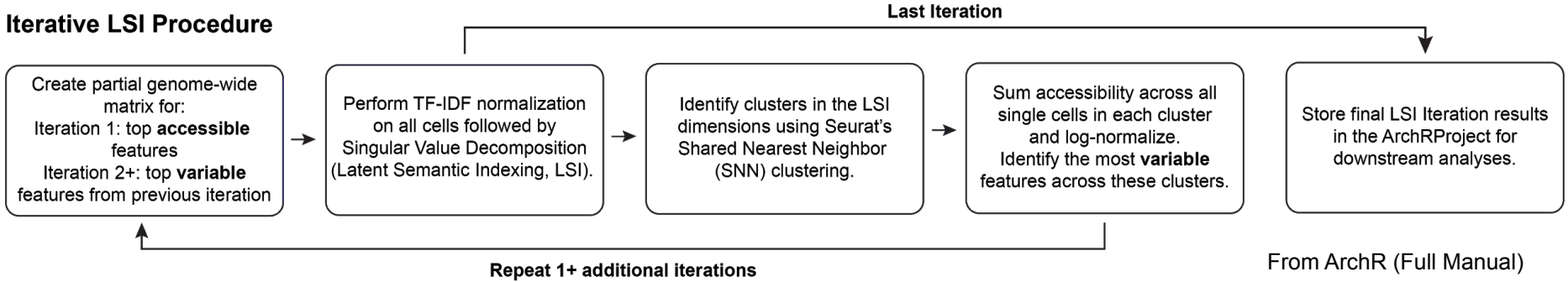
To remedy this, ArchR introduced the “iterative LSI” approach (Satpathy, Granja et al., 2019), which computes an initial LSI transformation on the most accessible tiles and identifies lower resolution clusters that are not batch confounded.

Iterative Latent Semantic Indexing (LSI)

1- This approach computes an initial LSI transformation on the most accessible tiles and identifies lower resolution clusters that are not batch confounded.
2- ArchR computes the average accessibility for each of these clusters across all features. ArchR then identifies the most variable peaks across these clusters and uses these features for LSI again.
3- In this second iteration, the most variable peaks are more similar to the variable genes used in scRNA-seq LSI implementations.

This approach minimizes observed batch effects and allow dimensionality reduction operations on a more reasonably sized feature matrix.

default values: iterations=2 ; varFeatures=25.000 ; resolution= 0.2 ; totalFeatures = 5e+05



From ArchR (Full Manual)

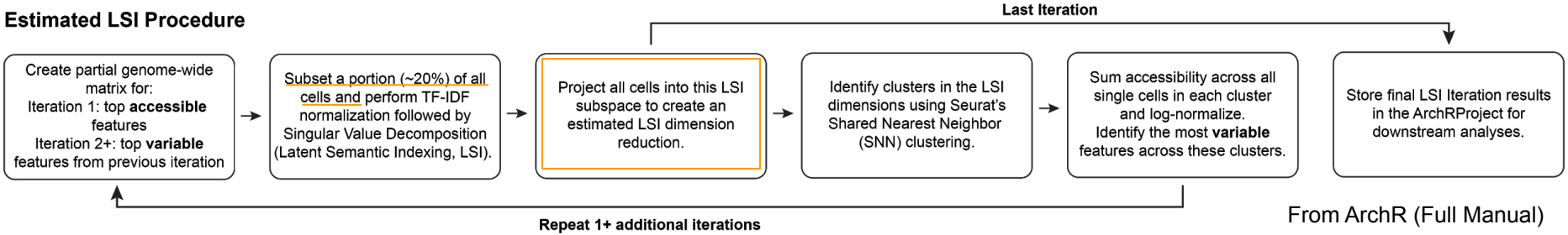
Iterative Latent Semantic Indexing (LSI)

Estimated LSI

For extremely large scATAC-seq datasets, ArchR can estimate the LSI dimensionality reduction with LSI projection.
The LSI procedure differs:
- First, a subset of randomly selected “landmark” cells is used for LSI dimensionality reduction.
- Second, the remaining cells are TF-IDF normalized using the inverse document frequency determined from the landmark cells.
- Third, these normalized cells are projected into the SVD subspace defined by the landmark cells.

This leads to an LSI transformation based on a small set of cells used as landmarks for the projection of the remaining cells.

default values: projectCellsPre=10.000

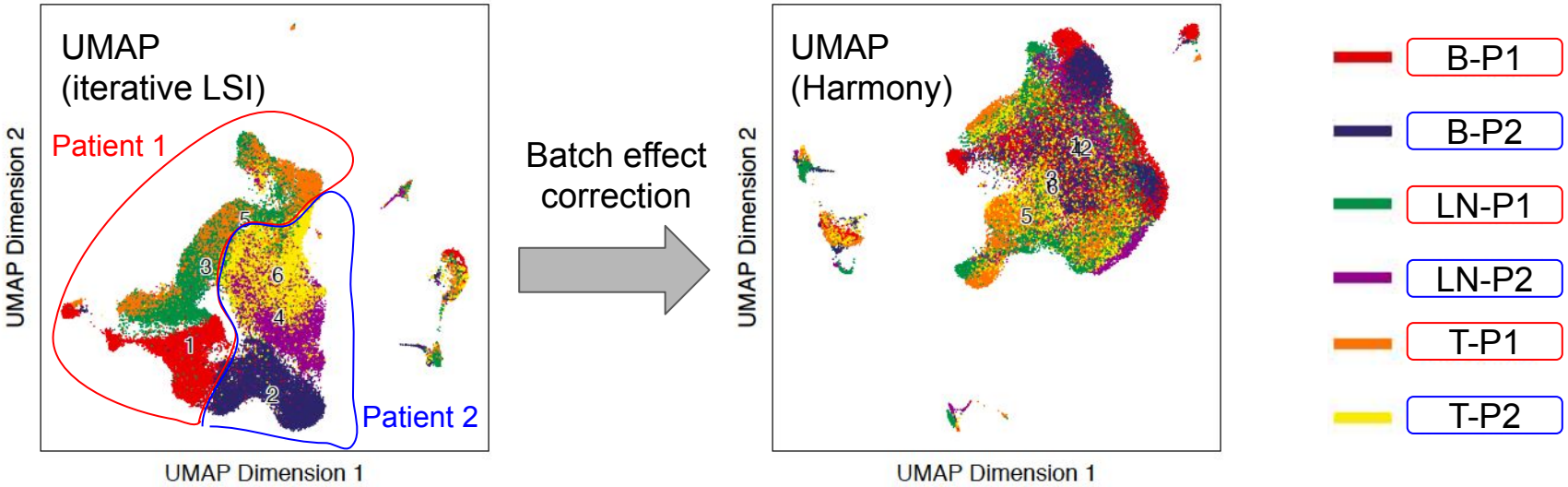


From ArchR (Full Manual)

Batch effect correction

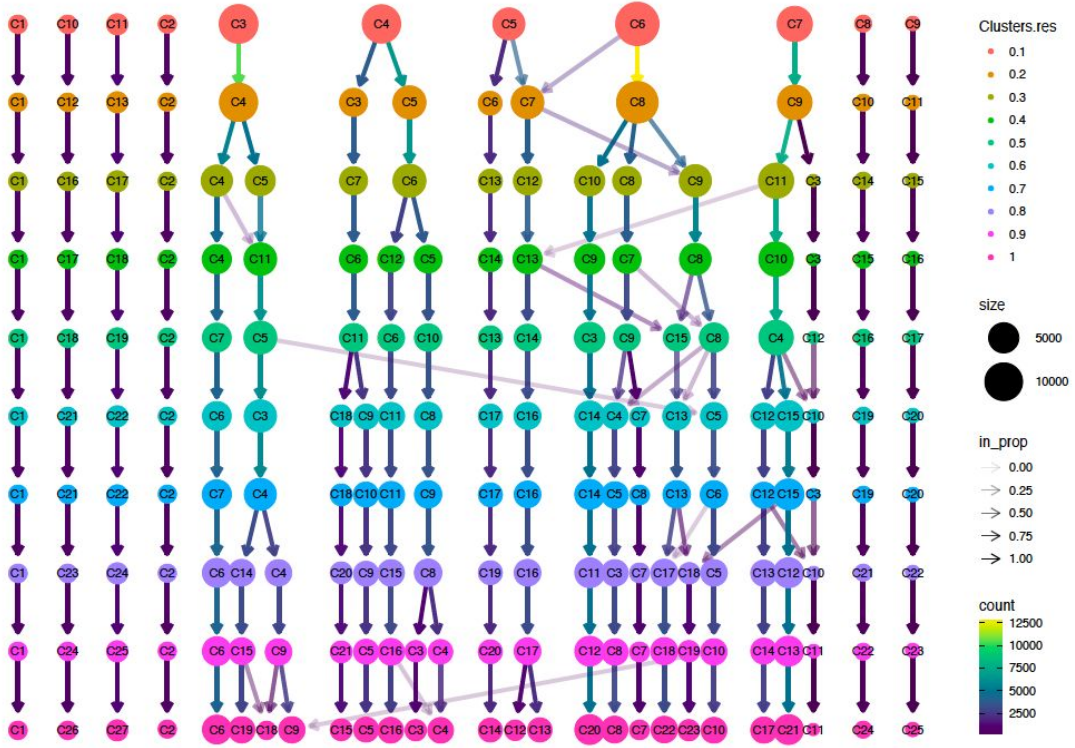
Sometimes the iterative LSI approach isn't enough to correct strong batch effect differences.

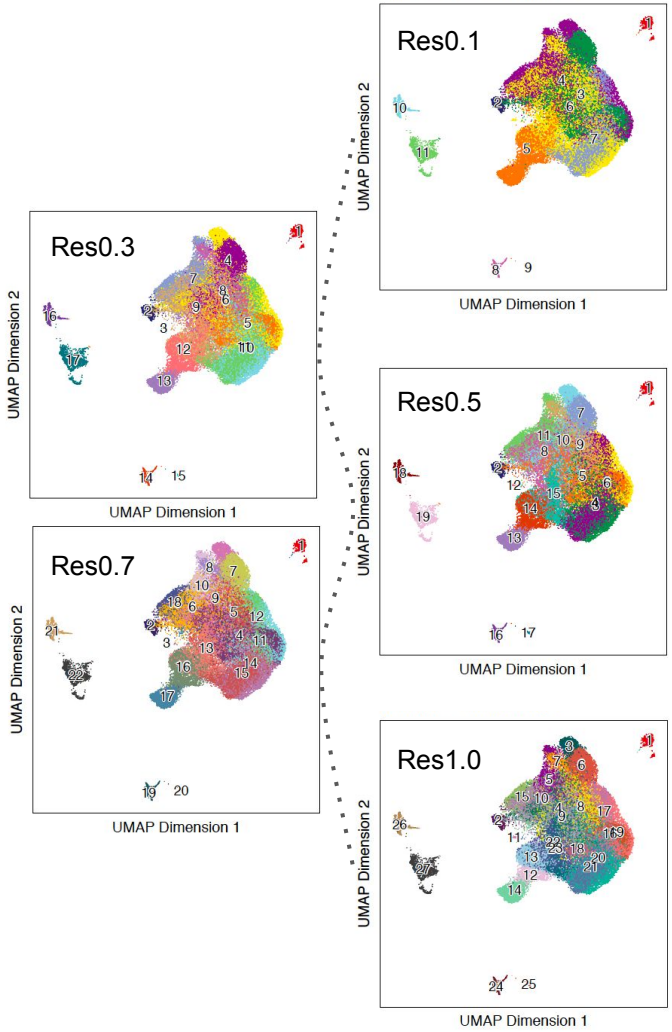
For this reason, ArchR implements a commonly used batch effect correction tool called **Harmony** which was originally designed for scRNA-seq.



Clustering

To identify clusters, ArchR allows to use same method as Seurat or Scran.





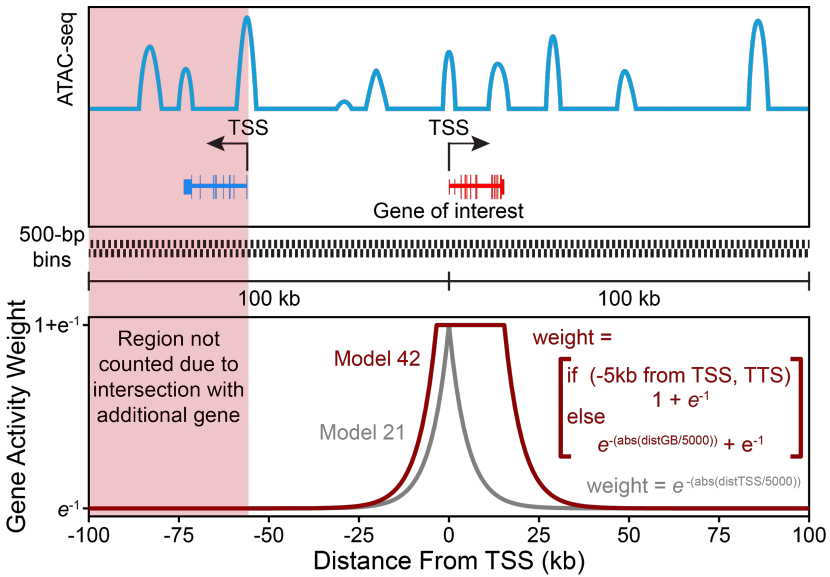
What is Gene Score?

GeneScore is a score that approximate gene expression based on the promoter and gene chromatin accessibility (modeled by the accessibility of bp fragments that correspond to the gene and promoter in the 100Kbp proximity -or lower if other gene is comprised- and weighted by the distance to the gene: closer bp fragments to the gene acquire higher weight in the gene score calculation)

⚠It is important to note that not all genes behave well with gene scores. In particular, genes that reside in very gene-dense areas can be problematic.

ArchR team tested over 50 different gene score models and identified a model that consistently outperformed the rest. It is implemented as the default in ArchR. It has three major components:

1. Accessibility within the entire gene body contributes to the gene score.
2. An exponential weighting function that accounts for the activity of putative distal regulatory elements in a distance-dependent fashion.
3. Imposed gene boundaries that minimizes the contribution of unrelated regulatory elements to the gene score.



distGB = distance to gene body distTSS = distance to TSS

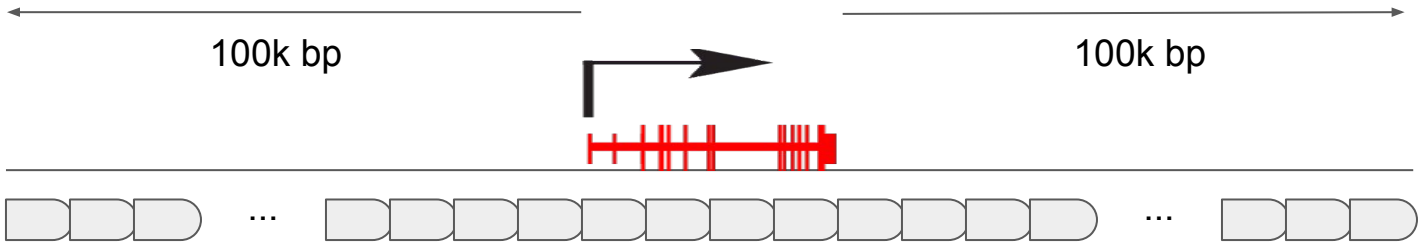
Illustration of the gene score Model 42, which uses bi-directional exponential decays from the gene TSS and the gene TTS while accounting for neighboring gene boundaries.

From Granja et al. 2020 (bioRxiv)

Calculation of Gene Score

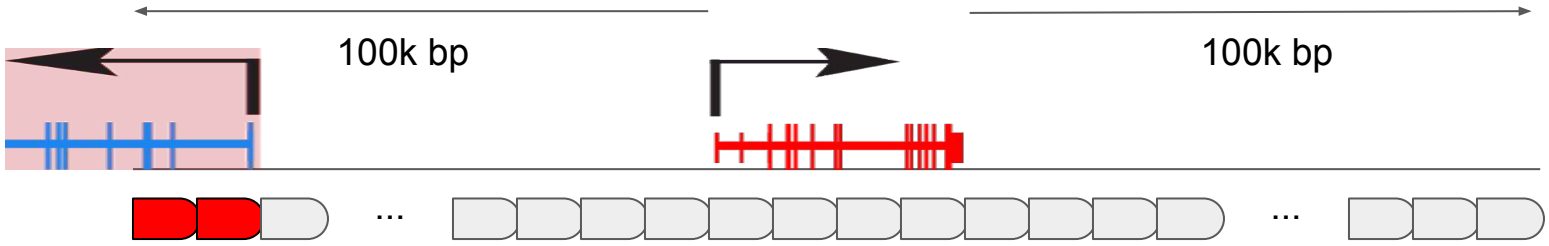
1) For each chromosome, ArchR creates a **tile** matrix using a user-defined tile size that is not pre-computed (default is 500 bp), overlaps these tiles with the user-defined gene window (default is 100 kb on either side of the gene), and then computes the distance from each tile (start or end) to the **gene body/gene start (default is gene body /useTSS=FALSE).**

They have found that the best predictor of gene expression is the local accessibility of the gene region which includes the promoter and gene body.



Calculation of Gene Score

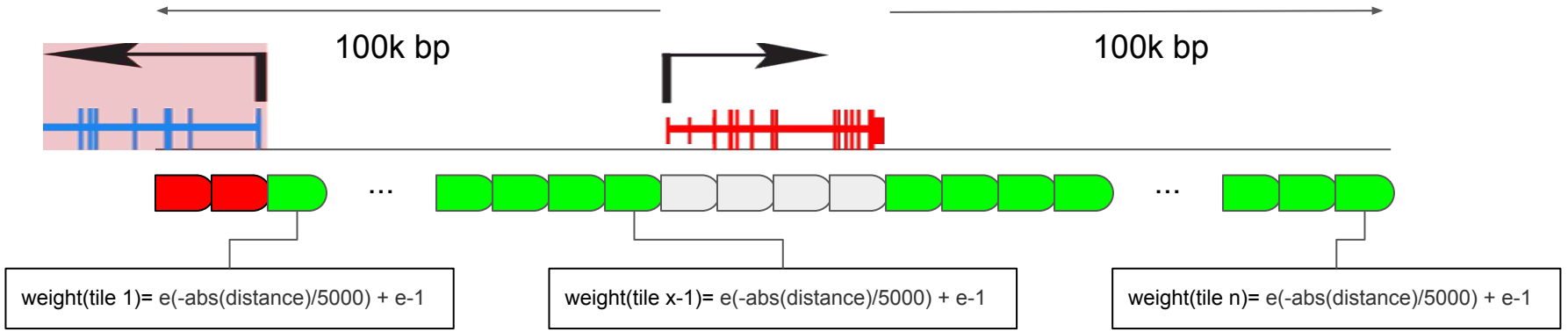
2) To properly account for distal accessibility for a given gene, ArchR identifies the subset of tiles that are within the gene window and do not cross another gene region. This filtering allows for inclusion of distal regulatory elements that could improve the accuracy of predicting gene expression values but excludes regulatory elements more likely to be associated with another gene.



Exclusion

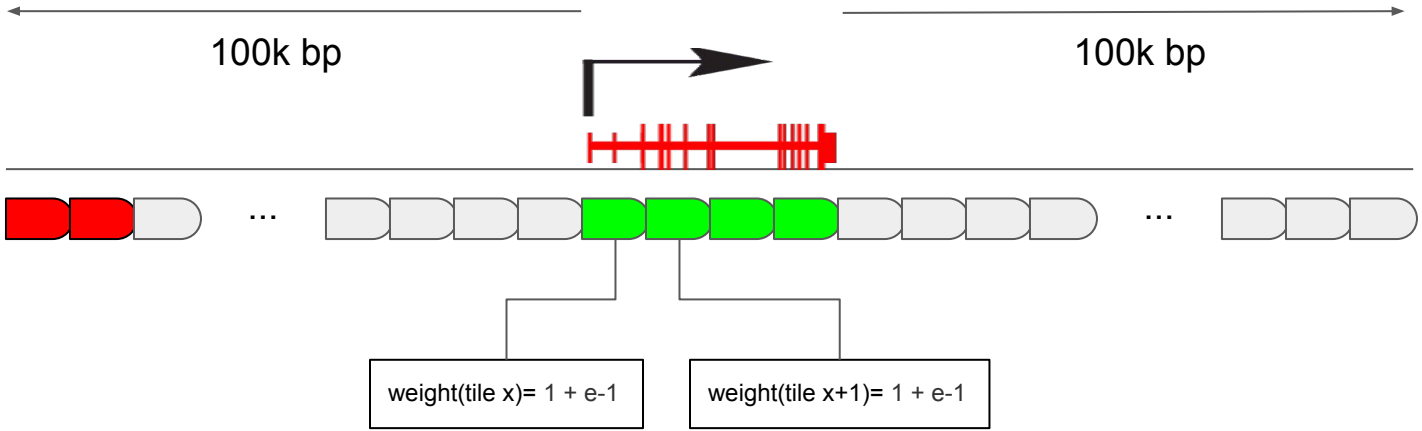
Calculation of Gene Score

3) The distance from each tile to the gene is then converted to a distance weight using a user-defined accessibility model (default is e(-abs(distance)/5000) + e-1).



Calculation of Gene Score

4) When the gene body is included in the gene region (where the distance-based weight is the maximum weight possible), extremely large genes can bias the overall gene scores (total gene scores can vary substantially due to the inclusion of insertions in both introns and exons).



To help adjust for these large differences in gene size, ArchR applies a separate weight for the inverse of the gene size (1 / gene size) and scales this inverse weight linearly from 1 to a user-defined hard maximum (default of 5).

Thus, smaller genes receive larger relative weights, partially normalizing this length effect.

Calculation of Gene Score

5) Corresponding distance and gene size weights are then multiplied by the number of Tn5 (enzyme) insertions within each tile and summed across all tiles within the gene window (without ambiguous tiles crossing another gene region).

This summed accessibility is a “gene score” and is depth normalized across all genes to a user-defined constant (default of 10,000).

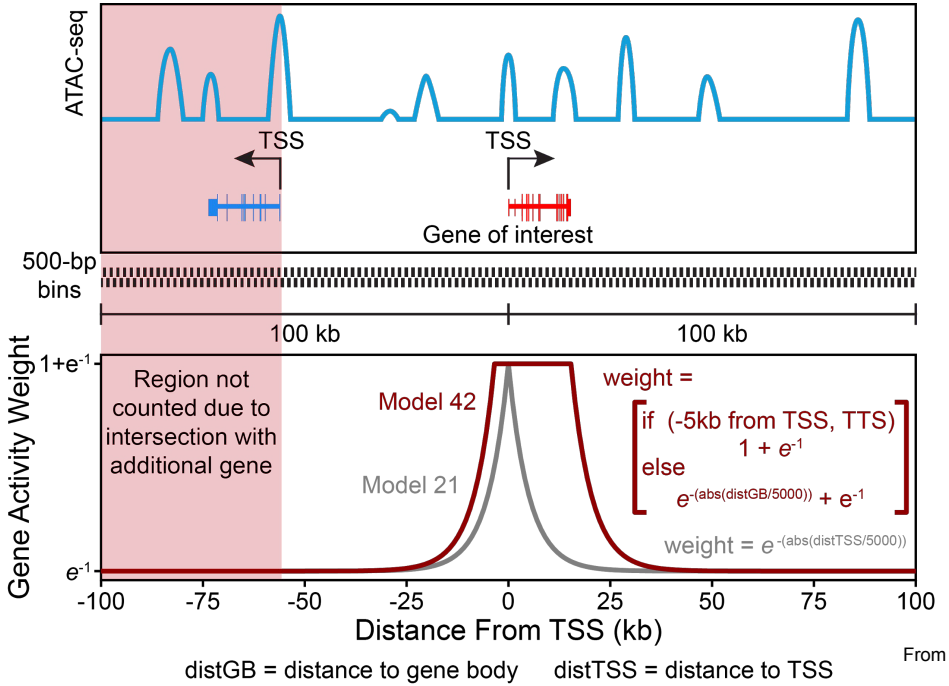


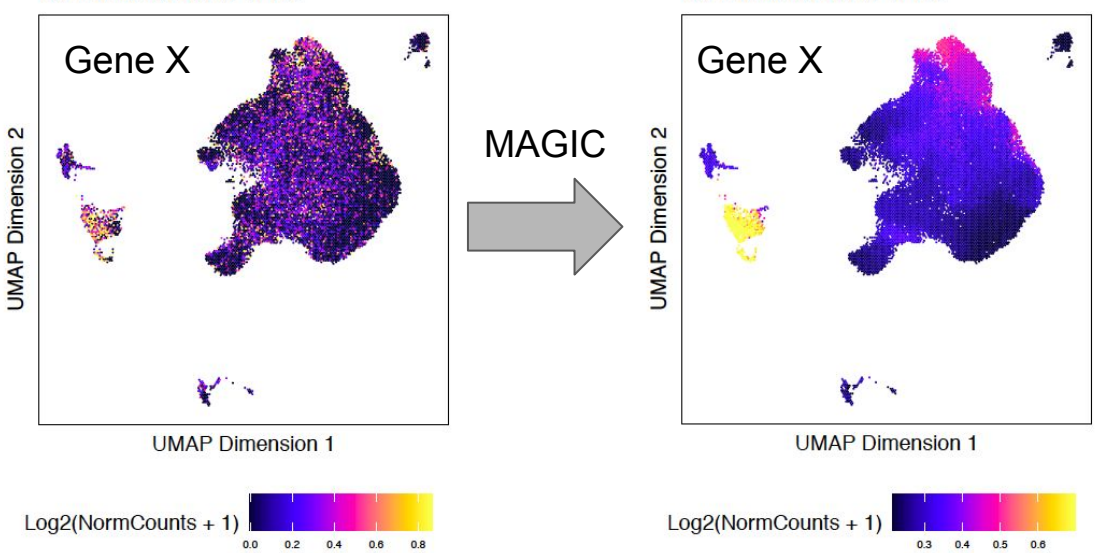
Illustration of the gene score Model 42, which uses bi-directional exponential decays from the gene TSS (extended upstream by 5 kb) and the gene transcription termination site (TTS) while accounting for neighboring gene boundaries. Model 42 is more accurate than other models (such as Model 21 which models an exponential decay from the gene TSS).

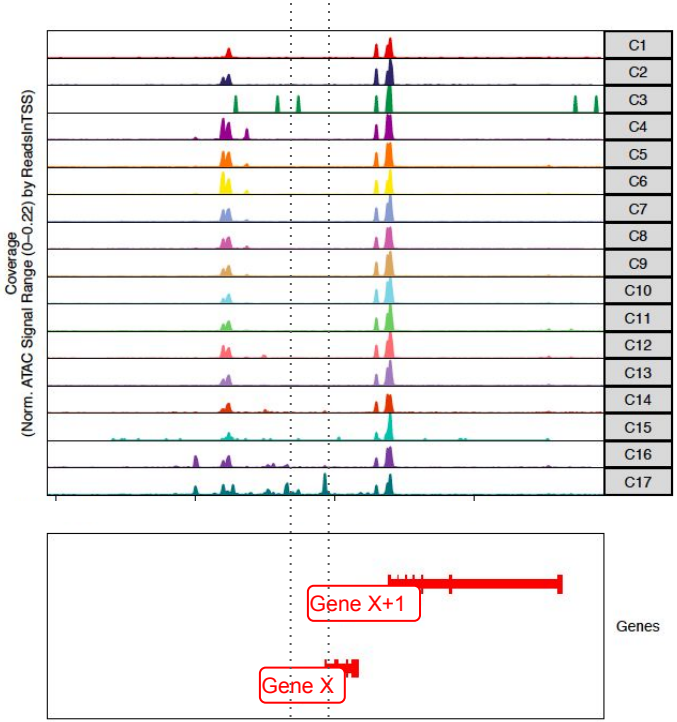
From Granja et al. 2020 (bioRxiv)

Visualization of Gene

Individual gene information can be visualized using two different methods:

- Track plot, details the peaks at proximity to the gene.
- Feature plot, overlays per-cell gene scores on our UMAP embedding.





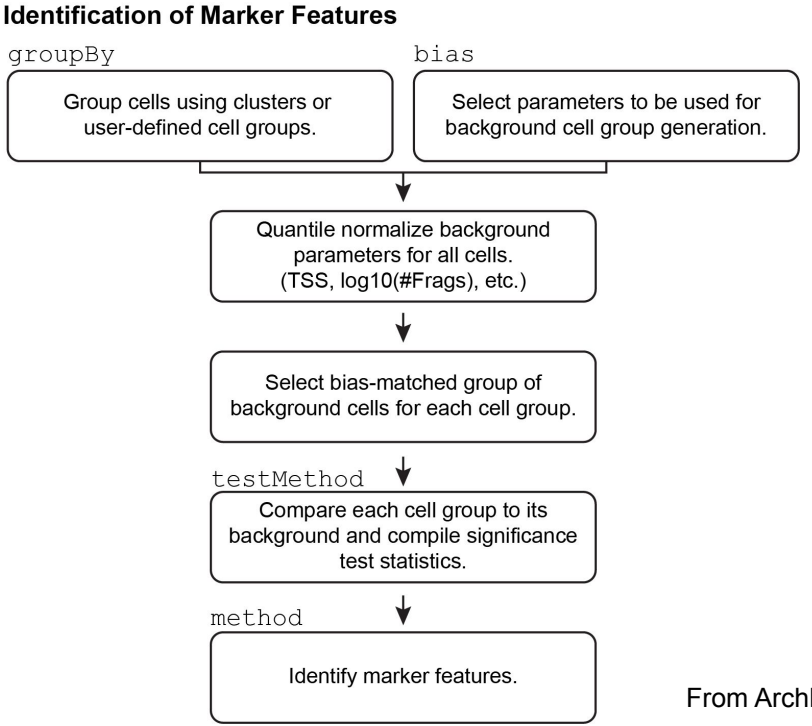
However, due to the sparsity of scATAC-seq data, gene score on UMAP can appear quite variable (and hardly interpretable).

To resolve that, ArchR proposes to use **MAGIC** **to impute gene scores by smoothing signal across nearby cells**.

Identification of marker features

In addition to using prior knowledge of relevant marker genes for annotation of clusters, ArchR enables unbiased identification of marker features for any given cell groupings.

ArchR can used **gene, peak or transcription factor motif** features. For example, ArchR could identify the **genes** that appear to be uniquely active in each cell type.

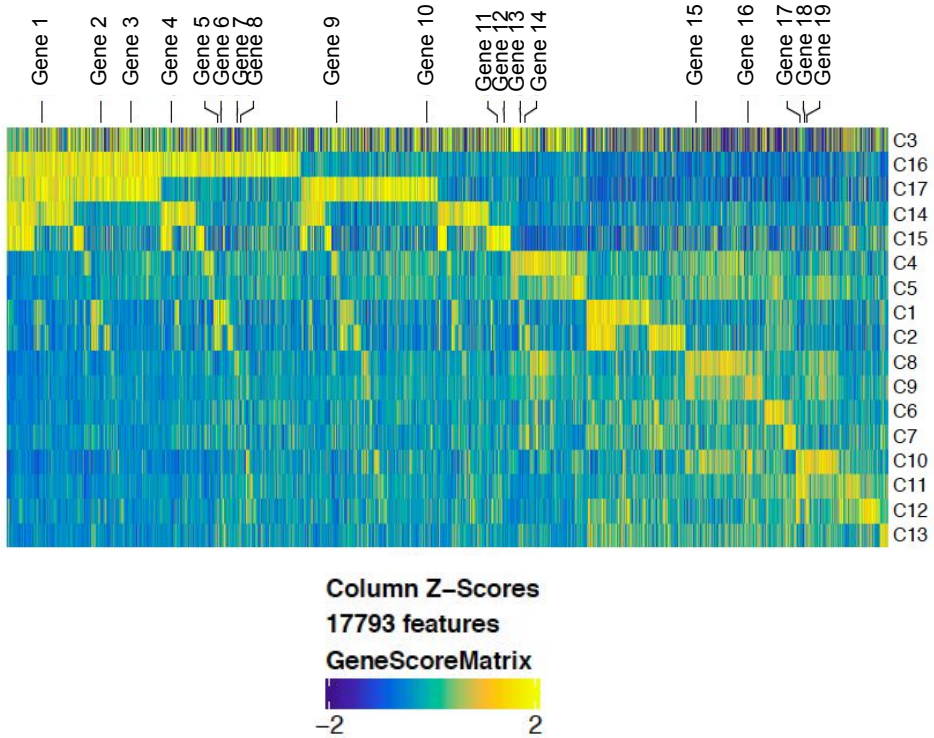


From ArchR (Full Manual)

Visualization of Marker features

Using this specific strategy, ArchR provides an unbiased way of seeing which genes are predicted to be active in each cluster and can aid in cluster annotation.

To visualize all of the marker features simultaneously, ArchR proposes an heatmap which can optionally supplying some marker genes to label.

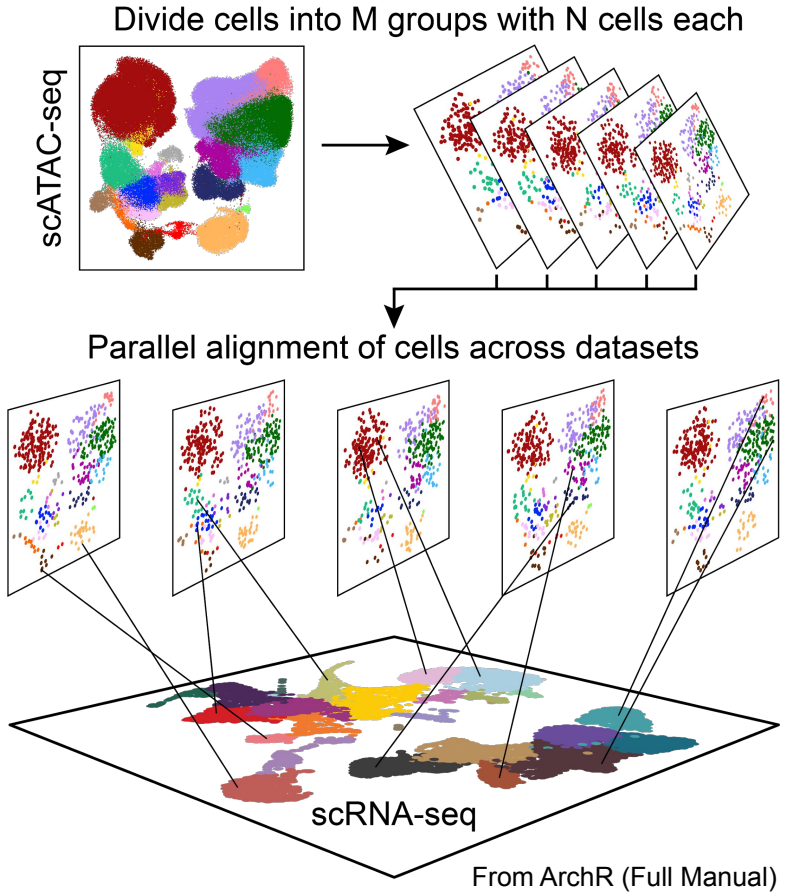


Cross-platform linkage: scATAC-seq with scRNA-seq

ArchR enables integration with scRNA-seq, offers the possibility to use clusters called in scRNA-seq space or use the gene expression measurements after integration.

The way this integration works is by directly **aligning cells from scATAC-seq with cells from scRNA-seq** by comparing the scATAC-seq gene score matrix with the scRNA-seq gene expression matrix. This alignment is performed **using the FindTransferAnchors() function from the Seurat package** which allows you to align data across two datasets.

However, to appropriately scale this procedure for hundreds of thousands of cells ArchR provides, a **parallelization of this procedure by dividing the total cells into smaller groups** of cells and performing separate alignments.



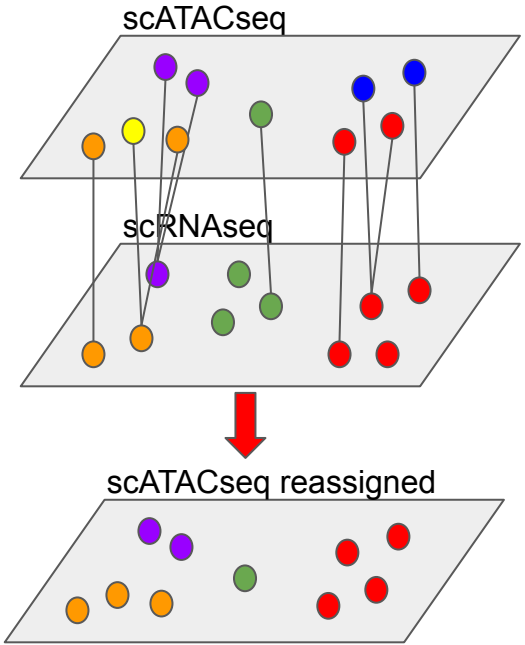
From ArchR (Full Manual)

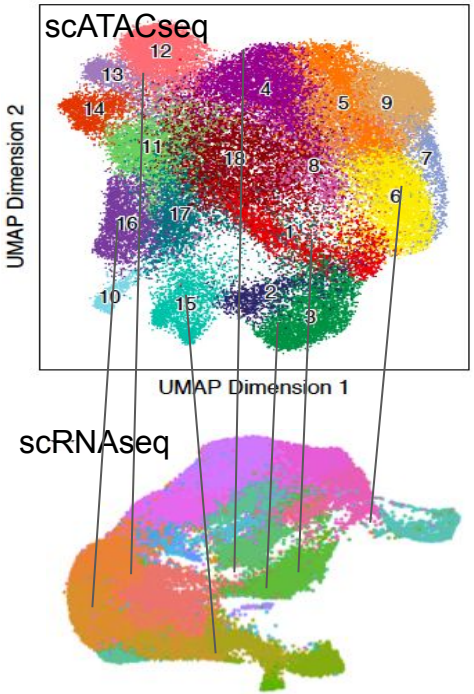
Cross-platform linkage: scATAC-seq with scRNA-seq

For each cell in the scATAC-seq data, this integration process finds the cell in the scRNA-seq data that looks most similar and assigns the gene expression data from that scRNA-seq cell to the scATAC-seq cell.

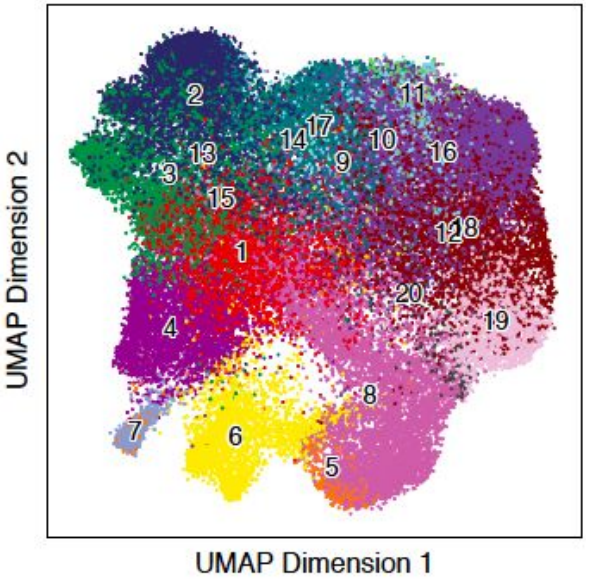
At the end, each cell in scATAC-seq space has been assigned to:

- a scRNAseq cluster ID
- a gene expression signature





scATACseq reassigned with scRNAseq annotation

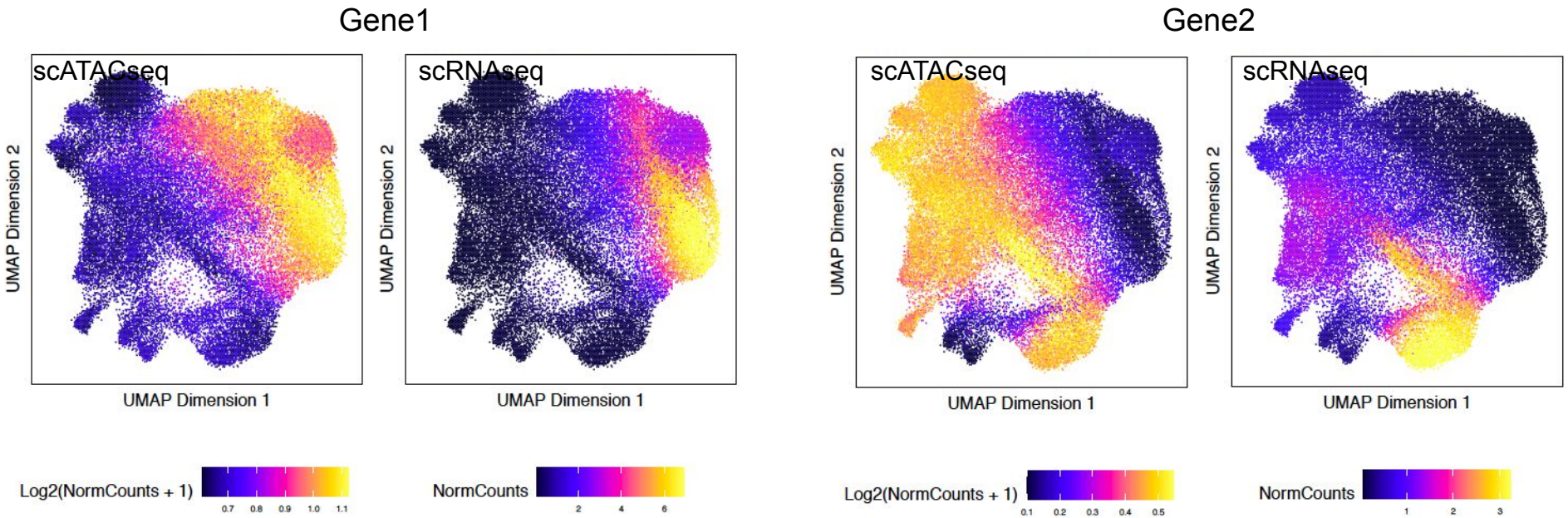


Cross-platform linkage: scATAC-seq with scRNA-seq

For each cell in the scATAC-seq data, this integration process finds the cell in the scRNA-seq data that looks most similar and assigns the gene expression data from that scRNA-seq cell to the scATAC-seq cell.

At the end, each cell in scATAC-seq space has been assigned to:

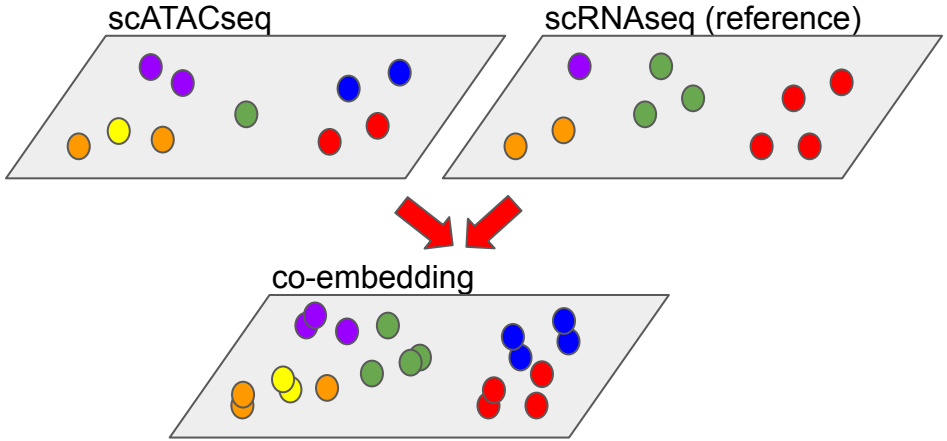
- a scRNAseq cluster ID
- a gene expression signature

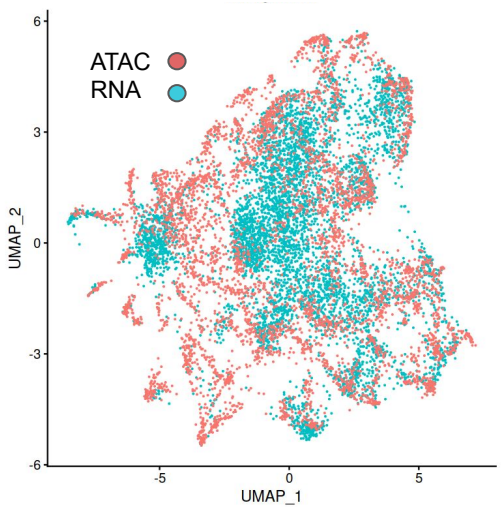


Co-Embedding scATAC-seq and scRNA-seq

To visualize scATAC-seq and scRNA-seq cells together, Seurat proposes to co-embed the scRNA-seq and scATAC-seq cells in the same low dimensional space. They proposes:

1) Identify anchors between the scATAC-seq dataset and the scRNA-seq dataset.
2) Transfer cell type labels to impute RNA-seq values for the scATAC-seq cells.
3) Merge the measured and imputed scRNA-seq data
4) Run a standard UMAP analysis to visualize all the cells together.





Work in progress

Peak Calling

ArchR applies a Iterative Overlap Peak Merging Procedure with the recommanded MACS2 peak caller (but it also possible to use the native peak caller of ArchR).

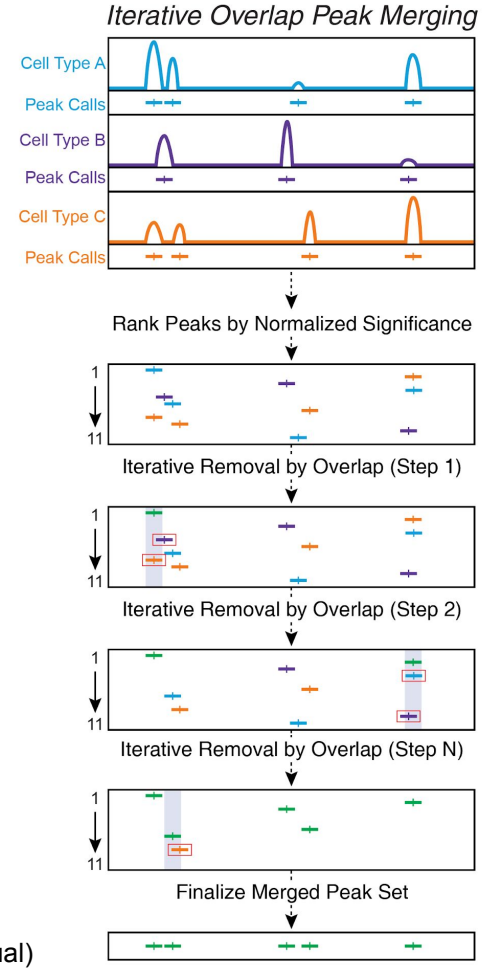
ArchR uses a function to perform this iterative overlap peak merging procedure:

-> First, ArchR would call peaks for each pseudo-bulk replicate individually.

-> ArchR would analyze all of the pseudo-bulk replicates from a single cell type together, performing the first iteration of iterative overlap removal.

-> After the first iteration of iterative overlap removal, ArchR checks to see the reproducibility of each peak across pseudo-bulk replicates and only keeps peaks that pass a threshold indicated by the reproducibility parameter.

-> At the end of this process, we would have a single merged peak set for each cell types.



From ArchR (Full Manual)

Work in progress

Downstream analysis after Peak Calling

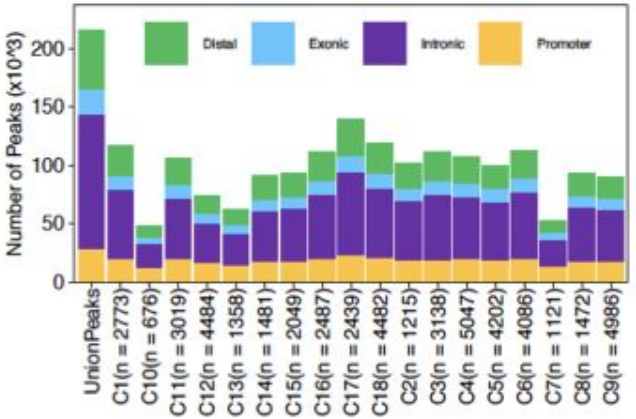
Once peak calling has done, ArchR offers several downstream analysis:

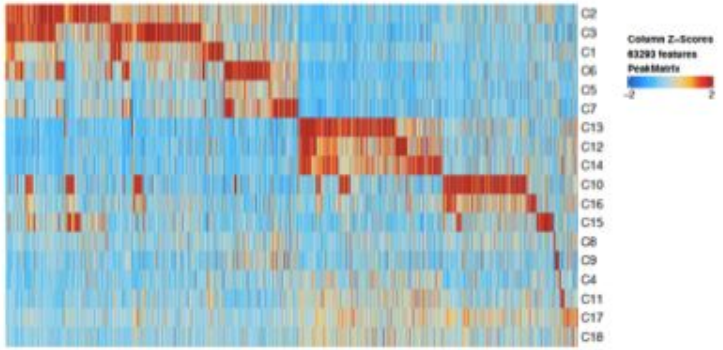
- Identification of Marker Peaks (and Marker Peaks unique to specific cell group)
- Prediction of specific Transcription Factor Binding Sites based on groups of peaks for a cell type (with specific accessible chromatin regions).
- TF motif enrichments (based on ChromVAR) and footprinting

ChromVAR is designed for predicting enrichment of TF activity on a per-cell basis from sparse chromatin accessibility data.

TF footprinting allows for the prediction of the precise binding location of a TF at a particular locus.

- Peak Co-accessibility, is a correlation in accessibility between two peaks across many single cells.
- Peak-to-gene linkage, leverages integrated scRNA-seqdata to look for correlations between peak accessibility and gene expression.





Work in progress

Peak Calling and Downstream analysis

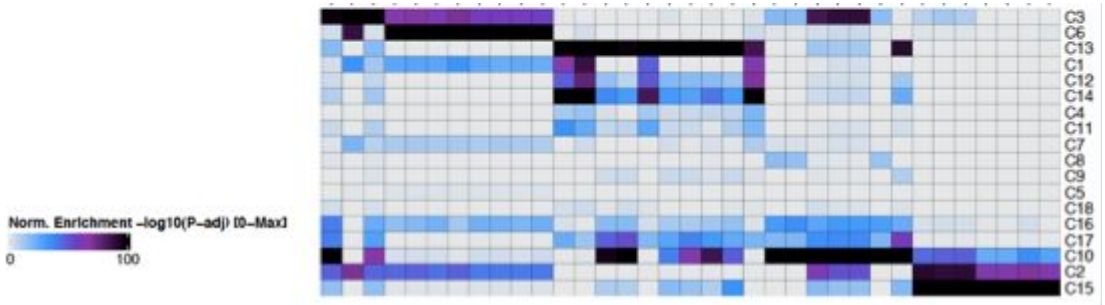
Once peak calling has done, ArchR offers several downstream analysis:

- Identification of Marker Peaks (and Marker Peaks unique to specific cell group)
- Prediction of specific Transcription Factor Binding Sites based on groups of peaks for a cell type (with specific accessible chromatin regions).
- TF motif enrichments (based on ChromVAR) and footprinting

ChromVAR is designed for predicting enrichment of TF activity on a per-cell basis from sparse chromatin accessibility data.

TF footprinting allows for the prediction of the precise binding location of a TF at a particular locus.

- Peak Co-accessibility, is a correlation in accessibility between two peaks across many single cells.
- Peak-to-gene linkage, leverages integrated scRNA-seqdata to look for correlations between peak accessibility and gene expression.



Work in progress

Peak Calling and Downstream analysis

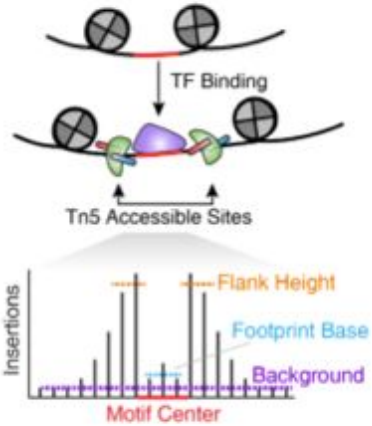
Once peak calling has done, ArchR offers several downstream analysis:

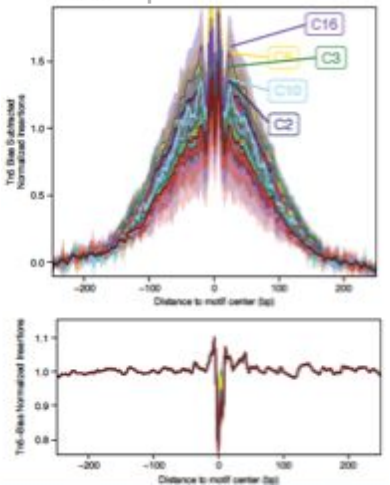
- Identification of Marker Peaks (and Marker Peaks unique to specific cell group)
- Prediction of specific Transcription Factor Binding Sites based on groups of peaks for a cell type (with specific accessible chromatin regions).
- TF motif enrichments (based on ChromVAR) and footprinting

ChromVAR is designed for predicting enrichment of TF activity on a per-cell basis from sparse chromatin accessibility data.

TF footprinting allows for the prediction of the precise binding location of a TF at a particular locus.

- Peak Co-accessibility, is a correlation in accessibility between two peaks across many single cells.
- Peak-to-gene linkage, leverages integrated scRNA-seqdata to look for correlations between peak accessibility and gene expression.





From ArchR (Full Manual)

Work in progress

Peak Calling and Downstream analysis

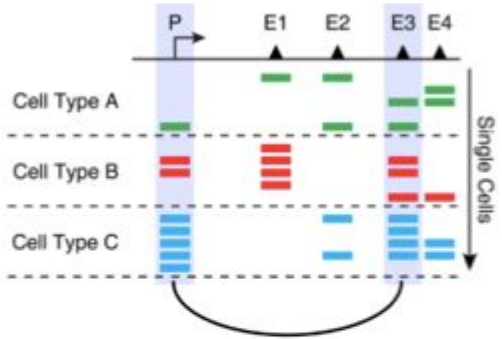
Once peak calling has done, ArchR offers several downstream analysis:

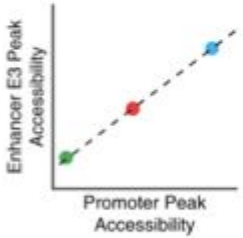
- Identification of Marker Peaks (and Marker Peaks unique to specific cell group)
- Prediction of specific Transcription Factor Binding Sites based on groups of peaks for a cell type (with specific accessible chromatin regions).
- TF motif enrichments (based on ChromVAR) and footprinting

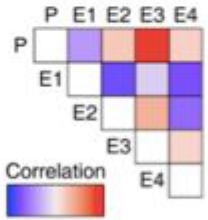
ChromVAR is designed for predicting enrichment of TF activity on a per-cell basis from sparse chromatin accessibility data.

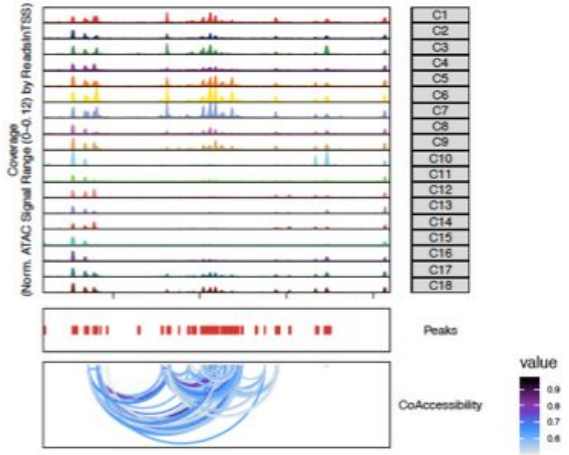
TF footprinting allows for the prediction of the precise binding location of a TF at a particular locus.

- Peak co-accessibility, is a correlation in accessibility between two peaks across many single cells.
- Peak-to-gene linkage, leverages integrated scRNA-seqdata to look for correlations between peak accessibility and gene expression.







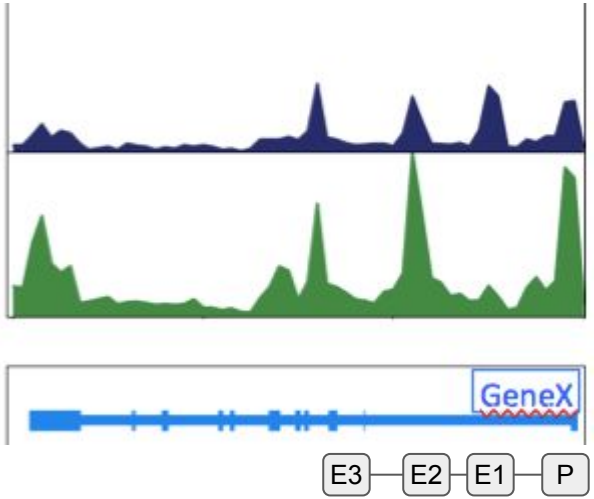


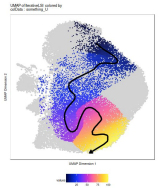
From ArchR (Full Manual)

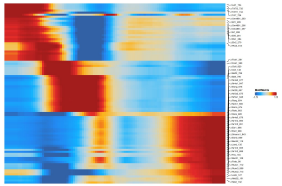
Perspectives / forthcoming analyses

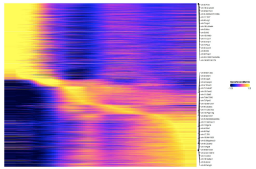
- Peak calling and downstream analysis:
  - TFBS prediction
  - TF Motif Enrichment
  - Footprinting
  - Peak co-accessibility
  - Peak-to-gene linkage

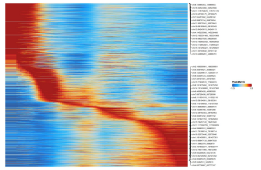
- Trajectory analysis











Acknowledgments

INSERM U932

TransIm Team

J Tosello

V Manriquez

L Niborski

E Piaggio

INSERM U830

RTOP Team

F Bourdeaut

IFGC Team

J Waterfall

NGS Platform