Developing a computational high-resolution scATAC-seq platform to prioritize non-coding genetic variants

Emine Ravza Gür
Green Templeton College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Michaelmas 2022

Acknowledgements

Personal

I would like to express my most profound appreciation to Prof Jim Hughes for his extraordinary supervision and mentorship during my PhD journey. I really appreciated that he gave me an opportunity to do a PhD at Oxford University and believed in me. The completion of my dissertation would not have been possible without his support. I am very grateful to Prof Gerton Lunter for his excellent supervision. I am thankful that he made my adaption to a new country and culture very easy and provided a very friendly environment for me to work in. I would like to acknowledge all Hughes and Lunter group members for providing a supportive environment and making my time here a pleasant experience. I would like to extend my sincere thanks to Simone, who provided me with not only technical support when I needed the most but also moral support; and Matthew Gosden, who was always there for me, supported me as best as he did and also thoroughly proofread half of my thesis work. I would like to thank Simone Riva, Martin Sergeant, Chris Cole and Liangti Dai for their technical support in developing the Avocato tool. Thanks also should go to my college advisor, Prof Keith Frayn and his wife, Theresa, for their moral support and for helping me to adapt to the country and PhD journey. I like to thank my thesis committee, Prof Thomas Milne and Prof Adam Mead, for their feedback during my thesis.

Also, I would like to offer my special thanks to my parents, Zeynep and Tahsin Öztürk and my siblings, Talha and Yahya Öztürk, for their support and belief in me. Especially, I am grateful for the tremendous moral support I received from my mother. I would like

to thank one of my dearest friends Betül Şahan, who has provided me with all kinds of morals, support and encouragement. Special thanks to Emily Georgiades for her friendship, kindness, and moral support when I needed it the most and for all our memories. She made my life in the UK a joyful experience. I would like to express my deepest and sincerest gratitude to my beloved husband, Doğan Gür, for supporting me spiritually and keeping me calm when I doubted myself throughout my PhD journey, especially during the writing-up period. This thesis would not have been possible without his outstanding support.

Institutional

I am more than appreciative of being supported by the Ministry of National Education Selection and Placement of Candidates Sent Abroad for Postgraduate Education (YLSY) scholarship. I want to thank the Republic of Türkiye and the Ministry of National Education for their support. I also would like to thank the Wellcome Centre for Human Genetics and the MRC Weatherall Institute of Molecular Medicine, particularly the MRC Molecular Haematology Unit and the Centre for Computational Biology, for providing me with a friendly environment for conducting this thesis work.

Abstract

A genome-wide association study (GWAS) is a standard approach for understanding the relationship between genetic variants and disease, resulting in a better understanding of disease mechanisms. However, despite the vast contribution of GWAS to our knowledge and understanding of human biology, our ability to interpret these findings has not yet been fully developed. This is primarily because most disease-disposed regions are found within the non-coding regions of the genome, which affect regulatory elements such as enhancers and promoters. There is currently no clear path to identify causal regulatory SNPs and to link these to the genes and cell types they affect.

This thesis work focuses on first understanding scATAC-seq as a technology and its outputs and then taking advantage of its potential power to prioritise non-coding genetic variants and cell types. I have deeply reviewed and explored single-cell open chromatin epigenetics, which led to discovering the effect of the Tn5 cutting pattern on providing different levels of information from scATAC-seq. I developed a mathematical strategy to remove nucleosomal background from scATAC-seq data, increasing the data resolution and enabling the accurate identification of TF-bound regions. We successfully built an analytical platform, Avocato, which provides a complete solution to understand the relationship between non-coding genetic variants and their affected diseases. Avocato is not only capable of analysing and visualising scATAC-seq data but also removes background noise resulting in high-resolution data to be used in prioritising non-coding genetic variants and cell types. Additionally, Avocato allows users to interact with their analysis and prioritisation results interactively in a user-friendly interface.

Together this thesis builds on existing technologies, bringing us one step closer to fully understanding GWAS findings and functional studies and elucidating the mechanisms of common human diseases.

Contents

List of Figures

List of Tables

List of Abbreviations

ATAC-seq	Assay for transposase-accessible chromatin using sequencing
Avocato	Analysis and visualisation of single-cell ATAC-seq observations
BAM	Binary alignment file
BED	Browser extensible data
ChIP-seq	Chromatin immunoprecipitation by sequencing
CITE-seq	Cellular indexing of transcriptomes and epitopes by sequencing
CRISPR	Clustered regularly interspaced short palindromic repeats
CTCF	CCCTC-binding factor
DAG	Directed Acyclic graph
DNA	Deoxyribonucleic acid
DNase-seq	DNase I hypersensitive sites sequencing
DR	Dimensionality reduction
Ery	Proerythroblast
FACS	Fluorescence-activated cell sorting
FAIRE-seq	Formaldehyde-assisted isolation of regulatory elements by sequencing
FDR	False Discovery Rate
GEM	Gel bead-in-Emulsion
GWAS	Genome-wide association study
H3K27ac	Histone modification – acetylation of lysine 27 on Histone 3
H3K4me1	Histone modification – monomethylation of lysine 4
H3K4me3	Histone modification – trimethylation of lysine 4
IFC	integrated fluidics circuits
LDA	Latent Dirichlet allocation
LSI	Latent semantic indexing

MAIT	Mucosal associated invariant T cell
MDV	Multi-Dimensional viewer
MDS	Multidimensional scaling
ML	Machine learning
MLV	Multi-Locus viewer
MNase-seq	Micrococcal nuclease digestion with deep sequencing
NMF	Non-negative matrix factorisation
NGS	Next-generation sequencing
NK	Natural killer cell
PBMC	Human peripheral blood mononuclear cell
PCR	Polymerase chain reaction
pDC	Plasmacytoid dendritic cell
PFM	Position frequency matrix
PL	Programming language
PP	Pre-processing
QC	Quality control
RNA	Ribonucleic acid
scATAC-seq	Single-cell ATAC-seq
scRNA-seq	Single-cell RNA-seq
SNP	Single nucleotide polymorphism
snRNA-seq	Single nucleus RNA-seq
T1D	Type-I diabetes
T2D	Type-II diabetes
t-SNE	t-distributed stochastic neighbour embedding
TCM	Central memory T cell
TEM	Effector memory T cell
TF	Transcription factor

TF-IDF	Term frequency-inverse document frequency
TSS	Transcription start site
TTS	Transcription termination site
UCSC	University of California Santa Cruz genome browser
UMAP	Uniform manifold approximation and projection
VAE	Variational autoencoder
VCF	Variant call format
WIMM	Weatherall Institute of Molecular Medicine

1 Introduction

1.1 The challenges of decoding human non-coding genetics

The genetic basis of common diseases differs from rare diseases. Rare diseases are typically caused by defects in one gene with a highly pronounced effect. In contrast, common diseases appear highly polygenic, with thousands of variants that contribute small effects individually to the phenotype of the disease. This makes understanding the mechanisms of common diseases difficult, as these interplaying genomic factors are often complex. This is where genome-wide association studies (GWAS) emerged. GWAS is a prevalent approach used in genomics to find weaker associations between a large sampling of genetic variants and diseases [1]. Understanding the underlying mechanisms of disease and the effect of genetic variants on these mechanisms plays a crucial role in their potential to develop targeted drug therapies to treat complex diseases [2].

Although GWAS had a significant impact in mapping genetic variants to specific diseases, several challenges still remain unmet in interpreting these GWAS associations [1]. An initial challenge is to understand the full haplotype structure of the genetic association. This is due to the fact that many GWAS studies rely on microarray platforms for genotyping individuals, where an individual SNP on the microarray actually represents a larger haplotype of genetically linked SNPs (or copy number or structural variants), any of which may be the true causal change detected via its linkage with the proxy SNP on the array. This difficulty in determining the actual causal

variant obviously hinders the design of downstream experiments from the basis of the genetic association [1], [2]. In practice, the linkage disequilibrium method is used to calculate a correlation between a genetic variant and its neighbouring variants within the same population to determine a complete understanding of the linked haplotype.

However, the analysis of such haplotypes has shown that most genetic variants are predominantly located in the non-coding part of the genome. This is one of the main reasons why it has been very slow to translate GWAS findings into clinical therapy, as interpreting the effects of non-coding variants is much more challenging than for coding variants [3], [4].

Studies of disease-associated regions have shown that many have variants that intersect with enhancers, promoters and other regulatory elements and affect their function [1]. This suggests that a major underlying mechanism in common disease genetics is the alteration of gene regulation rather than the coding region and resultant protein structure. However, understanding which gene or genes a specific enhancer element controls is still extremely challenging. Such elements are unpredictable in both number and distribution around the genes they control [1].

Similarly, genetic associations are identified by GWAS in a cell-type independent manner, although gene regulatory programmes are highly cell-type specific. Therefore, while present in the germline, the sequence variant is likely to exert its effect in only certain cell types or stages of differentiation in the lineage of these cell types. Additionally, it has been shown that cell types, which are likely the causal cell type, show statistically increased coincidence between GWAS variants and these cell

type-specific elements, which have been used to determine which cell types are relevant for a given GWAS [1].

Therefore, to decode the predominantly non-coding genetic associations found in common diseases, multiple challenges must be solved, often simultaneously, to generate hypotheses and guide functional genomics experiments:

1. Which part or parts of the haplotype is causal?
2. In which cell type, including stages of differentiation or cellular activation, are they acting?
3. Via which mechanism are they acting (regulatory, splicing etc.)?
4. Which genes are being affected?
5. How to solve these associations at a sufficient scale to map affected pathways?

While it has been shown that other mechanisms, such as splicing, can underlie GWAS associations, my thesis focuses on variations that likely exert their effect through an intersection with cell type-specific regulatory elements. Therefore, my thesis aims to address points 1, 2 and 3 at a sufficient scale to apply to point 5.

The overall goal is to generate a platform to guide scientists to design appropriate experiments in the right cell type and with a prioritised likely regulatory variant, to address point 4 and identify the affected genes. My strategy in finding causal SNPs and the cell types in which they are functioning leverages new single-cell epigenetic assays, particularly open chromatin data [1].

1.2 Functions of the noncoding genome that can be affected by sequence variation

Understanding the effect of regulatory variants is crucial to elucidate the mechanisms underlying common human diseases. The most informative data for understanding these comes from epigenetics, which aims to understand the nuclear function and, in particular, the mechanisms by which genes are controlled and regulated in different cellular and environmental contexts. Epigenetic changes are the physical and chemical modifications of DNA and its associated proteins (chromatin) that can affect nuclear behaviour, such as changing gene activity, without altering the underlying DNA sequence. The epigenome, therefore, refers to the collection of all epigenetic modifications in the genome and is known to play an essential role in regulating gene expression in cell differentiation and development [5], [6]. Gene regulation is the central process that directs cell differentiation, development, and cellular response to extrinsic cues. Epigenomics is, therefore, all physical and chemical modifications in the genome [6], [7] and their role in genome function.

Chromatin is DNA packaged into nucleosomes or bound by other chromatin-associated proteins. A nucleosome can be thought of as a fundamental unit of chromatin in which DNA is wrapped around proteins called histones, simultaneously packaging ~2m of DNA into the space of a single nucleus whilst also allowing dynamic regulation. Physical access to chromatin differs based on many factors, including nucleosome occupancy, organisation, and association of chromatin remodelling proteins. The level of chromatin accessibility ranges from closed chromatin, where nucleosomes are densely positioned, to open chromatin, where transcription factors

can bind to DNA and exert functional effects. The distribution of chromatin accessibility in the genome is remarkably dynamic across different cell types and states [8] (Figure 1.1).

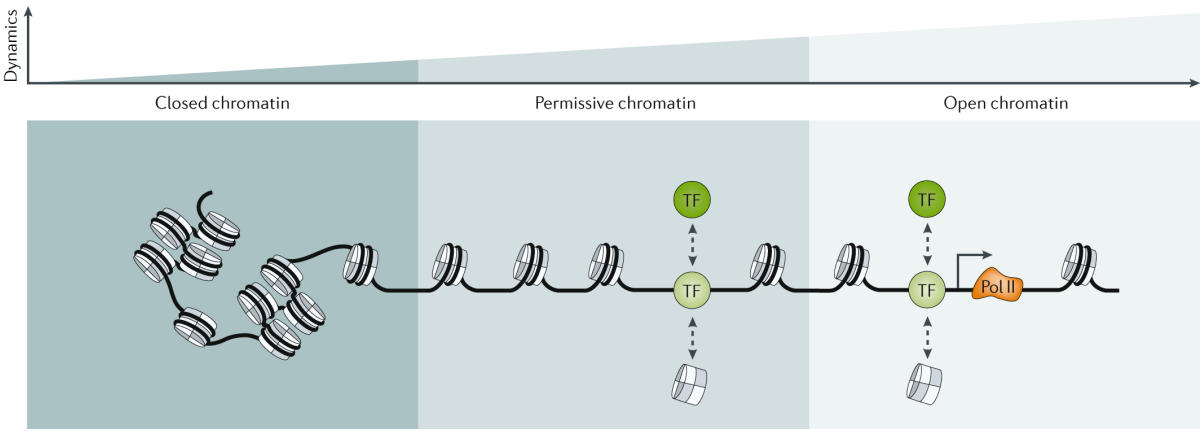


Figure 1.1: The activity of chromatin accessibility is variable across the genome. The level of dynamic activity is very low when chromatin is in a closed state. As chromatin starts to become more accessible (towards the right-hand side of the figure), chromatin shows high dynamic activity for TFs to bind the DNA and start transcription. TF, transcription factor; Pol II, RNA polymerase II. Figure adapted from Klemm *et al.* 2019 [8]

1.3 Mapping functional elements in the non-coding genome using open chromatin assays

Studying chromatin accessibility genome-wide allows us to profile the occurrence and position of regulatory elements at nucleosome-depleted regions, including enhancers, promoters, and insulators. These key elements lie within the non-coding part of the genome and regulate gene expression [8], [9].

Many assays have been developed to measure chromatin accessibility profiles at a genome-wide scale. The common principle behind profiling chromatin accessibility is based on marking more accessible regions of DNA due to their increased levels of fragmentation with nuclease enzymes or tagmentation with transposases. These

nucleosome-depleted regions allow the targeting enzymes to access the DNA more readily. DNase I hypersensitive site sequencing (DNase-seq) [10] is the first genome-wide next-generation sequencing-based open chromatin assay and uses the enzyme DNase I to fragment DNA, preferentially releasing fragments from open chromatin regions [11], which are then identified by NGS sequencing. This method has now largely been replaced by Assay for Transposase-Accessible Chromatin using sequencing (ATAC-seq) that uses a tagmentation process which is more progressive in open chromatin regions compared to closed [12]. Although DNase-seq and ATAC-seq are based on similar principles, ATAC-seq has many advantages over DNase-seq. In summary, the ATAC-seq assay is a quicker and much more straightforward protocol that uses a mutated hyperactive form of Tn5 transposase to simultaneously fragment DNA and tag it with adaptors within isolated cell nuclei, a process referred to as tagmentation.

On the other hand, since the DNase-seq is an enzyme digestion-based assay, the amount of enzyme directly affects data quality, where under-digestion and over-digestion both drastically affect the signal-to-noise ratio. Therefore, it needs thorough enzyme calibration to reach optimum sensitivity. This requirement for optimisation of enzyme concentration in DNase-seq means the protocol is long and takes several days to complete. In addition, the need to test multiple concentrations of DNase means that the required cell input for DNase-seq (1-10 million cells) is much higher than in ATAC-seq (500-50,000 cells). As well as DNase-seq and ATAC-seq, there are other chromatin accessibility assays: FAIRE-seq [13], which enriches for non-nucleosomal DNA via crosslinking, sonication and phenol-chloroform extraction, partitioning the less protein-associated DNA in the aqueous phase and nucleosomal DNA in the

phenol phase; MNase-seq [14] aims to identify nucleosome positioning in the chromatin using MNase, which cuts chromatin both at open chromatin and in nucleosomal linker regions [8], [9]. However, due to limited sensitivity (FAIRE-seq) and much larger requirements for sequencing (MNase-seq), these methods are infrequently used for the mapping of open chromatin sites in a cell type. ATAC-seq has now become the most widely used epigenetic assay in many fields because of its ease of performance and low cell number requirements, while DNase-seq remains restricted to more specialist labs [9].

1.4 Open chromatin epigenetics at the single-cell level

The use of such bulk epigenomics technologies has provided almost complete maps of regulatory elements in a variety of well-defined cell types in many organisms. However, due to their cell number requirements, samples are often homogenous populations of cells and their output, therefore, represents an aggregated average of all cells in the sample. Therefore, unless working with an already homogenous population such as, a cell line, cell isolation steps need to be completed prior to tagmentation by sorting or column-based isolation to ensure sample homogeneity. Bulk assays can, therefore, not be applied when cell purification is not possible or impractical and cannot detect unsuspected or poorly characterised cell populations from a mixture of cell types. Since population-based methods measure overall epigenomic information, aggregating the data from all cells, they ignore cellular and regulatory heterogeneity in cell types. This means the ability to isolate defined populations of cells from their heterogeneous tissue of origin is limited by the availability of methods [7], [15]. Recent technological and computational advances

have rendered it possible to map chromatin accessibility at a single-cell level, which has allowed for the simultaneous analysis of chromatin accessibility landscapes of heterogeneous mixtures of cells. ATAC-seq in particular was rapidly adapted to single cell platforms and the development of high-capacity commercial single cell ATAC-seq platforms has continued to fuel the rise in the use of ATAC-seq as a data type [16].

Although the data from any individual cell is very sparse, computational approaches have been developed to cluster cells with similar epigenetic landscapes within the analysed population of cells, which can then be aggregated to produce detailed epigenetic profiles from each cell cluster within the heterogeneous mixtures of cells. In this way, scATAC-seq technology can analyse cellular and regulatory heterogeneity within cell populations and detect new populations *a priori* [16].

Generating the accessibility landscape of every cell type in a tissue or particular niche with traditional Bulk ATAC-seq assay requires intense effort and money to complete chromatin accessibility maps across tissues as cell isolation needs to be done per cell type. However, this can be easily done with the scATAC-seq assay. It can cluster cells that show similar chromatin accessibility profiles within a complex tissue type or a mixture of heterogeneous cells population, which can lead to discovering new cell populations, identifying rare cell types and defining new chromatin accessibility states. The current approach is, therefore, to computationally annotate cells into clusters based on the similarity of their epigenetic landscape and then to aggregate the cells in these clusters (pseudo-bulking) to generate a genome-wide enrichment track equivalent to Bulk ATAC-seq [7], [9], [17].

1.5 The potential of scATAC-seq in interpreting genetics variants

Since ~90% of GWAS associations lie under the non-coding regions of the human genome, [18] further functional studies need to be done in order to expand our ability to interpret these GWAS findings functionally. The integration of GWAS results with functional genomics data can effectively address the challenges explained in section 1.1. Some existing approaches can be incorporated into an overall analytical approach to overcome the current difficulty of interpreting GWAS associations. Firstly, fine mapping is a statistical method popularly used to refine a set of potential causal genetic variants. Moreover, it is difficult to validate GWAS variants functionally without first identifying the cell types in which they act. This is further complicated by the fact that many tissues may contribute to the development of a trait. The causal cell types can be identified through the computational integration of genetics with chromatin marks or gene expression information. scRNA-seq and scATAC-seq are especially powerful tools that allow us to gain a deeper understanding of cellular and regulatory heterogeneity at the single-cell level. Lastly, the colocalisation of variants within GWAS haplotypes and open chromatin signal can be used to prioritise possible affected genes by causal variants [1], [2].

When the sequence variation in an individual's genome has been characterised, scATAC-seq is a powerful approach for linking sequence variation with variation in gene regulatory mechanisms. This can then be seen in many cell types while simultaneously mapping population composition as well as epigenetic variation in individual cell types (clusters) [1], [15]. Furthermore, genetic variants in the non-coding

regions can alter the regulation of genes. Most frequently, one allele may be affected by these variants. When the heterozygous genotype of a sample is known, then allele-specific imbalance in open chromatin signal can be detected in the ATAC-seq reads helping to confirm the effect and causality of the variant. Allelic skew can be both positive and negative, increasing or reducing the level of chromatin accessibility associated with the presence of a variant, and a single variant is capable of completely inactivating an element as judged by the open chromatin [19].

1.6 Different experimental methods to perform ATAC-seq at the single-cell level

There are different ways of capturing single cells in a scATAC-seq assay. Early methods were either based on using a microfluidic device to capture single nuclei (Fluidigm, C1) [20] or applying a combinatorial indexing strategy which relies on 2-step cellular barcoding rather than the physical separation of cells [21]. The microfluidic method relies on performing transposition and PCR on single cells inside integrated fluidics circuits (IFCs), followed by collecting single-cell libraries from the IFC and using PCR to apply cell-identifying barcodes. Then, those libraries are pooled and sequenced using high throughput sequencing, followed by deconvoluting data from the same cells based on the incorporated barcodes [20], [22]. On the other hand, combinatorial indexing approaches do not require single-cell compartmentalisation but use a 2-step molecular barcoding system based on two sets of 96 well plates to identify single cells. Populations of lysed cells are placed into individual wells of a 96-well plate where the tagmentation process and the addition of a well-specific barcode occur by tagmentation. These populations of nuclei are then pooled again, diluted and then

redistributed using FACS-sorting as new populations into individual wells of a second plate before introducing the second barcode during the amplification step on the second plate. After amplification, pools are combined and sequenced in the final step. As the vast majority of nuclei pass through a unique combination of plate wells, the data from an individual cell will contain a unique combination of the two barcodes. Each cell in the data can be easily identified [21]–[23].

When the scATAC-seq system using the microfluidic device system and combinatorial indexing first emerged, it came with several challenges, such as expense, the requirement for specialised equipment and the need for abundant tailored Tn5 [24]. In the recent version of sciATAC-seq (version 3), it does not require specially made Tn5, but instead works with commercially available Tn5 and can work with other mechanisms, namely well plates, flow cytometry etc., to reduce cost and increase compatibility. It contains a more sophisticated indexing strategy with three levels of indexing. It does not involve cell sorting; the first two indices are added via ligation, whereas the last, as in the original method, is added via a PCR step [25]. Even if the first version of the microfluidic-based system is simple and provides controlled experiments, it is low throughput and can only be performed in batches of 96 cells [23]. However, the recent improvement in microfluidics has enabled high throughput processing via commercialised and non-commercialised applications [22]. For example, Lareau *et al.* used a droplet-based microfluidic device to process 46,653 cells in their study [26].

Additionally, commercial companies like 10X Genomics provide a high-throughput droplet-based scATAC-seq assay [9]. 10X Genomics creates Gel bead-in-Emulsion

(GEM) micro reaction structures on a microfluidic device. Bulk transposed nuclei are incorporated into GEMs with unique barcodes in the microfluidics device, and barcodes are added to the transposed DNA fragments. The beads are then dissolved, and linear amplification is performed on the transposed DNA. In the final step, following the dispersal of the emulsion, DNA fragments are amplified via PCR prior to high-throughput sequencing [27].

1.5.1 Technical derivatives combined with other omics data

From these, there have been many modifications to the initial methods including; Perturb-ATAC, Pi-ATAC, and plate-based scATAC-seq as just a few examples [22].

Perturb-ATAC allows the investigation of the effect of genetic modification, either by knockout or CRISPR, on the chromatin accessibility [28]. Pi-ATAC merges epigenomic and proteomic profiling together in a new single-cell assay by measuring protein expression via FACS and then identifying chromatin accessibility within the same single cell [29]. Plate-based scATAC-seq sorts single cells and places them individually into either 96-well or 384-well plates [9], [24]. Due to the limited number of wells, the number of cells processed is low [9]. Takara Bio USA has developed a nano-well array (ICELL8 Single Cell System) to overcome low throughput; ICELL8 can process 5,184 nano-well arrays and generate equal or higher DNA fragments with reduced experimental cost coupled with fluorescence imaging [26].

1.7 Analytical challenges in scATAC-seq data analysis

While scATAC-seq is a very powerful method, it suffers several analytical challenges. The main challenge is that there is no consistent annotation across cell types to know where to quantify chromatin signals to represent the epigenomic landscape of a given cell type correctly. For scRNA-seq, gene annotation, which applies to any cell type, specifies where the genome data should be quantified to effectively represent the cell type-specific transcriptional activity and form the basis of clustering analysis. In contrast, for scATAC-seq, the relevant open chromatin landscapes vary from cell type to cell type, so no “one fits all” annotation for extracting the relevant signal can exist. This makes it challenging to form clusters from scATAC-seq data as, due to the sparsity of the signal in a single cell, it is unclear whether a signal represents the background or true open chromatin activity [22].

Moreover, since scATAC samples DNA rather than RNA, the chromatin accessibility profile is either zero for the closed state or one or two for the open state for diploid organisms. The maximum accessibility score of two for diploid organisms corresponds to the two alleles within a cell, unless the DNA is being replicated. Because of the binary nature of scATAC-seq, the data is very sparse, leading to difficulties in data analysis. In practice, this means that computational approaches developed for scRNA-seq could not be directly translated into the analysis of scATAC-seq data. Therefore, even as the technologies for data generation quickly improved, analysis lagged behind it and it remained unclear as to what was the most informative way to analyse scATAC-seq data. Since the scATAC-seq assay is a brand-new state-of-the-art technology, it lacked reciprocally sophisticated analytical tools to analyse, cluster and visualise

scATAC-seq data [16], [22]. However, the requirements for effective analysis of the new dataset can be broken down into several steps.

1.8 The computational requirements for scATAC-seq data analysis

Standard data analysis for scATAC-seq data should include the following steps:

1. Preliminary quality control: While this step is not compulsory, it is good practice to familiarise yourself with the raw data regarding adapter and GC contents, distribution of sequence length and so on [16]. FastQC is a frequently used tool for this purpose [30].
2. Adapter trimming: In the presence of sequence adapters, they should be trimmed from raw sequencing data as they introduce biases or prevent the alignment of raw reads [16]. The most commonly used adapter removal tools are the cutadapt [31] and Trimmomatic [32].
3. Alignment: Adapter-removed sequencing files should be aligned to a reference genome build of interest. Different aligners use different strategies to align the raw data, which could affect the output of the analysis. The most logistically important features of alignment tools are that they are fast and memory-efficient [16]. The commonly used ones for scATAC-seq data are Bowtie2 [33] and bwa-mem [34]. Aligners output an alignment file in SAM file format, which is a text file and colossal in size. Therefore, these alignment files are converted to a compressed sorted and indexed binary form (BAM) to save computational time and memory for post-processing steps. In addition to saving space, sorting and indexing mean data from a specific region of the data can be accessed without

reading the complete file, which greatly speeds up processing. SAMtools [35] is commonly used for efficient computational manipulation of these large files. Additionally, sequence alignment also provides information about alignment rate, which is an excellent initial criterion to determine sequence quality. The average expected successful alignment rate is ~80% [16], [22].

4. Quality control: After the alignment, further quality control analysis is recommended to generate more detailed information about data quality, including the number of duplicated and uniquely mapped reads. These criteria are important reflections of the complexity and, therefore, the potential information content of an NGS library. Picard tool [36] and SAMtools can produce such metrics [16]. Additionally, MultiQC enables a comprehensive interactive HTML report, including FastQC results, to assess basic statistics about alignment, look at reads that have been removed from the data via adapter removal and summarise reads counts by assigning them to genomic coordinates [37].
5. Filtering: After assessing post-quality control results and initial data quality assessment, further data cleaning steps are required before analysis. Reads with low mapping scores and unmapped pairs are removed as their correct position in the genome has low confidence, and discarding them decreases file size. PCR duplicates are removed to prevent bias quantitation. Mitochondrial reads are typically removed as mitochondrial contamination can be a large and highly variable component of any genomics experiment, especially ATAC-seq. They can drastically skew the normalisation of datasets by read count. ENCODE blacklisted regions should be excluded from the alignment file as these regions are prone to bioinformatic enrichment and false positives due to

copy number differences between the virtual genome used in the alignment process and the real genomes sequenced [16].

6. Post-alignment Quality control: Additionally, scATAC-seq specific plots can be generated to evaluate the alignment result biologically. In scATAC-seq data, reads are expected to enrich around TSS regions of genes, so TSS enrichment per DNA fragments plots can be created as a guide to general data quality. Another important metric is to look at the size distribution of DNA fragments. Data of good quality shows a reproducible pattern for size distribution which is reflective of the nucleosomal pattern in chromatin, with mono, di and tri nucleosomal peaks evident in the data [16].
7. Defining regions: After processing reads from the technical point of view, the most important part is identifying biologically informative regions to quantify data for cluster formation. Mainly, there are two strategies to describe open chromatin regions, namely using genomic coordinates and sequence content-based [17], [22].
    a. Using genomic coordinates: This method is mostly known as the peak calling approach, which tries to find enriched regions of accessible chromatin by peak calling aggregated scATAC-seq data or related Bulk ATAC-seq [17]. However, having equivalent Bulk ATAC-seq data is usually unavailable, so applying peak calling on aggregated scATAC-seq data is the popular approach to define open chromatin regions (see Figure 1.2.A). MACS2 [38] is the most commonly used peak caller for chromatin accessibility data. Another alternative is HMMRATAC [39], which uses a hidden Markov model and, to date, is the only peak caller developed specifically for the chromatin accessibility data [9], [16].

However, all the peak calling methods have a major drawback: Since the peak call is performed on the aggregated data from all the single cells, peaks from poorly represented cell types in the heterogeneous mix are difficult to peak call in the context of the overwhelming signal from the major cell types. This mean regions critical for the discernment of these clusters are not included in the quantification and clustering processes. This could lead to losing small but biologically important cell clusters, such as stem or progenitor cells. Therefore, an incomplete or skewed set of defined regions can lead to bias in the downstream analysis [40].

b. Using bin size approach: Another approach is to split the genome by fixed-bin size. This method splits the genome into bins and counts features within each bin, as is seen in Figure 1.2.B. This means that every region in the genome contributes to the computed matrix used for cell clustering. This is more reliable than the peak calling method, but until recently, it has been found to be too computationally heavy for general application [17], [40].

8. Feature matrix generation: After open chromatin regions are defined, a matrix is generated by counting the number of defined features per cell. The chromatin matrix (size = total number of cells * ~10^6 regions) has much higher dimensions compared to the gene expression matrix (size = total number of cells * ~20,000 genes in the human genome) [36].

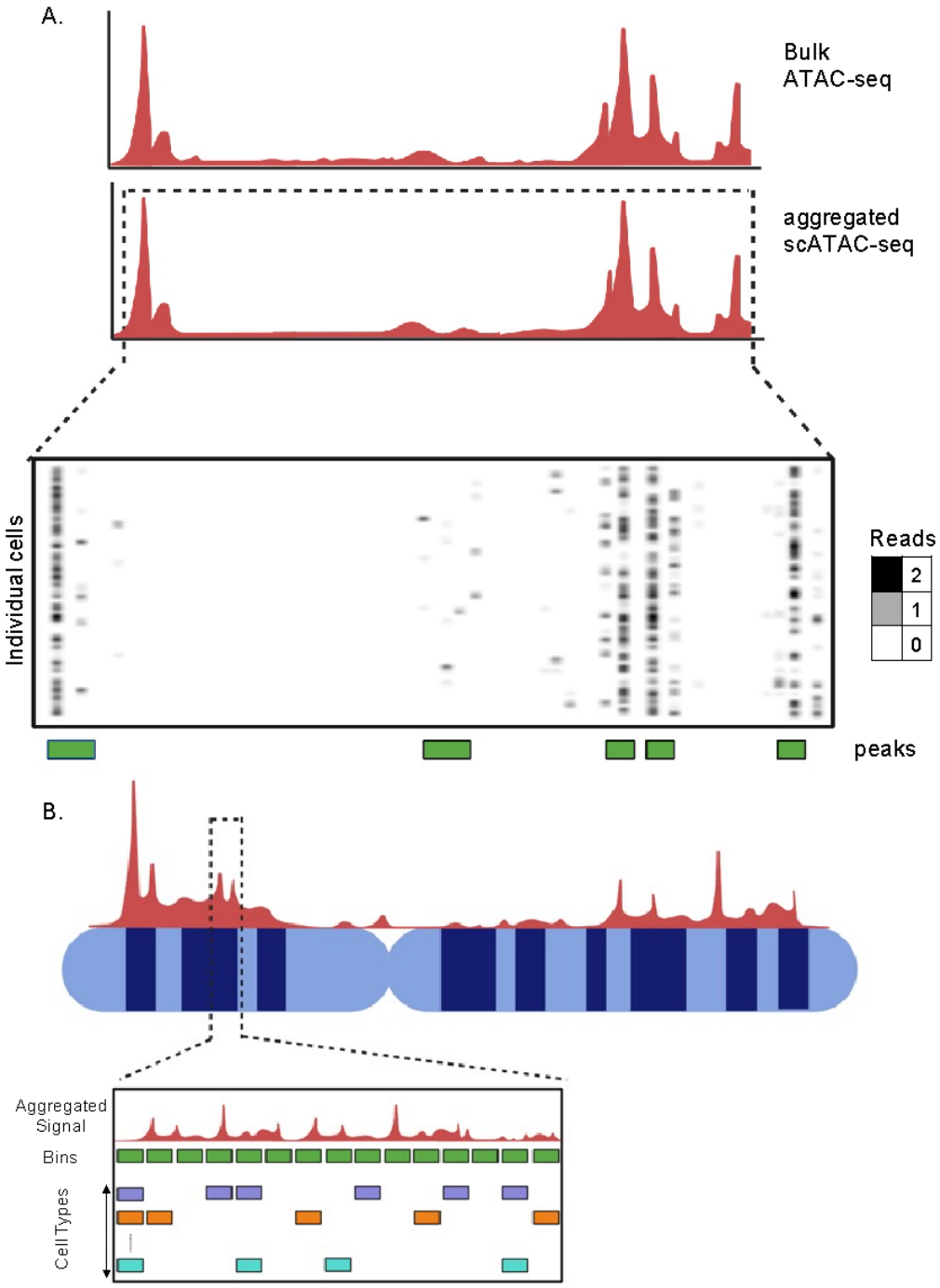


Figure 1.2: Methods for defining chromatin accessibility regions in scATAC-seq data analysis: using peak calling approach (A) and using bin size approach (B)
A. Red tracks are ATAC-seq signals: the top belongs to Bulk, whereas the lower is an aggregated signal. The box beneath the single-cell track represents the read count of the chromatin accessibility signal coming from individual single cells: 2, a signal from both alleles; 1, the signal from one allele; 0, no signal. Green blocks at the bottom show peak calling results. B. Red tracks here represent the scATAC-seq signal genome-wide. This method uses a window to bin the genome to obtain the matrix below. This binning method results in having a huge number of bins, shown as green blocks, each bin representing a set of chromatin accessibility signals. Only the read matrix in A was taken from Chen et al. [17]. The rest was created with “BioRender.com”.

9. Transformation: Considering the high dimensionality and sparse nature of the scATAC-seq raw feature matrix, data transformation needs to be performed to capture more informative regions and emphasise the epigenomic landscapes for rare cell types in the matrix. The typical approach for data transformation is first to binarise a raw feature matrix. The justification for the binarisation procedure is the sparse nature of scATAC-seq. In diploid organisms, the maximum value for accessibility score can be 2, so it makes sense to apply binarisation to decrease computational effort and avoid bias coming from sequence depth and PCR duplicates. After binarising the raw matrix, the binarised matrix can be used as it is as an input to the next step, or further transformation techniques can be applied to enhance signals which can be indicators for small cell populations. There are two popular methods for further data transformation based on different mathematical principles: applying the term-frequency-inverse-document-frequency on the binarised matrix and calculating Jaccard distances in the binarised matrix [17], [22]. The TF-IDF method is a very good statistical scoring that emphasises more informative features for small cell types [22], [41]. Measuring Jaccard distance can help to understand the relationship between two cells based on their accessibility level. It can tell whether each pair of cells are similar or not [22], [42].
10. Dimensionality reduction: This is a compulsory step to reduce the high dimensionality of the feature matrix to a low dimensionality by excluding redundant dimensions which do not contribute variable information to the data. In other words, DR methods remove redundant dimensions and decrease the level of noise and computational time for the next steps. Different DR methods use different mathematical or statistical strategies to reduce dimensions.

Commonly used DR methods in scATAC-seq are PCA, LDA, Latent Semantic Indexing (LSI) and its variants [22].

11. Visualisation: After reducing the dimension of the feature matrix, further non-linear dimensionality techniques can be applied to visualise scATAC-seq data in 2- or 3-dimensional space. T-distributed stochastic neighbourhood embedding (t-SNE) and uniform manifold approximation and projection (UMAP) are commonly used methods for visualising the data. Kobak *et al.* [43] summarise a detailed comparison between the two methods.
12. Clustering: The clustering step groups together cells with similar chromatin accessibility patterns as clusters. Louvain clustering is the most popular clustering method. Hierarchical, k-means and k-medoids clustering methods are also used [17]. Behind each clustering algorithm lies different mathematical strategies to cluster similar cells together. The k-means method is easy to implement, but it needs prior knowledge of a number of clusters which can lead to a false classification of cells when cluster distribution is imbalanced. For example, since rare cell type populations consist of a small number of cells, they can be easily hidden in one of the large clusters [44], [45]. Even though hierarchical clustering is very practical in showing the relationship between clusters, it performs poorly when data is too large and the number of features is too high [44]. Louvain clustering is a graph-based clustering method widely used in single-cell omics. Empirical testing of these in this study [15] showed Louvain clustering to outperform the other methods in clustering scATAC-seq data [17], [45]. Unfortunately, there are no one-fits-all clustering methods. Depending on the dataset and its properties, the choice of the clustering method is interchangeable [22], [45].

13. Cell type assignment: After forming clusters, biological cell type identity needs to be determined for each cluster. Since scATAC-seq profiles the epigenetic landscape of regulatory elements, which lie under the noncoding part of the genome, it is challenging to use gene markers to annotate cells [22]. Although there is a plethora of cell-type annotation tools for scRNA-seq, only a couple of methods have been developed to annotate cell clusters for scATAC-seq data. Cluster annotation can be done by using the following methods: inferring gene expression from scATAC-seq [40], integrating different modalities (data integration) [46], doing single-cell multi-omics experiments [47], [48] and using reference atlases like Azimuth [49].
    a. Calculating gene activity scores: Gene activity scores are calculated to infer gene expression more accurately from chromatin accessibility profiles. The correlation between gene activity scores and gene expression profiles is more reliable than those between promoter accessibility profiles [22]. The calculation of gene activity scores depends on the regions used, their length and their weight based on the distance to genomic regions. Once the gene activity score is calculated, if a list of the gene markers has been given, the score for those markers can be acquired to annotate clusters. If no marker list is available, all scores must be examined prior to the assignment [40].
    b. Data integration: While scATAC-seq is a very powerful technique to map regulatory elements like enhancers and promoters, it is not good at assigning cell clusters to cell populations. Therefore, cluster annotation can be improved by leveraging data integration with scRNA-seq data. The Seurat R package is one of the commonly used data integration

tools. It is robust to batch effects between experiments [46]. Secondly, reference-based data integration leverages transfer learning. It trains a reference dataset to learn certain cell types and then makes predictions to query the dataset based on the reference set. Azimuth is an excellent example of reference-based data integration as an atlas format. Azimuth learns cell type weights from a CITE-seq dataset that provides ground truth for cell identity via protein and gene expression information by identifying the relative importance of each modality [49]. The third option is to do a single-cell multi-omics experiment. It is now possible to generate snATAC and snRNA data from the same nucleus and link these modalities via the shared barcode ids [22].

14. Advanced downstream applications: After completing technical pre-processing steps and core biological analysis, several advanced analyses can be done using scATAC-seq data, such as finding differentially accessible regions [42] to classify cell type-specific regulatory regions using statistical tests, performing trajectory analysis [50], [51] to study differentiation, and calculating co-accessibility [40], [50] to study interactions between enhancers, promoters and others [17], [22].

1.8 Computational tools to analyse scATAC-seq data

Initially, when beginning my project, limited tools or approaches were available to analyse scATAC-seq data specifically. However, in the intervening period, there has been considerable progress in developing analytical approaches and tools to tackle the analytical challenges detailed above. Of course, as such approaches are very new,

there is little consensus in the field regarding their effectiveness and biases. Therefore, understanding the most effective analytical path to analyse scATAC-seq is critically important to my work. To date, several computational tools have been developed using different strategies to overcome the aforementioned challenges of scATAC-seq analysis. Chen *et al.* [17] benchmarked 10 computational tools for the scATAC-seq data analysis in 2019. Table 1 summarises all the currently available tools for scATAC-seq data analysis up to date.

It's worth mentioning that not all tools are designed to analyse chromatin data from end-to-end. The ones that do perform end-to-end analysis include SnapATAC [52], Signac [53], EpiScanpy [54], MAESTRO [47], scATAC-pro [55], Scasat [42], Destin [56], Cusanovich2018 [57] and BROCKMAN [58]. Other tools, such as those listed below, only perform some of the necessary steps in the data analysis hierarchy.

- scOpen is designed for scATAC-seq data imputation and denoising [59].
- SCALE [60] and PeakVI [61] developed new deep learning-based dimensionality reduction methods using VAE; they model the data differently: with a Gaussian mixture model and Bernoulli distribution, respectively. PeakVI also considers technical factors, such as sequencing bias and batch effect, and its model does not overfit [61].
- While Cicero focuses on calculating co-accessibility as well as trajectory analysis, it does not provide DR and uses the Monocle3 [62] tool to reduce the dimensions [50].
- cisTopic creates a Bayesian model based on topic modelling to perform dimensional reduction [63].

- scABC proposed a new clustering method based on a defined statistical model [64].
- chromVAR is an excellent tool to characterise motifs [65].
- Dr.seq2 focuses on producing detailed QC plots [66].
- SCRAT provides a web-based interface for performing downstream analysis with different options for the DR methods [67].

In contrast to these partial packages, ArchR not only offers a novel dimensionality reduction technique but also has enabled the most comprehensive end-to-end analysis framework, including calculating gene activity scores using a well-validated gene activity model, removing doublets, footprinting, motif enrichment, trajectory analysis, calculating coaccesibility, visualising genome interactively, integration of Bulk RNA-seq, and creating a peak-to-gene link [40]. Similarly, scATAC-pro also provides a complete analysis solution. It is a flexible pipeline with many DR and clustering methods options and produces an HTML report for analysis results [55].

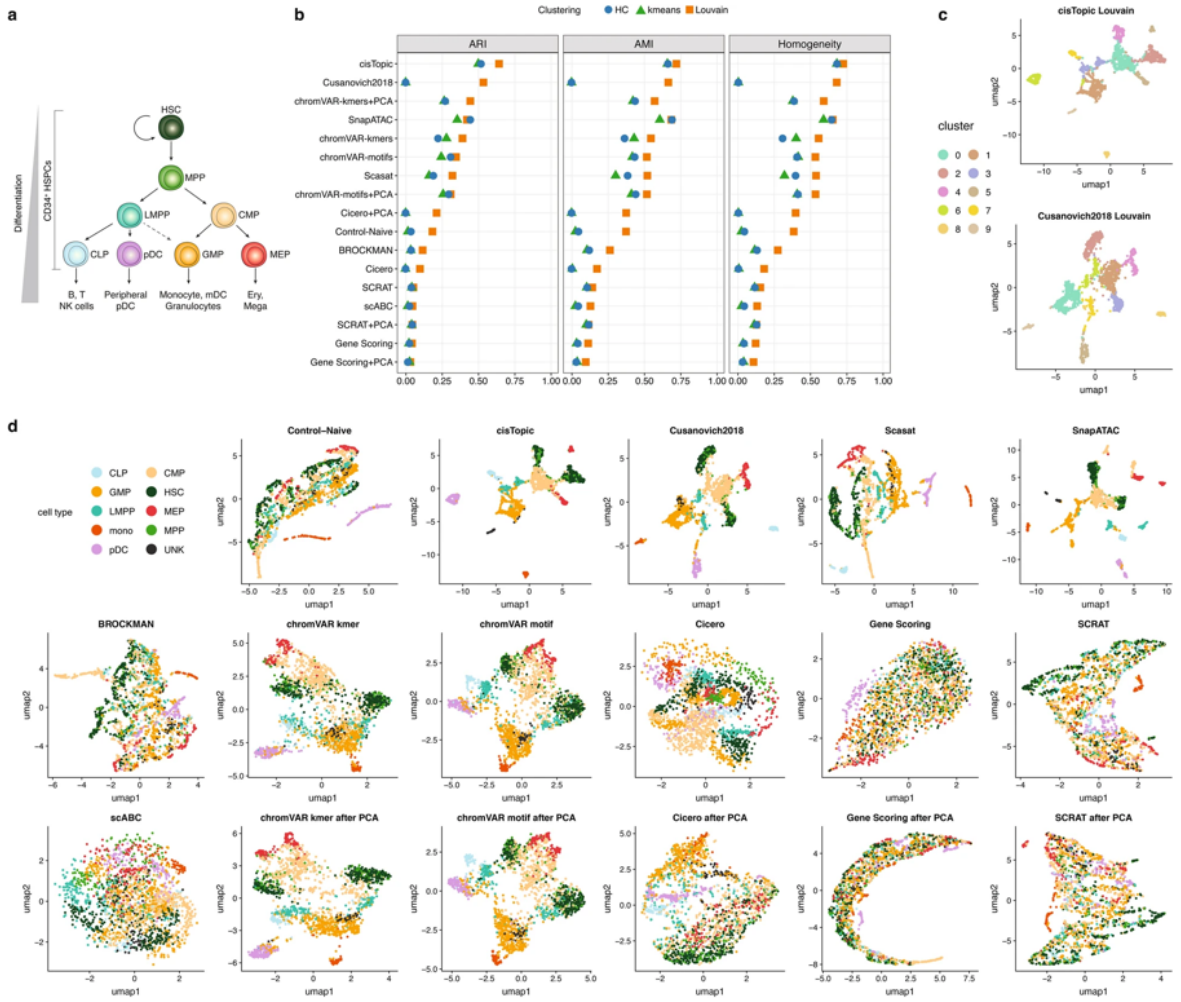


Figure 1.3 Benchmarking results of 10 computational pipelines [17] for analysing the Buenrostro2018 scATAC-seq dataset [68]
A. A differentiation arch describing the development of the cell types analyzed. B. The clustering performance of each analytical tool based on its ARI score is shown as a dot plot. C. Louvain clustering results from the two tools (cisTopic and Cusanovich2018) that had shown the best performance according to the score shown in b are shown as UMAP projections cells coloured by their related cell types shown in a. D. Louvain clustering results from all tools are shown as UMAP projections cells coloured by their related cell types.

1.8.1 Comparison of different analytical pipelines in terms of how they define chromatin accessibility regions

As previously described, a key challenge in scATAC-seq is the uncertainty about which regions are informative and can distinguish cell types within a complex mixture. One of the significant differences among these tools is how they extract signals from the data to form count matrices. This step is critical to all of the downstream analyses

as the statistical analysis of these count matrices allows for the identification of cell clusters in complex mixtures of cells. Therefore, if the extraction of data is incomplete or misdefined, then this will bias the clustering and identification of cell types within the sample.

Cicero, cisTopic, chromVar, scABC, Scasat, EpiScanpy, scATAC-pro, Signac and Destin use peak calling results either from aggregated scATAC-seq data or related Bulk ATAC-seq samples to produce peaks to define accessible regions, while ArchR and SnapATAC use the more comprehensive method of analysing the whole genome in fixed-bin sizes; 500bp and 5k, respectively. Generally, regulatory elements are regions that are 300-500 bp long [17], [22], [40]. Using a 5kb window can cause the merging of signals between regulatory elements, which could lead to misleading clustering results. Therefore, using a 500bp window provides a more fine-grained mapping of the accessibility of regulatory elements, which easily highlights where the TFs are bound [40]. There are other strategies to define regions. For example, BROCKMAN represents data by applying gapped k-mers around transposon integration sites [17]. Additionally, Cusanovich2018 uses an integration of two methods; splitting the genome into windows and then, calling peaks. First, it splits the genome into 5kb windows and performs LSA for each window, followed by forming pseudo-Bulk-clades and calling peaks within [57].

According to the results of benchmarking 10 scATAC-seq tools in Chen *et al.* [17], while SnapATAC, cisTopic and Cusanovich2018 outperform other tools, Louvain clustering is the best clustering method for scATAC-seq (Figure 1.3) [17]. However, cisTopic cannot deal with large datasets [59]. SnapATAC and Cusanovich2018 use a

5kb window to segment the genome, which can lead to over-clustering regulatory elements [40]. Wang *et al.* [47] benchmarks their tool, MAESTRO, against scABC, LSI, cisTopic and SnapATAC. This study benchmarks their tool, MAESTRO, with scABC, LSI, cisTopic and SnapATAC. The results showed that LSI-based methods have a high level of scalability, high clustering accuracy and high computing performance. MAESTRO can offer a gene activity model using scRNA-seq in the presence of single-cell multi-omics experiments [47]. Tools to analyse scATAC-seq are written mostly in R, some in Python [22]. MAESTRO provides a Snakemake environment for the analysis pipeline [47].

1.8.2 Comparison of different analytical pipelines in terms of how they calculate gene activity scores

When there is no equivalent scRNA experiment for the data, calculating gene activity scores is the only approach to annotate clusters in scATAC-seq. Different tools, however, use different models to define gene activity scores.

- Signac, a tool for analysing scATAC-seq data, uses open chromatin signals in the gene body and the 2-kb upstream region of a gene to calculate the gene activity score. Unfortunately, this assumes regulatory elements of a gene will always be near the gene, which is known not to be the case [53].
- On the other hand, Cicero uses only promoter regions, leading to an incomplete definition of gene activity score and ignoring informative signals in and around the gene [50].

- Since there is no well-characterised method to infer gene expression from scATAC-seq by calculating gene score, ArchR tested 56 different gene scoring methods to try to optimise this problem empirically. Based on the benchmarking results against ground truth gene expression, the best method to describe gene activity scores is model 42. It takes into account an extended gene body, the boundaries of other genes and the bi-directional exponential decay from TSS to TTS. This is the most comprehensive gene activity model to infer gene expression information from scATAC-seq data so far [40].

Table 1.1 A complete list of analysis tools for the analysis of scATAC-seq data
PP, pre-processing; DR, dimensionality reduction; PL, programming language

Tools	PP	Normalize	DR	Clustering	Purpose	Input
PeakVI	N	VAE with Bernoulli distribution and auxiliary neural network			DR	Matrix
scOpen	N	TF-IDF	NMF	-	Imputation	Matrix
Signac	Y	TF-IDF	LSI	Graph-based	End-to-end	FastQ
EpiScanpy	N	Binarize, LS norm	PCA	Louvain	End-to-end	FastQ, BAM or count file
SnapATAC	Y	Binarize, Jaccard similarity, regression-based norm	Eigenvector decomposition	Louvain	End-to-end	FastQ
ArchR	N	TF-IDF	iteLSI	Seurat	Downstream	BAM or fragment file
MAESTRO	Y	TF-IDF	LSI, cisTopic	Seurat	End-to-end	FastQ
scATAC-pro	Y	Binarize/TF-IDF	PCA/LSI/LDA	Louvain	End-to-end	FastQ
SCALE	N	VAE with the Gaussian Mixture model		K-means	DR	Matrix
Destin	Y	-	Weighted PCA	K-means	End-to-end	FastQ
cisTopic	N	Binarize	LDA	Density	DR	Matrix
Scasat	Y	Binarize, Jaccard dissimilarity	MDS	K-medoids	End-to-end	FastQ
Cicero	N	Binarize	- can use LSI from Monocle3 [57]		Trajectory, coaccesibility	Matrix
Cusanovich2018	Y	TF-IDF	LSI	Hierarchical	Workflow	FastQ
BROCKMAN	Y	Standardize, z-scores	PCA	Known cell type markers	End-to-end	FastQ
scABC	N	-	-	Iterative weighted K-medoids	Clustering	BAM and peak files
chromVar	N	Coverage bias correction and z-scores	-	Hierarchical based on correlation	Motif characterisation	BAM, peak files and motif PWMs
Dr.seq2	Y	-	PCA/t-SNE/ SIMLR	Hierarchical	End-to-end QC focused	FastQ
SCRAT	Y	LS and region length	PCA/t-SNE	Hierarchical/ k-means/ model-based/ DBSCAN	Web tool	BAM file

1.9 Thesis aims

The overall aim of this thesis is to use the novel capabilities of scATAC-seq to help functionally prioritise both the causal variants and cell types associated with common disease. While common disease has been the focus of my research its outputs are, in principle, applicable to any regulatory variant and have general applications to genome biology; in particular gene regulation.

A further goal was to develop a computational platform to break down the considerable barriers to the general use of these complex data types and computational analyses. This platform is intended to be an end-to-end solution to these analyses aimed at both experimental and computational scientists. A major feature of this platform has been the development of dynamic user interfaces to allow for human interaction with the analyses and multidimensional outputs of the platform.

The results of this thesis are laid out in three results chapters. The first results chapter (Chapter 2) deals with understanding the information content of scATAC-seq as a datatype, its comparison with the more established Bulk ATAC-seq methods as well as testing and validation of the appropriate methods for its analysis. The second results chapter (Chapter 3) deals with the development and validation of a method to increase the resolution of scATAC-seq to identify Transcription Factor bound regions in the genome and its potential application to GWAS analysis. The third chapter (Chapter 4) deals with the development of the end-to-end analytical platform (Avocato)

and JavaScript visualisation platforms (MDV and MLV) for the analysis of scATAC-seq data and its use to prioritise causal cell types and causal variants at high-resolution.

2 Understanding the technology for measuring chromatin accessibility at a single-cell level

2.1 Introduction

As explained in detail in Chapter 1, single-cell ATAC-seq technology is a powerful technique that has already improved our ability to decode non-coding genetics. In particular, it is able to determine the epigenetic landscapes of multiple primary cell types and stages within a heterogeneous population that would individually be difficult to isolate or are underrepresented, such as stages of differentiation of a given lineage [7]. Single-cell ATAC-seq assays profile comprehensive epigenomic landscapes of many cell types, resolving cellular heterogeneity and identifying rare cell types. By resolving a heterogeneous population in this way, it requires much less technical effort and money to resolve multiple cell populations than would be required with standard Bulk ATAC. Importantly, such maps will allow for the identification of which variants in a haplotype intersect with regulatory elements and in which cell types [1], [16].

Nevertheless, scATAC-seq is a very new technology and several questions need to be answered to understand its capabilities compared to the established Bulk ATAC-seq methods. These questions are as follows:

- One of the most important questions to be addressed first is whether the chromatin accessibility profiles from the different technologies are equally

interpretable. Can scATAC-seq data recapitulate the same signal structure as Bulk ATAC-seq data?

- If this is the case, then how many single cells are required to get quality, interpretable data from the scATAC-seq assay? This is because the final epigenetic profile is generated by Pseudobulking (the aggregation of the data from all cells in an assigned cluster). Therefore, signal strength will vary with the number of cells from a given cell type in a heterogeneous mixture [23].
- How biologically reliable is the current analysis of scATAC-seq and how far can we trust the analysis outputs? The interpretation of scATAC-seq completely depends on the ability to computationally identify different cell types within the heterogeneous input. This question has multiple aspects.
    - How well do clustering algorithms perform in identifying clusters and how many cells are needed to form a cluster?
    - Once clusters are formed, the cell types need to be identified. How reliable are current methods for doing this?

Over the last three years, computational development in the analysis of single-cell epigenetics has been very active, and there are now more than 15 specialist computational tools for its analysis (Table 1). However, they use different computational approaches with different assumptions and limitations. Hence, they need testing to see which ones really overcome the technical challenges in scATAC-seq regarding sparsity and high data dimensionality. Therefore, in this chapter, I will perform comprehensive data analysis on scATAC-seq of PBMC (peripheral blood mononuclear cells) and proerythroblast data using a panel of the most comprehensive

existing analytical tools; this will not only evaluate their performances but will also check each application to ascertain their fitness for interpreting non-coding genetics.

Furthermore, to test the parity between Bulk and scATAC data, in this chapter, I will compare the information content of Bulk ATAC-seq and scATAC-seq data from proerythroblast cells from the same individual. In a single-cell ATAC assay, the number of input cells is typically lower than in the Bulk ATAC method. However, it is unclear what the minimum number of cells required to annotate the genome of a cell cluster effectively is; therefore, I will investigate the effect of cell number on data quality. Ultimately, this chapter will act as general guidance on the single-cell ATAC-seq method from experiment to data analysis and its potential to interpret non-coding genetics.

2.2 Results

2.2.1 Do Bulk ATAC-seq and scATAC-seq provide the same information?

To test the parity of these approaches, an ideal experimental design should include results from Bulk ATAC-seq and scATAC-seq on the same material. In my recent publication [69], we generated such a dataset, which allowed me to compare results from both methods directly. We performed Bulk ATAC and single-cell ATAC sequencing (10X technology) on proerythroblasts derived from ex-vivo differentiated CD34+ haematopoietic stem cells from a normal human donor, with the caveat that for logistical reasons each experiment was formed an independent differentiation. Table 2.1 shows sequencing information for both sequencing experiments.

Bulk ATAC-seq and scATAC-seq data were analysed using custom scripts developed as part of my thesis in parallel with the cellranger-atac pipeline provided by 10X Genomics, respectively. The input cell number for BulkATAC-seq was ~80,000 cells, while for scATAC-seq was ~7,000 nuclei (4,485 cells recovered). As the population was considered to be essentially homogeneous, the aggregate of all the single-cell data was used in comparison with the Bulk ATAC data. After the data analysis, coverage tracks were uploaded to UCSC.

Table 2.1 Sequencing information for Bulk ATAC, scATAC and down-sampled versions of both

	Bulk ATAC-seq	**Bulk ATAC-seq down-sampled**	**scATAC-seq**	**scATAC-seq down-sampled**
Sample information	Don2 Male	Don2 Male	Don2 Male	Don2 Male
Sample type	Proerythroblast	Proerythroblast	Proerythroblast	Proerythroblast
Estimated number of cells	80,000 cells	**4,485 cells**	4,485 cells	4,485 cells
Sequence depth before quality control	42,060,324 reads	2,358,007 reads	231,097,116 reads	~42,060,324 reads
Sequence depth after quality control	17,586,680 reads	985,953 reads	160,957,352 reads	**~17,586,680 reads**
Alignment rate	94.33 %	94.33 %	97.49 %	97.49 %

Figure 2.1 shows chromatin accessibility at the alpha-globin locus in Bulk ATAC-seq (black track) and scATAC-seq (blue track). The orange and purple tracks are a down-sampled version of Bulk ATAC-seq to ~4,485 cells and scATAC-seq to ~17,586,680 reads (~11% of scATAC-seq reads), respectively, to make a fair comparison between the two methods. It is clearly seen in Figure 2.1 that the technological difference did not change the signal architecture. However, data quality and depth visually appeared superior in scATAC-seq over Bulk ATAC-seq in the same number of cells.

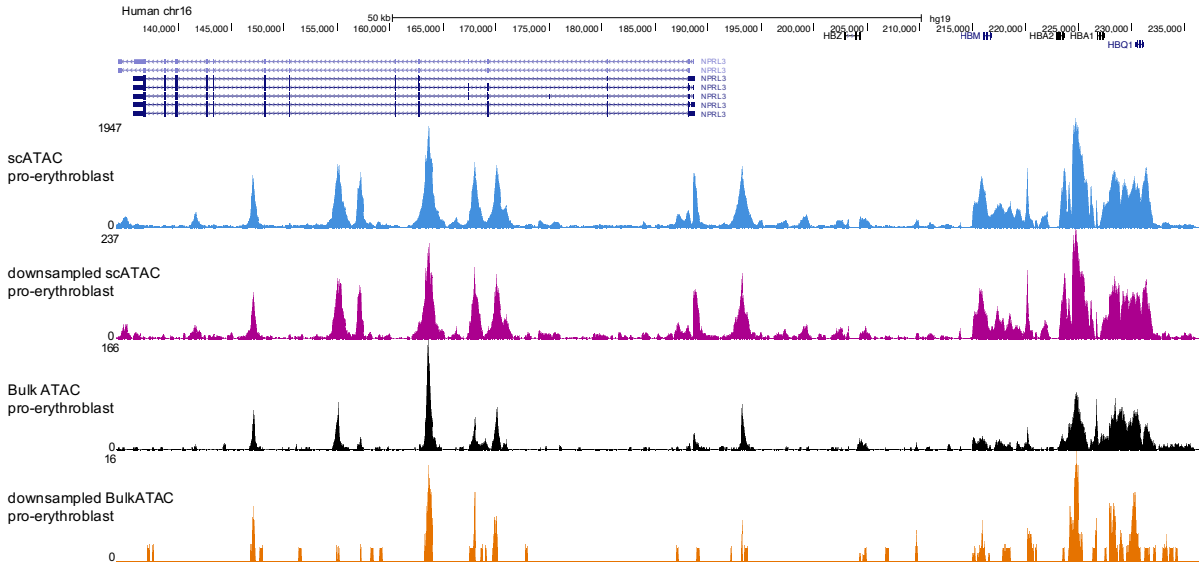


Figure 2.1: The scATAC-seq proerythroblast data shows high data quality than the Bulk ATAC-seq proerythroblast.
Comparison of data quality for proerythroblast dataset in scATAC-seq (blue track), down-sampled scATAC-seq (purple track, down-sampled from 160,957,352 reads to ~17,586,680 reads), Bulk ATAC-seq (black track) and down-sampled Bulk ATAC-seq (orange track, down-sampled from 80,000 cells to 4,485 cells) at the alpha-globin locus.

One of the common-sense checks is to look at the distribution of DNA fragments in both assays (Figure 2.2). As expected, the number of fragments in Bulk data is lower than in single-cell data. However, the distribution pattern of the two datasets appears similar, reflecting nucleosome protection (146 bp plus linker sequences), suggesting transposition behaves the same in both methods.

I then analysed the data at the genome level to compare the number of active regions detected in each dataset. To do this, I applied the Lanceotron peak calling method to both data sets (total aggregated scATAC and Bulk ATAC without downsampling). This shows a sizeable difference in the total number of peaks detected in Bulk compared to single-cell data, 13,727 and 71,266, respectively (using a Lanceotron cut-off of peaks score >= 0.5).

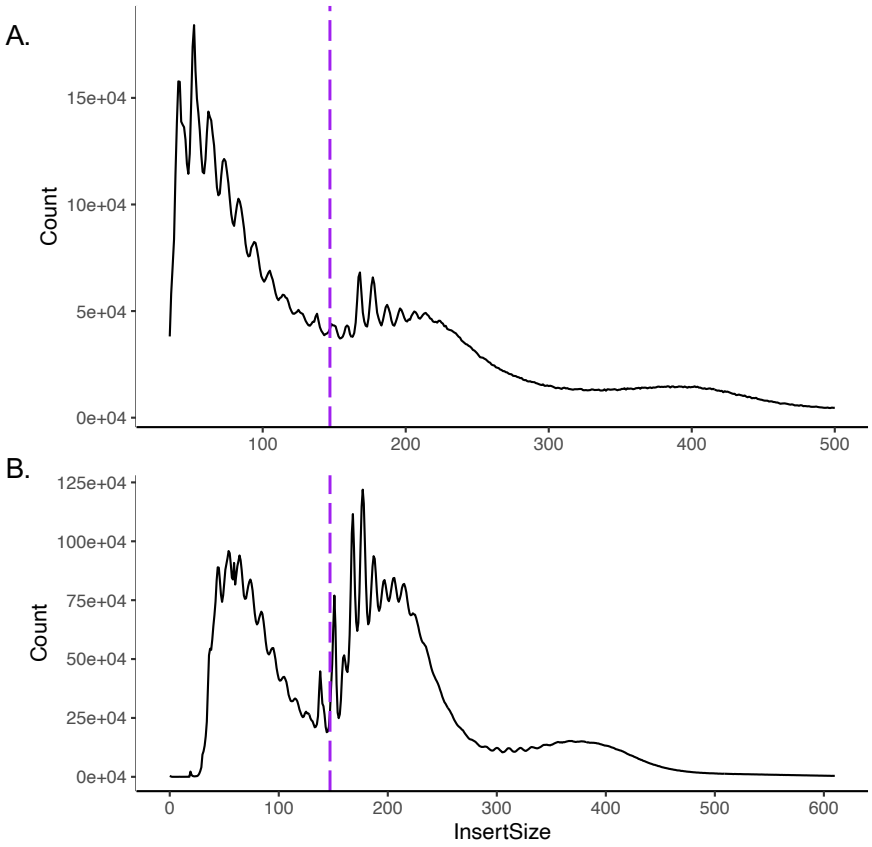


Figure 2.2: Distribution of DNA fragments for the proerythroblast dataset in Bulk ATAC-seq (A) and scATAC-seq (B) shows a similar distribution trend.
The x-axis of distribution plots represents the fragment size of tagmented DNA the y-axis shows a count frequency for those fragments. The purple dotted line refers to an average length of a nucleosome, 150bp. DNA fragments before the purple line are nucleosome-free DNA fragments, while DNA fragments after the purple line, in order, are mono-nucleosomal and di-nucleosomal DNA fragments.

I next annotated both Bulk and scATAC peaks based on genomic locations via an R package called ChIPseeker, (see Figure 2.3) to determine if certain classes of elements, including promoters and distal regulatory elements, were differentially detected by the two approaches. In the distribution of Bulk ATAC peaks, promoter (<=1kb) peaks were most abundant (Figure 2.3.A), while in scATAC peaks, distal intergenic and promoter (<=1kb) peaks account for the most (Figure 2.3.B). I next checked how many scATAC-seq peaks overlapped the Bulk ATAC-seq peaks (see Figure 2.3.C). A total of 71,313 peaks called, 57,586 peaks are only detected by scATAC-seq data, 47 peaks are unique to bulk ATAC-seq data and 13,680 overlap between both data sets. Therefore, almost every Bulk ATAC peak is also found in the scATAC

set. This suggests that single-cell ATAC-seq may simply be more sensitive than Bulk, while the enrichment for promoter distal peaks would suggest that it is more effective in detecting enhancer and/or CTCF sites than Bulk. To confirm this, I used ChIP-seq datasets to determine regulatory element classes: H3K4me1, primed enhancer marker; H3K4me3, promoter marker; H3K27ac, transcriptional activity marker; and CTCF, boundary elements. With these data, I annotated both Bulk and scATAC open chromatin peaks as they are enhancers, promoters or CTCF sites.

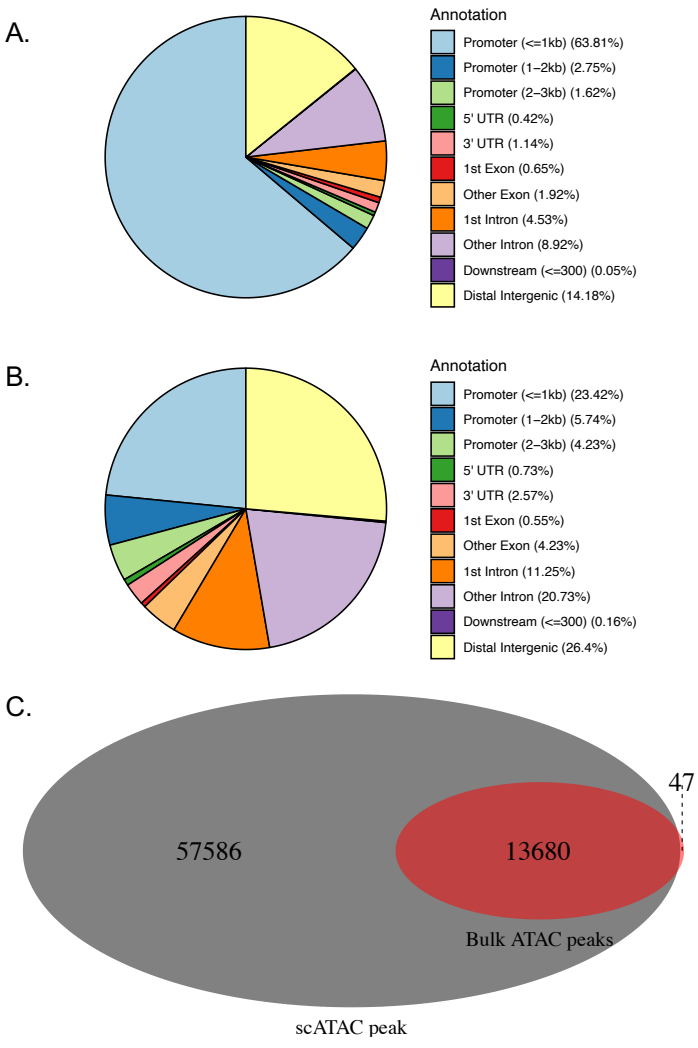


Figure 2.3: scATAC-seq peaks contain more information about distal regulatory elements than Bulk peaks and overlap with almost all peaks of Bulk ATAC-seq.
Genomic location-based peak annotation for A. Bulk ATAC and B. scATAC peaks. C. Difference between Bulk ATAC and scATAC peaks is shown as a Venn diagram representing the total number of peaks in each sample. Related MLV session can be accessed through the link, https://mlv.molbiol.ox.ac.uk/projects/multi_locus_view/6274.

Figure 2.4 shows that only a very small proportion of called peaks in Bulk ATAC-seq are marked as CTCF sites, with the majority of peaks epigenetically marked as either enhancers or promoter classes. However, the same analysis in scATAC peaks (Figure 2.5) shows a very large increase in the detection of CTCF sites and, to a lesser degree, weakly active enhancer-like sites. This shows that Bulk ATAC has a much lower sensitivity than scATAC for the detection of weak open chromatin sites, such as CTCF sites and weakly active enhancers like elements (also see supplementary Figure 1).

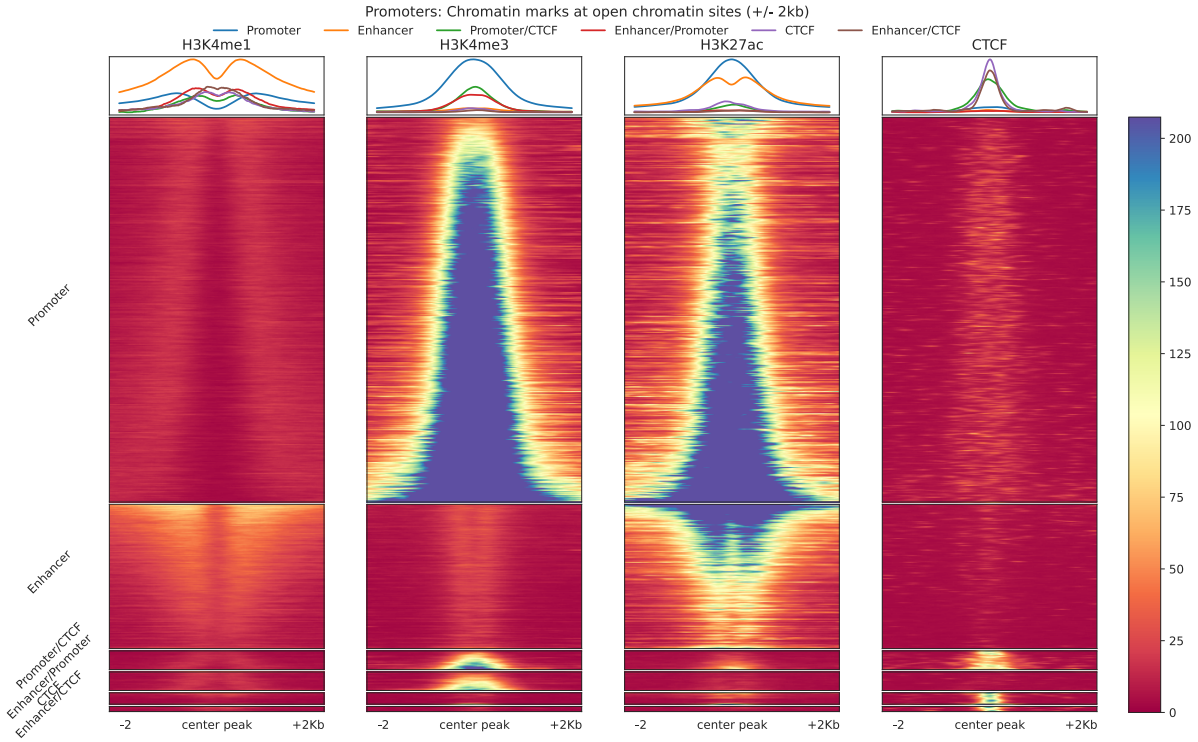


Figure 2.4: Peak annotation result for Bulk ATAC-seq peaks by using in-house ChIP-seq datasets shows Bulk ATAC-seq is not sensitive enough to identify CTCF sites. Peaks in Bulk ATAC-seq were centred and expanded as 2kb from each side. Read count for those fixed regions was acquired from, respectively, H3K4me1, H3K4me3, H3K27ac and CTCF. Then, peaks are categorised based on their read coverage values.

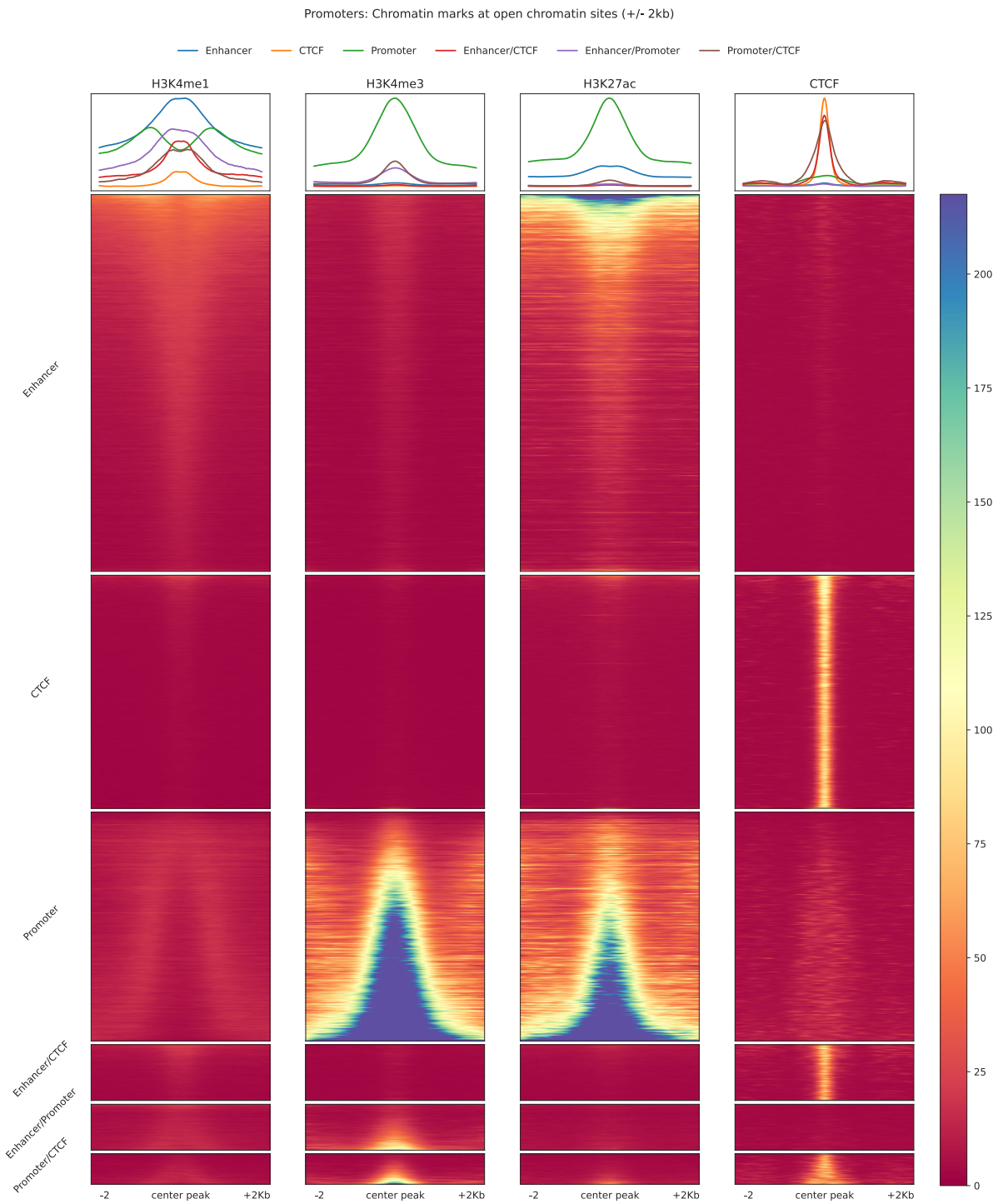


Figure 2.5: Peak annotation result for scATAC-seq peaks by using in-house ChIP-seq datasets shows that scATAC-seq is deep and sensitive enough to be able to detect CTCF sites

Peaks in scATAC-seq were centred and expanded as 2kb from each side. Read count for those fixed regions was acquired from, respectively, H3K4me1, H3K4me3, H3K27ac and CTCF. Then, peaks are categorised based on their read coverage values.

There are also some enhancer and promoter peaks that are unique to scATAC-seq data, which is likely due to minor biological differences between the two independent differentiations. Manual inspection of the small number of peaks unique to Bulk ATAC (N=47) showed them to be due to minor noise in the peak callers' behaviour (Supplementary Figure 2). In summary, the analysis in this section of matched bulked and scATAC-seq showed that, after pseudo-bulking, scATAC-seq data contained similar data but of superior sensitivity to Bulk ATAC-seq and can therefore be used analogously. This, therefore, opens up the potential to use scATAC-seq data to very sensitively annotate the epigenetic landscapes of multiple cell types simultaneously in heterogeneous cell populations.

2.2.2 Analysis of open chromatin in homogeneous cell samples using scATAC-seq

The ability to identify clusters of similar cells within complex heterogeneous populations is a critical but computationally complex step in scATAC analysis. This is a prerequisite for the aggregation of sparse data from each cell to be aggregated into the pertinent cell types to generate an interpretable epigenetic profile.

Bulk ATAC-seq of primary cells depends on the assumption that the method used to isolate them actually yields a homogenous population. In contrast, scATAC-seq can detect *a priori* differences in a presumed homogenous population. This can be clearly seen in my dimensionality reduction and clustering analysis performed as part of Truch *et al.* [69]. The cells are derived from the ex vivo differentiation of CD34+ stem cells. They have historically been considered a homogenous population of proerythroblasts,

where the only expected differences would be related to the cell cycle. However, dimensionality reduction analysis using the cisTopic method and visualisation in 2D space clearly showed two distinct clusters of cells (Figure 2.6). To determine what the potential underlying biological reason might be, I further analysed these two clusters as individual cell types.

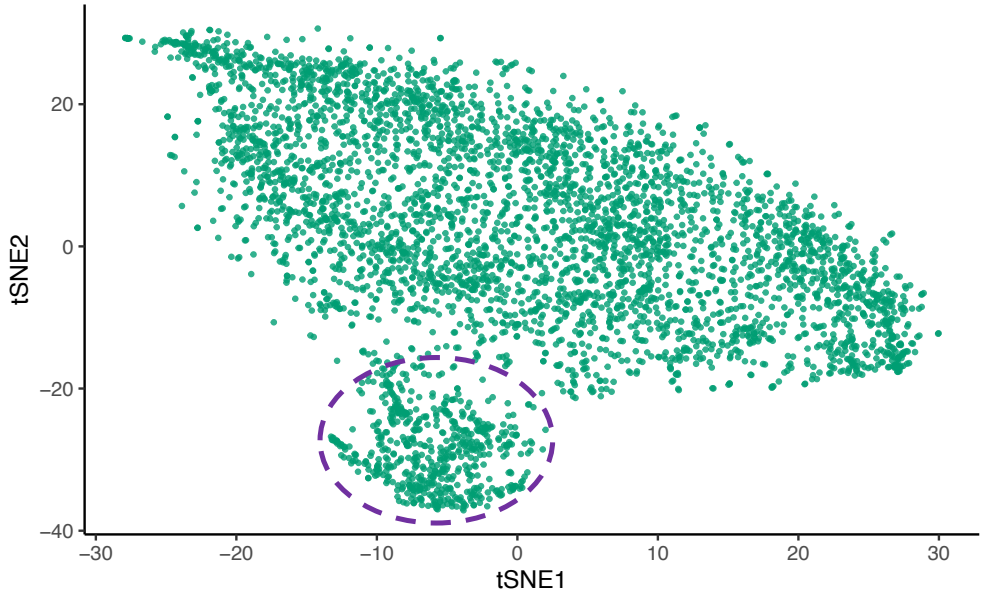


Figure 2.6: Dimensionality reduction analysis for scATAC-seq proerythroblast is shown as a t-SNE projection, each dot representing each individual cell.
Cells marked with a dotted purple circle demonstrate a different epigenomic landscape than the rest of the cells, as explained in section 2.2.2, Figures 2.7 and 2.8.

After pseudo-bulking the large and small clusters, I applied peak calling via MACS2 on each cluster. The small cluster has 614 cells and 8612 peaks, whereas the large cluster has 3871 cells and 64503 peaks. I annotated these peaks using an R package called ChIPseeker. Figure 2.7 shows where those peaks lie within the genome. This analysis showed a profound loss of distal peaks to TSSs in the smaller cluster. I, therefore, performed gene and pathway enrichment analysis (Figure 2.8) using the clusterProfiler R package (p-value < 0.05) to try and find a biological reason why the distribution of peaks in the small cluster might have changed drastically.

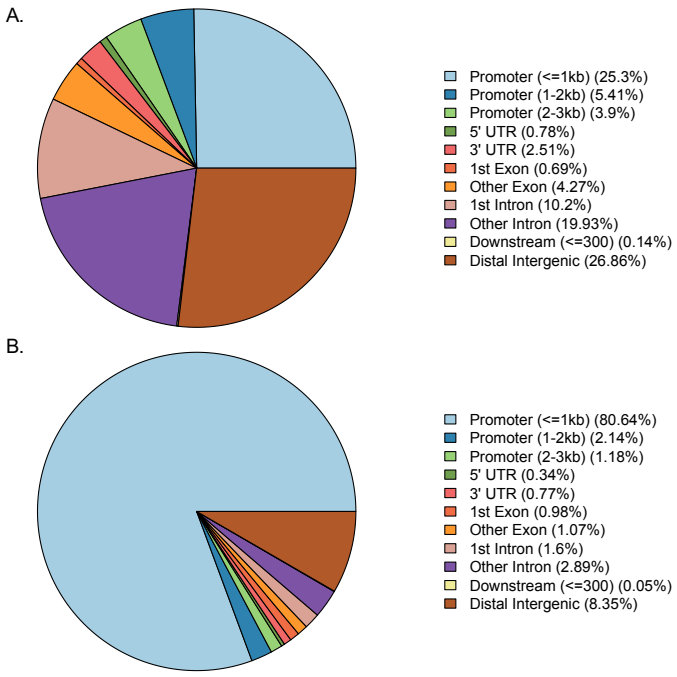


Figure 2.7: Peaks in the small cluster from Figure 2.6 are identified as very close promoters, whereas the activity of distal regulatory elements seems to diminish.
Genomic location-based peak annotation for A. the large cluster and B. the small cluster from Figure 2.6.

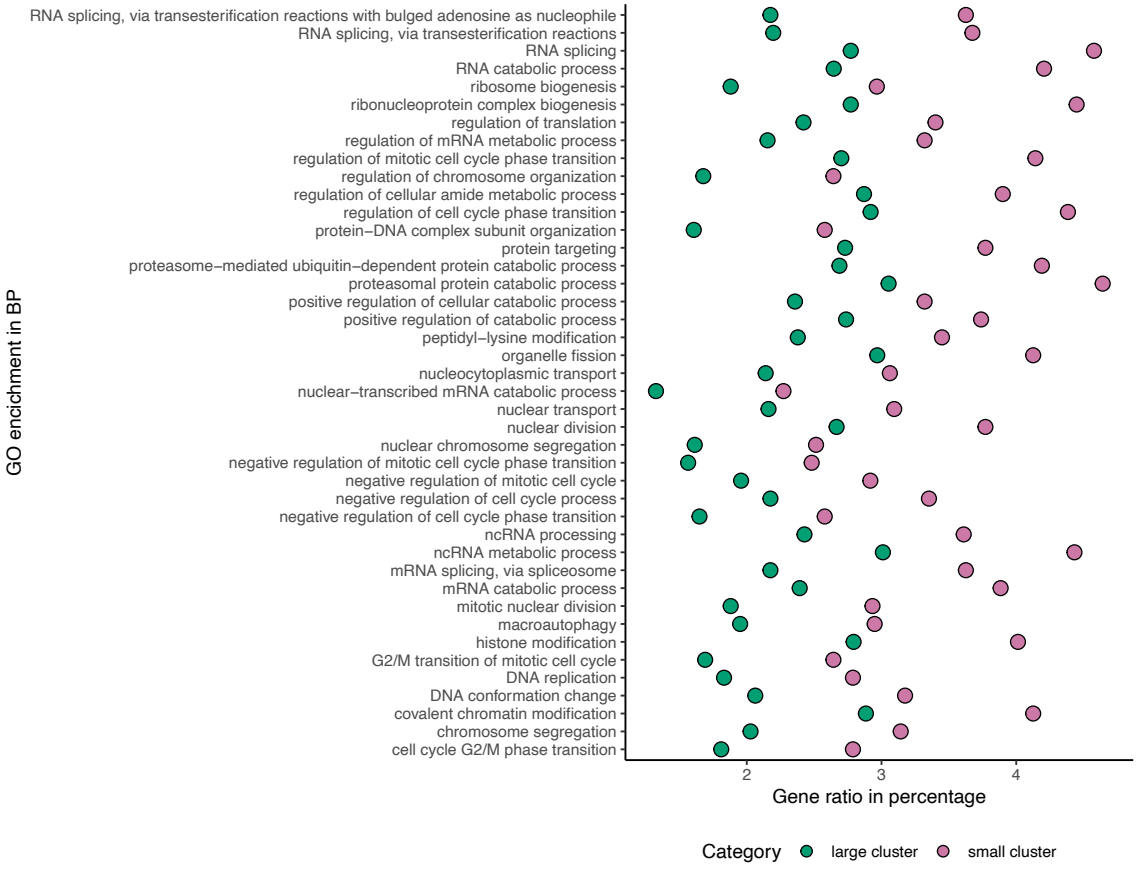


Figure 2.8: Gene enrichment result of the small cluster shows up-regulation in the processes involved in the ejection of the nucleus than of the large cluster.
Gene enrichment results of BP in the small cluster (pink dots) and the large cluster (green dots) are shown as a dot plot. The x-axis shows the gene count ratio as a percentage, whereas the y-axis refers to enriched biological processes that are found by the algorithm after FDR control.

This analysis showed a strong up-regulation of processes such as autophagy, macroautophagy, catabolism and chromatin modification. These are all key processes in pyknosis that precede the expulsion of the nucleus and mitochondria during red cell maturation [70]. Therefore, it strongly suggests that this cluster represents stages of differentiation downstream of the proerythroblasts, which form the bulk of the cell in this sample [71].

To sum up, scATAC-seq not only recapitulates the same signal architecture as Bulk ATAC-seq but also reveals cellular heterogeneity among and within presumed homogenous populations and allows for their separate analysis after pseudo-bulking of the detected clusters.

2.2.3 How many cells are required to get useful genome annotation after pseudo-bulking?

This analysis, however, highlights another key question in scATAC-seq: considering an obvious difference between the large and small clusters is cell number, how many cells are enough to generate interpretable data? This is an important practical question in the design and performance of scATAC experiments. The goal in epigenetics genome annotation is to determine how many cells are required within a cluster to get sufficiently deep, usable data. Therefore, I down-sampled the scATAC-seq data (proerythroblast) from 4,485 cells to 2000, 1000, 500, 100 and 50 cells to see how

aggregated open chromatin annotation scales with cell number. Figure 2.9 shows six different UCSC tracks with varying numbers of cells, starting from the original data with 4,485 cells down to 50 cells. This visual assessment clearly suggests that data quality is relatively robust, down to 500 cells. At 100 cells, although the active regions are clearly discernible, the structure of the data is beginning to degrade. While the extraction of useful data is likely to be still possible, it will become more computationally challenging, and losses in sensitivity would be expected. In addition, Table 2.2 shows the total number of peaks in each downsampled dataset and the number of peaks that overlap the original peaks from 4,485 cells. As the sensitivity diminishes, the number of overlaps between downsampled data and the main data decreases.

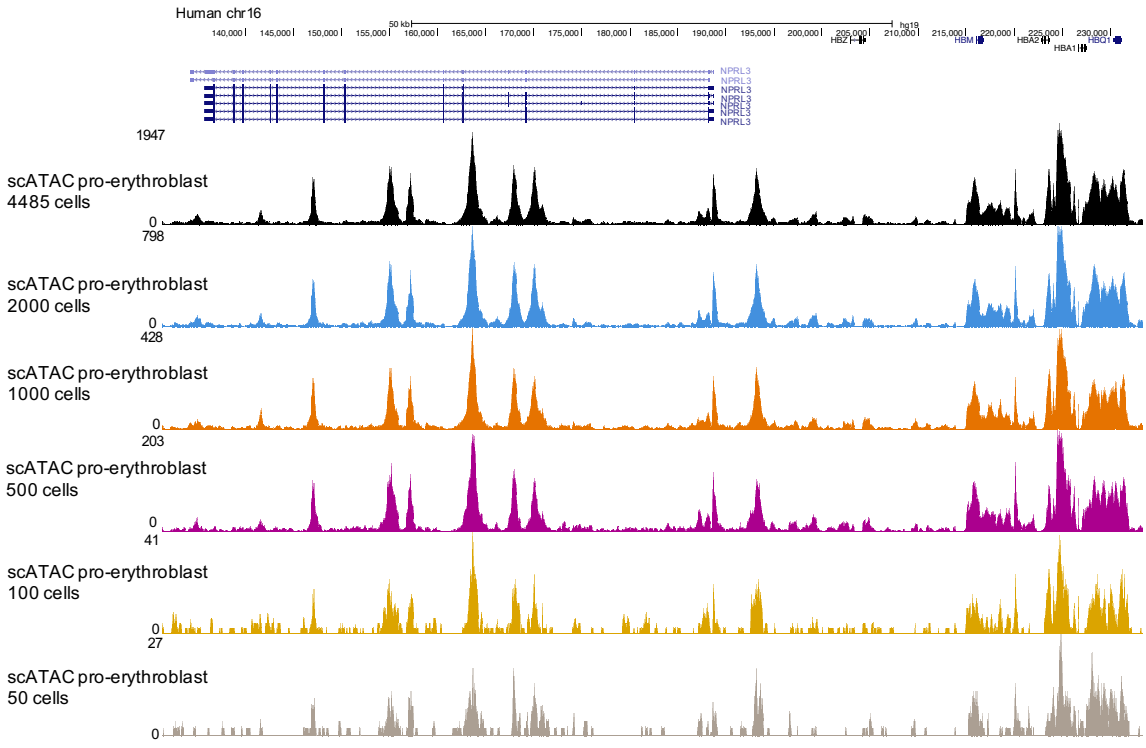


Figure 2.9: Data quality in scATAC-seq proerythroblast data is high, with even 500 cells. Comparison of data quality for scATAC-seq proerythroblast data with respect to cell number at the alpha-globin locus. The black track, 4,485 cells; the blue track, 2,000 cells; the orange track, 1,000 cells; the purple track, 500 cells; the yellow track, 100 cells; the grey track, 50 cells.

To investigate the difference between the original scATAC-seq proerythroblast data and scATAC-seq proerythroblast downsampled data containing 500 cells, I categorised their peaks as promoter, enhancer or CTCF using our in-house chromatin data. This analysis allowed me to observe what class of regulatory elements peaks were lost when sensitivity diminishes. Figure 2.10 shows the annotated peak set in the downsampled data. When the peak annotation results in Figure 2.10 were compared with the result in Figure 2.5, it is seen that the majority of peaks lost in the downsampled peak set are CTCF sites. In the case of reducing the sensitivity of scATAC-seq data, small and sensitive regulatory elements like CTCF are less enriched in the result of the peak calling method.

Table 2.2: The total number of overlapping peaks between scATAC proerythroblast and its downsampled versions

scATAC query data	Number of peaks in the query set	scATAC subject data	Number of peaks in subject set	Number of overlaps
2000 cells	50676 peaks	4485 cells	72065 peaks	50093
1000 cells	41224 peaks	4485 cells	72065 peaks	40631
500 cells	26541 peaks	4485 cells	72065 peaks	26000
100 cells	5164 peaks	4485 cells	72065 peaks	4533
50 cells	3970 peaks	4485 cells	72065 peaks	2224

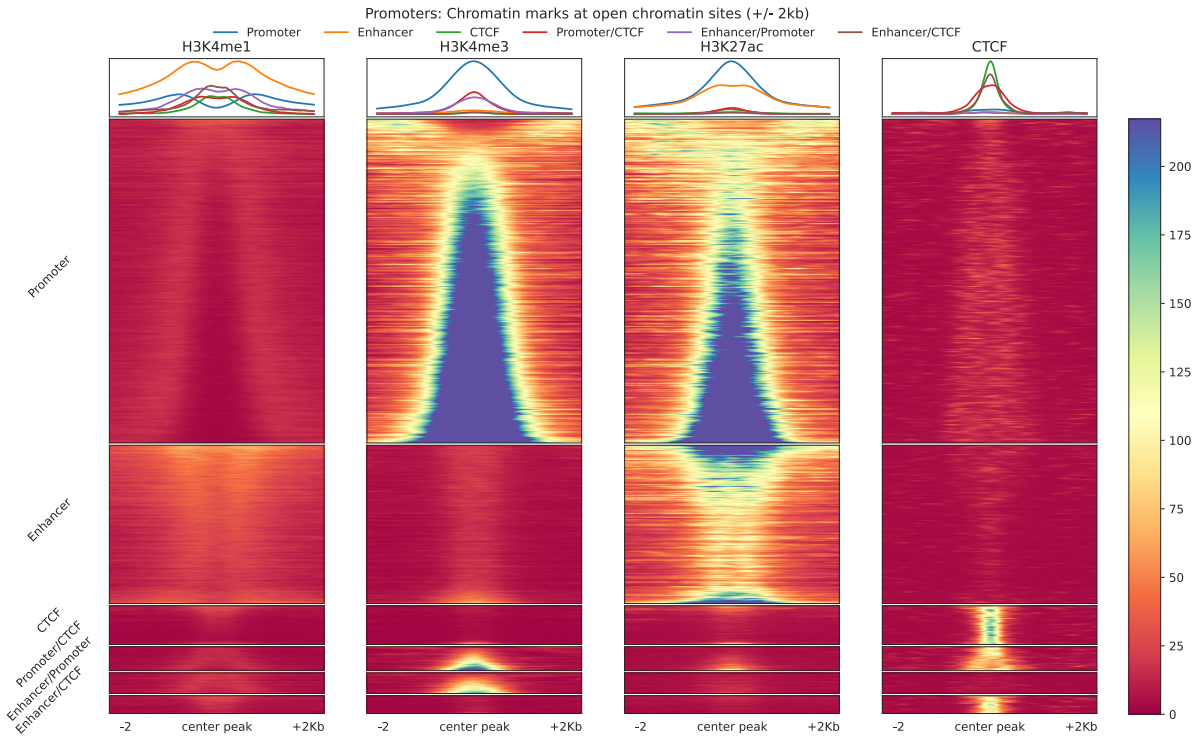


Figure 2.10: Peak annotation result for peaks from scATAC-seq proerythroblast downsampled data containing 500 cells by using in-house ChIP-seq datasets shows that as the number of cells reduces, the sensitivity of scATAC-seq data to identify CTCF sites diminished.
Peaks in scATAC-seq data, containing 500 cells, were centred and expanded as 2kb from each side. Read count for those fixed regions was acquired from, respectively, H3K4me1, H3K4me3, H3K27ac and CTCF. Then, peaks are categorised based on their read coverage values.

Table 2.3: The total number of overlapping peaks between scATAC PBMC and its downsampled versions

scATAC query data	Number of peaks in query set	scATAC subject data	Number of peaks in subject set	Number of overlaps
4485 cells	84391 peaks	~10000 cells	87555 peaks	79368
2000 cells	65866 peaks	~10000 cells	87555 peaks	64674
1000 cells	51185 peaks	~10000 cells	87555 peaks	50476
500 cells	31495 peaks	~10000 cells	87555 peaks	31066
100 cells	11448 peaks	~10000 cells	87555 peaks	10427
50 cells	6479 peaks	~10000 cells	87555 peaks	4943

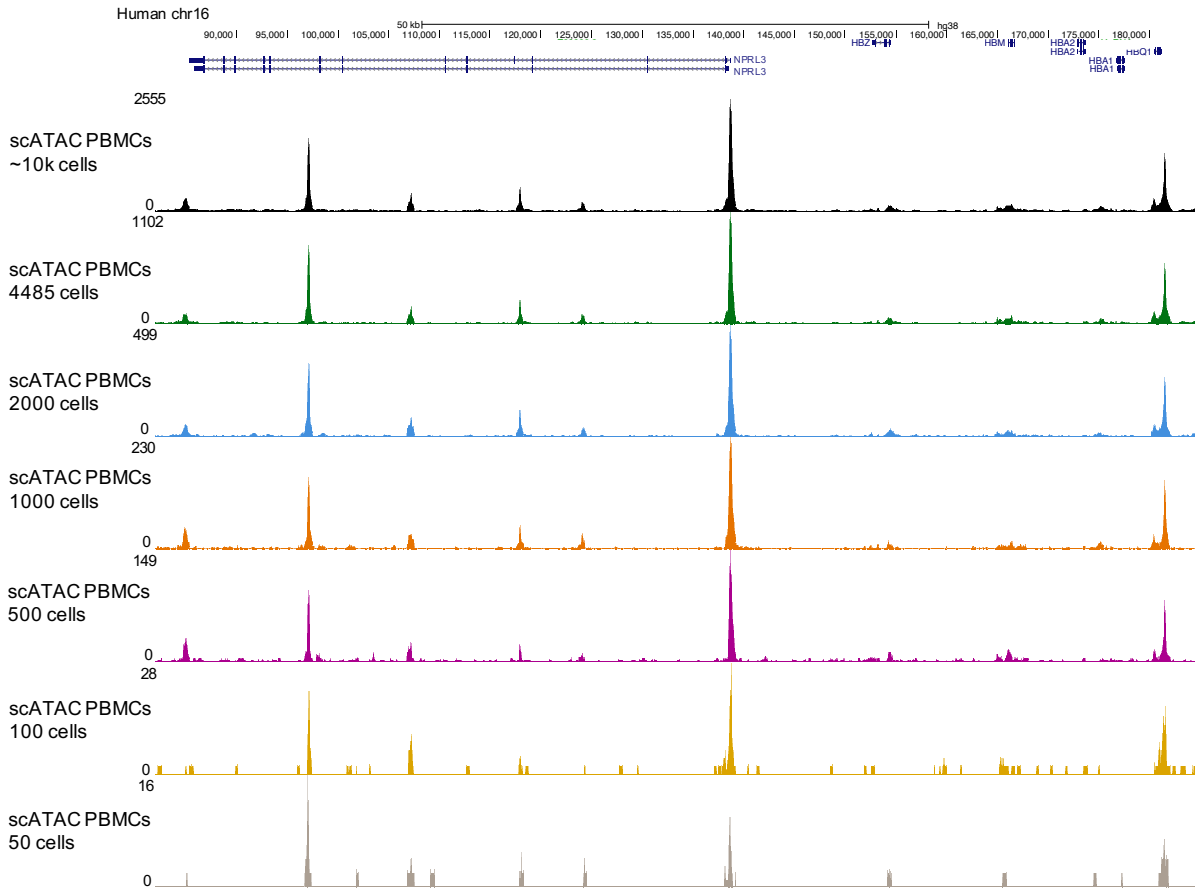


Figure 2.11: Data quality in scATAC-seq PBMC dataset is high, with even 500 cells. Comparison of data quality for scATAC-seq proerythroblast data with respect to cell number at the alpha-globin locus. The black track, ~10,000 cells; the green track, 4,485 cells; the blue track, 2,000 cells; the orange track, 1,000 cells; the purple track, 500 cells; the yellow track, 100 cells; the grey track, 50 cells.

This general robustness down to ~100 cells was confirmed using downloaded publicly available 10X scATAC-seq of aggregated total PBMCs (Figure 2.11). The same low sensitivity pattern applies to the number of peaks as data shrinks (Table 2.3). The same peak annotation cannot be performed in the PBMC dataset, as the data is very heterogeneous, but the general relationship between cell number and signal is similar.

2.2.4 How many cells are required to form clusters robustly?

In this analysis of cell number versus aggregated data quality, I have used the total aggregated data. However, in practice, we would analyse data from specific clusters identified within the data. It is, therefore, important to know not only that a cluster of a given number of cells will yield usable data but also that a cluster of this size can be robustly identified during the dimensionality and clustering steps to allow its subsequent aggregation and analysis. I, therefore, investigated the effect of the total number of cells on the formation of distinct clusters within the data to observe at which level I would start losing the clustering structure. To do this, I chose to spike in data from cells of a highly defined cell population (proerythroblasts) into data from a highly heterogeneous population (PBMCs) at different levels. Both data were derived from the 10X platform, and proerythroblasts do not already exist in PBMC isolates. The general idea is to see the lowest number of proerythroblasts cells required to form a distinct and identifiable cluster in the context of a highly complex sample. I chose to extract cell data from the large cluster shown in Figure 2.6 to represent homogenous proerythroblasts and then spiked in 300, 150, 80, 40 or 20 cells into the PBMC data. Each spike-in dataset then went through the same analytical process with the same parameters. It was important in these analyses to be able to label the cellular identity of the clusters robustly so I could distinguish the erythroid cluster from the PBMC clusters and identify the cell types within the PBMC. The erythroid cells are easily identified as they are artificially added and trackable by their cell IDs. To identify likely cell identities within the PBMCs, I took advantage of the fact that both the erythroid and PBMC dataset I used were 10X multiome, so each cell also has matched snRNA in addition to snATAC data. This fact means that cell clusters can be identified using

a publicly available CITE-seq database of PBMCs, which transfers cluster labels based on cell surface markers via scRNA-seq data. I performed this analysis using the Azimuth platform, which labels cells identified via their snRNA profile by comparison with these reference datasets (see section 2.2.6 in this chapter for a fuller explanation of these analyses).

In this analysis, the publicly available 10X multiome PBMC dataset, with snATAC and snRNA, which contains 3,000 cells, was used to reduce computational time. The snATAC component of these multiome datasets was analysed independently using ArchR, and then clusters were annotated via the CITE-seq cell type identification from Azimuth. Monocyte, B cells, CD4 T cells, CD8 naive, CD8 T other, NK and pDC clusters were chosen from the PBMC dataset because they are clustered distinctly. The idea is to avoid ambiguity in the identity of cells within each cluster. Since erythroid cells are not normally found in PBMC isolates and have very distinct epigenetic landscapes, I hypothesised that they would form a unique cluster from the pre-existing cells.

This analysis, shown in Figure 2.12, clearly showed that the added proerythroblast cells indeed formed a distinct cluster and were highly robust in terms of cell number. With 300, 150, 80 and 40 cells, the proerythroblast cells were clustered together and far apart from the others. However, with 20 proerythroblast cells, the clustering algorithm failed to group them all together. The algorithm instead assigned them to multiple clusters (Monocytes, NK cells, CD4 T cells and CD8 naive cells clusters). This *in silico* dilution experiment suggests that clustering methods can detect distinct clusters robustly with even 40 cells. Importantly, 40 cells are much lower than the

number of cells I determined necessary to annotate a cell cluster's epigenome via Pseudo-bulking; therefore, effectively, it would suggest that if a cell cluster has sufficient cell numbers to be effective in this regard, it will also likely have sufficient cell numbers to be identified as a coherent cluster.

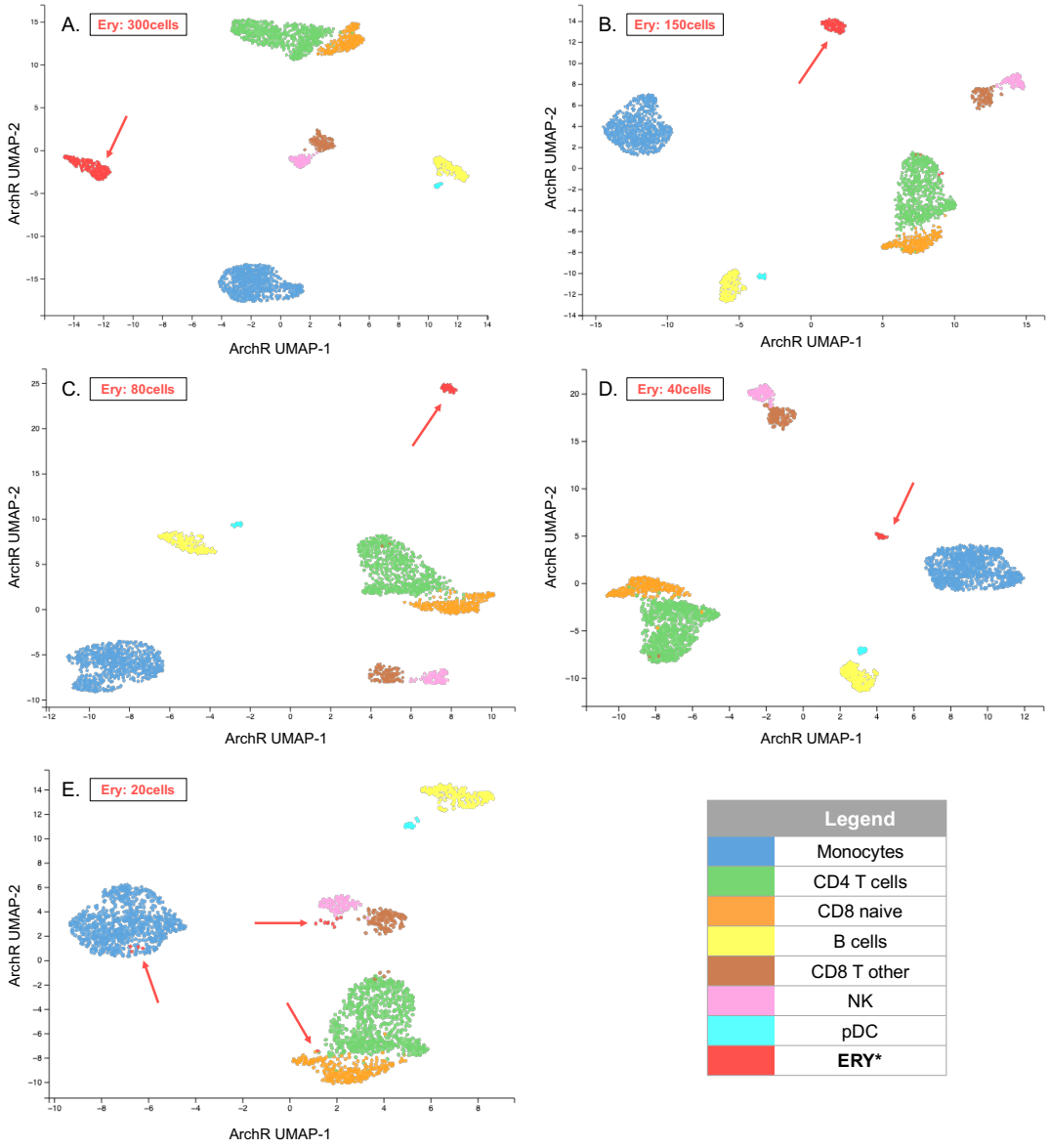


Figure 2.12: In-silico experiment, deciding the optimum number of cells to form a cluster shows that clustering is still possible even with 40 cells.
Each graph displays clustering results on different cell numbers for proerythroblast as UMAP projections using ArchR. Each dot represents an individual cell and is coloured by cell type, as seen in the legend. A, 300 cells; B, 150 cells; C, 80 cells; D, 40 cells; E, 20 cells. Related MLV session can be accessed through the link.
https://mlv.molbiol.ox.ac.uk/projects/sc_atac_seq/5970

2.2.5 Tool comparison

As I explained in the main introduction, there is now an ever-increasing number of analytical tools to analyse scATAC-seq data. They mainly differ in how they define regions in the chromatin landscape to extract data to perform downstream analysis and in which dimensionality reduction techniques and clustering methods are applied. In my work, I have used a published large-scale analysis of the effectiveness of these approaches to guide my initial choices of which to use [17]. In this analysis, CisTopic, SnapATAC and Cusanovich2018 were shown to have the best performance among 10 computational tools for the analysis of scATAC-seq data in the benchmarking study [17]. Furthermore, the MAESTRO study [47] showed that LSI-based dimensionality reduction techniques perform better than other DR methods. When I started my PhD, that benchmarking study was just published, and ArchR had not been developed yet. Therefore, I first started testing cisTopic and SnapATAC tools as they were two of the three best performing tools for analysing single-cell chromatin. The initial version of SnapATAC proved to have many technical problems and was difficult and unreliable to run. These errors were fixed, and tool performance was improved in SnapATAC2, released this year. In this thesis, I initially used cisTopic for my analysis but switched to ArchR upon its release, which has been my main analytical tool for the majority of my thesis. The reason I switched to using ArchR for analysing scATAC-seq data is explained in detail in section 1.8. ArchR provides robust downstream analysis to get clusters and has the most effective gene activity model to model gene expression from scATAC-seq data. However, for completeness, I also tested the performance of the new SnapATAC2 in comparison with cisTopic and ArchR using our proerythroblast dataset (Figure 2.13) and ArchR and SnapATAC2 on the more complex scATAC

PBMC data (Figure 2.14). Reanalysis of our proerythroblast data using cisTopic and ArchR showed a very consistent pattern of two clusters, which, as previously discussed in our analysis, appear to represent two very different chromatin landscapes relating to the stage of differentiation.

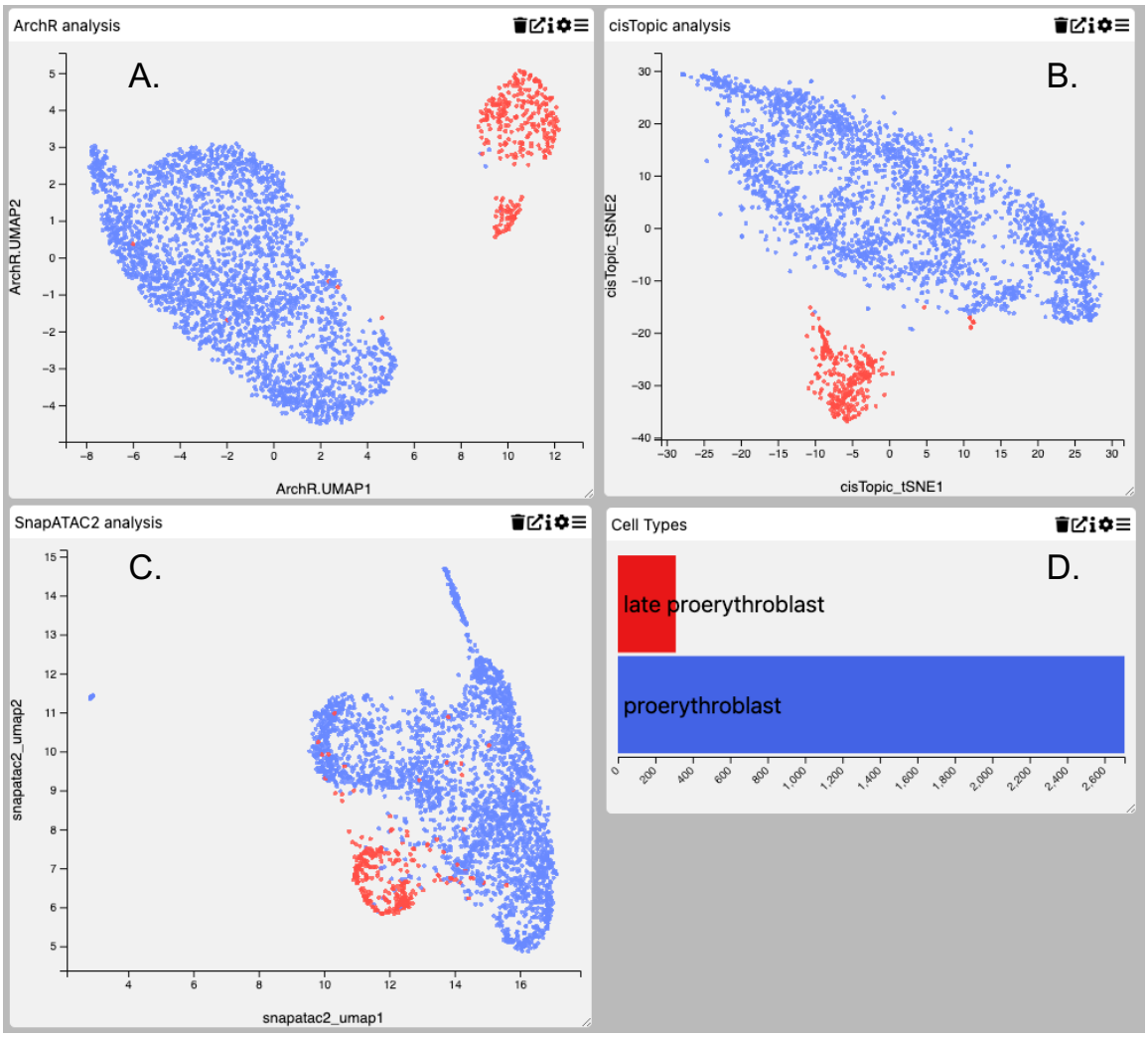


Figure 2.13: Comparative analysis results of scATAC proerythroblast data indicate that although ArchR and cisTopic analysis produced a very similar and clear separation between different cell populations, SnapATAC2 did not have the same clear clustering structure.
Each graph displays different clustering results using ArchR (A), cisTopic (B) and SnapATAC2 (C). Each dot represents an individual cell and is coloured by cell type, as seen in the bar chart on D. Related MLV session can be accessed through the link.
https://mlv.molbiol.ox.ac.uk/projects/sc_atac_seq/5991.

However, it seems clear that these two populations are poorly defined in the SnapATAC2 output. This is possibly due to the fact that this dataset is very atypically homogeneous for a scATAC-seq dataset. I therefore directly compared SnapATAC2 with ArchR, which defined this population most clearly, using the multiome PBMC dataset. This had the advantages of not only being much more heterogeneous but also providing an orthologous approach to labelling the ground truth identity of the clusters using its snRNA data coupled with CITE-seq based Azimuth analysis. Identifying biologically relevant cell clusters is crucial in understanding how well each tool can identify these clusters from an open chromatin-based signal, and Azimuth allows us to annotate the cell clusters using their known immunophenotypes (see section 2.2.6).

Using the CITE-seq-based annotation superimposed onto the clustering generated from the scATAC-seq as guidance, it appears that both approaches generate sensible and coherent clustering. However, SnapATAC2 appears to define many populations more clearly with less mixing and with more separation. This can be seen particularly in the NK, CD8 TEM and CD8 Naïve populations, which appear as distinct and highly separated clusters in the SnapATAC2 output compared to the ArchR output. Overall, in light of these results, there appears not to be a one-fits-all analytical tool for analysing scATAC-seq data. SnapATAC2 may be a better choice for highly heterogeneous samples going forward.

In contrast, ArchR may be a better choice for more homogenous populations, such as the differentiation of a single cell lineage. It does, however, highlight the importance of the use of orthologous data types and approaches to understand the ground truth

composition of samples to test the effectiveness of the specific dimensionality and clustering approaches used prior to analysis.

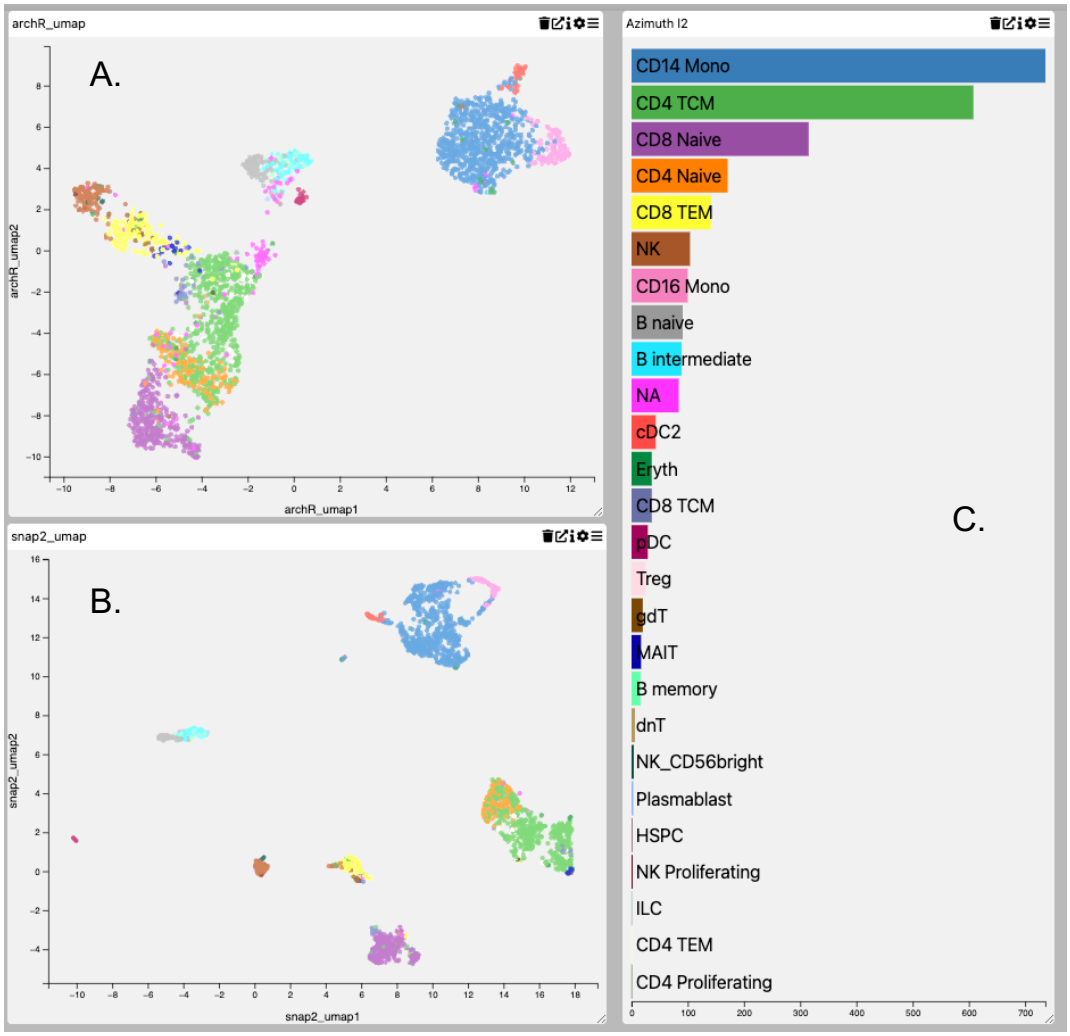


Figure 2.14: Comparative analysis results of scATAC PBMC data indicate that SnapATAC2 produced more clear clustering structure than ArchR.
Each scatter plot displays different clustering results using ArchR (A) and SnapATAC2 (B). Each dot represents an individual cell and is coloured by Azimuth cell types, as seen in the bar chart on D. Related MLV session can be accessed through the link.
(https://mlv.molbiol.ox.ac.uk/projects/sc_atac_seq/5933)

2.2.6 What is the optimum method for annotating cell clusters?

In biology, cell types are typically identified by some defining property, often leveraged to isolate them. In the immune system, this typically relies on the presence of cell surface markers, followed by FACS isolation and functional analysis. This means that

most standard definitions of immune cell identity have clear immune phenotypes that can serve as ground truth as to their identity and provides methods to purify these cell types to determine their transcriptomic and epigenetic signatures. This, in turn, provides some ground truth to determine how well algorithms perform in the challenging task of identifying these cell types via their sparse epigenetic or transcriptomic profiles in complex and heterogeneous cell mixtures. This step is critical to determine how well they can be relied upon in complex disease states such as cancer or poorly defined cellular niches where such ground truth is not readily available. Even today, annotating clusters is still challenging, and there is not one-fits-all approach to do this. scRNA uses genes as markers to annotate clusters and directly assays gene activity in cells. This has a large advantage in that gene annotation provides definable regions in the genome to quantify data to infer gene activity. Gene transcription also largely correlates with protein production, so the activity of ground truth cell surface marks can be inferred directly from gene activity.

However, annotation of cells is even more challenging in scATAC-seq as most of the cell-specific activity relates to distal regulatory elements, which are very difficult to relate to gene activity directly and hence to the pertinent cell surface markers for cell type identification. The CITE-seq assay can profile cell surface markers and gene expression from the same cell. It also provides a direct linkage between immunophenotypes and scRNA-seq profiles to annotate clusters robustly. This has been a real advance in annotating scRNA-seq datasets, but a similar assay does not exist for scATAC. CITE-seq datasets are also limited to specific cellular niches, particularly the immune niche. Therefore, current approaches in scATAC-seq tools

typically depend on building gene activity models from open chromatin profiles to try and infer gene expression and link epigenetic signals with marker genes (Figure 2.15).

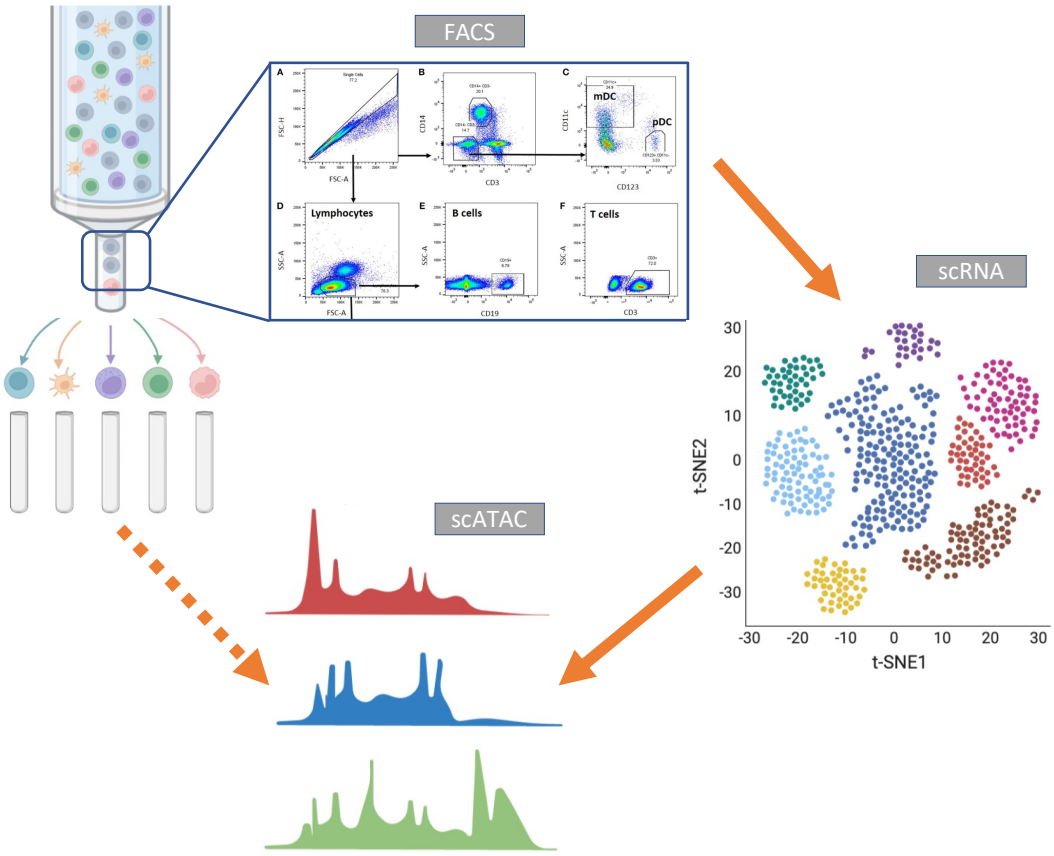


Figure 2.15: The best way to identify cell types is using cell surface markers, which can be directly inferred from scRNA but not scATAC.
There is no direct link between protein information and scATAC-seq (FACS figure was taken from [72]). The rest was created with "BioRender.com".

The absence of any ground truth dataset generated from either transcriptomic or proteomic data to use as cell markers in epigenomic data makes assigning cell identity to clusters in scATAC-seq very challenging. There are a couple of approaches currently used in the field to designate cell types to the clusters. Since epigenomic data profiles the regulatory landscape of a cell population, which fall into the non-coding regions of the genome, usage of gene annotation is not available. However, gene expression can be inferred using some gene activity models in scATAC-seq, as

discussed for ArchR. Another method is to integrate scRNA-seq data with scATAC-seq data, which relies on label transferring but has some disadvantages (see section 1.7 for a detailed explanation). The final way of annotating clusters is having multiome sequencing, where transcriptomic and epigenomic information is obtained from the same nuclei avoiding using label transferring. In the following sections, I will discuss each approach and their comparison in annotating cell clusters. For this comparison analysis, I used PBMC, which contains ~10k cells, a single-cell multiome dataset from 10X Genomics.

2.2.6.1 Using only gene activity scores from scATAC-seq with a list of known marker genes

Standard data analysis protocol in scATAC-seq includes cell type annotation. The most common method to perform this is first to establish a gene activity model for inferring gene expression from scATAC-seq data and then calculate gene activity scores based on the defined model to determine cellular identity via the activity of marker genes. As stated, ArchR has empirically optimised and tested the gene activity model [40], by comparing 56 different gene activity models in two different datasets. They model the inferred gene activity from scATAC-seq data from three aspects of the regulatory landscape. It takes the accessibility signal along the whole gene body. It considers boundaries of genes preventing overlapped chromatin information from neighbouring genes while it takes into account the effect of distance-weighted distal regulatory elements. I followed the ArchR method to analyse only the scATAC-seq data of the multiome PBMCs dataset. Pliner *et al.* [73] provide ground truth marker genes. I used ArchR-calculated gene activity scores to annotate clusters using those gene markers. Figure 2.16 shows the UMAP projection of scATAC-seq cells with cell

type annotation information in Figure 2.16.B. Cell identity assignment was made by taking into account gene activity scores for marker genes (listed in Supplementary Table 1) shown as a heatmap in Figure 2.16.C. In the heatmap, T cell markers (CD3D, CD3E, CD3G) are enriched for clusters 9, 10, 11, 13, 14, 15, 16, 17, and 18, which corresponds to more than half of the data suggesting they are some varieties of T cells (Supplementary Figure 5). However, further manually annotating the T cell family as CD4 TEM cells, CD8 TEM cells, CD4 TCM, CD8 TCM, CD4 Naive, CD8 Naive, Treg cells and MAIT cells based on complex combinations of the marker gene is extremely challenging, especially for a bioinformatician or analyst, and requires in-depth knowledge of immunophenotypes. Therefore, reliable automated methods of cell annotations are desirable and necessary.

Information from different modalities can be used to improve and automate cell identity assignment. Before the advent of multiome technologies, computational integration of scRNA-seq data from the same cellular mixture with scATAC-seq data has been used for the more accurate cluster annotation, which I will discuss in the next section.

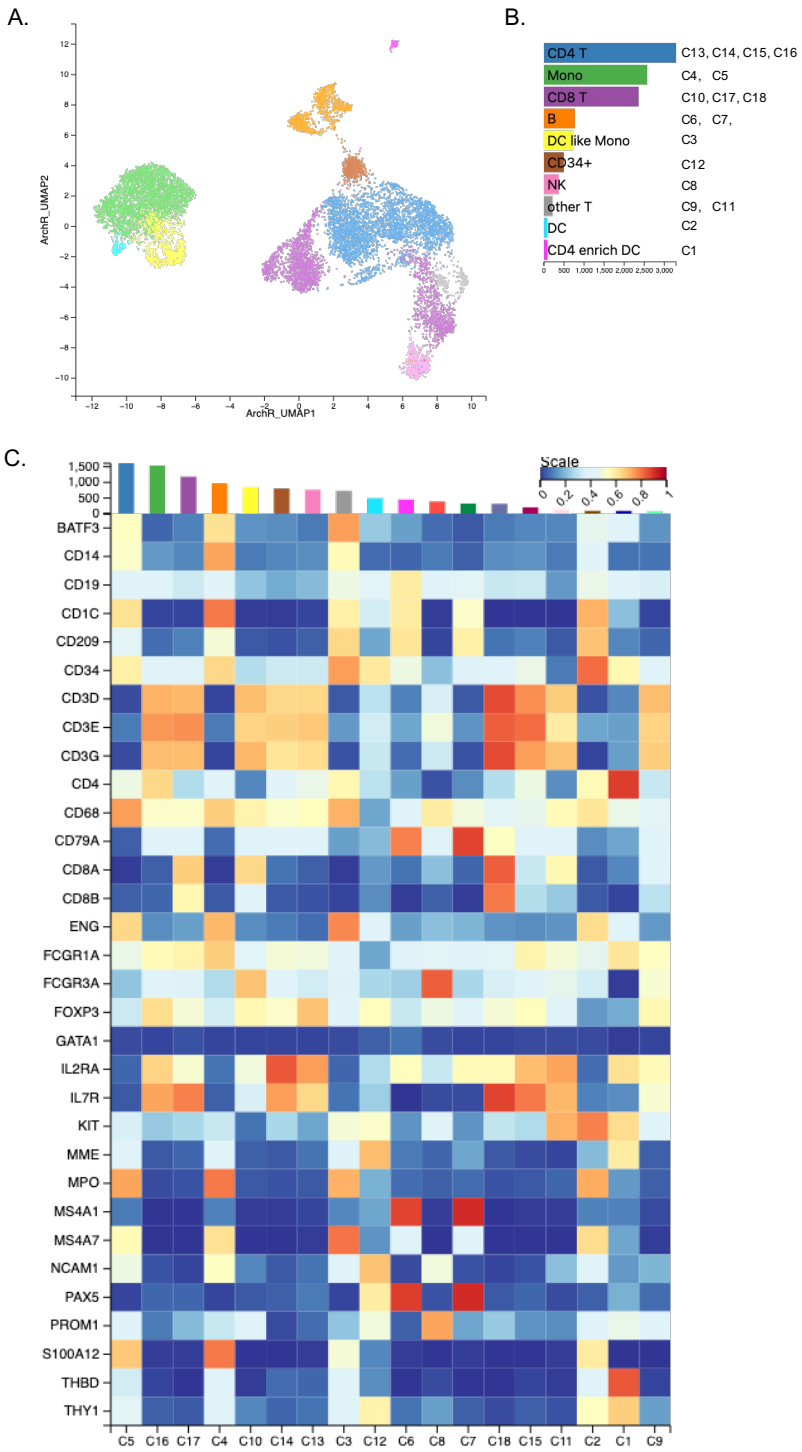


Figure 2.16: Cell-type annotation by using known gene markers is insufficient to assign an identity to clusters properly.
ArchR analysis of scATAC-seq PBMC, followed by cell type annotation using known gene markers from Pliner *et al.* [73], are shown. A. ArchR UMAP projection of single cells. B. Bar chart showing cell types with the number of cells. C. Heatmap of imputed gene activity scores for marker genes. Related MLV session can be accessed through the link.
https://mlv.molbiol.ox.ac.uk/projects/sc_atac_seq/6349

2.2.6.2 Integration of independent scRNA-seq with snATAC-seq for cell type assignment.

For integrating scRNA into scATAC-seq, I analysed publicly available 10X scRNA PBMC data, which contains 10000 cells, using the Seurat package. Comparison of annotation results from using gene activity scores of known marker genes and integration of scRNA data showed this approach helped to annotate clusters more accurately with a more refined clustering (Figure 2.17, panels C and D). These clearly identified subclusters such as pDC, CD16 Mono, CD4 TCM, CD8 Naïve, MAIT and Treg cells.

The integration method is based on transferring labels from one modality to another, and while it is both automated and provides more extensive information, it has drawbacks. When applying label transferring, the method assumes that query and subject datasets have exactly the same number of clusters and that the number of cells within each cluster is balanced. In practice, this can be difficult to achieve in two independent experiments of different modalities, which can cause problems in accurately transferring the labels.

The development of multiome has therefore proved to be a real advance in this regard as both snATAC and snRNA datasets are generated from exactly the same cells and so are inherently balanced.

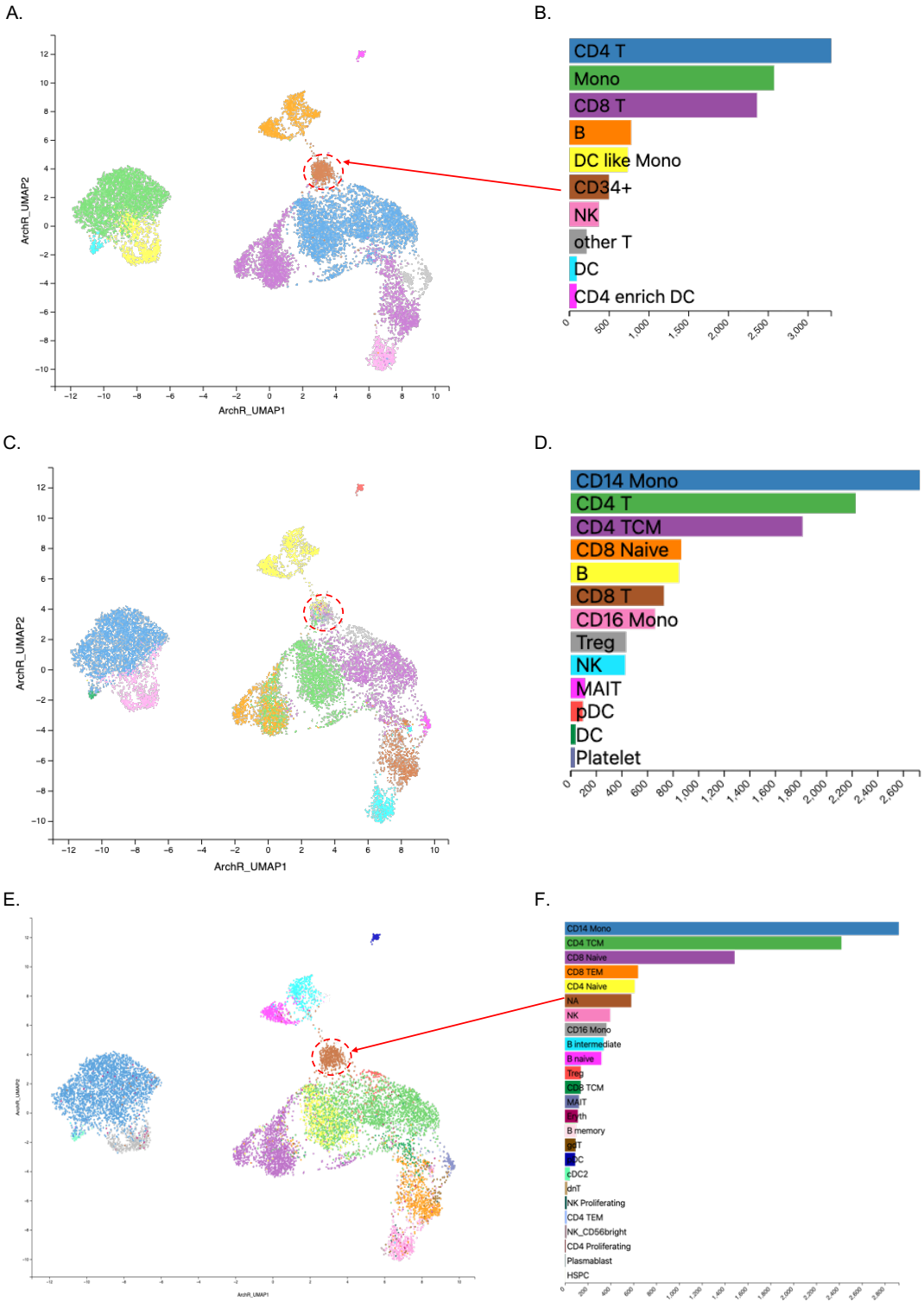


Figure 2.17: Comparative results of different cell type annotation methods on the scATAC-seq PBMC dataset suggest that there is no consensus on which method performs best.
ArchR analysis of scATAC-seq PBMC (A, C and E), followed by cell type annotation using known marker genes (B), integrating independent scRNA (D) and using multiome data (snATAC + snRNA) with Azimuth (F). Related MLV session can be accessed through the link. https://mlv.molbiol.ox.ac.uk/projects/sc_atac_seq/6359

2.2.6.3 Using multiome snATAC and snRNA for cell label transfer to identify cell types.

As previously discussed, the Azimuth annotation approach utilises CITE-seq datasets to provide cell-type ground truth through immunophenotyping combined with scRNA-seq data generation. This holds out the potential to link the ground truth cell annotation with snATAC-seq data of multiome via the integration of the common RNA-seq data generated in both CITE-seq and multiome datasets.

The underlying ground truth annotation uses the protein and transcriptome CITE-seq data to generate a gene expression matrix's cell identity. Combining both modalities to annotate clusters is a powerful method, as they compensate for each other when one fails to separate certain clusters. As also shown in the paper [49], gene expression information alone was unable to separate CD8 T+ and CD4 T+, whereas protein information perfectly distinguished these clusters. Hence protein data can enhance the accuracy of cell type annotation when transcriptomic data fails to do so and vice versa. Similarly, in Figures 2.16E and F., we also see, as they reported in their manuscript, that this approach also effectively separates the highly related CD4 T cells family and CD8 T cells, and B clusters were further stratified as B naive, B intermediate and B memory. Therefore, at least for the immune niche, the CITE-seq data produces a highly robust and granular set of cellular labels, which we can transfer from the RNA-seq signal to the ATAC-seq signal in multiome data (Figure 2.17 panel F).

Of course, the use of reference atlases like Azimuth for annotation is limited to the characterised cell types in a dataset. If a query dataset has a unique cell type that a reference dataset does not have, then the reference dataset cannot assign a cell type

to that particular cluster. Cluster 12 in Figure 2.17 (circled as red) is a perfect example of this limitation. Azimuth designated this cluster as N/A since it does not have such a cell type in its reference PBMC data. The same limitation applies to the scRNA-seq integration method. We can see that for cluster 12, the integration method also could not find any appropriate cluster type. Similarly, cluster 12 was labelled as a mixture of cell types when the integration method was used as an annotation approach since scRNA-seq data also does not seem to have that cell type. Therefore, this approach, while powerful for certainly cellular niches, cannot be used in many other cellular niches and also in situations such as cancer where novel malignant cell types exist.

To summarise, there is not a well-characterised method to annotate scATAC-seq clusters. Using only known marker genes over gene activity scores is insufficient for cluster annotation. Because first, the gene activity score from scATAC-seq is a proxy of gene expression, which is not suitable for scATAC as the data structure is entirely different for the two modalities. Secondly, using marker genes to identify clusters is very challenging if one is not an immunologist. Next, scRNA integration relies on label transferring, for which accuracy can be low when the total number of cells per cluster is unbalanced and in the two modalities [74]. Finally, having a multiome sequencing of snATAC and scRNA is the best option the field has at the moment. Since two assays have exactly the same cells, we can avoid label transferring. One caveat here is that the snRNA modality is nucleus RNA. Compared to the latest scRNA protocols, snRNA is heavily 3' biased (Figure 2.18), and because it is nuclear has a more nascent profile as the stable cytoplasmic pool is excluded. I recommend combining three annotation approaches for a robust cell identity assessment if possible.

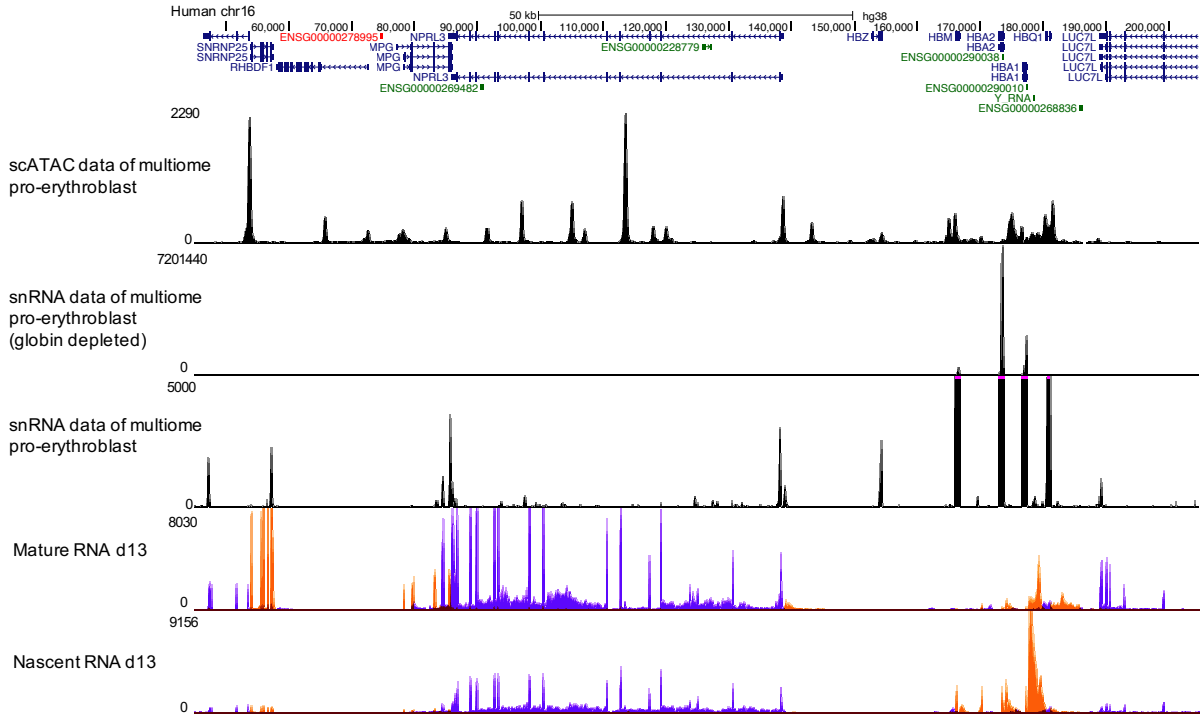


Figure 2.18: UCSC tracks for proerythroblast show that snRNA, nuclear RNA, is 3' biased compared to scRNA at the alpha-globin locus.
The top track belongs to snATAC of multiome. The second and third tracks show globin-depleted snRNA and scaled snRNA of multiome. The fourth and fifth tracks are mature Bulk RNA and nascent Bulk RNA at day 13 of erythropoiesis.

2.3 Discussion

Since Bulk ATAC only provides an average measurement of chromatin accessibility profiles, the emergence of single-cell ATAC technology has had a large impact on the number and purity with which the regulatory landscape of cell types can be mapped. The scATAC-seq assay can provide cell type-specific chromatin landscapes at the single-cell level in a heterogeneous cell population, including at the organ level [75]. Importantly, my analysis showed that upon pseudo-bulking signal architecture technology does not change between scATAC-seq and Bulk and, therefore, can be used in the same way. In fact, it was clear that, for at least the 10x platform, scATAC can provide deeper data resolution and can better detect very sensitive chromatin

accessibility patterns such as those associated with CTCF binding. In combination with the new and effective approaches to cluster cells with the same chromatin accessibility profiles, these approaches can now produce highly detailed and cell-type specific regulatory maps from heterogeneous mixtures of cells.

As in my analysis of proerythroblast data, scATAC-seq can also find hidden cell populations within cell isolates previously assumed to be homogenous and even detect changes associated with cell cycle stages.

Another important finding of this chapter is that the scATAC-seq assay does not require an excessive number of cells to produce informative and useful regulatory data. My analysis shows that a scATAC cluster with 500 cells still provides good data quality. Data quality starts to fade when there are 100 cells in scATAC data, and this finding is consistent across datasets with different levels of complexity. Importantly, current clustering approaches can effectively cluster cells with fewer cells than this, ~40, so this step does not limit its use. This was a real concern and technical challenge since small cell populations contribute the least amount to aggregated scATAC-seq data, and larger cell populations can hide their signals (see Supplementary Figures 3 and 4 as examples). A tiny signal in a particular location of the genome can be an indicator of a rare cell type or noise. This result is, therefore, crucial as one of the important goals for scATAC-seq is to identify rare cell type populations like stem cells or progenitor cells.

Data quality relies on multiple factors, including the reproducibility of the methodology, throughput in terms of cells, sequencing depth, sample complexity, and technique

used to capture single cells. While other methods, such as plate-based or combinatorial indexing based, also exist, I have concentrated on the 10X Genomics platform. This is due to the larger throughput in cell numbers, as our ultimate goal was to generate regulatory genome annotation rather than just cluster information. In light of the analyses in this chapter, my host laboratory has decided to invest in the 10X platform for its future work, in addition to the added benefits of the availability of reproducible reagents, technical platforms as well as high levels of both technical and computational support.

Accurate and automated annotation of cell clusters is an ongoing challenge in single-cell epigenetics. This is particularly challenging in poorly characterised cellular niches and in complex models such as cellular differentiation and cancer. However, the main target for the genetics that my lab focuses on is likely active in the immune and erythroid niches, which have been well characterised. In my work, I have shown that the use of multiome datasets combined with annotation platforms, such as Azimuth, can transfer cellular labels determined by CITE-seq immunophenotyping to scATAC-seq data via the shared RNA-seq data generated by CITE-seq and multiome. It is hoped that the scope of such a reference-based database will only increase. At the time of writing, Azimuth now contains reference datasets for lung, motor cortex, pancreas, fetal development, kidney, bone marrow, tonsil, adipose and heart tissue, as well as a first mouse reference dataset of the motor cortex.

Few tools existed to analyse scATAC-seq data when I started my DPhil, and I initially set out to develop novel approaches and pipelines for its analysis. However, very rapidly, multiple approaches were developed and benchmarked against each other.

From my testing of these computational approaches, I decided to proceed with the ArchR package as the current best end-to-end package for scATAC-seq analysis. However, I have also shown that the recently released SnapATAC2 may also be usefully employed and even beneficial in certain circumstances, and I will continue to employ both in parallel in the future.

The rapid development of these approaches has allowed me to refocus away from developing a basic scATAC-seq platform and the ultimate goal of my work was to confirm the nature and utility of scATAC-seq data to annotated cell-type specific regulatory landscapes. This would then allow the development of a platform to use these data to functionally prioritise regulatory variants in common human diseases, which I will describe in the following chapters.

While I ultimately decided to abandon my development of a basic analytical platform at the end of my first year, I realised that the central observation that was to be the core of this approach could also be used to increase the resolution of scATAC-seq, better defining the regions bound by transcription factors. Accurately determining which variants or SNPs lie within TF-bound regions is obviously a critical step in functionally fine-mapping potentially causal variants, so I decided to investigate further and confirm this observation which I will lay out in the following chapter.

3 Computationally increasing the resolution of scATAC-seq data to define better transcription factor-bound regions.

3.1 Introduction

As is explained in Chapter 1, a challenging data analysis step in scATAC-seq is defining where to informatively extract open chromatin signal, to drive the dimensionality reduction and clustering processes in the absence of a guiding annotation, such as gene annotation in scRNA-seq analysis [40]. Initially, the commonly used approach was to aggregate the data from all single cells as a pseudo-Bulk and then call peaks on the aggregate data to define regions to extract data from each cell. This approach has the potential to lose cell type-specific signals from rare but important cell type populations as the signal from these cells is swamped in the aggregate by the background of the more abundant cells so they can be missed in the peak calling step [22]. This means regions important for the definition of these small populations are left out of the DR and clustering processes. During my DPhil, this was superseded by the computationally demanding but less skewed approach of extracting data from the entire genome using a fixed bin size. Regulatory elements are generally 300 - 500 bp long; specifically, therefore, bin size should be small enough to avoid merging different regulatory elements, and this level of resolution was achieved by ArchR [40]. As I have shown in my analysis in Chapter 2, this approach can cluster cell populations of around 40 cells in a heterogeneous population and is a very effective solution to this problem.

Another potential approach, which was the path I initially started on in my DPhil, was to find a way to decrease the general background in the aggregated data to allow for the more effective identification of informative regions from the pseudo-Bulk.

I, therefore, started looking for parts of the ATAC signal that would define accessible regions with minimum noise and maximum sensitivity to meet this deficit. I first started to consider how single-cell epigenomics technology works, specifically in terms of how Tn5 behaves when it cuts chromatin. As is seen in Figure 3.1, Tn5 is engineered to have as little sequence specificity as possible. Its cutting DNA pattern is, therefore, mostly determined by accessibility and chromatin structure, which is why it works as a chromatin accessibility assay. It transposes DNA poorly when wrapped around nucleosomes but can transpose in linker regions. This causes the clear mono and dinucleosomal patterns seen in the size distribution patterns of Tn5 transposed chromatin (Figure 3.2).

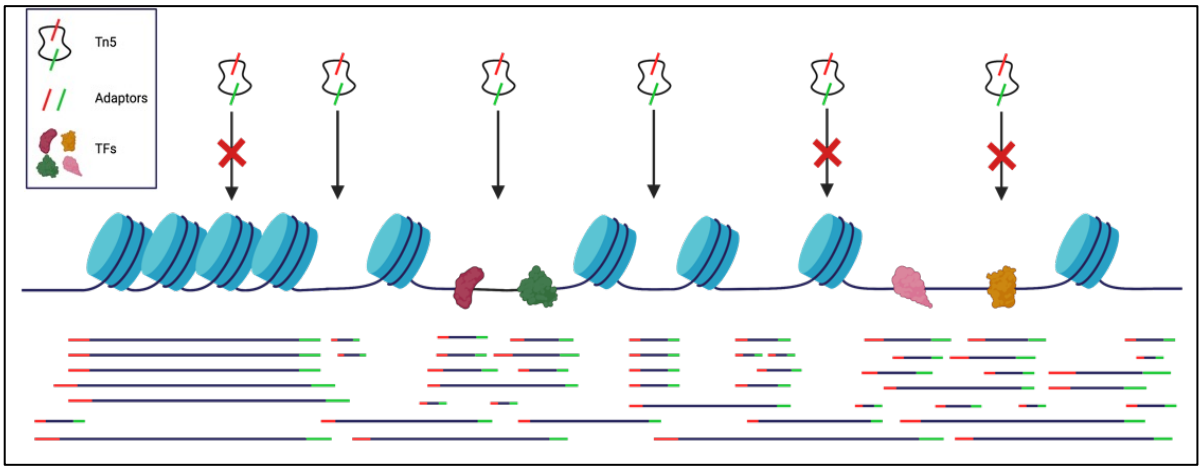


Figure 3.1: How Tn5 transposes chromatin.
Tn5 cannot cut nucleosomes or TFs. TF, transcription factor

On the contrary, it transposes DNA very frequently in regions where there are no nucleosomes, which would account for the subnucleosomal fragment sizes seen in Figure 3.2. This behaviour of Tn5 made me examine whether the different lengths of fragments contribute to different levels of information to decrease the overall nucleosomal background and emphasise regions bound by TFs.

In this chapter, I will investigate the behaviour of the Tn5 enzyme in detail in terms of the patterns in transposed chromatin it produces and, using other genomics data sets, how this relates to chromatin function and features. I will assess which DNA fragment length contributes to which level of information. I will demonstrate how the results of the *in-silico* analysis of size fractionation reduce background noise, particularly nucleosomal noise and separates this signal from a mixture of complex signals. I will explain why fragment size filtering is powerful as it transforms scATAC-seq data into high-resolution data and discusses its potential in solving fundamental problems in genomics.

3.2 Result

3.2.1 Size fractionation analysis of scATAC-seq proerythroblast data

Since Tn5 cuts the DNA differently depending on the nucleosome content, creating different fragment sizes with a clear nucleosomal protection pattern (see Figures 3.1 and 3.2). This suggested a working hypothesis that ATAC-seq patterns are composed of multiple signal types, some of which are nucleosomal and some of which are related to TF binding. Also, the pattern in Figure 3.1 is derived from the analysis of fragment sizes from the ATAC-seq data itself, that the signal may be separatable bioinformatical

based on the size of the resultant fragment. I, therefore, looked at the genomic open chromatin patterns of DNA fragments across different bins of fragment sizes in scATAC-seq from CD34+ proerythroblast (Figure 3.2).

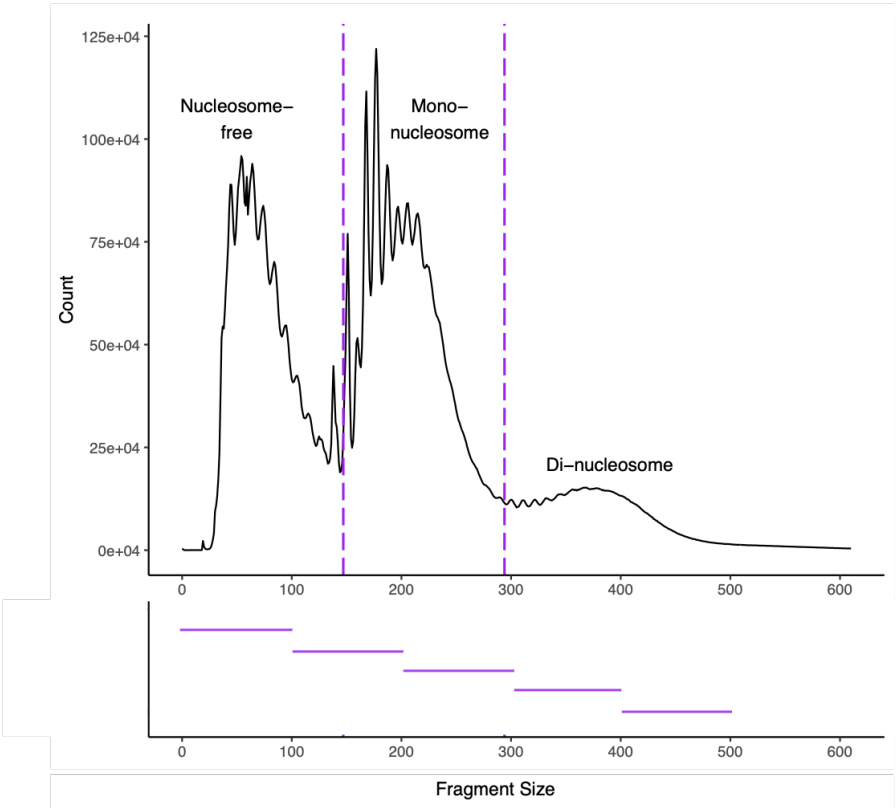


Figure 3.2: The distribution of DNA fragments for the scATAC-seq proerythroblast dataset clearly shows patterns for nucleosome positioning.
The x-axis represents the fragment size of tagmented DNA, whereas the y-axis shows a count frequency for those fragments. The purple dotted line refers to an average length of a nucleosome, 150bp. DNA fragments before the purple line are nucleosome-free DNA fragments, while DNA fragments after the purple line, in order, are mono-nucleosomal and di-nucleosomal DNA fragments. Five short purple lines beneath the distribution plot represent different ranges of fragment size.

I thought these varied distributions might provide different information about the contribution of these size ranges to signal and background in the assay, so I divided scATAC-seq CD34+ proerythroblast into five equally spaced size bins, as is seen in Figure 3.2. Figure 3.3 shows the genome browser tracks for these size bins over the alpha-globin locus, which is highly active in these cells, including their known

enhancers in the body of the NPRL3 gene. Firstly, it showed that our original hypothesis was correct, and the first bin <100bp is almost devoid of background between the peaks compared to the raw, unfiltered data (black track). Most of the background appears to be relegated to the >100bp bins.

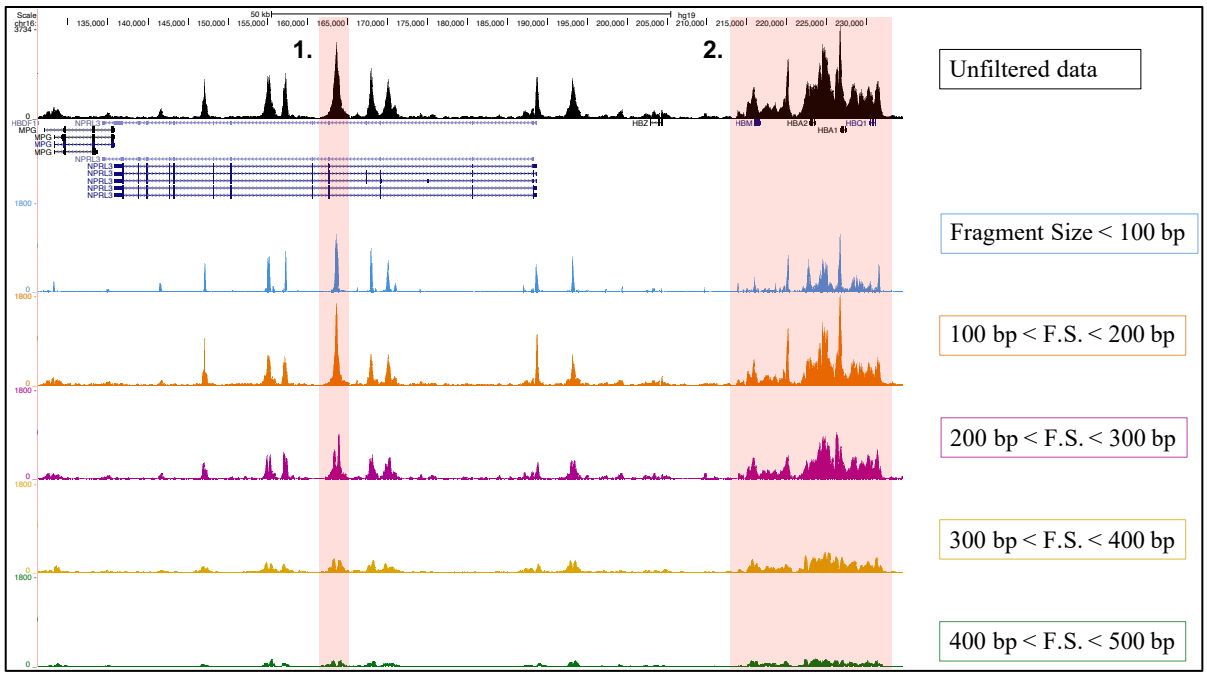


Figure 3.3: Fragment size analysis of scATAC-seq proerythroblast at the alpha-globin locus indicates nucleosome-free regions are the most informative signals.
The black track on top is an unfiltered scATAC-seq proerythroblast. Tracks beneath the black track refer to five different fragment size ranges, as seen in Figure 3.2, in order. Highlighted regions are explained in the text.

However, it also showed a surprising and intriguing result. It is also clear that the larger fragment size also strongly contributed to the peaks in the signal rather than just the overall background. This is clearly seen in the 200-300 bp bin, which retains the same general peak structure even though all the fragments are greater than mono-nucleosome size, and this can be seen in even larger bin sizes (300-400 and 400-500). The important effect of this is that in the first bin (100-200bp), the peaks while retaining their position, can be seen to be of a much higher resolution.

Conversely, in the larger bin sizes of 200bp and greater, strong peaks (see first highlighted region as an example, the R2 enhancer region) start to separate, forming two individual peaks. This strongly suggests that these two separate peaks represent flanking nucleosomes composed of large, transposed fragments. In contrast, the highly punctate peak in bin <100bp, which lies between them, represents smaller transposed fragments in the heart of these enhancer regions. As these fragments are much smaller than a protecting nucleosome, these could represent a pattern caused by smaller protecting proteins, i.e., transcription factors. This general pattern of increased resolution can be seen at all the discrete peaks in the tracks, in fact, genome-wide.

Another very interesting effect can also be seen in the large domain of open chromatin associated with the highly active alpha globin genes (highlighted region 2). Due to the high level of transcriptional activity, the open chromatin signal covers almost the complete genomic locus over the genes. It is generally identified as one block when the peak is called (see unfiltered data track). Furthermore, it is also impossible to clearly identify promoter-associated open chromatin peaks for many of these genes, particularly HBA2. However, in the track of bin size <100bp, a clear and punctate peak can now be seen at the HBA2 gene, with the previously confounding signal now relegated to the larger bin sizes. This identification of hidden peaks in highly active regions is not limited to the highly active alpha globin genes and can be seen at many complex loci across the genome.

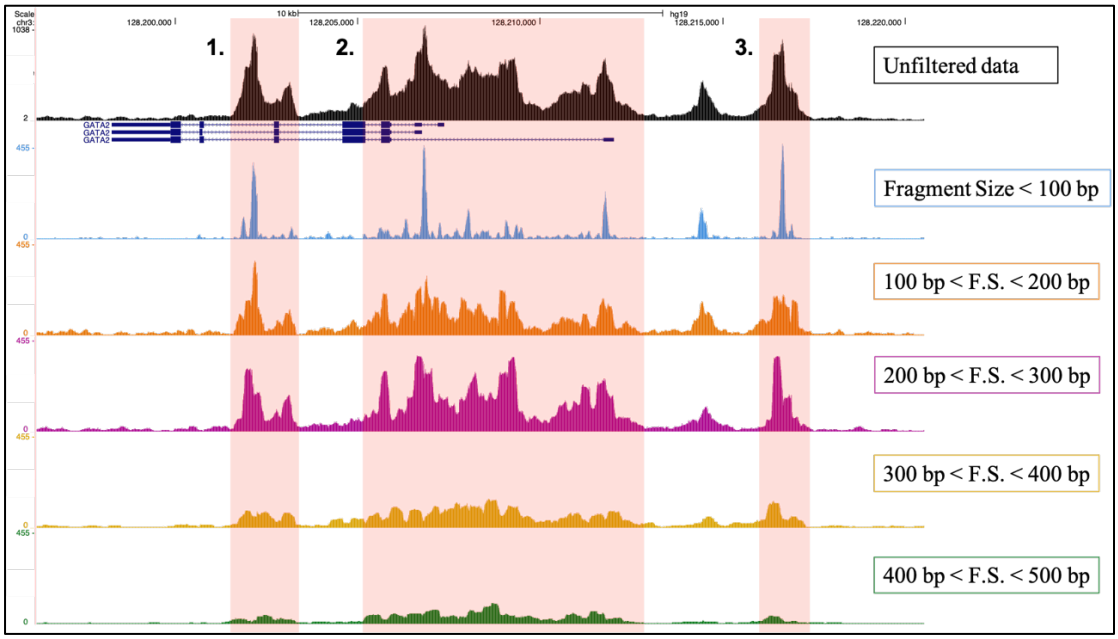


Figure 3.4: Fragment size analysis of scATAC-seq proerythroblast at the GATA2 locus displays that nucleosome-free regions reflect super-enhancer precisely.
The black track on top is an unfiltered scATAC-seq proerythroblast. Tracks beneath the black track refer to five different fragment size ranges, as seen in Figure 3.2, in order. Highlighted regions are explained in the text.

Figure 3.4 shows another example at the GATA2 locus, expressed at more typical transcriptional levels. At the 1st and 3rd highlighted regions, size fractionation analysis <100 bp identified highly punctate peaks of smaller fragments within the larger unfractionated peaks, similar to the enhancers in the alpha-globin locus. In the middle 2nd highlighted region over the body of the GATA2 is a signal reminiscent of that found over the alpha globin genes, where the complexity of the signal makes it difficult to identify discrete signals associated with the known annotated promoters of GATA2. However, in size fractionate bin <100bp, clear punctate peaks can now be seen precisely at the annotated promoters.

To investigate this observation at the level of the whole dataset, I analysed the distribution of fragment sizes only within peak regions called from the unfractionated data and compared these to those outside of peak regions, see Figure 3.5.

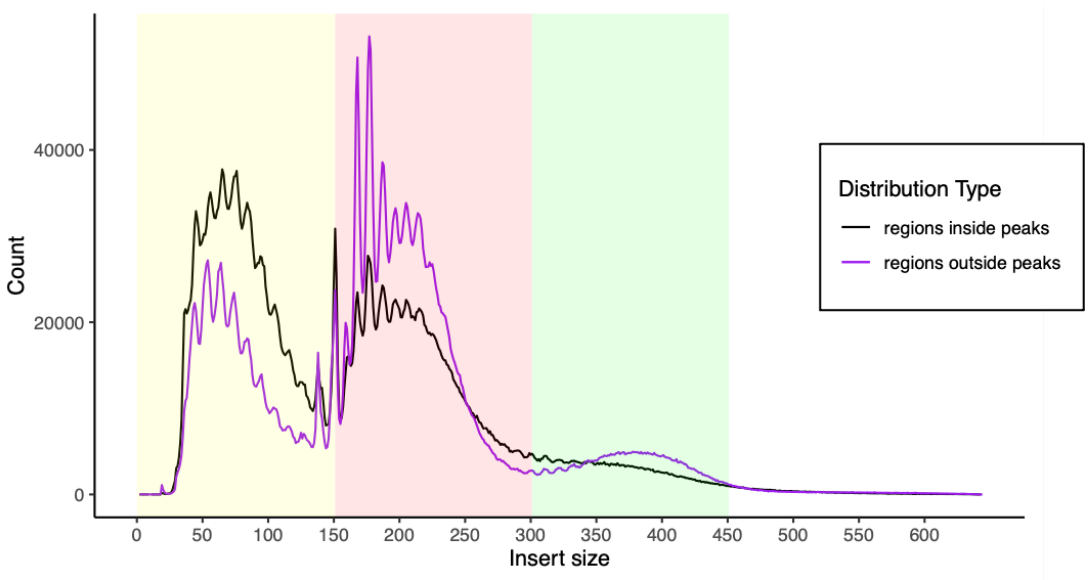


Figure 3.5: The distribution of DNA fragments for the scATAC-seq proerythroblast dataset in regions inside peaks and outside peaks suggests that the two distributions have opposite patterns, providing different levels of information.
The purple distribution belongs to regions outside peaks, while the other belongs to regions inside peaks. Three colours represent fragment sizes with different nucleosomal contents; the first region is nucleosome-free, the second is mono-nucleosomal, and the third is di-nucleosomal. The purple distribution shows high frequency in nucleosomal regions, whereas the black distribution displays high enrichment in the first region, where there is no nucleosome.

In Figure 3.5, there are three main distributions; less than 150 bp (nucleosome-free), ~150-300bp (mono-nucleosomal) and ~300-450 (di-nucleosomal). For the distribution in the peak regions, which is shown as a black line, there is an enrichment for short DNA fragments (nucleosome-free). Whereas outside the peak regions (purple line), we see a depletion of subnucleosomal and an enrichment for both mono and dinucleosomal.

Taken altogether, all these analyses suggest that strong nucleosomal signals contribute not only to the general background but also to the informative peak signals in these assays. This shows that nucleosomes in and around active elements and sites of transcription are more susceptible to Tn5 transposition and contribute strongly

to open chromatin signals. Conversely, it shows that TF-bound regions can be more clearly identified at a great resolution within these data via their associated subnucleosomal size transposed fragments.

Furthermore, this confounding nucleosomal signal can be removed by the simple step of size fractionating the signal to isolate subnucleosomal fragments. We see strong enrichment for peaks of subnucleosomal-sized fragments within the cores of enhancer elements and promoter elements, even to the degree where this distribution can clearly pinpoint the position of known promoters where this was not possible in the unfiltered data.

Obviously, the fractionation of the data into 5 bins is very unwieldy and, from our analysis, unnecessary. Therefore, in light of the size fractionation analysis and distribution pattern of peaks and outside of peaks, going forward, we decided to size fractionate the data into only two bins <=150bp and >150bp. Data with reads whose fragment length is <=150bp represents regions enriched for TF binding sites, whilst data with >150bp can be used as an indicator of nucleosome enriched.

3.2.2 What can high-resolution scATAC-seq data provide?

The role of open chromatin assays such as ATAC-seq is to provide a general map of TF binding activity genome-wide. It is currently unfeasible to generate such maps via methods such as ChIP-seq due to incomplete knowledge of which TFs are active in a particular cell type, the lack of antibodies against all known TFs and the Herculean effort required. Open chromatin therefore acts as a generalisable proxy for TF binding

which is applicable across all cell types. Therefore, an approach that can identify the TF bound signal more accurately and reliably within this proxy signal would be of great value for all downstream applications.

My previous analysis has shown that ATAC-seq signal is highly contaminated with nucleosomal signal and size fractionation can partition this nucleosomal signal away from the TF signal, which results in obtaining high-resolution scATAC-seq data.

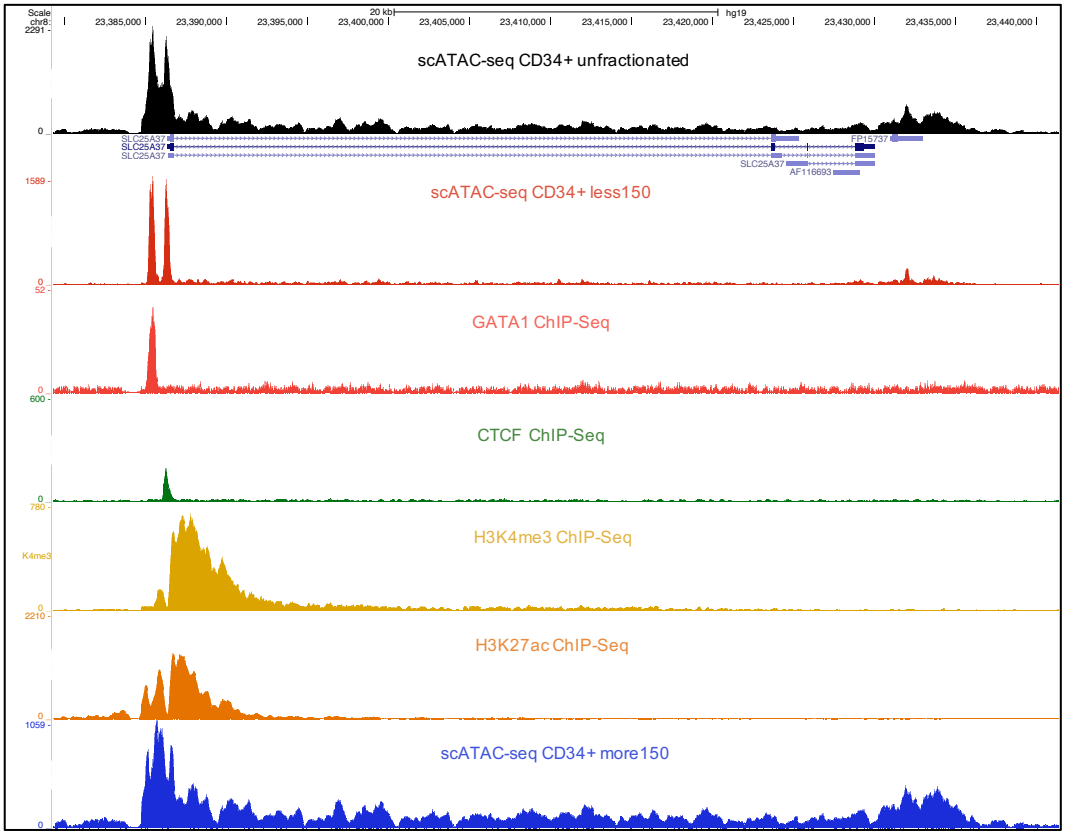


Figure 3.6: Fragment size analysis of scATAC-seq proerythroblast at SLC25A37 locus indicates that size fractionation is able to separate different regulatory elements precisely.
Unfractionated aggregated scATAC-seq data is shown in black at the top, and this data fractionated <150bp is shown below the gene annotation in red. The >150bp fraction is shown in blue at the bottom of the figure. ChIP-seq data for the erythroid TF GATA1 is shown below the <150bp track and ChIP-seq for the chromatin-associated protein CTCF is shown below this. The active promoter marks H3K4me3 and H3K27ac are shown below that.

Figure 3.6 shows a great example to demonstrate the practical advantage of how high-resolution data can more completely and precisely annotate the activity of a genomic locus. Figure 3.6 shows a compendium of genomics data at the SLC25A37 locus, which is highly active in proerythroblast cells.

In the unfractionated ATAC-seq, a large open chromatin site of complex structure is clearly evident at the promoter element of the SLC25A37 gene. Size fractionation analysis (red track) separates this into two clearly defined individual peaks. The first peak precisely coincides with a GATA1 bound region and the second with a CTCF bound region. Fragment size filtering is, therefore, capable of deconvoluting complex functionality previously merged into a single element in the ATAC-seq data. On the other hand, the blue track nucleosomal track appears generally enriched in the body of this highly active gene and, therefore, may be of use in highlighting regions where nucleosomes are disrupted by active processes such as transcription.

Similarly, at the previously described GATA2 locus, the highly punctate peaks revealed from the highly complex ATAC-seq signal by the fractionation analysis can be seen to align precisely with peaks both in the GATA1 and CTCF ChIP-seq, confirming these regions as TF-bound elements (Figure 3.7). Looking at the highlighted region in Figure 3.7, fragment size filtering identified not only a GATA1 binding region in the promoter but also a weak CTCF site adjacent to it. Size fractionation appears to be not only a very sensitive method for precisely identifying TF-bound regions in promoters, enhancers, and CTCF sites but is of particular advantage when they are in close proximity to each other and their open chromatin signals merge.

In particular, the advantage of the extra resolution is very evident at regions that are defined as so-called super-enhancers, where active elements are, by definition, in close proximity and in the examples below (KLF1 and NFE2 Figures 3.8 and 3.9) even lie over the bodies of active genes with promoter elements and adjacent CTCF bound elements.

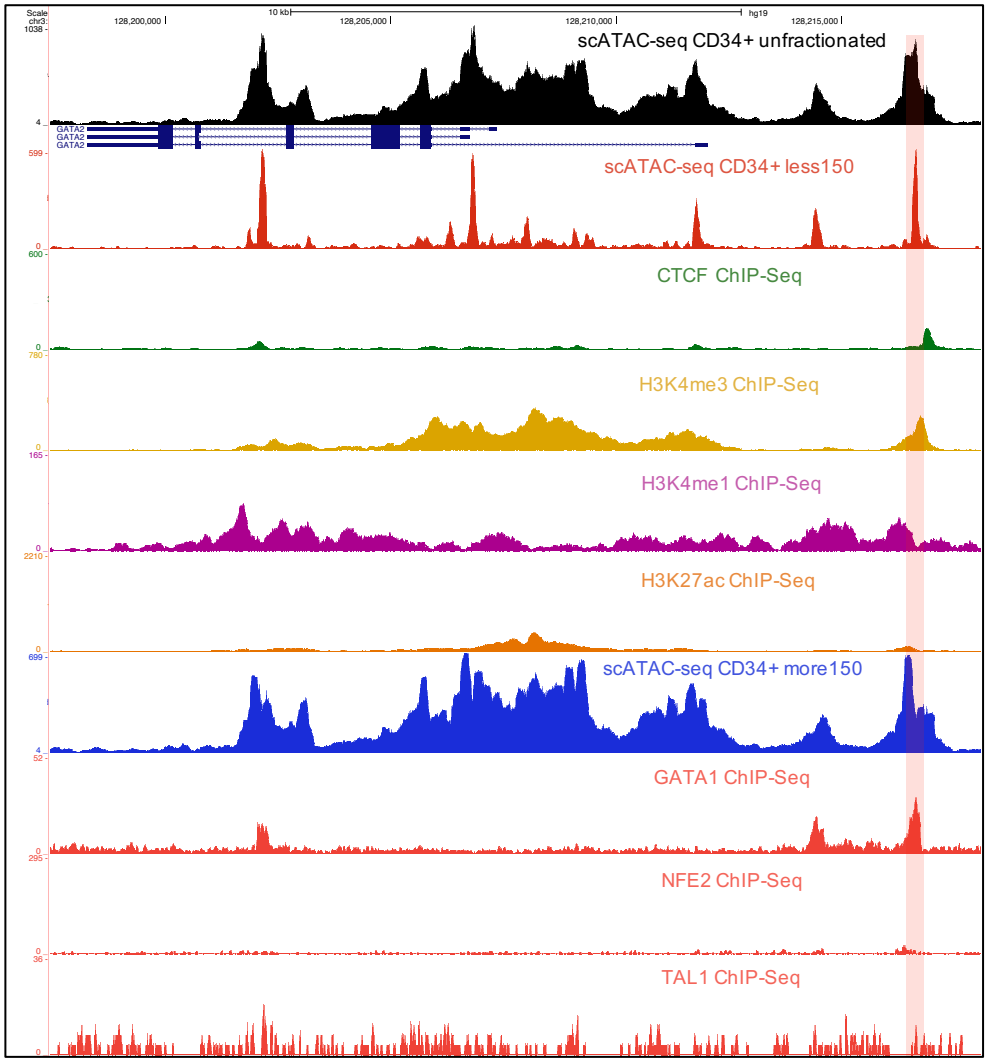


Figure 3.7: Fragment size analysis of scATAC-seq proerythroblast at GATA2 locus indicates that size fractionation is able to separate different regulatory elements precisely and can identify super enhancers.
Unfractionated aggregated scATAC-seq data is shown in black at the top, and this data fractionated <150bp is shown below the gene annotation in red. ChIP-seq for the chromatin-associated protein CTCF is shown below that, followed by the ChIP marks H3K4me3 (yellow track), H3K4me1 (purple track) and H3K27ac (orange track). The >150bp fraction is shown in

blue after the ChIP marks. ChIP-seq data for the erythroid TF GATA1, NFE2 and TAL1 in order are shown below the <150bp track in red. Highlighted region is explained in the text.

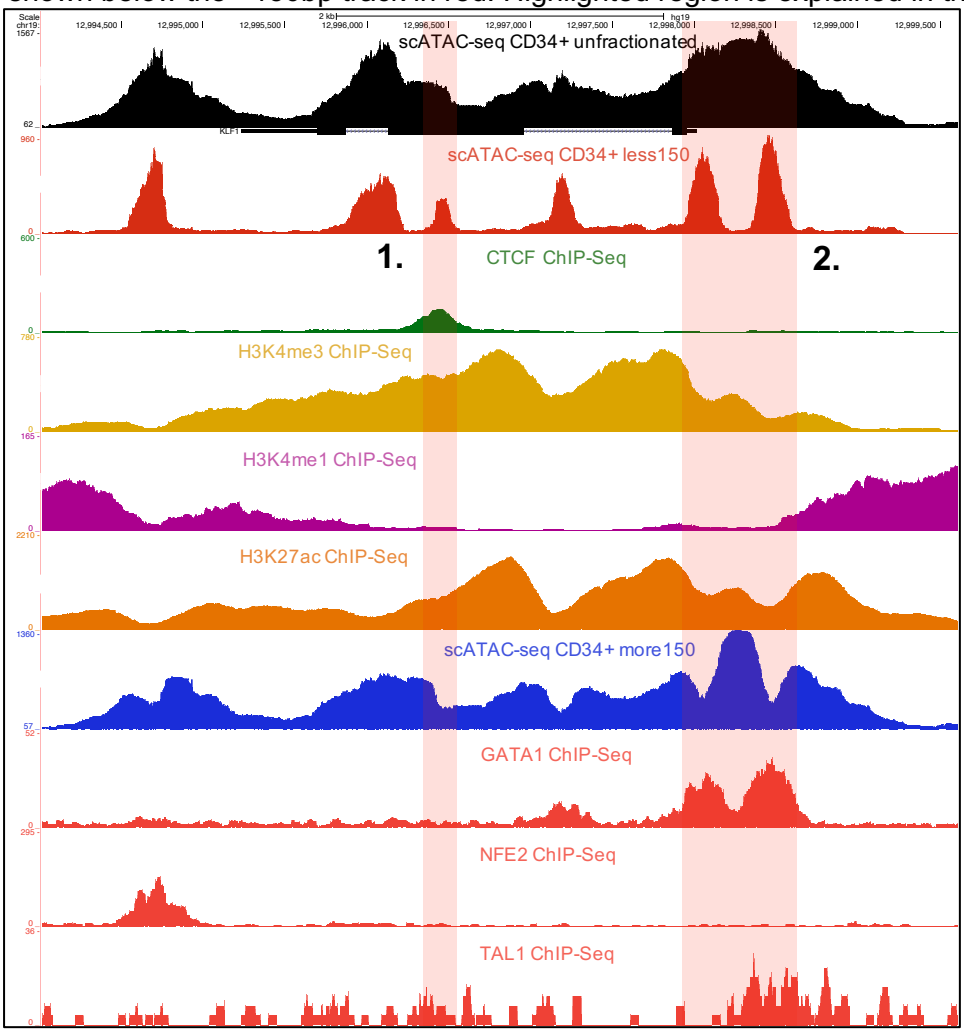


Figure 3.8: Fragment size analysis of scATAC-seq proerythroblast at KLF1 locus indicates that size fractionation is able to separate different regulatory elements precisely.
Unfractionated aggregated scATAC-seq data is shown in black at the top, and this data fractionated <150bp is shown below the gene annotation in red. ChIP-seq for the chromatin-associated protein CTCF is shown below that, followed by the ChIP marks H3K4me3 (yellow track), H3K4me1 (purple track) and H3K27ac (orange track). The >150bp fraction is shown in blue after the ChIP marks. ChIP-seq data for the erythroid TF GATA1, NFE2 and TAL1 in order are shown below the <150bp track in red. Highlighted regions are explained in the text.

In both these examples, peak callers call the whole region a single block, although some elements embedded in the block are evident to the naked eye. However, some clearly defined structures in the size-fractionated track would be impossible to discern in the raw data (see highlighted regions 1 and 2 in Figure 3.8), revealing a CTCF-

bound element in the body of the gene (region 1) and a secondary peak upstream of the promoter element of KLF1 (region 2) both bound by the TF GATA1. In the NFE2 locus (Figure 3.9), as in the GATA2 locus, size fractionation precisely identifies the two annotated alternative promoters, the first of which precisely align with peaks of binding for the TFs TAL1 and GATA2 (region 2) and multiple CTCF bounds sites in the body of the gene (see region 1 as an example).

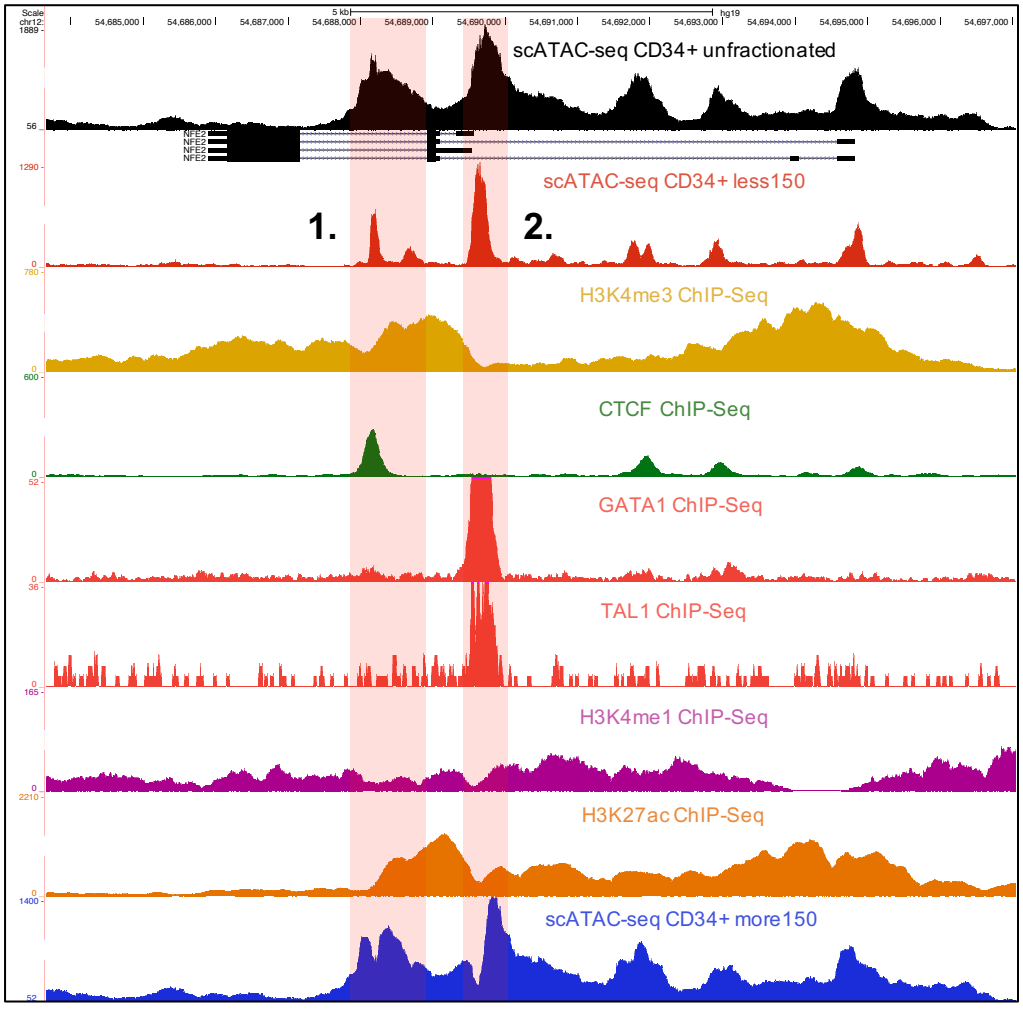


Figure 3.9: Fragment size analysis of scATAC-seq proerythroblast at NFE2 locus indicates that size fractionation is able to separate different regulatory elements precisely.
Unfractionated aggregated scATAC-seq data is shown in black at the top, and this data fractionated <150bp is shown below the gene annotation in red. The >150bp fraction is shown in blue at the bottom of the figure. The ChIP-seq for H3K4me3 is shown in yellow after fractionated <150bp track. The ChIP-seq for the chromatin-associated protein CTCF is shown in green after the H3K4me3 track. ChIP-seq data for the erythroid TF GATA1 and TAL1 are

shown in red below the ChIP-seq for the chromatin-associated protein CTCF in green. Below that, there are the ChIP marks H3K4me1 (purple track) and H3K27ac (orange track).

These locus-based examples show a clear correlation between the binding of TFs and the regions identified by fractionation analysis. In the next section, I will extend this analysis genome-wide to look at the genome-wide effect of size fractionation on peak structure and the association between these high-resolution peaks and TF binding motifs and genomic features linked to TF binding, such as sequence conservation.

3.2.3 Correlation between high-resolution peaks and features of TF-binding at the genome-scale.

To confirm at the genome-scale that the fragment size filtering method provides high-resolution scATAC-seq data that more clearly identifies TF-bound regions, I evaluated how peaks from size-fractionated scATAC-seq data correlate with sequence conservation, motif enrichment, and Tn5 cut size compared to standard peaks calls from the same but unfractionated data.

3.2.3.1 Distribution of high-resolution peaks in scATAC-seq data

To understand the general effect of size fractionation on the structure of the data, I decided to visualise the effect across all open chromatin sites in the data in standard ATAC-seq and how this changed upon size fractionation. To make this a fair comparison, I repeated the comparison using standard peak calls from unfractionated data and using high-resolution peak calls from the fractionated data.

To do this, I called peaks both on unfractionated and size-fractionated data using Lanceotron and filtered peaks by the peak score > 0.5 parameters to generate a

moderately stringent set of high confidence peaks in both. This analysis showed that irrespective of which peak call I used, the open chromatin signal in the size-fractionated ATAC-seq data (Figure 3.10, right-hand panels, B, D and F) was more focused and less dispersed than in the unfractionated data (Figure 3.10, right-hand panels, A, C and E). This is most evident in the meta-pileup of these data when the high-resolution peak calls are used with the size-fractionated data (comparing green line fractionated signal with blue line unfractionated signal in panels A and B). This shows that across the how genome-wide data set size fractionation increases the resolution of the data.

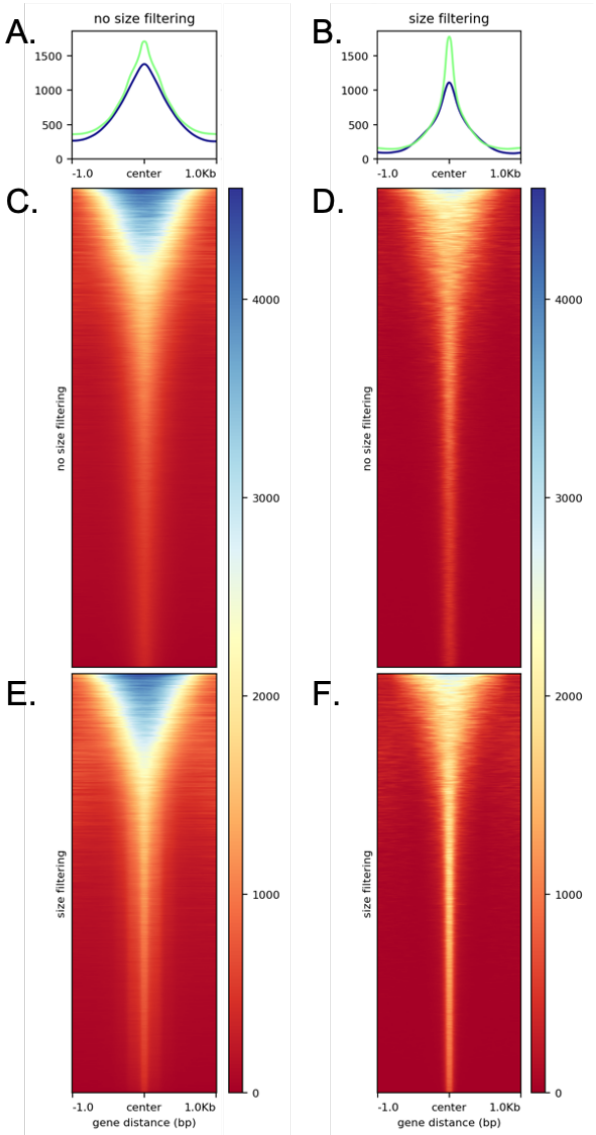


Figure 3.10: The distribution of low-resolution and high-resolution peaks over the unfractionated and fractionated data suggests that high-resolution peaks are more pronounced and narrower, resulting in capturing only informative chromatin accessibility.
A. The green line refers to the high-resolution peak coverage distribution over the standard scATAC-seq data. In contrast, the blue line shows the standard peak coverage distribution over the standard scATAC-seq data. B. The green line refers to the high-resolution peak coverage distribution over the high-resolution scATAC-seq data. In opposition, the blue line shows the standard peak coverage distribution over the high-resolution scATAC-seq data. C. The distribution of the standard peak count in the standard scATAC-seq data is shown as a heatmap. D. The distribution of the standard peak count in the high-resolution scATAC-seq data is shown as a heatmap. E. The distribution of the high-resolution peak count in the standard scATAC-seq data is shown as a heatmap. F. The distribution of the high-resolution peak count in the high-resolution scATAC-seq data is shown as a heatmap.

3.2.3.2 The correlation between high-resolution peaks and conservation data

I next wanted to determine whether the regions called in the high-resolution data better defined regions likely bound by transcription factors. I first decided to look at whether sequence conservation (as a general proxy for the position of functional TF binding motifs) correlates better with the regions in the high-resolution peak set or the standard low-resolution peaks. My logic is that the regions within the ATAC signal responsible for directing TF binding are much more likely to be conserved in evolution than just nucleosome-associated regions. It is also important to consider precisely how such meta-analysis works as open chromatin elements are enriched for sequence conservation and so should always show an enriched association.

Metaplots work by cumulatively piling up signals on top of each other across many examples of a class of regions in the genome, open chromatin sites in this instance. Data is piled up relative to a fixed position in the object (usually the centre), and so the signal in the same relative position in each example is cumulatively added to the signals in the same position as the preceding examples. As a by-product of this, the more precisely the centre of the regions used is associated with the signal that is

accumulated, and then the sharper and more enriched will be the meta-peak. Our hypothesis, therefore, going into this analysis, was that if the high-resolution ATAC peaks more precisely identified TF-bound regions in a much smaller genomic region, then they would constrain any signal associated with TF binding (such as sequence conservation) more precisely to the centre of the plot. This would have the effect that when piling up the same data over more precisely defined TF-associated regions would produce much more enrichment over the centre of the plot compared to a set of less well-defined regions.

Figure 3.11 shows the result of piling up base-pair resolution conservation data (PhyloP evolutionary conservation scores for 100 vertebrates) over standard peaks derived from the unfractionated ATAC-seq data (blue) and over high-resolution peak call derived from size-fractionated <150bp data (orange) using Lanceotron as previously described. The general association of open chromatin and conservation can be clearly seen in the blue standard peaks with a cumulative increase in signal stretching around 1kb from the centre of the peak calls. The structure of the meta-plot in the high-resolution analysis is, however, very different. It shows a very large increase in peak height around the 0 positions in the metaplot and an extremely punctate peak of conservation enrichment at position 0. This shows that the high-resolution peak calls are very enriched for sequence conservation and more precise define the position of sequence conservation and, therefore, functional TF binding motifs.

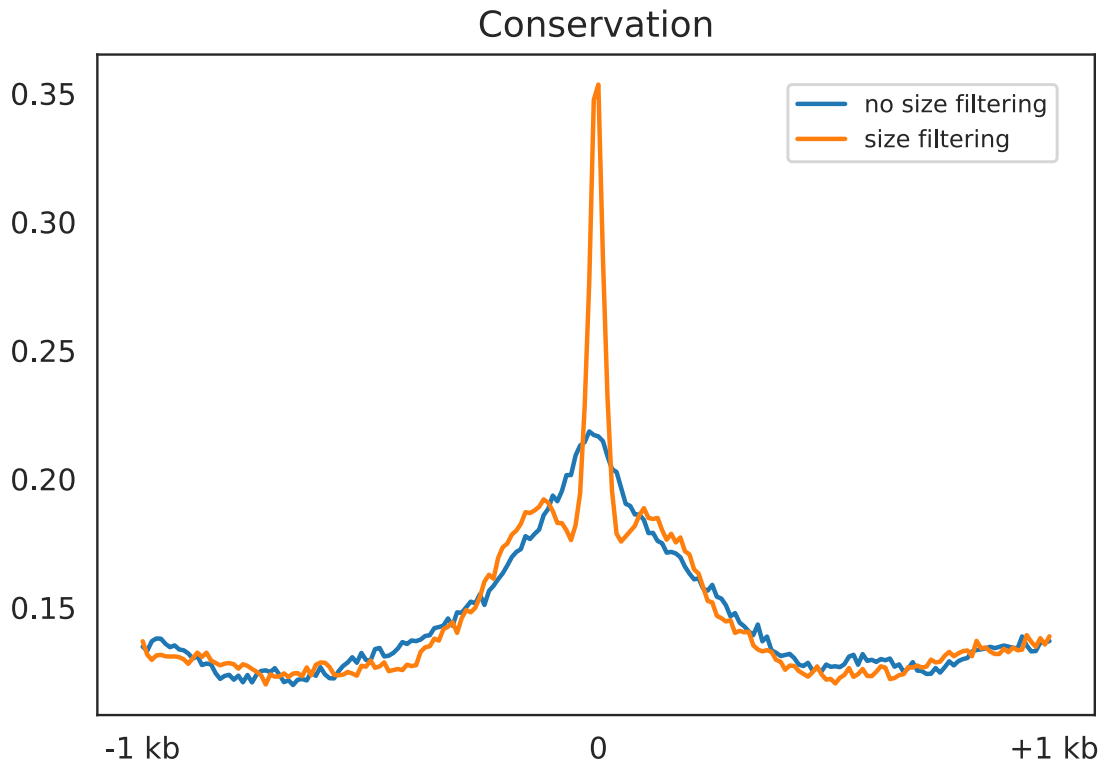


Figure 3.11: PhyloP evolutionary conservation distribution shows that peaks from the high-resolution scATAC-seq data can precisely express the exact location of sequence conservation, having a punctuated signal.
Peaks from the standard (blue) and high-resolution (orange) data are accumulated to the centre of each peak and extended as 1kb from each side. Conservation scores were acquired for both extended peak sets.

3.2.3.3 Distribution of high-resolution peaks in motif enrichment

In this previous analysis, we used sequence conservation as a proxy for the position of TF binding motifs. We so did not require precise information about the sequence motifs being used in this cell type. However, the erythroid system is one of the best-understood cell types in terms of gene regulation, and many of the key TFs and their binding motifs are well-known and highly characterised. This, therefore, gives us the opportunity to extend this analysis using the position of sequence motifs of key TFs known to be active in these cells. The analysis is very similar to the conservation analysis and uses the same high-resolution and standard peak calls. I replaced the conservation score input track with an input track where positions in the genome which

match the motif in question are given an arbitrary score of 10, while unmatched positions have a score of 0. In this way, motif occurrence can be analysed in the same manner as the conservation scores and will behave similarly in the meta-plots. Therefore, we would expect a larger and more punctate meta-peak if motif positions are defined more precisely in the high-resolution peaks compared to the standard peak call.

I chose KLF1, NFE2, GATA1, GATA2 and GATA1_TAL1 motifs to do this analysis as they have previously been shown to be enriched in open chromatin sites in this cell type [76]. As can be seen in Figure 3.12, in each case, there is a more pronounced and punctate peak in the high-resolution peak set compared to the standard peak set, showing that the position of the motifs for these key erythroid TF is much more precisely defined in the size-fractionated peak set.

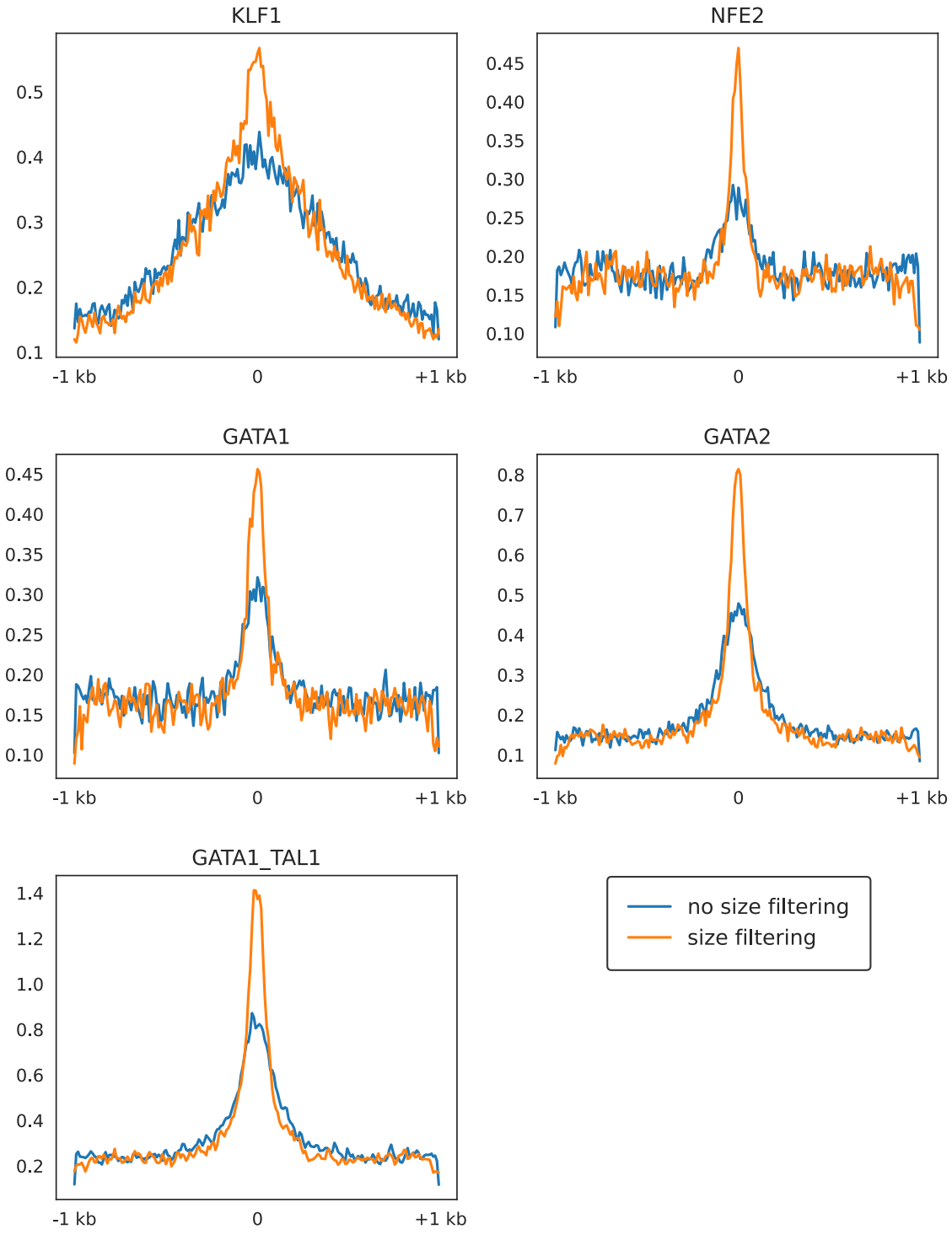


Figure 3.12: The high-resolution peak set can identify the locations of important motifs, including KLF1, NFE2, GATA1, GATA2, and GATA1 + TAL1, more precisely than the standard peak set as the data is piled up on the right location of motifs.
Peaks from the standard and high-resolution (orange) data are accumulated to the centre of each peak and extended as 1kb from each side. Read coverage from related motifs was acquired for both extended peak sets.

3.2.3.4 Distribution of Tn5 cut sizes around high-resolution peaks and standard peaks.

I next wanted to ask if the meta-analysis of where the Tn5 actually binds to the chromatin would show a different pattern between the high-resolution peak set and the standard peak calls. This was inspired by the previous analysis in this chapter that showed at discrete elements such as the R2 globin enhancer that the high-resolution peaks could be seen to be flanked by phased nucleosomes, seen as two flanking peaks in the larger bin sizes (Figure 3.3). Therefore, while the centre of the high-resolution peaks may more precisely define TF binding sites, as shown in the previous sections, conversely, the edges of these peaks may more precisely define the positions of phased flanking nucleosomes. This may be evident in a meta-analysis of Tn5 insertion sites as they have depleted in the nucleosome-bound region and form a distinctive pattern in ATAC-seq data.

However, Tn5 binds as a homodimer. When it binds to DNA, it forms two insertion events for each Tn5 molecule, separated by 9 bases. Thus, the Tn5 binding site actually lies in the centre of the Tn5 dimer, not at each Tn5 insertion point. To overcome the Tn5 insertion bias, 4 bp was added to the + strand, whereas 5 bp was subtracted from the – strand [69]. As before, a quantitative track of the positions of these insertion events was used as input in a similar manner as conservation scores using the same standard and high-resolution peak calls.

Figure 3.13 shows the results of this analysis with cumulative enrichment for corrected Tn5 insertions centred on standard peak calls in blue and high-resolution peak calls in orange. As expected, both peak sets show meta-enrichment for Tn5 insertions as

all regions are hypersensitive to Tn5 insertion. However, only the high-resolution peak call produces a strong stepped nucleosomal pattern in the insertion site meta profile. This can only be due to the more efficient phasing of the Tn5 insertions showing that the high-resolution sites represent a more precise map of the TF bound regions in between the flanking phased nucleosomes.

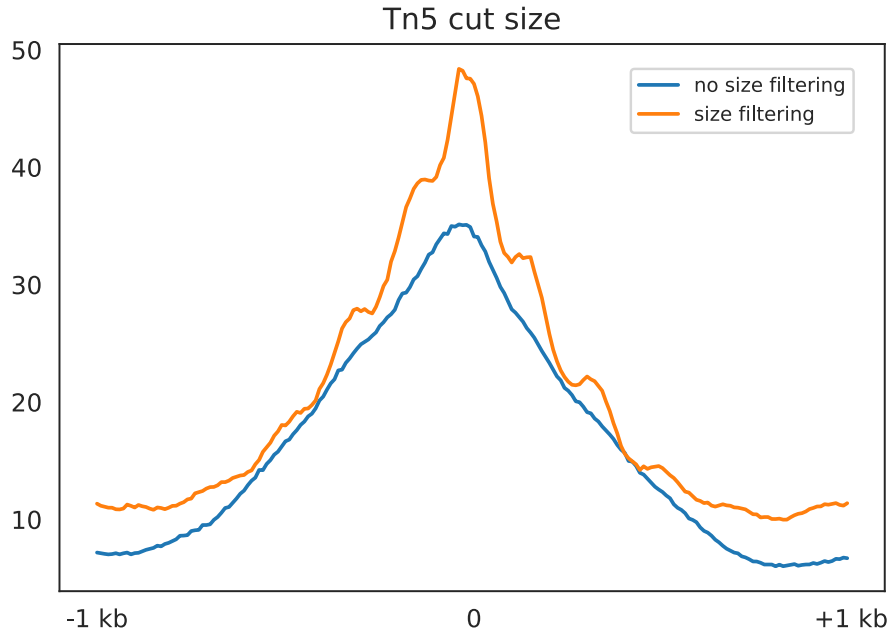


Figure 3.13: The high-resolution peak set reflects the Tn5 cut size better than the standard peak set.
Peaks from the standard and high-resolution (orange) data are accumulated to the centre of each peak and extended as 1kb from each side. Tn5 insertion bias was excluded from the data by adding 4 bp to the + strand and subtracting 5 bp from the – strand. Read coverage from the bias-corrected data was acquired for both extended peak sets.

3.2.4 Does the high-resolution strategy work on Bulk ATAC-seq data?

The obvious question is to investigate whether the fragment size filtering method provides similar benefits to Bulk ATAC-seq data. The same size fractionation was applied to Bulk ATAC-seq equivalents of proerythroblast data. Figure 3.14 displays a comparison of how the fragment size filtering approach works in Bulk ATAC-seq and

scATAC-seq data. It is clearly seen that obtaining high-resolution in Bulk ATAC-seq is entirely applicable even if the benefit is not as high as in a single cell. Still, the size fractionation method can be applied to Bulk ATAC-seq data to remove some level of background noise. The main reason the filtering works better in scATAC-seq is that scATAC-seq technology is deep and sensitive. If Bulk ATAC-seq data is sequenced very deep, the possibility of this method's working will increase.

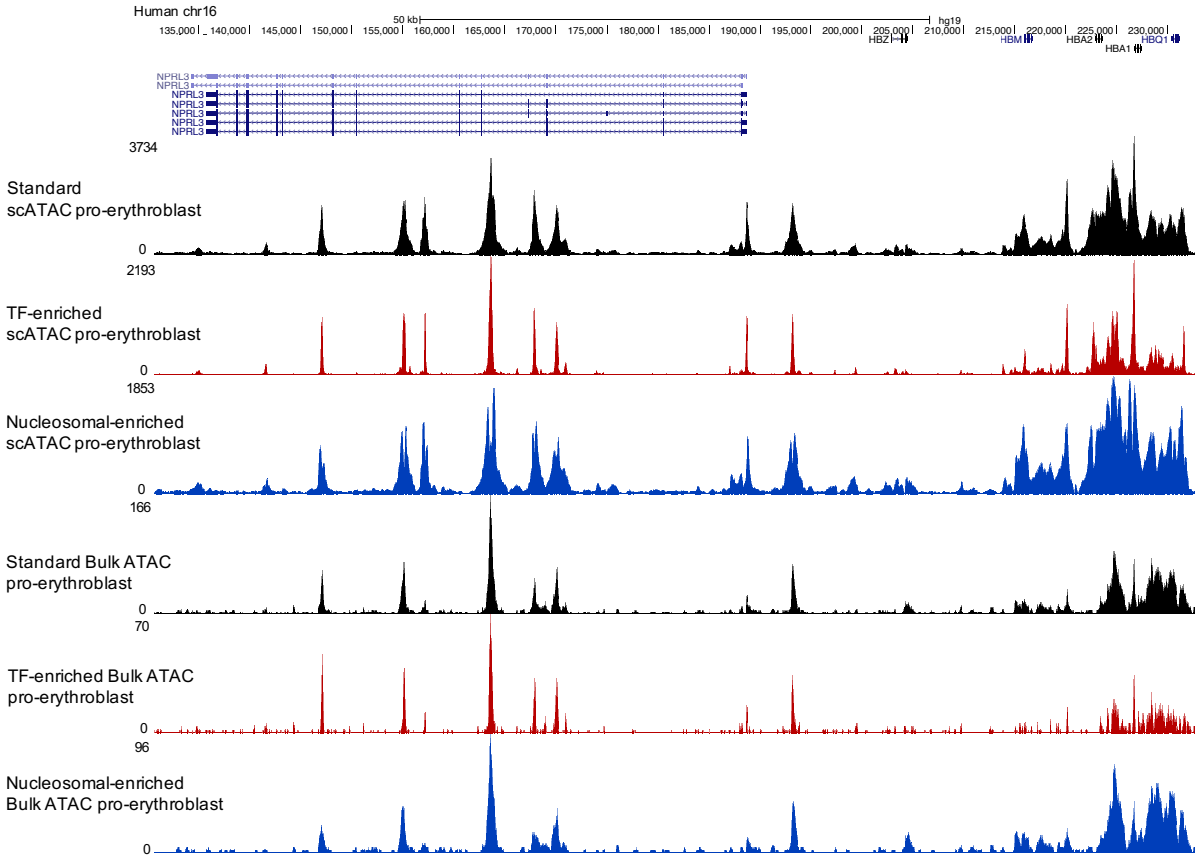


Figure 3.14: The size fractionation method works better in scATAC-seq data compared to Bulk ATAC-seq data, even if the method can be applied to the Bulk ATAC-seq data. The first three tracks belong to scATAC-seq; the black is standard scATAC-seq data, the red is TF-enriched, and the blue is nucleosomal-enriched data. the last three tracks belong to Bulk ATAC and follow the same annotation as scATAC tracks.

3.3 Discussion

In summary, using both analyses at individual loci with matched ChIP-seq data for TFs and genome-wide meta-analysis for signals related to functional TF binding, I have shown that size fractionation of ATAC-seq data can significantly increase the resolution of open chromatin data and more clearly define regions that are bound by transcription factors. Considering the use of the ATAC-seq signal to generate fundamental maps of TF binding and activity across cell types, this ability has beneficial impacts on all downstream applications of ATAC-seq. As I have shown, it can be applied to specific genomic loci to uncover greater detail in the regulatory landscape of genes and even regulatory elements hidden in the normal signal. It could therefore be applied dynamically to look for subtle changes in TF binding associated with cellular differentiation or cellular activation by extrinsic signals, an application particularly suited for scATAC-seq, which can characterise multiple stages of cellular differentiation in a single assay. Also, it could be applied more generally for statistical motif analysis to understand the complement of TF motifs used in a given cell type, where the greater definition of TF-bound regions will increase signal-to-noise to increase the sensitivity of these analyses.

However, it has been clear to me that these data can have a large impact on our ability to prioritise and functionally interpret non-coding human genetics. This has been the main drive behind my work with single-cell epigenomics, and trying to exploit this potential makes up the remainder of my thesis. I will lay out this potential application in this discussion and will describe my computational development of a platform to leverage this potential in the subsequent chapter.

3.3.1 The potential for high-resolution ATAC-seq for the prioritisation of causal non-coding variants.

GWAS has generated huge datasets detailing the statistical association between genetic variation and an ever-increasing range of diseases. However, its findings are complex and difficult to interpret with confidence and at scale. The ultimate requirement to leverage these data to understand disease biology is to know which genes in which cell types are affected, the direction of the effect (increased or decreased expression), and address this at a sufficient scale to understand the biology at the pathway level. All of these ultimately depend, as the initial step, on finding the causal variants in large linkage haplotypes and the cell types they affect.

My development of high-resolution ATAC-seq analysis is beneficial to three fundamental aspects of this. Firstly, to determine which variants in a haplotype lie precisely in the regions bound by TFs and so can affect their binding. This can be achieved by the intersection of imputed haplotypes from a GWAS with high-resolution ATAC-seq. This would prioritise only variants that fall within regions that direct TF binding rather than the large open chromatin domains that represent active nucleosomal domains (Figure 3.15). As seen from the loci in Figure 3.7 – 3.9, this represents a very large increase in the level of granularity with which this can be done, particularly at highly active or complex regions, such as super-enhancers.

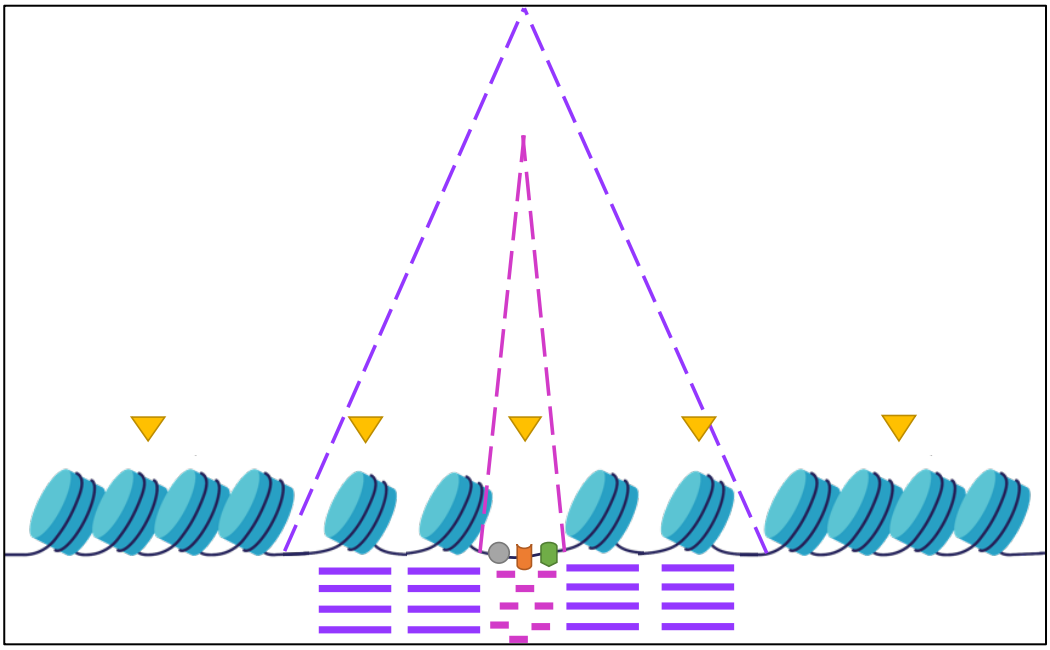


Figure 3.15: The high-resolution data can identify SNP(s) that are affecting TF binding sites at an example locus.
Standard potential Tn5 insertion is shown as dotted purple lines, whereas the high-resolution is displayed as dotted pink lines. The five yellow triangles represent hypothetical SNPs. The purple and pink bins refer to DNA fragments resulting from Tn5 tagmentation.

Secondly, due to its ability to also tease apart elements that have merged due to proximity, it can guide the generation of more appropriate hypotheses to test downstream experimental analysis. Using the previous example of the SLC25A37 promoter and three hypothetical variants (Figure 3.16). It can be seen that the middle variant, which would normally be prioritised due to its intersection with a promoter peak, can be deprioritised due to its lack of intersection with a high-resolution peak. Both of the remaining variants do intersect with high-resolution peaks, but the likely mechanism is different for each. The left-hand variant clearly intersects with a promoter element bound by GATA1 and so is likely to affect promoter activity directly. However, the right-hand variant intersects with an adjacent CTCF-bound element and so could affect enhancer interactions. Obviously, these different mechanisms require different downstream approaches to test; for example, chromatin conformation

capture would seem of limited use in the promoter variant, while it would be critical in the CTCF variant example.

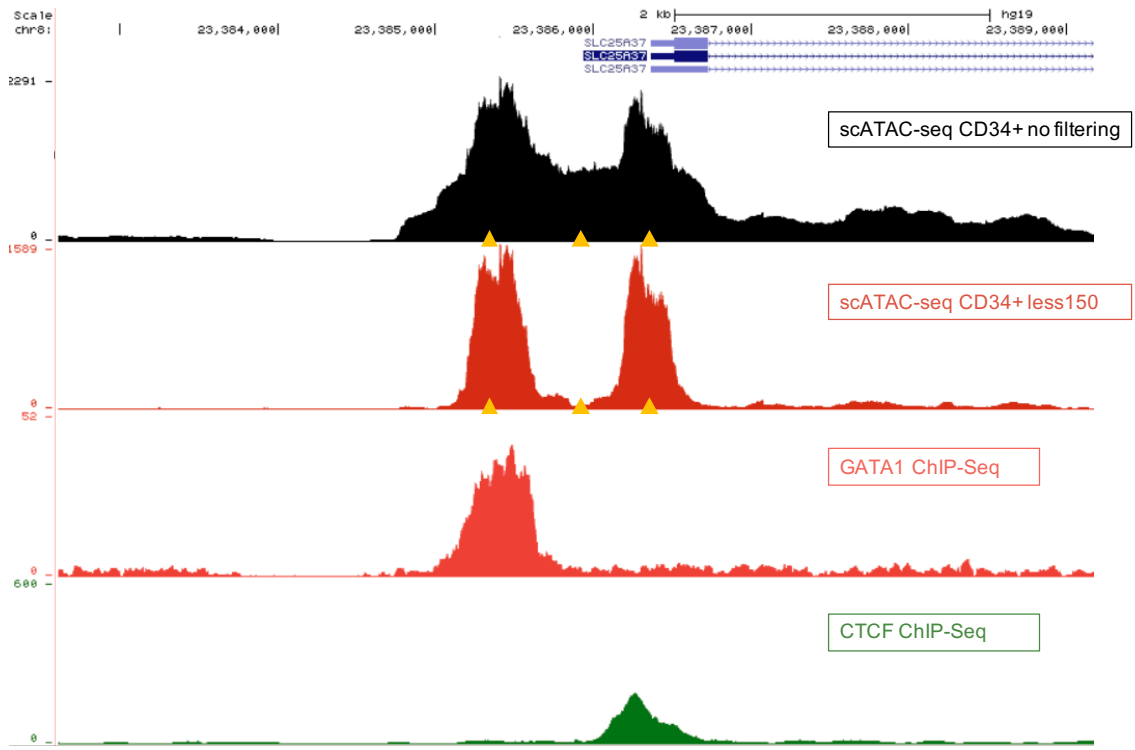


Figure 3.16: The interpretation of the exemplary SNPs (yellow triangles) at SLC25A37 locus has three different explanations in the high-resolution data, identifying promoter, GATA1 and CTCF binding sites. In contrast, Bulk ATAC data has the same interpretation for those SNPs as they affect the promoter of the gene.
Unfractionated aggregated scATAC-seq data is shown in black at the top, and this data fractionated <150bp is shown in red. The ChIP-seq data for the erythroid TF GATA1 is shown below the <150bp track. The ChIP-seq for the chromatin-associated protein CTCF is shown in green at the bottom of the figure. Three yellow triangles beneath the data are exemplary SNPs.

Thirdly, as the approach uses scATAC-seq, it can analyse the intersection between genetics and open chromatin across multiple likely effector cell types in a single experiment which obviously greatly increases analytical throughput. However, it can also prioritise the most likely effector cell types in a heterogeneous mix, for example, autoimmune genetics in PBMC scATAC-seq, via statistical enrichment of intersection between variants and the high-resolution chromatin landscape produced by my

approach. The strategy has been used in previous studies [77], [78]. The principle is based on the reasonable assumption that a large component of GWAS genetics is regulatory. Therefore, while open chromatin and the haplotypes of any sufficiently large GWAS study will show some level of background intersection, the degree of the intersection will show a statistical enrichment in cell types in which the genetics are interfering with gene regulation.

In the following chapter, I will describe the development of analytical, computational and visualisation platforms to enable the application of these three principles to large-scale genetic datasets.

4 Developing Avocato: scATAC-seq data analysis platform for prioritising non-coding genetics

4.1 Introduction

In this chapter, I aim to make use of both the computational approaches and principles described in the previous chapters to develop a computational platform to prioritise regulatory variants. Prioritisation will be based on their intersection with transcription factors bound regions defined by the high-resolution ATAC-seq analytical approach described in Chapter 3.

The goal of this work is to overcome the practical barriers to using these approaches at scale due to the high dimensionality of the data and the complexity of the multiple analytical approaches required to analyse it (Chapter 1). The challenge of complexity and scalability of scATAC-seq data and genetics can be tackled by developing an automated and reproducible analytical platform. In this chapter, I will discuss the development of an analytical platform called Avocato (**A**nalysis and **V**isualization **O**f single-**C**ell **AT**AC-seq **O**bservations) to analyse and visualise scATAC-seq data, identify disease-relevant cell clusters as well as perform functional fine-mapping of genetics using scATAC-seq data.

First, I will give an overview of the design of the tool's workflow, including analysis steps and visualisation interface. The aim is to generate a robust, feature-rich, but

easy-to-use analytical platform which is useable by experimental scientists as well as trained bioinformaticians. Subsequently, I will use publicly available and in-house datasets to demonstrate the functionalities, user features and outputs of the platform.

Design criteria and requirements

Building such a generalisable pipeline is a very challenging task and needs diverse expertise and inputs.

In brief, these are:

1. a deep understanding of the analysis of scATAC-seq and the best tools and approaches to performing analyses of key steps in the pipeline (Chapter 2);
2. the generation of a novel codebase to perform the required steps in the generation and analysis of scATAC-seq at high resolution (Chapter 3);
3. the generation of a novel codebase to intersect human genetics with high-resolution ATAC at scale;
4. the testing and validation of statistical approaches used as part of the pipeline;
5. the generation of test and validation datasets and the use of these in iterative testing and validation of each step of the pipeline;
6. the defining of the various “use cases” at each stage of the pipeline. To define the appropriate analyses, interfaces, and outputs for both the pipeline and the graphical interfaces that users will require;
7. computer science expertise to integrate all the functionality into a robust pipeline, including installation;
8. computational expertise to optimise all the processes for computational performance;

9. and the development of JavaScript interfaces based on the *use cases* defined in point 6 allows for dynamic and intuitive methods of exploring and filtering of the data.

Therefore, Avocato has been developed as a multidisciplinary collaboration that includes a computer scientist and JavaScript developer who built and optimised the backend of the pipeline (steps 7 and 8) and visualisation (step 9) web apps, respectively. My responsibilities focused on steps 1 – 8, although we all ultimately had input at each stage.

4.2 Results

4.2.1 Overview of the platform's functionality and outputs

4.2.1.1 Stage 0

This is the initial step of the pipeline after installation (see section 4.2.2.1 for the description of installation). Its role is to input user-specified metadata (file location, output names) and key parameters, which is managed through a user-friendly interface (see section 4.2.2.1) and proceeds the running of the Avocato package properly. The platform design is composed of two stages with different goals (Figure 4.1).

4.2.1.2 Stage 1

The first stage deals with processing from either fastq raw data or processed BAM files up to the generation and annotation of cell clusters. It includes an interactive JavaScript viewer capable of visualising the cluster structure in 3-dimensions (Multi-

Dimensional Viewer or MDV) and superimposing relevant data such as QC metrics or gene activity on the cluster structure. As part of the final step of stage 1, for all of the clusters, the most informative genes are found to help users to annotate clusters.

4.2.1.3 Stage 2

At the beginning of Stage 1, all of the clusters have been converted into cluster-specific high-resolution ATAC-seq genome tracks. These tracks are used to calculate the statistical enrichment of the intersection of the supplied file of genetic changes (in VCF or BED format) with the high-resolution ATAC peaks in each cluster. The most enriched clusters can be prioritised for downstream analysis as the most likely effector cell type represented in the single cell data. This statistical enrichment approach is an established approach in the field [77], [78]. Still, we used high-resolution data to generate more precise intersections to calculate the statistical enrichment (see section 4.2.5.2 for the description of the statistical enrichment approach and section 6.10.2).

The second stage uses high-resolution ATAC data to prioritise variants in the supplied genetics, which are enriched for signal in the high-resolution ATAC-seq. The aim is to identify which variants in a given haplotype are in regions with the most evidence of being in TF-bound regions and, so, most likely causal.

4.2.1.4 Requirements for each stage

<u>For stage 1 with the Multi-Dimensional Viewer (MDV).</u>

1. Provide an embedded end-to-end -solution for analysing scATAC-seq data, including quality control metrics, to generate informative clusters.

2. Provide an interactive graphical interface for these clusters to understand the cellular composition of the data.
3. To generate gene activity scores (using the ArchR gene activity model) to aid the users in the identification of cellular clusters using known marker genes using dynamic visualisation of gene scores across clusters.
4. To use statistical testing to generate lists of genes that drive cluster formation to *a priori* identify genes important for cluster identity and visualise these dynamically across clusters to aid the identification of cellular clusters.
5. To provide dynamic visualisation of aggregated scATAC-seq from each cluster in an integrated genome browser to guide, at the level of the regulatory landscape, the difference between clusters.

<u>For stage 2 with the Multi-Locus Viewer (MLV).</u>

1. To develop a scoring method based on the enrichment of coverage over variants in the TF-enriched high-resolution ATAC fraction (<150 bp) relative to the nucleosome enriched fraction (>150 bp, see Chapter 2).
2. To develop interactive methods in the JavaScript front end to allow users to prioritise variants based on this score.
3. To develop appropriate genome-based views to allow users to dynamically inspect the prioritised variants in an integrated genome browser view in the context of both the high-resolution ATAC-seq (<150 bp) and nucleosomal signals (>150 bp).
4. To develop mechanisms to export prioritised variants and metadata for further downstream analysis.

4.2.2 A flexible and modular workflow

Avocato is meant to be an automated end-to-end reproducible pipeline for the analysis of scATAC-seq. However, I decided that it would be important for the 2 stages to be capable of running independently and analysing scATAC-seq data in the absence of genetics input (Stage 1) and reusing previous outputs of stage 1 to analyse different sets of genetic data in stage 2.

The first stage can therefore be run as a stand-alone module to interrogate cluster composition, annotate cell clusters and generate high-resolution ATAC-seq tracks for these clusters. We also thought it is important that stage 1 is capable of using previously aligned data to facilitate the use of merged data from separate scATAC-seq experiments (merged BAM file as input). A capability I make use of in this chapter in section 4.2.5.2 to test the cell cluster prioritisation steps of the pipeline.

The second stage is obviously dependent on the availability of a stage 1 output. Still, it can be reinitiated with a new VCF or BED input, which allows for the on-the-fly refinement of the input genetics, such as pre- and post-imputation of lead SNPs. A major use case for this functionality is, of course, the analysis of different sets of genetics which are considered likely to affect a similar cellular niche, such as different autoimmune genetics on a scATAC-seq dataset of immune cells. Both stages output independent and interactive visualisation of the analysis results. In the next section, I will explain each stage and its analysis steps in detail.

4.2.2.1 A detailed description of the currently implemented Avocato setup and workflow

Avocato uses Snakemake as a python-based workflow management system which is now a widely used and well-supported computational platform for the generation of automated analysis pipelines with reproducible results.

The Snakemake system controls the different analysis steps of each stage, both as independent workflows as well as linking the two workflows into a single analysis run. The first stage focuses on quality control and data management, followed by upstream processing of fragment files using ArchR as a processing engine, including applying the dimensionality reduction method and clustering algorithm and calculating gene activity scores. Prior to providing an interactive multidimensional view called MDV, it finds the most descriptive genes per cluster based on imputed gene activity scores using an iterative decision tree algorithm (Figure 4.1, Stage 1).

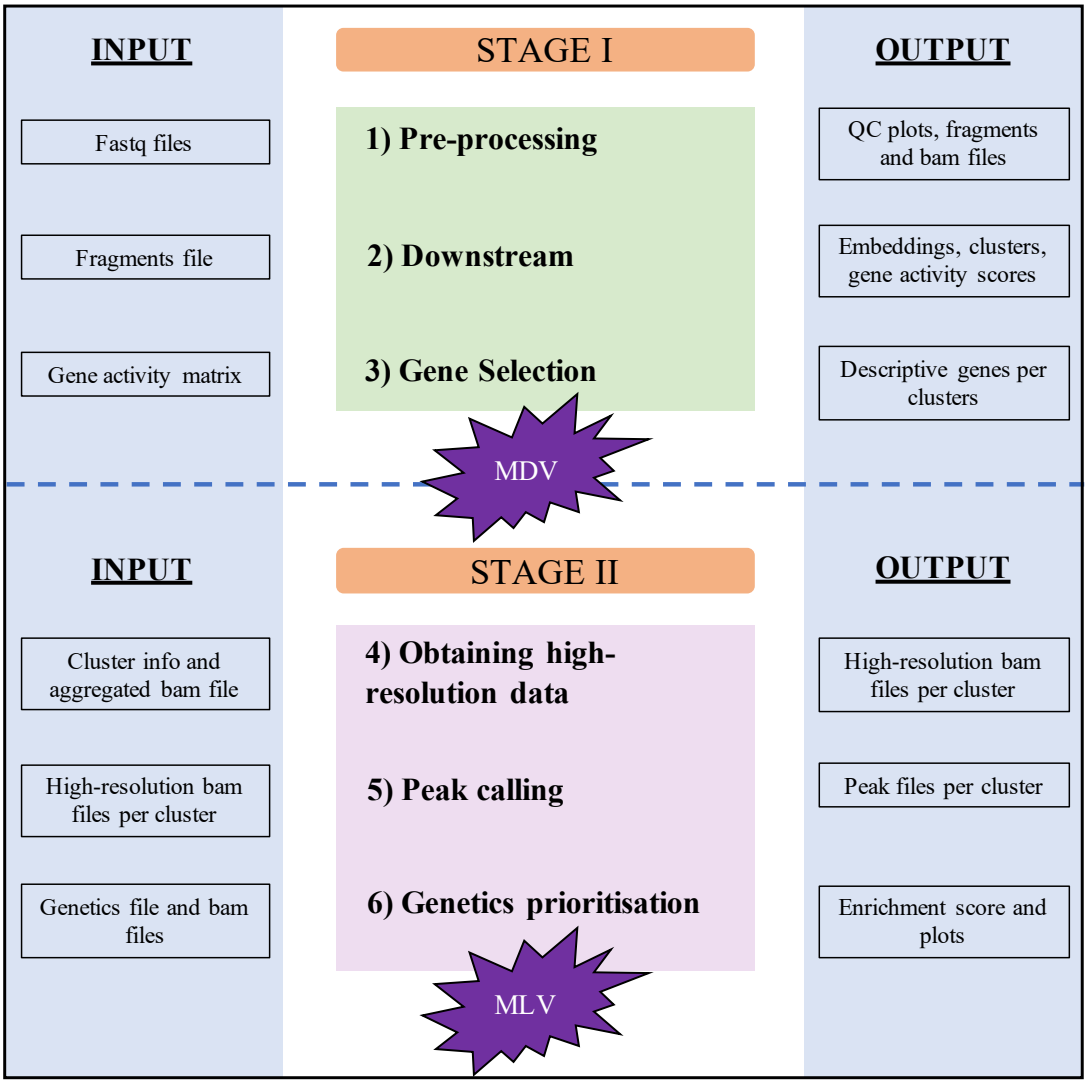


Figure 4.1: Avocato refers to the analysis and visualisation of single-cell ATAC-seq Observations, which consists of two stages.
Stage 1 includes the following processes; pre-processing the raw data, forming clusters and performing gene selection to find the most descriptive genes per cluster. Stage 2 contains specialised analysis steps to prioritise genetics variants using the high-resolution scATAC-seq data. Each stage produces an interactive visualisation session, MDV and MLV. Inputs to and outputs from each step are shown in blue rectangular regions.

Technical descriptions of the analysis steps for each stage can be found below.

Installation

The key goal for Avocato is to be as widely used as possible, including by experimental scientists. I realised that a major usage barrier would be the installation of the package itself. Considering all of the novel code and multiple codes and package dependencies

that Avocato requires, this would be a daunting task for even computer scientists. To overcome this, we decided to integrate all of the installation steps and setting up of the required computational environment for all stages of Avocato into a Conda package. Conda is perfect for this task as it is an open-source (and so freely available), cross-platform and language-agnostic package management system originally designed by Python data scientists to solve difficult package installation and management problems. This means that Avocato can be robustly installed simply by following the below instructions.

1. For cloning the Avocato repository from our GitHub page, run the below commands:

```
> git clone git@github.com:luntergroup/avocato.git
> cd avocato
```

2. To install the Avocato Conda environment, run the below commands:

```
> conda install -c conda-forge mamba
> mamba env create --file=envs/avocato.yml
> conda activate avocato
```

Set up of analysis run

The Snakemake workflow that runs Avocato is controlled using a standard input file written in the YAML markup language, which specifies key aspects of the workflow, including input files, output files and key user-driven parameters and their default settings. To again decrease usage barriers in terms of computation knowledge, we developed a user-friendly interface that allows for the generation of this initial input file

that requires no knowledge of the YAML language. To do this, we developed a local (driven through the local server or computer on which Avocato is installed) web page that allows users to specify all of the basic metadata, such as where the raw data is, whether the input is fastq or BAM, names for output and the ability to change basic parameters. This web interface then generates a correctly formatted YAML file with all of the user-supplied input included, which then serves as the input to the Snakemake pipeline. The layout of the YAML formatting page can be seen in Figure 4.2, and the standard inputs and editable parameters are described in the figure legend. The pipeline is subsequently initiated using this file with a simple command.

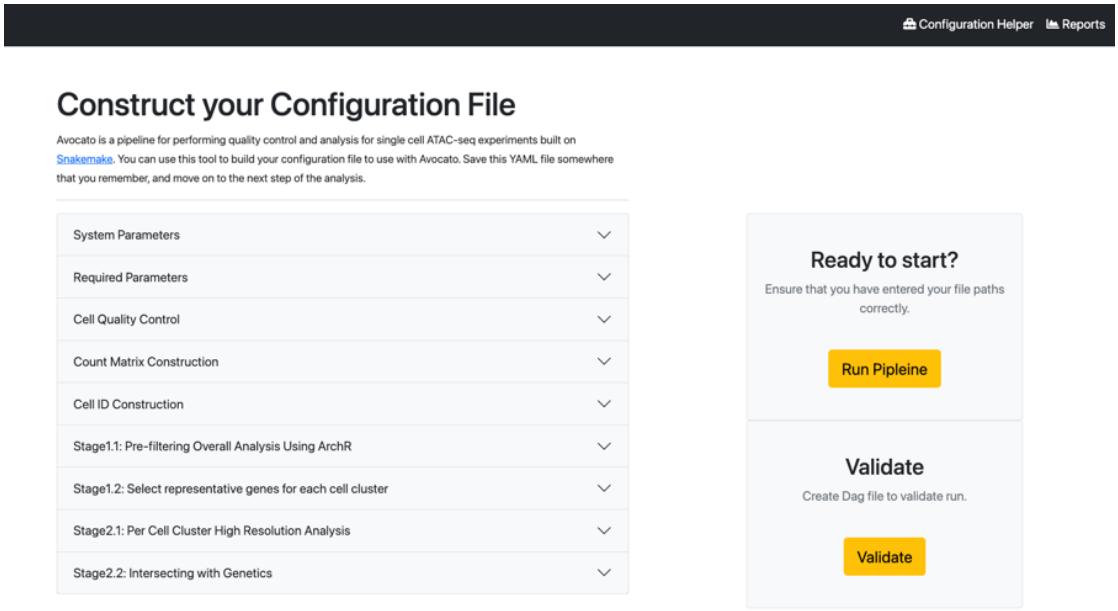


Figure 4.2: Configuration formatting page for Avocato analysis allows users to change pipeline parameters online easily.
Config parameters for each step can be accessed and changed by expanding each category (9 categories in total). Config files that users create can be validated by clicking validate icon on the right-hand side.

For running Avocato analysis, run the below command:

```
> snakemake --configfile=config/config.yaml --cores n (e.g., 4)
```

After the above command is run, Avocato now automatedly runs through all of the processes of stages 1 and then 2 (if a genetics file is provided), which are laid out in detail below.

Analysis steps for stage 1 are as follows:

1. Adaptors are removed from the raw sequencing files.
2. Raw reads are mapped to the reference genome.
3. An error-correction approach is applied to a BAM file to decide which cell barcodes are real cells. Error-corrected BAM file is then used as the input used in the rest of the analysis.
4. A fragments file is created from the error-corrected BAM file. Some QC plots are produced.
5. The fragment size method is applied to the error-corrected BAM file to split the file by fragment size.
6. Coverage files are generated for unfractionated BAM, TF-enriched BAM and nucleosome-enriched BAM files.
7. The ArchR method is applied.
   7.1. An arrow file is created, including the creation of a 500bp tile matrix and gene activity score matrix.
   7.2. Doublets are calculated
   7.3. An ArchRProject is created.
   7.4. Calculated doublets are removed from the project.
   7.5. An iterative LSI dimensionality reduction is applied to the tile matrix.
   7.6. UMAP embeddings are calculated.
   7.7. Graph clustering is applied.

7.8. Scores in the gene activity matrix are imputed using MAGIC.

7.9. The ArchRProject is saved for reuse by users if required.

8. An iterative decision tree method is applied to the imputed gene activity scores to find the most discriminative genes per cluster.
9. A multidimensional view (MDV) is created for interactive visualisation.

As described, Avocato can take two different forms of files as input: fastq files and BAM files. If the input is a BAM file, Avocato bypasses the pre-processing step and runs from the “downstream step” (Figure 4.1). After stage 1 is completed, stage 2 is initiated. Stage 2 generates high-resolution, cluster-specific scATAC-seq data, followed by generating bigwig files (coverage tracks) for genome browser visualisation. MACS2 or Lanceotron peak calling methods (the specific caller is a user-driven parameter) are then used to generate high-resolution peak calls for each cluster track. The final analysis of stage 2 is where genetics meets scATAC-seq data for fine-mapping using the high-resolution data. Additionally, it produces a heatmap of SNP enrichment per cluster (Section 4.2.5.2, Figures 4.15, 4.16, 4.17 and 4.18).

<u>Analysis steps for stage 2 are as follows:</u>

1. Aggregated high-resolution scATAC-seq data is split from the sized fractionated aggregated BAM file by cluster information coming from stage 1.
2. Genome coverage files are generated for cluster-specific high-resolution BAM files.
3. The specified peak calling method is applied.
4. SNP prioritisation in the high-resolution scATAC-seq data is performed.
5. MLV session is created for interactive visualisation.

4.2.3 Data visualisation as a central feature of Avocato

Considering the high intrinsic dimensionality of single-cell data, with the complexity of the output of each analysis stage and the scale of the genetics data used as input, the ability to facilitate effective human interactions with the output is critical to Avocato's use case. Central to this is the dynamic interaction and visualisation of the data at each stage. Below I have laid out the basic visualisation modes of each stage and will demonstrate their outputs with real data in the subsequent sections.

An important property to note in Avocato visualisations in stages 1 and 2 is that they are interactive, which means that either drawing around the features can select cells within any plot to select those cells or drawing a square, such as in a FACS gate and moving this around the data projection to select cells. The second point is that the plots are all interactively linked so that if data selection is performed on a particular plot, that selection is automatically reflected in all other displayed plots.

4.2.3.1 Stage 0

Basic QC plots informative in scATAC-seq data are loaded into the Web page that is used to generate the initial YAML file. These are now in the process of being integrated into the MDV viewer of stage 1. These include the size distribution of fragments produced in the experiment as well as read counts for cell barcodes which are basic metrics to assess the raw output of scATAC-seq experiments (see 4.11 and section 4.2.3.1 as examples).

4.2.3.2 Stage 1

The main functionality of this stage is to assess cluster structure after analysis and to guide cluster identification.

Figure 4.3: Overview of a multidimensional viewer (MDV) showing Stage 1 results
This is the default view for MDV. Left-panel contains 4 generic figures. These include: UMAP projections of single cells coloured by the imputed gene score for a chosen gene (NPRL3) are shown in 1. Graph 2 is the 3-D view of 2-D UMAP, coloured by cluster id. Descriptive genes per cluster and their gene activity scores are shown as a heatmap in 3. The bar chart in 4 shows clustering results along with the total number of cells in the clusters. In contrast, the right panel contains a list of human gene annotations and a genome browser view. The panels can be linked through a "plus sign icon" marked in a dotted red square.

At the end of stage 1, a summary file that contains 2D and 3D UMAP embeddings, clustering information, coverage tracks and imputed gene activity scores per single cell to be used as input to the web app is generated (Figure 4.3). Figure 4.3 displays the results of stage 1. On the left panel are 2-D (Figure 4.3.1) and 3-D (Figure 4.3.2) versions of UMAP projections followed by imputed gene scores per cluster for genes most descriptive in the gene selection step (Figure 4.3.3) and a row chart showing the total number of cells per cluster (Figure 4.3.4). The viewer allows interaction with

the data in 2-D and 3-D UMAP projection at the same time. 3D visualisation is effective as it explores adds an extra dimension to separate closely related cell clusters. Figure 4.4 shows zoomed versions of 2-D and 3-D UMAP embedding for scATAC-seq proerythroblast data coloured by cluster id. In 3D representation, the separation between the large cluster and the small cluster is clearer than in the 2D projection. All such graphs can be “popped out” as an independent window to make the most use of the screen space available and to rearrange graphs for better comparison.

A gene annotation list where properties of genes of interest (for example gene activity score) can be chosen on the right panel (Figure 4.3.5). These genes properties can be used to colour the cell clusters on the left panel by clicking the icon marked as red rectangular on the top-left-hand side. MDV not only colours 2D or 3D UMAP projection using that gene's imputed gene score. At the moment, selection can be made only using one gene, but we are still working on extending this so that users can choose a subset of genes. Figure 4.3.6 demonstrates a powerful dynamic property of the MDV browser in that it can dynamically generate genome coverage tracks using the current cluster annotation. This gives the ability to inspect the regulatory landscapes of known marker genes in the dataset based on the current cluster annotation to help determine the identity of the clusters. As this data is streamed from the underlying BAM file it can also be filtered dynamically by fragment size, shifting between the standard and high-resolution view of the ATAC-seq data at will. At the initiation of stage 2 these tracks will then be produced as individual BigWig coverage files suitable for viewing in genome browsers such as UCSC.

Available plots in MDV

1. Cluster structure in Avocato can be visualised by dimensionality reduction projection into either t-SNE or UMAP. Projections can be made in either standard 2D or interactive 3D. The importance of the 3D project is that it provides an extra dimension for human eyes to determine new cluster structures that would be masked in 2-Dimensions (Figure 4.4). As described, the plots are linked so that cells can be selected in either plot.

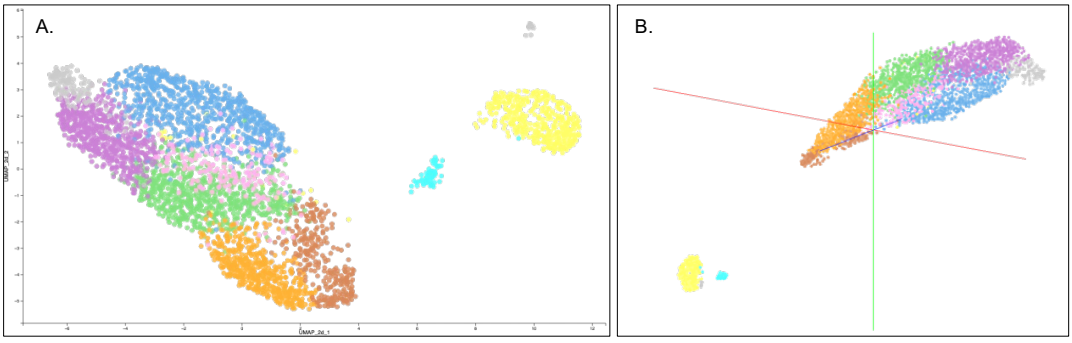


Figure 4.4: 3-D visualisation of data brings a different perspective to understand the data structure.
2-D UMAP projections of scATAC-seq proerythroblast is shown in A, whereas 3-D UMAP projections are shown in B. Cells in both graphs are coloured by cluster id.

2. Avocato calls potential clusters within the data using its internal cluster calling algorithms (see section 4.2.2.1). These clusters can be superimposed dynamically onto the 2D and 3D plots to understand the relationship between the statistically called clusters and the structure of the data (Figure 4.5).

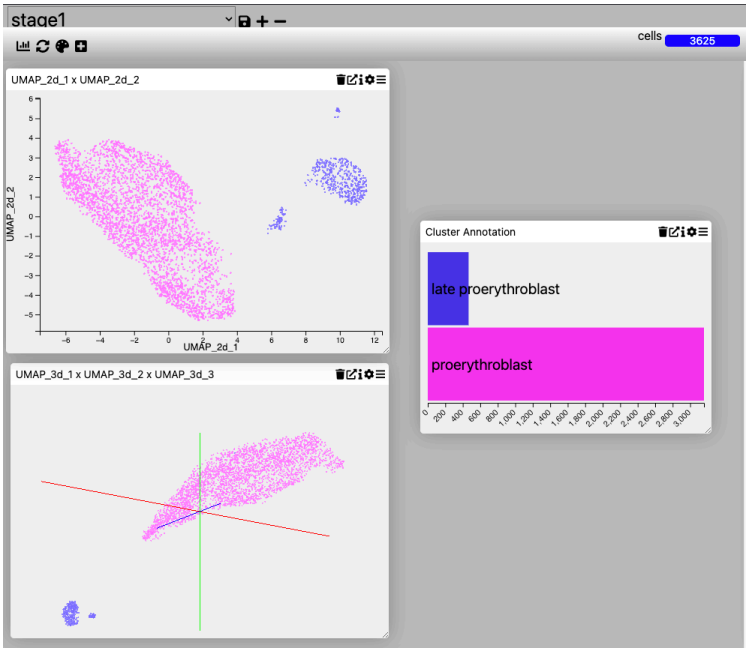


Figure 4.5: UMAP projections in 2-D and 3-D are coloured by cell type information.
2-D and 3-D UMAP projections of scATAC-seq proerythroblast are coloured by cluster annotation, seen as the bar chart on the right-hand side.

3. Avocato provides a wide range of QC plots and is not only limited to default plots. Users can create other plots in the current session by clicking the icon marked as a blue rectangle in Figure 4.6. Alternatively, they can create a new session, as shown in Figure 4.6, followed by creating new plots in the new session by clicking the icon marked as a red rectangle. Figure 4.6 created a new session called *'QC PLOT'*, and five quality control plots per cell were added, including a total number of fragments (Figure 4.6.1), promoter ratio (Figure 4.6.2), TSS enrichment scores (Figure 4.6.3), nucleosome ratio (Figure 4.6.4), and reads in TSS (Figure 4.6.5). There are a pie chart showing cell type annotation and bar chart showing cluster ids (Figure 4.6.6 and 4.6.7). The icon marked as a yellow rectangle shows the number of cells, in this case it shows the total number of cells since there is no selection applied. However, Figure 4.7 displays 467 cells on the same place when late proerythroblast was

selected in chart 6 from Figure 4.6. As explained before, any selection affects and update all the plots added in the current session.

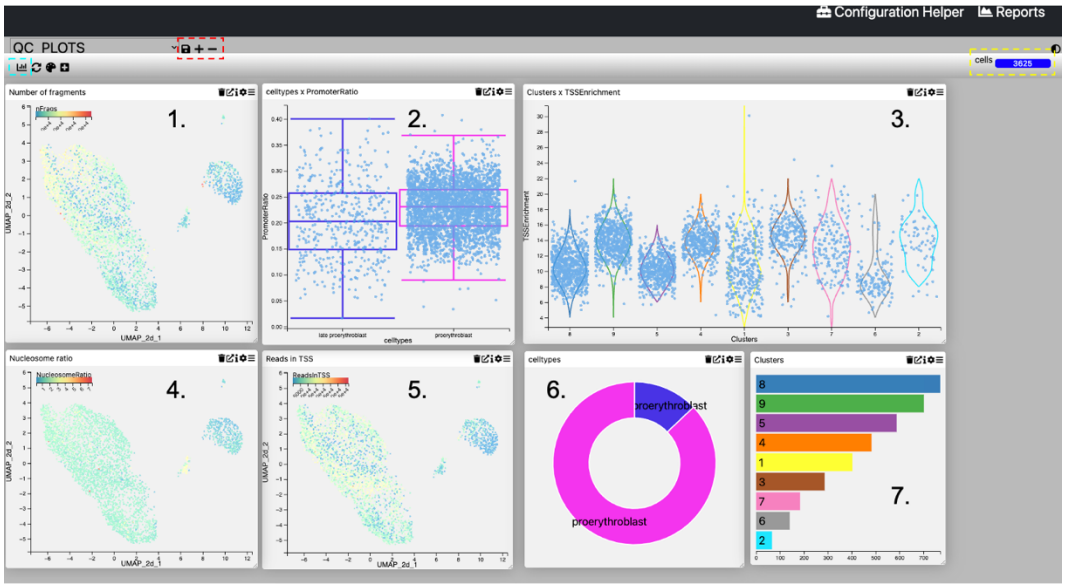


Figure 4.6: Avocato is capable of generating different kinds of metadata in the current session.
This session shows QC plots over cells for proerythroblast scATAC-seq data. 2-D UMAP projections of cells coloured by different metrics in 1. the number of fragments, 2. promoter ratio, 3. TSS enrichment scores, 4. nucleosome ratio, 5. reads in TSS. The pie chart in 6 shows different cell populations, whereas the bar chart in 7 shows clustering results. A new session can be created or removed via the panel marked as red-dotted rectangular. The graph icon below the panel, which is marked as blue-dotted rectangular, is used to create a new plot in the current session. Additionally, the total number of cells of the related scATAC-seq data can be seen on the top right-hand side position marked as dotted yellow rectangular.

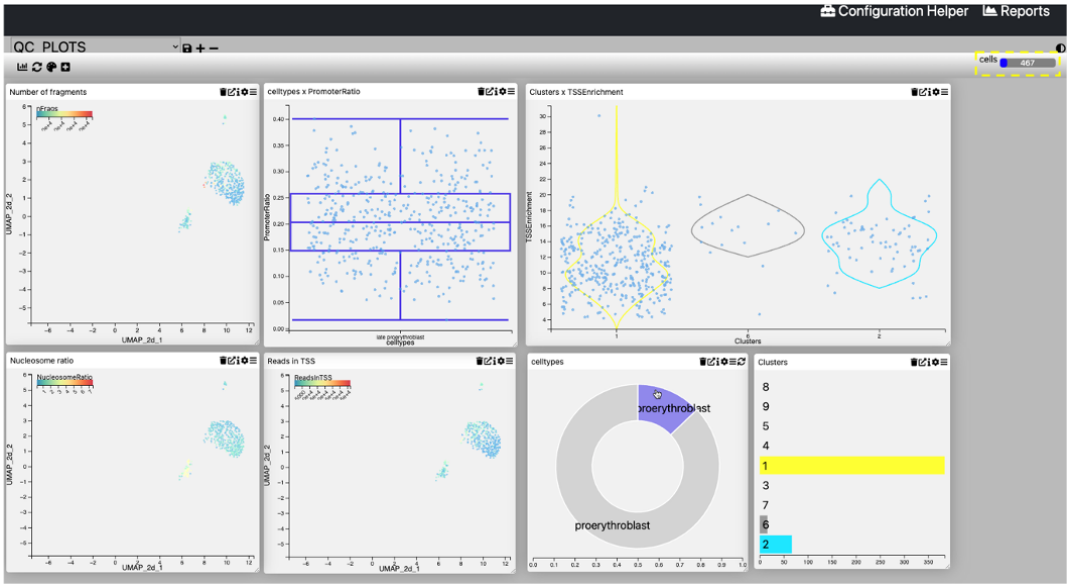


Figure 4.7: Avocato is capable of showing the effect of the selection which is made in one plot on the others.
This view shows the same view in Figure 4.6 except with one difference. Only cells from the late proerythroblast population are selected and this selection updates other plots.

4.2.3.3 Stage 2

Stage 2 enables users to explore their GWAS associations interactively on TF-enriched and nucleosomal-enriched scATAC-seq data. Figure 4.8 represents how the multi-locus view (MLV) works after stage 2 is completed. The list of genetic variants can be accessed on the right panel in Figure 4.8. Read counts for those variants from various scATAC-seq providing different levels of information can also be viewed.

Different graphs can be plotted by clicking the 'Add Chart' icon on the top right-hand side. Figure 4.9 is one of the examples of possible graphs that can be plotted. Here, the y-axis represents read coverage coming from nucleosomal-enriched scATAC-seq (> 150 bp, large), whereas the x-axis refers to read coverage from TF-enriched scATAC-seq (< 150 bp, small). This scatter plot is very useful because using high-resolution scATAC-seq provides precisely TF-enriched open chromatin regions by removing nucleosomal background, so SNPs can be filtered by making a selection on the plot like in Figure 4.10 based on the ratio of TF-enriched and nucleosomal-enriched data. I expect coverage from high-resolution data to be higher than nucleosomal-enriched data, so the example selection in Figure 4.9 works well.

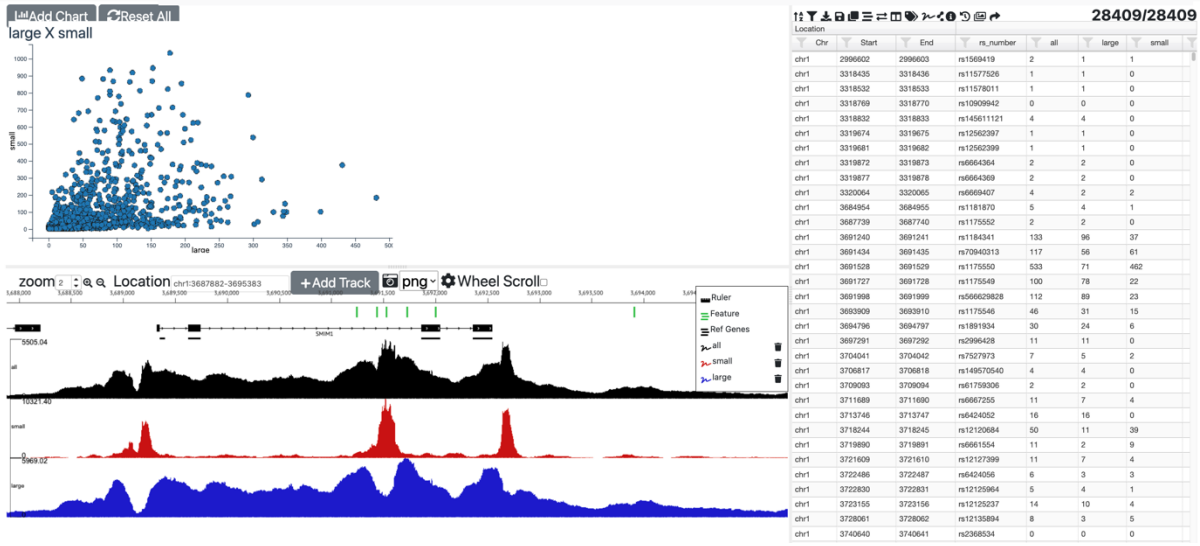


Figure 4.8: Overview of an MLV showing stage 2 results
MLV is a region-based visualisation interface allowing users to interact with their genetic data. On the right panel, genetic data is shown in a table. In contrast, different charts can be added on the top left panel. As an example, there is a scatter plot, each dot representing SNP from the table. These SNPs are shown as green blocks on the bottom of the left panel along with the standard scATAC-seq data in the black track, the high-resolution data in the red track, and the nucleosomal enrich data in the blue track. Related MLV session can be accessed through the link. https://mlv.molbiol.ox.ac.uk/projects/multi_locus_view/5884

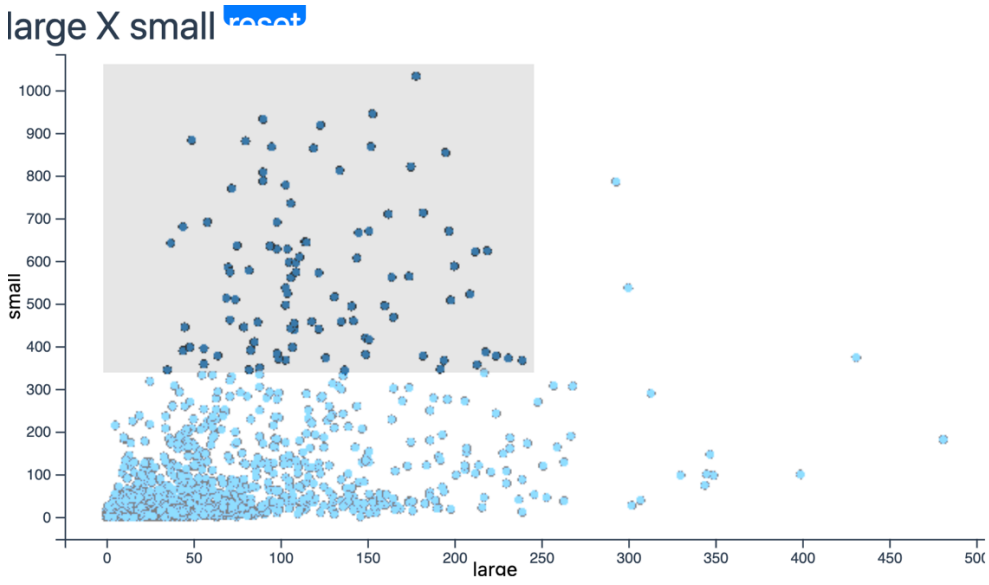


Figure 4.9: The scatter plot can easily filter SNPs with high read coverage from TF-enriched data.
The rectangular selection shows SNP having higher read coverage in the high-resolution data than nucleosomal-enriched data. The x-axis shows read coverage from nucleosomal-enriched data, while the y-axis display read coverage from the TF-enriched data. Each dot represents SNP from the table.

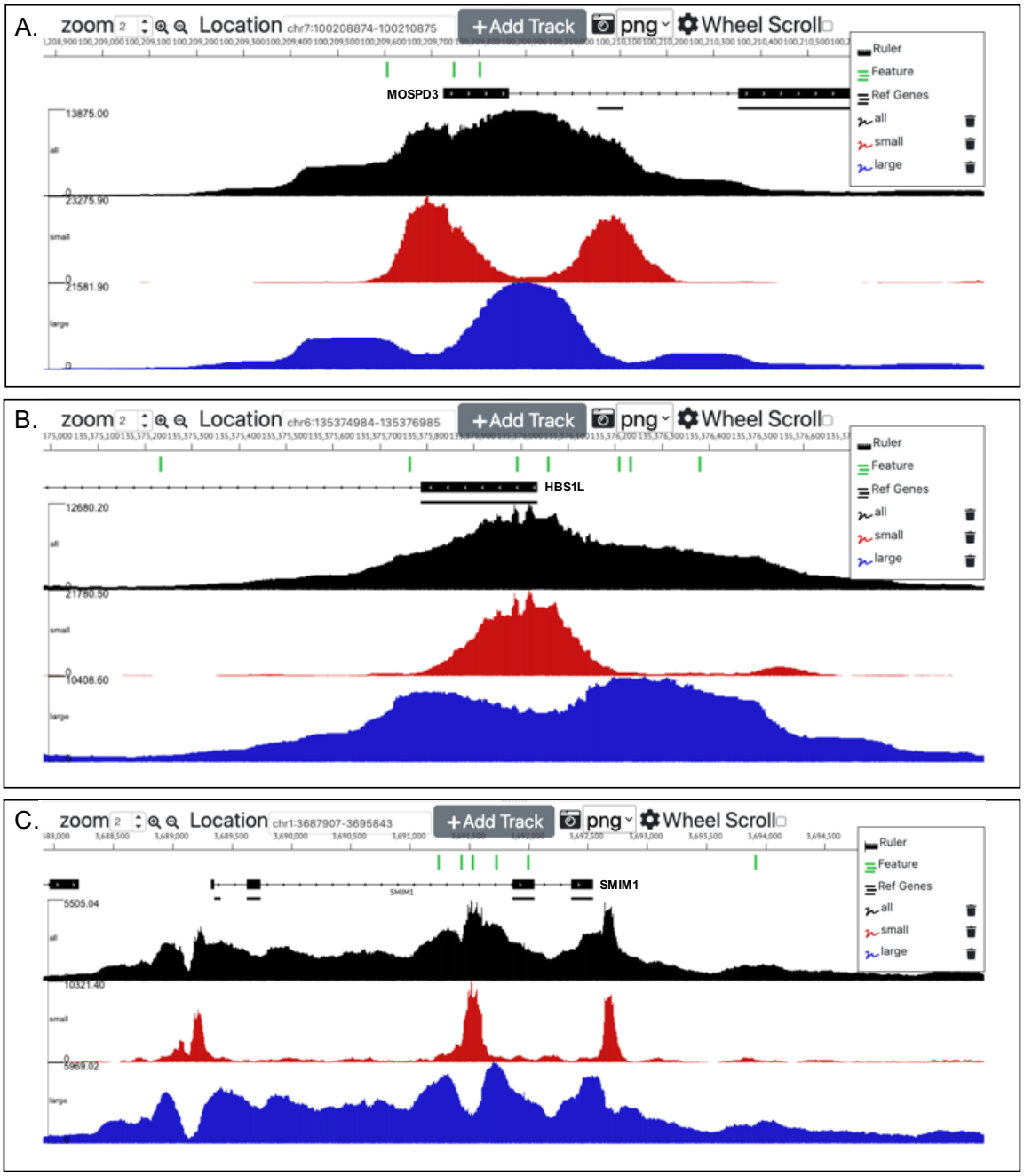


Figure 4.10: Three examples show the power of our prioritisation method to identify SNPs that affect regulatory elements.
A. SNPs are shown as green blocks at the bottom of the figure, along with the standard scATAC-seq data in the black track, the high-resolution data in the red track, and the nucleosomal enrich data in the blue track at the MOSPD3 locus. B. SNPs are shown as green blocks at the bottom of the figure, along with the standard scATAC-seq data in the black track, the high-resolution data in the red track, and the nucleosomal enrich data in the blue track at the HBS1L locus. C. B. SNPs are shown as green blocks at the bottom of the figure, along with the standard scATAC-seq data in the black track, the high-resolution data in the red track, and the nucleosomal enrich data in the blue track at the SMIM1 locus.

As explained in detail in the previous chapter, using high-resolution scATAC-seq data helps precisely identify the TF binding regions within open chromatin peaks that are overlapped with genetic variants. This example of stage 2 results shows SNP prioritisation using red blood cell traits on proerythroblast scATAC-seq data. The MOSPD3 locus in Figure 4.10 A has three SNPs shown as green blocks. When using standard scATAC-seq data (black track), these SNPs are interpreted similarly as they are affecting the promoter of the gene MOSPD3. However, high-resolution scATAC-seq can clearly deprioritise the first SNP as it does intersect with a high-resolution peak. Additionally, the second SNP can be seen to intersect the precisely defined promoter peak of the gene. Finally, the last SNP also intersects the same promoter peak but has a lower chromatin accessibility signal than the second one. It is worth noting that standard scATAC-seq data shows one broad peak for the upstream of the gene, whereas high-resolution data separate that broad peak into two different regulatory elements, one being the promoter.

A similar scenario is present in Figure 4.10 B, six SNPs (green blocks) intersect the promoter of HBS1L at low resolution. At high-resolution, there are only two SNPs intersecting (see high-resolution data in the red track). The rest of the SNPs fall into nucleosomal-enriched regions, as judged by the high signal in the nucleosomal track (blue track).

The final example locus, SMIM1, has 5 SNPs (green blocks) over the gene body of the SMIM1 gene. With standard scATAC-seq, the interpretation for these SNPs would have the same conclusions as they affect a likely enhancer region in the gene SMIM1. At high-resolution, only the second and third SNPs intersect the TF bound region of

this enhancer in the SMIM1 gene (red track) while the others intersect with nucleosomal enriched regions (blue track).

4.2.4 Testing Avocato outputs using scATAC-seq proerythroblast data.

In this section, I used scATAC-seq proerythroblast data as a control to check the correct processing of the data and output by Avocato. I have analysed the same data using many different platforms (Figure 2.13), so the structure of this highly homogeneous structure is well understood (Chapter 2).

I modified only the following config parameters to run Avocato as follows:

- `the aligner = bwamem2`;
- `the peak caller = Lanceotron`;
- `Fragment size cut-off = 150bp` (see Chapter 3).

I used as test genetics the Astle red blood GWAS traits [79] as these are the most relevant genetics for this cell type taken into account. It must be noted that as this is a highly homogeneous erythroid population, the cell type prioritisation of cell type will not work due to the lack of an appropriate background. However, it works as a technical control for the performance of the pipeline and for the prioritisation of SNPs in stage 2. I will reuse this dataset in the context of an appropriate background in the latter section of this chapter to demonstrate the cell type prioritisation function of Avocato (see section 4.2.5.2).

A successful Avocato analysis should generate the output files for 5 main analysis steps: (1) quality control and data management; (2) forming clusters; (3) transitioning to high-resolution data; (4) the non-coding genetic variant intersection with the landscape of chromatin accessibility profiles; and (5) interactive visualisation.

4.2.4.1 QC and data management

Avocato produces adaptor-free sequence files prior to aligning the raw data to the reference genome. It applies a cell barcode correction algorithm to discard non-biological cells and keeps error-corrected and eliminated barcodes in the *03_cellids folder*. The final position sorted BAM file, and its index file can be accessed in the *05_bam folder*. Avocato also generates a fragment file and its index to be used as input to ArchR. As stated, these preliminary quality control graphs plot the distribution of DNA fragments and read coverage for every single cell. Figure 4.11 shows quality graphs for proerythroblast data. The fragment size distribution matches the pattern I demonstrated in Figure 2.13 using cisTopic, ArchR and SnapATAC2 tools to analyse the same data, with comparable read count distribution per barcode. This shows that the data alignment, feature extraction and barcode correction in Avocato are working correctly.

As a final analysis of this section, Avocato creates an RPKM normalised coverage track for the unfractionated BAM file to be uploaded onto UCSC. Which represents the standard scATAC-seq track of aggregated data.

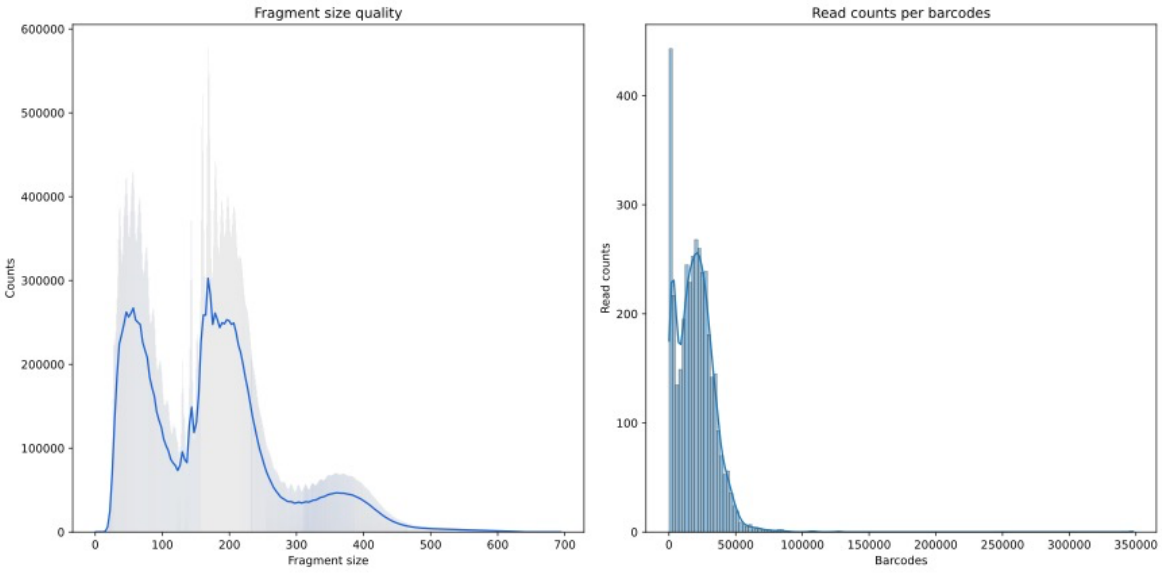


Figure 4.11: Avocato QC plots have a similar pattern, as seen in Figure 3.2.
The distributions of DNA fragments and read count per cell barcode are shown in A and B, in order.

The same quality control graphs on scATAC-seq PBMCs data can be found in supplementary Figure 6. The QC plots for PBMC as a heterogeneous population also have the same pattern as proerythroblast (homogenous population), showing consistency of performance across samples.

4.2.4.2 Calling clusters

After generating the fragment file, ArchR uses it to create a 500bp tile matrix as a definition of accessible chromatin regions. It uses the iterative LSI method to reduce the high dimension of the matrix, followed by applying graph-based clustering to group cells. Even if the sample is a homogeneous population, the clustering algorithms will generate more than 1 cluster. These clusters are sometimes the products of technical over-clustering due to the clustering method or parameters used. Forming clustering is a very challenging and complex task as there is no one-fit-all method to apply. Clustering algorithms differ based on the technical nature of the data, data type,

sample complexity and hypothesis. Therefore, clustering results need to be evaluated carefully. There is a couple of practical ways to investigate and refine scATAC-seq clustering result.

First, if the cell types are well characterised with known marker genes that distinguish specific clusters, then the gene activity scores calculated by Avocato for these genes can be sequentially superimposed dynamically onto the cluster plots to assigned cell types. How to do this in the absence of such data was one of the key reasons for developing in Avocato the statistical approaches (for iterative decision tree algorithm, see section 6.10.1) to identify the list of loci which are important for determining cluster identity. These can act as *a priori* marker genes to decide whether to retain or merge clusters.

Secondly, cluster-specific coverage and peak files can be visualised in the UCSC to compare subtle differences between clusters at the level of the regulatory landscape. This, again, can be guided by knowledge of known marker genes or *a priori-determined* set of gene loci that drive cluster formation. The inspection of the regulatory landscapes in a genome browser view over these key loci can guide the operator as to whether there are convincing enough differences to keep the clusters distinct or to merge them into a single cluster.

Lastly, clusters from the scRNA-seq experiment for the same sample can be transferred to the scATAC-seq analysis result (see Chapter 2, section 2.2.6.2, Figure 2.17). At present, such analyses, including the use of tools such as Azimuth, are performed outside of Avocato. However, the database behind MDV is designed so

that external data can be easily merged into it and viewed in the context of the other analysis. However, this is, at present, still a more computationally demanding step than just running the pipeline. I will demonstrate some of these approaches from the Avocato analysis using the erythroid dataset.

As shown previously (Chapter 2, Figure 2.6), this scATAC-seq dataset contains two main biological clusters: proerythroblast and late-proerythroblast. There is, however, a lack of known marker genes for these closely related late stages of differentiation. Therefore, I uploaded coverage tracks of the small and large clusters from Figure 2.6 into USCS to investigate signals at well-characterised loci. Figure 4.12 shows UCSC tracks at the alpha-globin locus. Analysis in section 2.2.2 revealed that the distal peaks were greatly diminished in the smaller cluster. Figure 4.12 supports this finding as chromatin accessibility of regulatory elements at the alpha-globin locus decreased relative to the promoter signals, and that effect can be seen genome-wide. This would support maintaining the two clusters as distinct based on the real detectable difference in their regulatory landscapes.

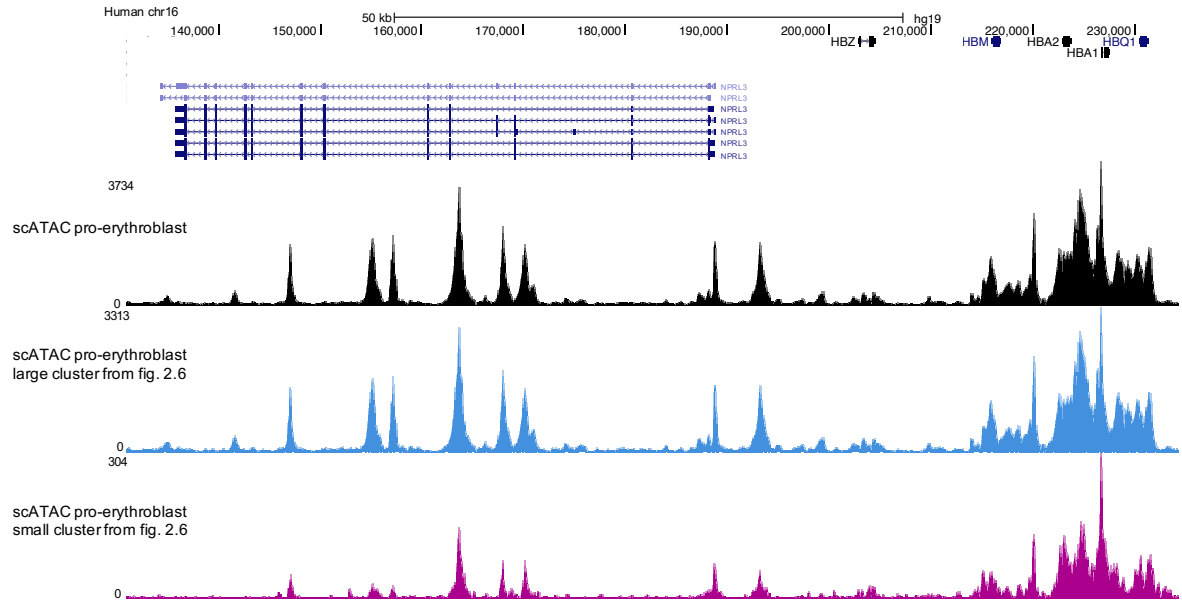


Figure 4.12: Coverage tracks of two clusters of scATAC-seq proerythroblast data from Figure 2.6 indicate that in the small cluster, there is a decrease in chromatin accessibility of regulatory elements at the alpha-globin locus, supporting the result of the clustering method based on biological differences.
The scATAC-seq proerythroblast data is shown on the top black track. The large cluster is on the middle blue track, whereas the small cluster is on the bottom of the figure in the purple track.

One of the more complicated decisions to make in deciding to maintain clusters as distinct or merge them is whether to include cell cycle clusters. This is very much dependent on the question being asked, but it does demonstrate the power of integrating scRNA-seq data as an extra layer into the Avocato visualisation platform. scRNA-seq for the same sample was generated and analysed as part of Truch *et al.* [69]. Figure 4.13 shows the same clustering pattern in Figure 2.6 in these scRNA-seq results. However, the cells in these clusters are now coloured in MDV as to their position in the cell cycle via enrichment for sets of known cell cycle genes. This shows that even in a very homogenous cluster structure can be driven by the cell cycle but also the facility with which additional external analysis can be integrated into Avocato output to guide cluster annotation. For evaluating scATAC-seq PBMC clustering results, see section 2.2.6.

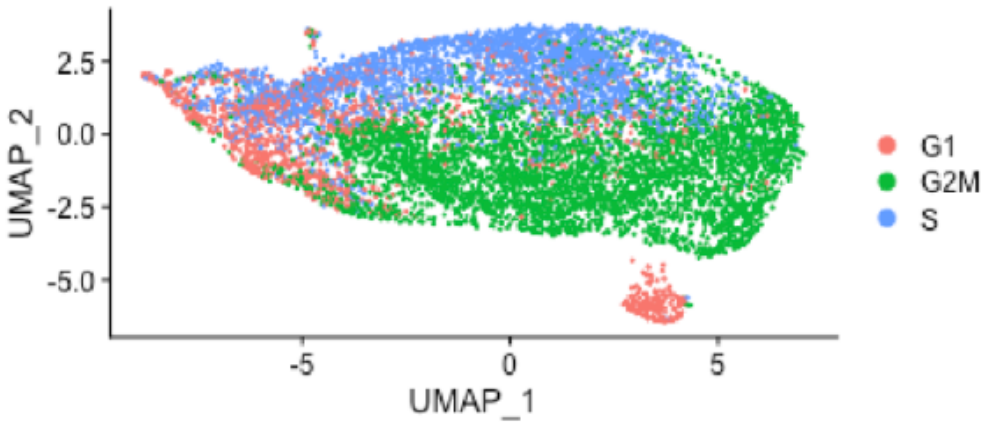


Figure 4.13: UMAP analysis of scRNA-seq on the same sample, proerythroblast, displays that the clustering method formed clusters based on cell cycle stages.
Each dot represents individual single cells coloured by the gene expression of cell cycle genes. The figure was taken from supplementary Figure 16 from Truch *et al.* [69]

4.2.4.3 Transitioning to high-resolution data

Avocato generates TF-enriched position-sorted BAM and nucleosome-enriched position-sorted BAM files as well as high-resolution cluster-specific BAM files. It uses high-resolution cluster-specific files to perform peak calling using Lanceotron. High-resolution coverage files can be uploaded to the genome browsers.

4.2.5 Testing Avocato cellular prioritisation of genetics using merged data from proerythroblasts and PBMCs.

The most logistically important choice in the functional dissection of genetic variants is to determine which cell type or types is the most likely effector cell type. As previously discussed in the field, statistical enrichment in the intersection between genetic variants and regulatory signals, such as open chromatin, has been used to prioritise cell types [77], [78]. I felt this was a critical function to build into Avocato, utilising not only the complex mixture of cell types typically found in scATAC-seq

experiments to form a coherent background for the statistical analysis, but also the high-resolution intersection between variants and TF-bound regions to increase the signal to noise in this analysis potentially. To demonstrate the utility of this feature, I have constructed a ground truth dataset scATAC-seq from a cell population (proerythroblast) which is known to be strongly linked to a set of a comprehensive set of genetic traits (red blood traits). To form a credible background, I used Avocato's ability to work from BAM files to merge the proerythroblast data into a scATAC-seq dataset of PBMC, which has no known link to these traits. This test case, therefore, consists of running Avocato with this merged scATAC data set, with red cell trait as the genetics input (Astle *et al.* [79]) and see if it can convincingly pick out the red cell clusters as the most likely effector cell and so validate this approach.

4.2.5.1 Basis of the statistical testing for the identification of likely effector cell types

As stated previously, the theoretical basis of this approach is that if a cell type is a likely effector cell type for the genetics in question, there should be a great chance of an intersection between variants and regulatory elements. It is, therefore, critically important to generate an appropriate background set against which to judge enrichment statistically. Obviously, scATAC is intrinsically suited to this as it determines the regulatory landscapes of multiple cell types in a single experiment and this facility is increased in Avocato due to its facility in merging data from multiple experiments to increase the complexity of the background (see section 6.10.2 for the mathematical description of how the statistical test works).

4.2.5.2 Testing of cell type enrichment analysis.

In addition to using red cell traits as a control to show the ability to identify the appropriate cell types in our merged erythroid and PBMC scATAC-seq dataset, I will also test immune traits on the same merged set, in this case, to highlight enriched immune cells. The list of traits in this analysis is laid out in the following table.

Table 4.1: A list of genetic variants and scATAC-seq dataset used to test the efficacy of statistical SNP enrichment method (link to GWAS studies see section 6.11.3)

Genetic variants	scATAC-seq data	Number of lead SNPs
Multiple sclerosis	PBMCs + proerythroblasts	436
Type-I diabetes	PBMCs + proerythroblasts	255
Type-II diabetes	PBMCs + proerythroblasts	3643
Astle only red blood traits	PBMCs + proerythroblasts	2246
Astle only red blood traits	proerythroblasts	2246

Figure 4.14 displays the UMAP projection of the merged scATAC-seq dataset, showing Avocato successfully separated the erythroid cells from the PBMC cell types, while maintaining their individual cluster characteristics. In this UMAP Each dot represents a single cell; clusters circled in black belong to PBMC, while clusters (0,1,2,3) circled in dotted black belong to proerythroblast.

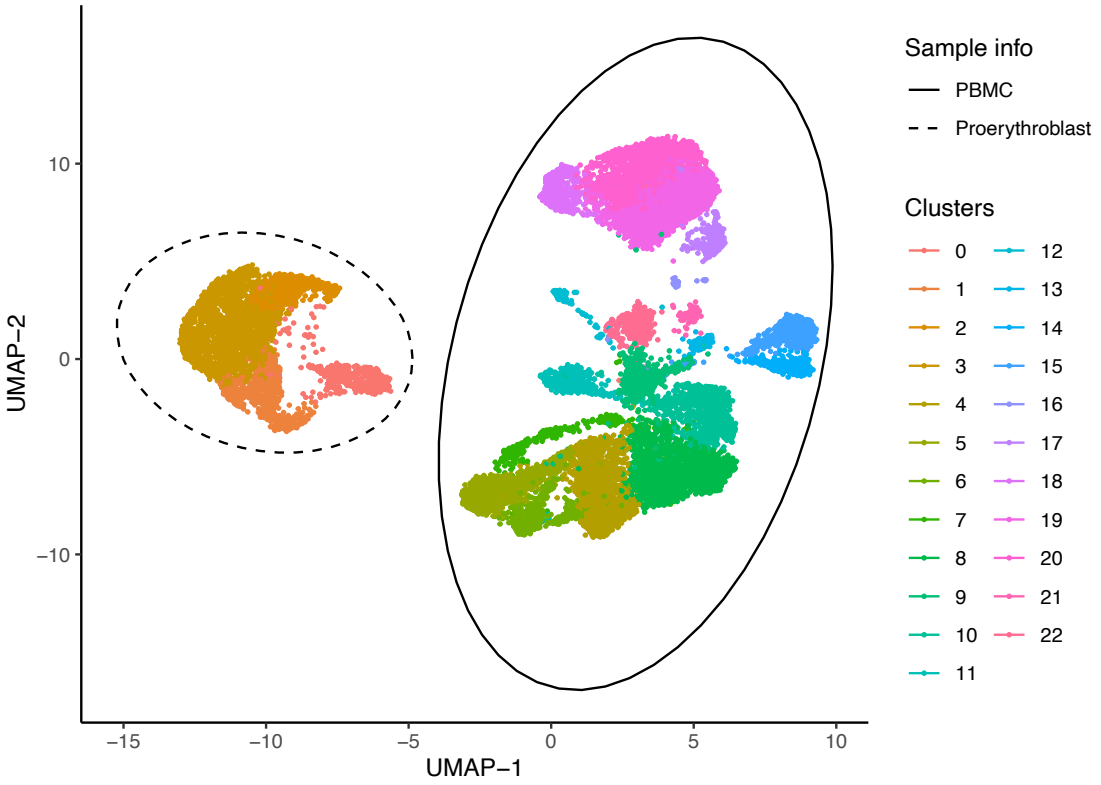


Figure 4.14: Avocato is capable of distinguishing red cells and immune cells.
Avocato Stage 1 result on in-silico merged scATAC-seq data (PBMC circled black and proerythroblast circled dotted black) is shown as UMAP projections, each dot representing single cells coloured by clustering id.

Figure 4.15 shows the statistical enrichment scores as a heat map for all of the clusters shown in Figure 4.15. It can be seen that, by far, the most statistically enriched clusters are clusters 1, 2 and 3, which represent the erythroid clusters in this merged dataset. Interestingly, cluster 0, which represents the very late pyknotic erythroid cluster, is not really enriched for intersection for erythroid traits. This controlled experiment shows clearly that this statistical approach can identify the most likely effector cell type from a complex mixture of cells and can even sub-divide highly related cell clusters such as erythroid cluster 0 in the context of the erythroid clusters 2, 3 and 4. The number of cells in each cluster under Figure 4.15 points to a current limitation of using scATAC-seq data over intersections with bulk data in these types of analyse. For the statistical

enrichment to work, the clusters have to have enough cells to effectively map the regulatory landscape for the genetics to be intersected with.

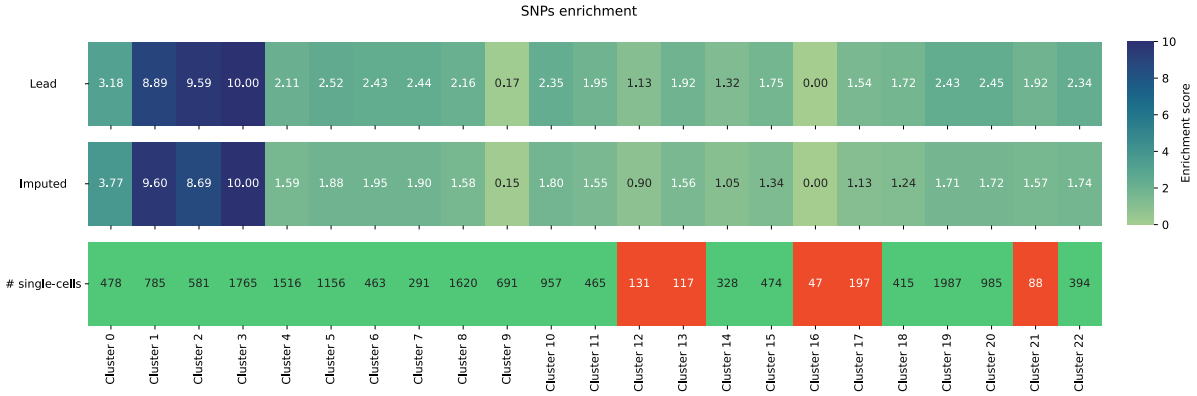


Figure 4.15: The statistical enrichment scores suggest that the three erythroid (1-3) clusters should be focused on prioritised based on red blood cell traits.
The statistical enrichment scores are shown as a heat map for all of the clusters, the first belonging to the lead SNPs, whereas the second is imputed SNPs. The last heatmap shows the total number of cells per cluster. Clusters to be excluded due to lack of sufficient cell are coloured red.

Apart from using red blood cell traits as genetic variants on the in-silico merged scATAC-seq data, I assessed SNP enrichment scores across the same clusters for other genetic variants, such as type-I diabetes (T1D) and multiple sclerosis (MS). The reason for using these genetics is that those variants will more likely affect immune cells and it would we would expect to observe high enrichment scores in PBMC clusters. Figure 4.16 represents higher SNP enrichment scores on T1D for clusters 8, 7, 5, 4, and 6 (the highest value to the lowest), which belong to PBMC data, while lower scores were observed in proerythroblast clusters. Furthermore, in Figure 4.17, multiple sclerosis GWAS associations were used to find the biggest target on the clusters of the merged scATAC-seq data by taking into SNP enrichment results account. The same high enrichment pattern was observed here, as in Figure 4.16.

Clusters from red blood cells have very low SNP enrichment scores, whereas PBMC clusters have high scores suggesting my method works well.

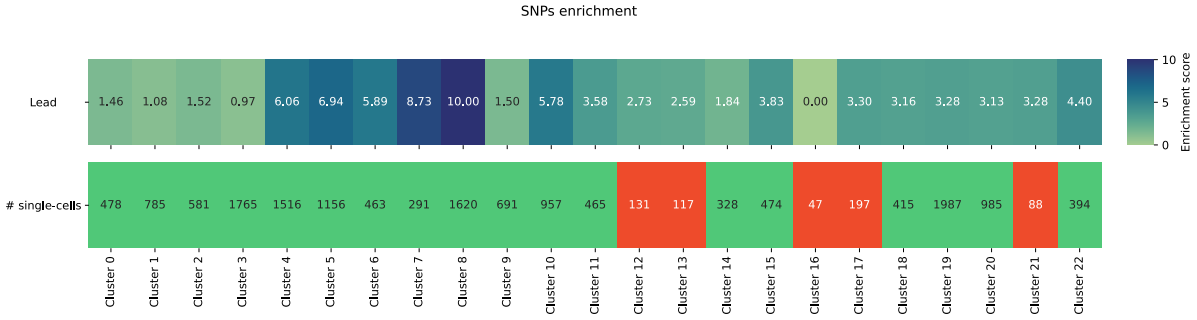


Figure 4.16: The statistical enrichment scores suggest that clusters 8, 7, 5, 4 and 6 should be focused on prioritising based on type-I diabetes.
Those enriched clusters are from the PBMC dataset. The statistical enrichment scores are shown as a heat map for all clusters. The last heatmap shows the total number of cells per cluster.

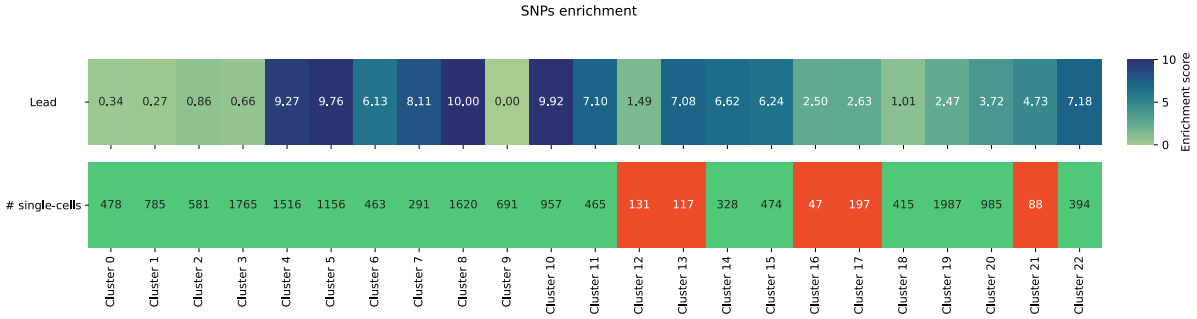


Figure 4.17: The statistical enrichment scores suggest that clusters of immune *niches* should be focused on prioritising based on MS.
Those clusters are clusters 8, 10, 5, 4, 7, 22, 11 and 6. The statistical enrichment scores are shown as a heat map for all clusters. The last heatmap shows the total number of cells per cluster.

Next, I tested the SNP prioritisation method using a different set of genetic variants that are not expected to be relevant to either red cell or immune biology. For this purpose, I used GWAS SNPs from a Type-II diabetes (T2D) GWAS. Since T2D is mostly thought to be related to pancreatic function, I expected SNP enrichment scores should be almost zero or at least of a similar value across clusters.

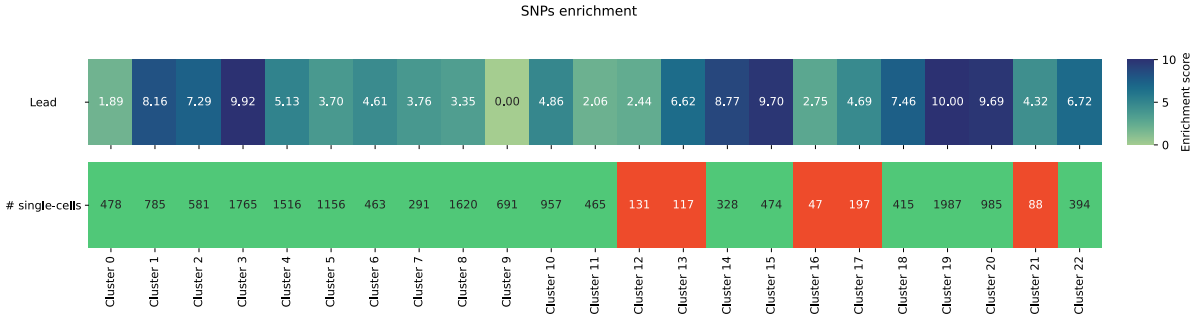


Figure 4.18: The statistical enrichment scores suggest that there are no statistically enriched clusters to be focused on prioritising based on type-II diabetes.
The statistical enrichment scores are shown as a heat map for all clusters. The last heatmap shows the total number of cells per cluster.

Figure 4.18 displays the different levels of T2D SNP enrichment scores across all clusters in the merged erythroid-PBMC merged dataset. The output shows relatively high scores across both erythroid and immune clusters and shows a general limitation of these statistical models. Like all scores for statistical enrichment, having an appropriate background to estimate true enrichment is critical. The analysis of the red cell and immune traits works well as each merged population is different enough to form an appropriate background for each other. In the case of T2D genetics, the appropriate cell type was missing as no matched 10X dataset for the pancreas is currently available. In practice, this could lead to confusion when genetics are tested against a population of cell types that have no relationship to genetics. However, recently scATAC-seq data from 222 human cell types from both adult and fetal tissues has been released [75]. I will, in future work, use this to generate a more general background. This background can be used consistently to normalise the degree of statistical intersection for any analysis and so scale the degree of statistical enrichment consistently across datasets in these analyses and so overcome this problem.

4.3 Discussion

There are many challenging factors in understanding non-coding genetic variants with the integration of epigenomics. In the previous chapters, I have shown that scATAC-seq as a technology has real potential to aid the prioritisation of genetics that affect gene regulatory elements. The potential for its use will only increase as its unique capabilities are used to generate data from more and harder access human cell types, including stages of cellular differentiation and human development. All these data will be able to use in the high-resolution approach that I have developed as part of my thesis.

However, the depth and complexity of the data combined with the complexity of its analysis generate major barriers to its general use beyond specialist laboratories. In this chapter, I set out to generate a platform to break down these barriers. The main focus was to generate an easy-to-use but the complete pipeline for the analysis of scATAC-seq that incorporates the additional advances of our high-resolution analysis.

Therefore, Avocato can produce generalisable and robust scATAC-seq analysis results along with the functionality of performing SNP prioritisation in high-resolution cluster-specific data. We leverage the high-resolution maps to perform the critical step of prioritising both cell types as well as causal SNPs to help develop hypotheses and guide the design of downstream experimental analysis. One of the major features of Avocato lies in the development of user-friendly interactive multi-dimensional and multi-locus viewers to interrogate the multiple outputs of the two stages of analysis. This is because we felt the ability of humans to interact with the analysis data was a

critical component of what was needed to break down the barriers to the use of these approaches and data.

The development of Avocato is still in progress. At the moment, we are working on MDV and MLV, refining the most important “use cases” and improving aspects such as a more generalisable statistical test for the cluster-specific SNP enrichment algorithm.

5 Discussion

5.1 Overview

Understanding non-coding genetic variants in functional genomic studies play a crucial role in understanding disease biology and therefore exploiting genetics to guide the development of drug targets. Three aspects need to be understood when interpreting these variants from GWAS associations. Firstly, in GWAS, it is hard to identify the causative SNPs among many thousands of genetic variants in linkage in these haplotypes. Secondly, which cell type(s) they are functioning in is difficult to determine as these germline variants are found in every cell type. The final challenge is determining which genes are affected by these SNPs as they affect regulatory elements which are difficult to link to genes without extensive experimental analysis. Taken together, these challenges make it very difficult to interpret GWAS associations in terms of the biological mechanisms and pathways underlying these complex diseases.

The work in my thesis concentrates on the potential of scATAC-seq data to help functionally prioritise causal variants and cell types, which I briefly summarise below. Subsequently, in the remainder of this discussion, I would like to explore its use in combination with other technologies, both computational and experimental, to help refine its prioritisation and solve subsequent challenges.

The first thesis Chapter acts as a foundation for understanding scATAC-seq technology widely from different aspects. These include the different experimental

methods to generate the data and testing the different computational approaches to perform the necessary analytical steps of dimensionality reduction, clustering and cluster annotation.

In the second Chapter, I developed a new computational method to increase the resolution of scATAC-seq data to define regions in the genome bound by TFs more precisely. I demonstrated and validated this approach both at the level of individual loci and genome-wide using orthologous signals linked to TF binding. I discuss the potential for the use of high-resolution ATAC-seq for the prioritisation of non-coding genetic variants.

The final Chapter combines the finding from the previous chapters together to establish an analytical platform, Avocato, to analyse and visualise scATAC-seq data as well as prioritise both cell types and non-coding genetic variants. The aim was to overcome the substantial barriers to use caused by the multiple complex analyses required, combined with the high dimensionality of the data and the genetics involved.

5.2 Outputs of this work

In this thesis, I have deeply reviewed and explored single-cell open chromatin epigenetics. I demonstrated the superiority of scATAC-seq over Bulk ATAC-seq. I evaluated different analytical pipelines in terms of the ability to analyse both homogenous and heterogeneous populations to analyse scATAC-seq data. I displayed the efficacy of diverse methods to annotate cell populations and then combined these to provide a practical and reproducible solution for this task.

In addition, I thoroughly explored the behaviour of the Tn5 enzyme and its effect on the length of DNA fragments. I developed a novel analytical method to increase resolution in scATAC-seq data by diminishing nucleosomal background so that TF-bound regions are identified more accurately. While my ultimate goal was to better prioritise causal variants and the cell type(s) they act in, this approach can be generalised to any question where a more precise understanding of TF-bound regions in a cell type is an important consideration.

Finally, we developed an automated, generalisable and robust tool, Avocato, capable of analysing scATAC-seq data end-to-end, visualising its result interactively in a user-friendly interface and providing a platform for prioritising non-coding genetic variants.

With everything taken into account, this thesis has expanded the ability of functional studies to interpret GWAS associations by providing a fuller solution to the prioritisation of causal SNPs.

5.3 Future Work

5.3.1 The potential to refine and use machine learning approaches to add additional levels of prioritisation

Avocato prioritises potential causal SNPs in a haplotype based on the intersection between the SNP position and a high-resolution map of TF-bound regions. However, although it can determine more precisely if the variant is in a region that directs TF binding, it cannot yet identify whether the SNP actually affects TF binding or function.

So, the next step for me is to find an extra prioritisation method to distinguish which variants in such TF-bound regions also are predicted to alter TF binding. One such class of prioritisation approaches are the machine learning (ML) methods. ML can predict activity, such as open chromatin or TF binding from genome sequence. As they predict from sequence, they potentially can be used to predict the effect of a change in the sequence on genome activity, either positively or negatively [78], [80]. Importantly, ML methods also predict activity from a sequence in a cell type-specific manner. These methods are particularly suited to predicting genome activity as they require many examples, such as open chromatin sites, to learn the sequence rules that underlie the activity. These can be easily generated using genome-wide ATAC-seq or scATAC-seq approaches. An important advantage of my work is that typically, the better defined the training set is, the better an ML algorithm learns to predict. This opens the potential to train ML algorithms, using the output of my high-resolution approach rather than standard ATAC-seq to more precisely define and so better learn the sequence patterns that determine open chromatin in a cell type. These ML algorithms could then be fused with the output of stage 2 in Avocato to add an extra layer of prioritisation to identify variants that not only are in the right region to affect TF binding but are also predicted by ML algorithms trained on the same data, to identify which ones are also predicted to affect genome activity in the presence of the SNP. Figure 5.1 shows a real-life example of how the ML approach can be a powerful tool to predict the gain/loss of function in the genome, utilising an existing deepHaem ML algorithm developed by the Hughes group.

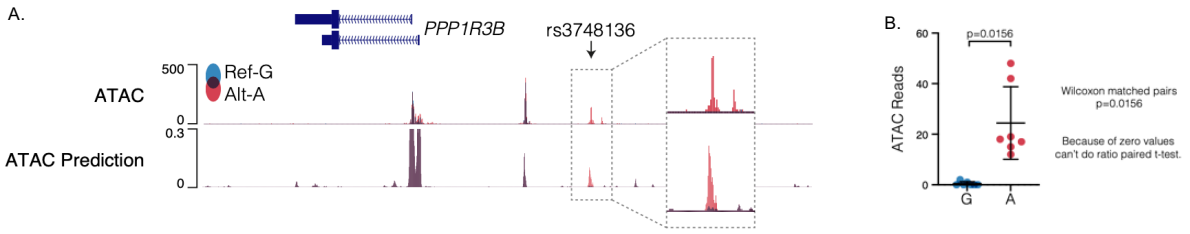


Figure 5.1: ML prediction is accurate when looking at actual Bulk ATAC-seq data at the PPP1R3B locus.
Bulk ATAC signals are shown at the top of the figure; the ML prediction of ATAC signals is below that. Signals as shown black means two alleles overlap while red signals mean signal comes from only one allele. A. The genome track shows results from two homozygous individuals. Allelic skew analysis for seven individuals who are heterozygotes is shown in B. G allele has zero ATAC reads. In contrast, the A allele has different levels of ATAC reads across 7 heterozygotes individuals, showing how much the skew is for the SNP.

The track on top shows the Bulk ATAC-seq signal of proerythroblast data at the PPP1R3B locus, whereas the below track displays ML prediction results for the same locus. In each track, the signals are normalised so that they have the same scale (an individual homozygous for rs3748136 in red and an individual who is referenced at this position in blue). The black signal means the two tracks (red and blue) perfectly overlap, while the red signal means the signal comes from the allele with rs3748136. Interestingly, the ML model predicted a complete gain of function relative to the reference when rs3748136 is present and this was validated by the two ATAC-seq tracks (see inset box).

This example was also chosen to show when Avocato would fail to prioritise a real functional variant. As can be seen in Figure 5.2, not only is the presence of an open chromatin site, but also active marks such as H3K27ac are completely linked to the presence of the rs3748136 SNPs. Therefore, unless the scATAC-seq analysed in Avocato was from an individual with the correct genotype (donor 2), no peak would exist in this region (donor 3) and so the variant would not be prioritised.

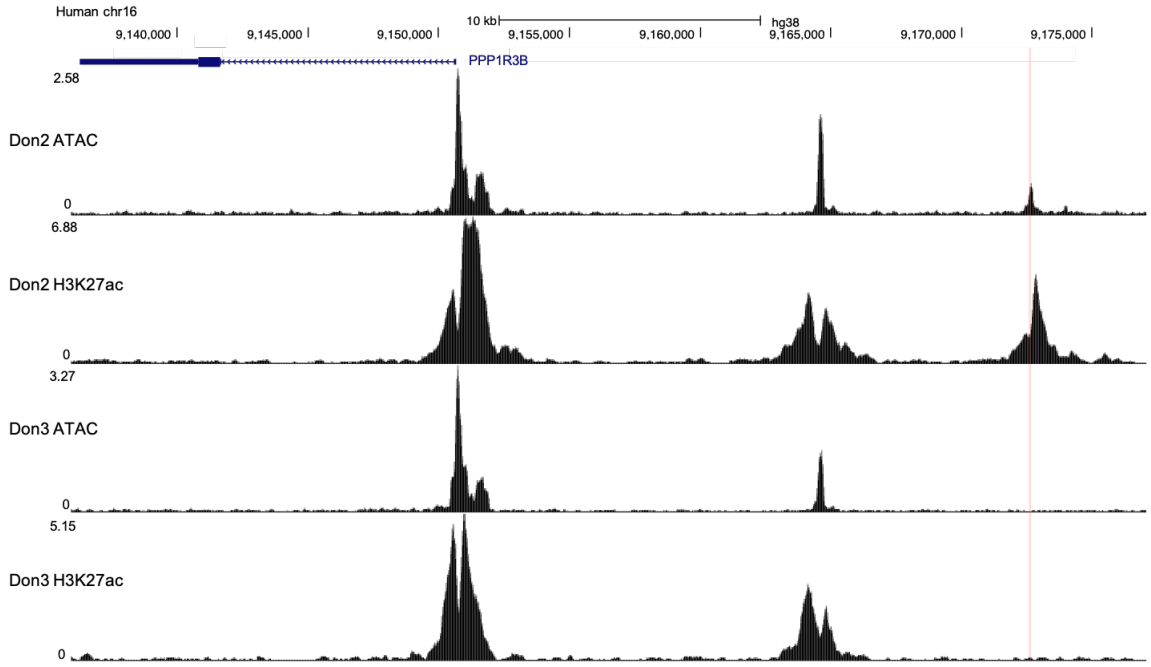


Figure 5.2: Bulk ATAC tracks for Don 2 and 3 show that Don 2 gained the new property in the presence of the highlighted SNP (rs3748136, Figure 5.1) at the PPP1R3B locus. This finding is also confirmed with the H3K27ac ChIP. The first two tracks belong to Don 2, the first being Bulk ATAC and the other being the ChIP for H3K27ac, whereas the last two are to Don 3, the first being Bulk ATAC and the other being the ChIP for H3K27ac. The highlighted region shows the location of SNP, rs3748136.

However, as seen in Figure 5.1 ML approaches can predict and prioritise such variants and the analysis of complete GWAS studies using the same trained network (Figure 5.3), suggest they may be more common than previously thought and so important to include in a comprehensive prioritisation pipeline.

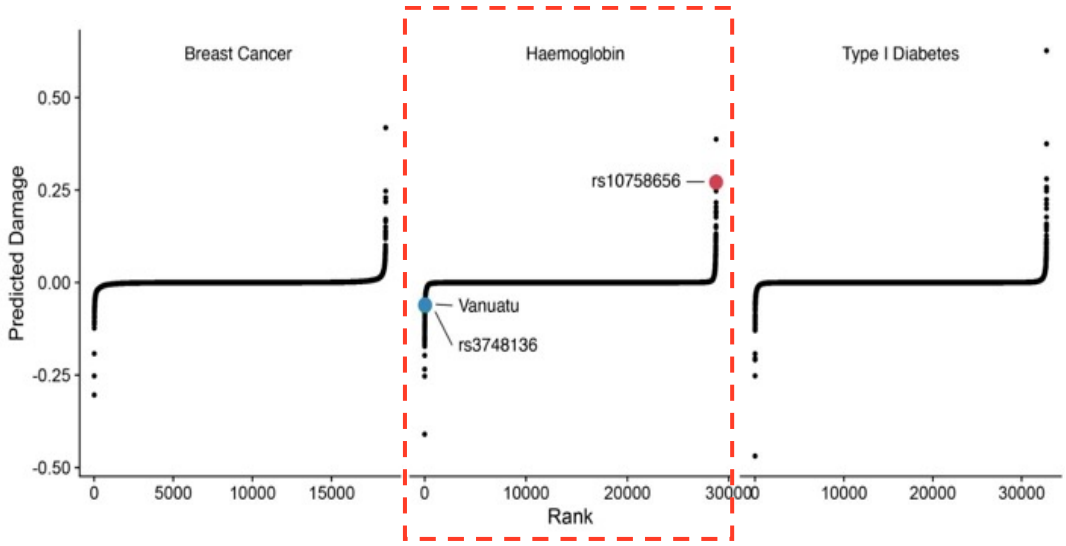


Figure 5.3: ML model can predict whether an SNP is a gain or loss of function or not causal.
The overall total damage score for haemoglobin data on Bulk ATAC proerythroblast is shown in the middle dotted red rectangular. Positive damage scores represent losses of function, while negative damage scores refer to function gains. SNPs with zero damage scores are the ones that do not do anything.

In Figure 5.3, the middle panel represents the prediction results of an imputed GWAS for Red Cell traits trained on proerythroblast data (day 10 from the proerythroblast differentiation phase). As the y-axis is, Damage Score, the top tail of the S-shape plot shows losses of function (positive damage), whereas the bottom tail shows gains of function (negative damage). As expected for a fully imputed GWAS, the vast majority of SNPs do not do anything and their predictions lie around zero. Interestingly analysis of breast cancer predisposing variants (predictions based on breast epithelia open chromatin) and Type-I diabetes (based on CD4 open chromatin) show very similar patterns, suggesting gains of function are common across diverse genetics.

Avocato's approach however, is focused on finding losses or gains of function in TF bound regions and so will miss such complete gains of function (relative to the genotype of the individual from whom the data is generated). Therefore, such ML

methods represent a highly complementary analysis that can be integrated into Avocato in the future as a 2nd step prioritisation for not only refining its prioritisation, but also to prioritise potential complete gains of function relative to the genotype of the data being analysed.

On the other hand, such ML approaches are not fully validated yet and it is still not known how well these methods perform in general and across different cell populations. Therefore, these approaches still need to be tested and validated extensively to observe their strengths and weaknesses. Such approaches are trained to predict feature such as open chromatin sites from, which obviously can be validated using methods such as ATAC-seq. However, as they predict from sequence, they are also being used to predict the effect of sequence variation which cannot be validated from a single ATAC-seq track. What is needed is open chromatin data, for a given cell type from, multiple individuals with known genome sequence to be able to validate the effect of sequence variation on prediction. However, such gold standard datasets do not exist for the systematic testing and validation of the ability of such networks to predict the effect of sequence variation.

My next step would be creating such datasets in multiple cell types and different individuals, to create some gold-standard datasets in which I will can directly test the association between the presence of a SNP and open chromatin for a cell type. For this purpose, my lab has already generated Bulk ATAC-seq datasets acquired on day 13 of erythropoiesis differentiation from 50 normal donors. These Bulk ATAC datasets are uniquely suited for this purpose as their genomes are fully sequenced. This will

allow me to create gold-standard datasets that I can use to test and validate the current ML models and refine them if necessary.

This dataset allows me to perform a first-level validation, assessing the ML predictions in one cell type using the variation present in many individuals. I would then like to follow up with scATAC-seq for PBMCs from the same individuals to allow me to perform a secondary but critical evaluation to test the ML predictions across multiple cell types. This data generation is now underway in the Hughes lab and I aim to use these resources to develop and validate the next version of Avocato, which will integrate these ML approaches using the gold standard sets of functional variants they will provide.

5.3.2 Studying differentiation in scATAC-seq

While the generation of these data will generate much need validation for ML approaches, these datasets will focus mostly on terminally differentiated cells. This is not the only point that such genetics can work on. It seems highly reasonable that such variants can also affect the differentiation processes that precede terminal differentiation. It will therefore also be interesting to understand the relationship between the genetics of such traits across each stage of differentiation.

My lab has been working on red blood cell differentiation; we have already generated a single-cell multiome dataset from 10X Genomics to cover the whole differentiation arch seen in Figure 5.4, in donors 2 and 3. We generated multiome datasets for cells from day 5 and day 10 of differentiation and will have day 0 next year. This will give

me the scATAC-seq derived regulatory landscapes of each stage of erythroid differentiation in these individuals. I can use these to prioritise using Avocato to investigate how much of the genetics of red cell traits effect earlier stages of differentiation. Also, as their genome sequences are known I can generate preliminary gold standard sets of functional variants in the manner described above and use these to validate the ML networks. Based on these prediction results, I will retrain the model for each stage of differentiation. The overall goal is to test and emphasise the need, not only for data from terminally differentiate cells, but also from the cells they are derived from, including states of cellular activation, to fully understand regulatory variation in genetic traits

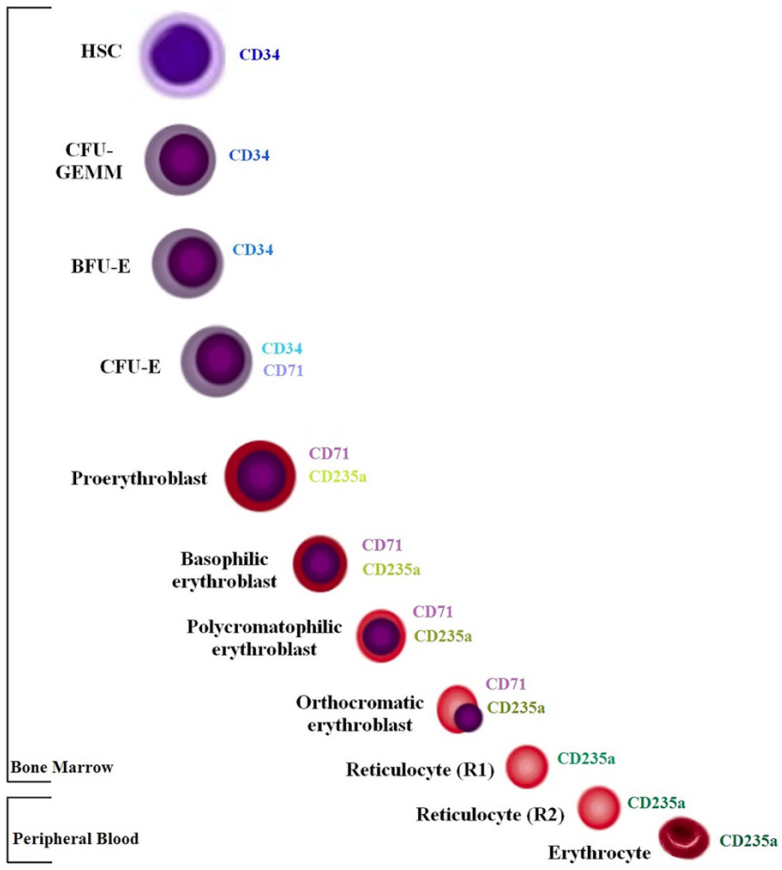


Figure 5.4: The erythropoiesis lineage, along with the expression of related surface markers, are shown. The figure is taken from Macri et al. [81].
The colour of the surface markers reflects the expression level.

5.3.3 Testing a combined Avocato-ML platform

Ultimately, after testing, validating and refining the ML model, I will build a scATAC-combined Avocato-ML platform (Figure 5.5) to comprehensively study how sequence variation affects the formation of active chromatin in the haematological niche and its associated common diseases.

As an exemplar application, I intend to study in collaboration with the Mead Lab in the WIMM, a haplotype in the JAK2 locus that predisposes to Myeloproliferative Neoplasms. Work in my lab has already shown that this haplotype is not associated with structural variation (checked by bionano optical mapping), splicing effect (spliceAI and transcriptomics), no coding mutations, no intersection with UTRs or previously mapped regulatory elements in erythropoiesis differentiation (using published data). Interestingly, currently trained ML networks to predict gains of function for two SNPs (combined haplotype) in an early population in differentiation. As part of this work, we intend to generate scATAC-seq data across erythroid differentiation from an individual with this haplotype. We will use it as a clinically important test case for the extended scATAC-combined ML platform.

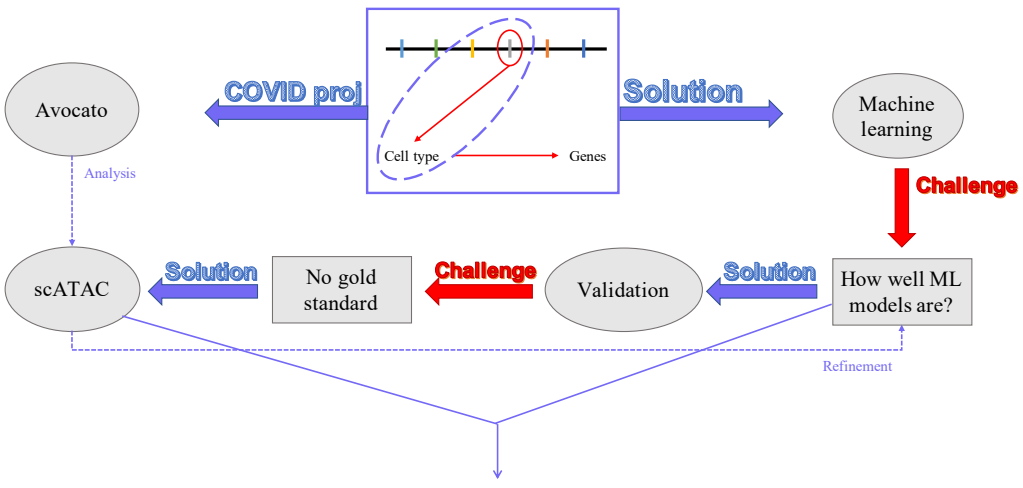


Figure 5.5: A scATAC-Combined ML platform provides a full solution to prioritise the non-coding genetic variants, which helps to understand disease mechanisms.
I focus on solving two challenges in interpreting GWAS findings; finding causal SNPs and finding the right cell type in which causal SNP affect. These problems can be solved using two approaches; Avocato and ML. Each grey shapes represent a step in the combined platform.

5.3.4 Going beyond variant

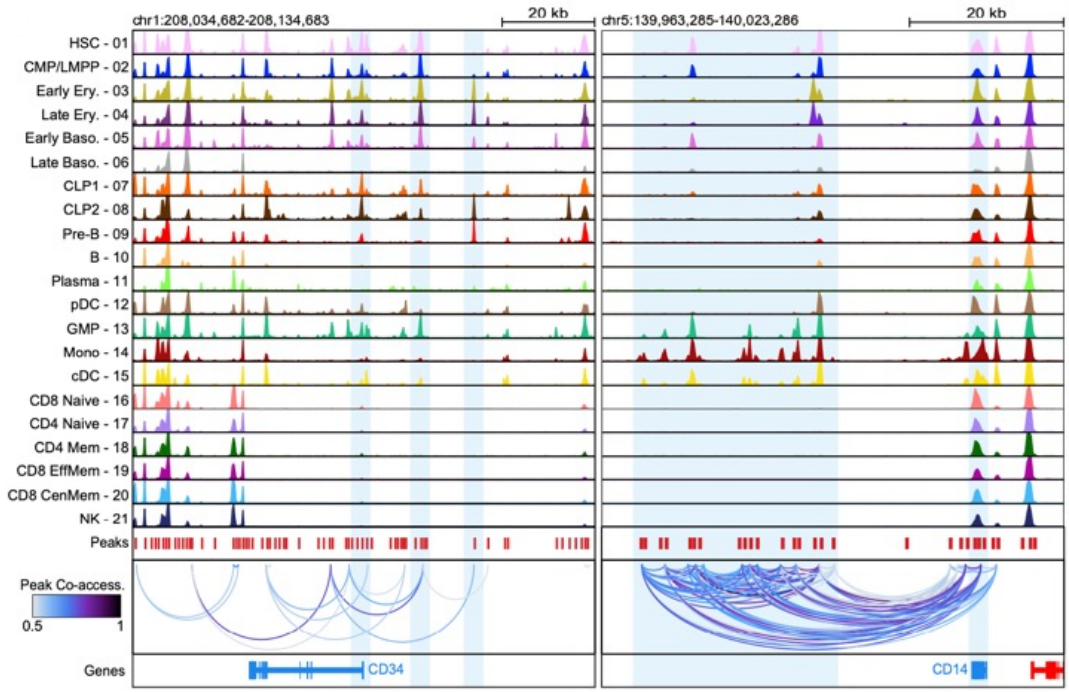
I have been acutely aware that my work only forms part of the solution for the decoding of non-coding genetics, namely variant and cell type prioritisation and validation. In the future I would like to build on my work as well as the experience and training that I gained during my DPhil to address the remaining challenges. The obvious next challenge is of course to identify the genes such regulatory variants effect. My group is expert in Chromosome Conformation Capture (3C) approaches to link regulatory elements with the genes they control and working in this environment has suggested another possible route I could try in the future that could address this.

Co-accessibility analysis in scATAC-seq is an intriguing new concept. It hypotheses that when the accessibility two peaks are statistically linked together in a locus they are functionally linked. In other words, if when Peak A exists within a single cell, it is also statistically likely that Peak B exists within that cell as well, because one functionally effects the other [40], [50]. This could suggest that the relationship between a regulatory element and a gene promoter could be derived from scATAC-seq data alone as they are obviously functionally linked. Figure 5.6.A shows an example of the linkage between peaks produced by co-accessibility analysis at the CD34 and CD14 loci reproduced from Granja *et. al* [40].

The credibility of this hypothesis is however not well investigated or validated. So, in future work I would like to test the reliability and accuracy of calling of links between regulatory elements and genes via the co-accessibility analysis of scATAC-seq data.

Therefore, I would like to validate if and to what degree the co-accessibility linkage between gene and regulatory elements obtained from scATAC-seq produces reliable results by validating them using cutting edge 3C data.

A.



B.

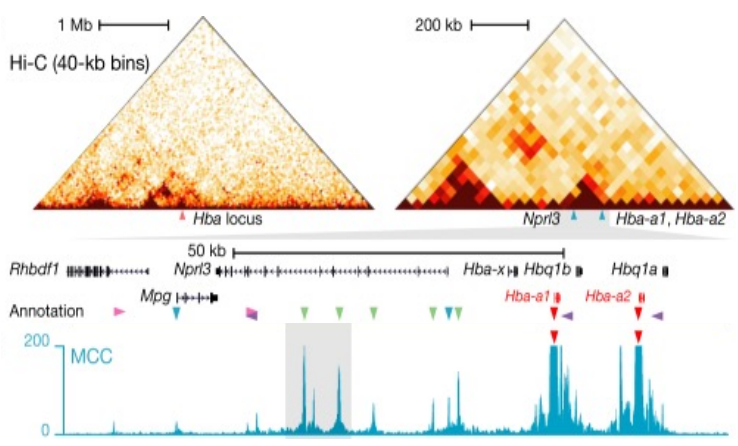


Figure 5.6: Comparison between co-accessibility analysis in scATAC-seq and MCC.
A. Peak co-accessibility analysis on scATAC-seq PBMC data shows a correlation between peaks at CD34 and CD14 locus using chromatin accessibility, indicating they are accessible together or not. This is a computational approach. Figure A is taken from the study [40]. In contrast, MCC experimental shows physical regulatory interactions between different regulatory elements. B. An example MCC track at the alpha-globin locus (blue track) with standard HIC data at two resolutions is shown above for comparison. The grey connecting bar show the position of the MCC data relative to the HIC data. B. An example MCC track at the alpha-globin locus. Figure B is taken from the study [82].

Micro-capture-C (MCC) data (Figure 5.6.B currently provides the highest resolution (base-pair) and most interpretable data of physical, regulatory interactions between regulatory elements and the genes that control. Figure 5.6.B displays the MCC track at the HBA locus (from the promoters of the HBA genes) from a recent publication from the group [82] and shows the resolution with which the known regulatory elements are identified. MCC data is presently being generated from large numbers of genes within the group across multiple cell types in both erythroid and immune cells. I am very excited to use these MCC data to test whether co-accessibility linkage can replicate the same relationship as these complex 3C experiments. Not only because of the saving in cost, effort and increased throughput but also as co-accessibility linkage can be performed in numerous critical cell types and stages of differentiation in which these molecular methods cannot.

6 Methods

6.1 Assay for transposase-accessible chromatin with high-throughput sequencing (ATAC-seq)

6.1.1 Data generation

Bulk ATAC-seq on proerythroblast from a normal human donor was generated as a part of a collaboration [69]. See Truch *et al.* [69] for a detailed description of Bulk ATAC-seq data generation.

6.1.2 Data analysis

The quality of raw sequencing files was evaluated using `FastQC` to check if the adapter sequences were present, and adapter sequences were trimmed using cutadapt. Furthermore, filtered fastq files from the Bulk ATAC-seq experiment were aligned to the human genome `hg19` using `Bowtie2` and low-quality reads (`MAPQ<30`), singleton reads, and PCR duplicates were removed from aligned data. Peak calling was then performed via `Lanceotron`, called peaks were filtered by peak score (`>0.5`), and blacklisted regions were excluded from the peak file. Finally, the coverage track was generated using `deepTools BamCoverage`. Data were visualised on the UCSC genome browser (Figure 2.1). The described custom script to run all explained processes is available on GitHub (see section 6.11.4 for code availability).

Aligned and filtered Bulk ATAC-seq data was downsampled to correspond to the total cell number of scATAC-seq data, which is 4,485 cells in total, using `samtools`. The coverage track was generated using `deepTools BamCoverage`.

6.2 Single-cell ATAC-seq

6.2.1 Data generation

6.2.1.1 Proerythroblast data

scATAC-seq on proerythroblast from a normal human donor was generated as a part of a collaboration using 10X Genomics Chromium™ i7 Multiplex Kit N Set A (1000084). See Truch *et al.* [69] for a detailed description of scATAC-seq data generation.

6.2.1.2 Peripheral Blood Mononuclear Cells (PBMCs)

Different versions of the PBMC dataset were downloaded from the 10X genomics website. These are as follows:

- PBMC from a healthy donor – granulocytes removed through cell sorting (10k cells) Single Cell Multiome ATAC + Gene Expression, source link;
- 10k PBMCs scATAC-seq from a healthy donor (Next GEM v1.1), source link;
- PBMC from a healthy donor – granulocytes removed through cell sorting (3k cells) Single Cell Multiome ATAC + Gene Expression, source link.

6.2.2 Data analysis

6.2.2.1 Proerythroblast data analysis

10X analysis

`The cellranger-atac count` (v1.1.0) pipeline by 10X Genomics was used to analyse scATAC-seq data, and the human genome `hg19` was used as the reference genome. 10X pipeline generates a sorted and indexed BAM file. Peak calling was then performed via `Lanceotron`, called peaks were filtered by peak score (`> 0.5`), and blacklisted regions were excluded from the peak file. Finally, the coverage track was generated using `deepTools BamCoverage`. Data were visualised on the UCSC genome browser (Figure 2.1).

cisTopic analysis

After running the 10X count pipeline on scATAC-seq proerythroblast, the `cisTopic` tool was used to analyse the data more comprehensively. The `cisTopic object` was created from 10X results, followed by running the `runWarpLDAModels` function with the following parameters:

- $\alpha = 50$;
- $\beta = 0.1$;
- `iterations = 500`;
- `a number of topics = x`, where `x` is in the range of 2 and 100 (2, 10, from 20 to 60, 1 by 1; from 70 to 100, 10 by 10).

With the `selectModel` function, the number of topics was determined as 48 after assessing the second derivative of the likelihood curve and perplexity. Then, the

`cluster_louvain` function was used to perform Louvain, a community detection algorithm, after the cell-topic matrix was retrieved using the `modelMatSelection` function. Finally, for visualisation purposes, 2D t-SNE embeddings were calculated using the `Rtsne` R package (Figure 2.13 and 2.16).

ArchR analysis

After running the 10X count pipeline on scATAC-seq proerythroblast, the `ArchR` tool was used to analyse the data more comprehensively. `ArchR` analysis used a fragment file resulting from the 10X count pipeline as an input to create an `arrow file` with the following parameters:

- `filterTSS = 4`;
- `filterFrags = 1000`;
- `addTileMat = TRUE`;
- `addGeneScoreMat = TRUE`.

The `ArchR project` was created after calculating and removing doublets by filtering generated doublet scores (using the `addDoubletScores` function with default parameters). Here is the workflow.

1. The iterative LSI dimensionality reduction was implemented using the `addIterativeLSI` function on the tile matrix.
2. UMAP embedding was added via the `addUMAP` function.
3. The `plotEmbedding` function was then used to visualise data in 2D space and colour cells by cluster information.
4. Finally, the `addClusters` function was used to form clusters (Figure 2.13).

For a full explanation of how the ArchR tool analyses scATAC-seq data, see their website, https://www.archrproject.com/bookdown/index.html.

SnapATAC2 analysis

The following tutorial was followed for SnapATAC2 analysis: https://kzhang.org/SnapATAC2/main/tutorials/pbmc.html, as the most recent version of the tool, is not published yet. A fragment file from the result of the 10X count pipeline was used as input to create the `AnnData object`. A quality control step was applied using the following parameters:

- `min_counts = 2,000`;
- `min_tsse = 5`;
- `max_counts = 50,000`.

After filtering cells, a tile matrix was created using a 500bp size window containing insertion counts. Before calculating and removing doublets from the data, features were selected. The dimensionality reduction method was first performed by computing the similarity matrix and then applying normalisation and decomposition. Leiden algorithm was used as a clustering method, and the outcome was visualised by calculating UMAP projections. The clustering assignment and UMAP embedding were merged in R using the `tidyverse` R package.

After finishing different analyses, all results were integrated into **R** and uploaded onto MLV for further visualisation and comparison (Figure 2.13).

6.2.2.2 PBMC data analysis

10X analysis

`The cellranger-atac count` (v2.0.0) pipeline by 10X Genomics was used to analyse PBMCs containing approximately 10,000 cells. Results from the pipeline, including DR, clustering, and t-SNE projections, were merged in R to be uploaded to MLV. 10X pipeline uses the LSA technique as a DR technique and graph-based clustering as a clustering method. See their website for a full explanation of the algorithm the 10X pipeline uses, https://support.10xgenomics.com/single-cell-atac/software/pipelines/latest/algorithms/overview.

The analysed scATAC-seq PBMC data via the 10X pipeline was split according to cluster information from the 10X clustering result. After cluster-specific BAM files were indexed, the coverage tracks were using `deepTools BamCoverage` and uploaded onto UCSC.

To reduce computation running time, a small PBMC multiome dataset containing approximately 3,000 cells was used to demonstrate a comparison between different analytical tools for scATAC-seq analysis. Therefore, ATAC per fragment information file and filtered feature barcode matrix, analysed by Cell Ranger ARC (v2.0.0), were downloaded from the 10X genomic website (Section 6.2.1.2).

ArchR analysis

The same analysis workflow for ArchR analysis, explained in section 6.2.2.1, was used to analyse the PBMC dataset (Section 2.2.5, Figure 2.14).

SnapATAC2 analysis

The same analysis workflow for SnapATAC2 analysis, explained in section 6.2.2.1, was used to analyse the PBMC dataset with the following parameter:

- `min_counts = 2,500;`
- `min_tsse = 10;`
- `max_counts = 50,000.`

After finishing different analyses, all results were integrated into R and uploaded onto MLV for further visualisation and comparison (Section 2.2.5, Figure 2.14).

6.3 Peak Annotation

6.3.1 Genomic coordinates-based peak annotation

Peak files from Bulk ATAC and scATAC were imported into R separately using the `tidyverse` R package as data frames and converted into `GRanges` objects with the `GenomicRanges` R package. `annotatePeak` function from the `ChIPseeker` R package was used to annotate peaks based on UCSC genome annotation. Pie charts were generated using the `plotAnnoPie` function from the same R package (Figure 2.3).

6.3.2 Peak annotation using in-house ChIP-seq marks

Another way of annotating peaks is using specific ChIP-seq marks, such as H3K4me1, H3K4me3, and H3K27ac, to detect promoter, primed enhancer, and higher transcription activity. In-house ChIP-seq data of H3K4me1, H3K4me3, H3K27ac, CTCF, and background input were used for this purpose. For visualisation in UCSC,

coverage tracks were generated using `deepTools BamCoverage` for ChIP-seq BAM files. Both peak files were extended as 2kb from each side (in python) to capture the ChIP-seq signal correctly. Coverage information was acquired from BAM files for those, extending peaks via `bedtools multicov`. Once read counts were obtained, extended peak files were sorted by the difference between H3K4me1 and H3K4me3. For plotting, `computeMatrix reference-point` from `deepTools` was used to create matrices for Bulk and single-cell data between ChIP-seq marks and peak files using the following parameters:

- `--referencePoint = center`;
- `--beforeRegionStartLength = 2000`;
- `--afterRegionStartLength = 2000`;
- `--sortRegions = keep`;
- `--skipZeros`;
- `--numberOfProcessors = max`.

Once matrices were generated, `plotHeatmap` from `deepTools` was performed to create a heatmap of each peak in each peak file. Heatmaps were modified in python using `pandas`, `NumPy`, `matplotlib` and `seaborn` packages. Then, peaks were categorised based on the read coverage from ChIP-seq data (Figures 2.4 and 2.5 and supplementary Figure 1).

6.4 In-silico downsampling experiment

6.4.1 Homogenous data, proerythroblast

scATAC proerythroblast data, which contains 4,485 cells, was downsampled to 2,000, 1,000, 500, 100, and 50 cells using a custom pipeline with python packages, including `pandas`, `pysam`, `tqdm`, `os`, `seaborn`, and `matplotlib` (see section 6.11.4 for code availability).

The pipeline includes the following steps:

- extracting cell barcodes (CB tags) from the analysis folder of the 10X count pipeline;
- plotting read distribution for the tags;
- downsampling the total number of cells to 2,000, 1,000, 500, 100 and 50 cells by randomly selecting CB tags in turn;
- plotting read distribution for randomly selected tags;
- sub-setting the data by the total number of CB tags, creating new BAM files;
- sorting and indexing new BAM files.

After generating downsampled BAM files, the coverage tracks were generated using `deepTools BamCoverage` and uploaded onto UCSC (Figure 2.9).

6.4.2 Heterogeneous data, PBMC

The same workflow (Section 6.1.1) was used to do downsampling on scATAC-seq PBMC data but starting from 9,988, 4,485, 2,000, 1,000, 500, 100 and 50 cells (Figure 2.11).

6.5 In-silico dilution experiment

In-silico experiment integrates two filtered scATAC-seq datasets: PBMC and proerythroblast. Analysis of scATAC-seq proerythroblast showed that data actually consists of two clusters: proerythroblast and late proerythroblast (Figure 2.6). The late proerythroblast cluster was extracted using CB tags. scATAC-seq analysis on PBMC demonstrated that the annotation of clusters is not clear to distinguish similar clusters. To avoid ambiguity in cell identity assignment, clusters that are grouped clearly apart were taken into consideration. Selected clusters and their total number of cells are as follows:

- Monocytes cluster: 842 cells;
- CD4 T cluster: 758 cells;
- CD8 naïve cluster: 288 cells;
- B cells cluster: 198 cells;
- CD8-other cluster: 129 cells;
- NK cells cluster: 106 cells;
- pDC cells cluster: 29 cells.

A custom-made pipeline was used for the rest of the study, where its steps are shown as follows (see section 6.11.4 for code availability):

- extracting cell barcodes (CB tags) for homogenous proerythroblast data;
- extracting cell barcodes (CB tags) for PBMC data;
- plotting read distribution for those tags;
- selecting CB tags randomly from the list in point 2, so the total number of diluted cells is 300, 150, 80, 40, and 20 cells;

- plotting read distribution for the randomly selected tags;
- sub-setting the data by the total number of CB tags, creating new BAM files;
- sorting and indexing new BAM files;
- merging subsetted homogenous proerythroblast data with curated PBMC data;
- indexing merged BAM files;
- performing ArchR analysis on the merged data;
- merging different results from ArchR into one metadata;
- saving metadata into a file in text format (Figure 2.12).

6.6 Assigning cell identity to clusters

For this section, ATAC per fragment information file and filtered feature barcode matrix, analysed by Cell Ranger ARC (v2.0.0), were downloaded from the 10X genomic website (Section 6.2.1.2). scATAC-seq PBMC of multiome was analysed using the `ArchR` tool, as explained in section 6.2.2. After following the analysis steps from section 6.2.2, the following analyses have been performed.

1. After plotting the result in 2D space with UMAP, the `addImputeWeights` function, which uses `MAGIC` as the engine, was used to add imputation weights to gene scores (calculated during forming of the arrow file) to reduce noise associated with the dropout.
2. Finally, different annotation methods were applied to assign cell types to these clusters, using gold standard data (Section 6.3.1), integrating independent scRNA-seq results (Section 6.3.2) and using multiome data (Section 6.3.3).

6.6.1 Using only gene activity scores from scATAC-seq with a list of known marker genes

After imputing gene scores, gene markers of PBMC from Pliner *et al.* [73] were used to designate cell types to the clusters by looking at gene score activity for those genes. Cluster information, UMAP embeddings, and cell-type annotation results were merged together in R using the `tidyverse` R package and uploaded onto MLV (Figures 2.16, 2.17.A and 2.17.B).

6.6.2 Integration of independent scRNA-seq with snATAC-seq

Filtered feature cell matrix of 10k Human PBMCs, 3' v3.1, Chromium Controller, which was analysed by `Cell Ranger` (v6.1.0.), was downloaded from the 10X Genomics website. `Seurat` R package (v4.1.1) was used first to filter scRNA-seq PBMC data using the following parameters for quality control and data normalisation by using the `LogNormalize` function with a scale factor equal to 10,000:

- `min.cells = 3`;
- `min.features = 200`;
- `nFeature_RNA > 200`;
- `nFeature_RNA < 5,000`;
- `percent.mt < 10.`

Highly variable genes were identified via the `FindVariableFeatures` function using `vst` as a statistical method, and `nfeatures = 3,000`. Data was scaled using the `ScaleData` function, which applies a linear transformation, prior to performing PCA as a dimensionality reduction method. The number of PCA components was

determined using the elbow rule. Then, the graph-based clustering technique was applied to group single cells into groups using the `FindNeighbors` function. UMAP was calculated as a non-linear dimensional reduction method for visualisation purposes only. The most informative genes among clusters were found using Wilcoxon statistical test. Assigning cell type identity to clusters was done by using gene markers, which can be found in the `Seurat` tutorial (https://satijalab.org/seurat/articles/pbmc3k_tutorial.html). `Seurat object` was saved as RDS file format to be integrated into `ArchR` result and assign cell type identity to the clusters. The result of independent scRNA-seq PBMC data analysis was integrated into the `ArchR` result via the `addGeneIntegrationMatrix` function with unconstrained integration. Cluster information, UMAP embedding, and cell type annotation results were merged in R using the `tidyverse` R package and uploaded onto MLV (Figure 2.17.C and 2.17.D).

6.6.3 Using multiome snATAC-seq and snRNA-seq

`Azimuth app` (https://azimuth.hubmapconsortium.org/) was used to analyse snRNA of multiome dataset using gene expression matrix as an input. Azimuth performs cell identity assignment to clusters by leveraging the reference-based mapping method after normalising the count matrix. Annotation result provides three different levels of annotation depending on their granularity.

Results from three different annotation methods were integrated together in R using the `tidyverse` R package and visualised in MLV (Figure 2.17.E and 2.17.F).

6.7 Obtaining high-resolution scATAC-seq data

After obtaining the BAM file from section 6.2.2.1, it was split based on different fragment lengths using `awk` filtering on the 9th column, which contains fragment length information and the `samtools view` command. The range of varying fragment lengths of the BAM file is as follows:

- 1-100bp,
- 101-200bp,
- 201-300bp,
- 301-400bp, and
- 401-500bp.

Using `samtools`, reads with a mapping quality lower than 30 and PCR duplicates were removed from the BAM file obtained in section 6.2.2.1, and reads with only proper pairs were considered. The filtered BAM file was further filtered based on different fragment lengths.

List of BAM files generated:

- no fragment size filtering (unfractionated scATAC-seq data);
- reads with fragment lengths only equal to or smaller than 150bp (TF-enriched scATAC-seq data or high-resolution scATAC-seq data);
- reads with fragment lengths only bigger than 150bp (nucleosomal-enriched scATAC-seq data).

6.8 Graphical analysis

6.8.1 ATAC DNA fragment distribution

The distribution of DNA fragments in both assays was calculated with R packages called `rtracklayer` and `Rsamtools`. The `ggplot2` R package was used to plot the distribution. The described custom script to plot distributions is available on GitHub (see section 6.11.4 for code availability).

6.8.2 Venn diagram for the intersection of Bulk ATAC and scATAC peak files

`intersectBed` from `Bedtools` was used to overlap scATAC-seq peaks with Bulk ATAC-seq peaks using the `-wao` parameter. `VennDiagram` R package was used to show the number of overlapping peaks between two peak files.

6.8.3 Distribution of regions inside and outside the called peaks

`MACS2` was applied on the BAM file (generated in section 6.2.2.1) for peak calling. Then, the peak file was used to subset that BAM file so that it would contain only peak regions using the `samtools view`. The peak file was also used to subset that BAM file so that it would not contain peak regions using the `samtools view` with the `-L` parameter. For plotting the distribution, the same script explained in section 6.8.1 was used.

6.9 Correlation between high-resolution peaks and features of TF-binding at the genome-scale

Meta plots were used to compare peak calls from size-fractionated scATAC-seq data with those from unfractionated data in terms of chromatin accessibility (Section 6.9.1), sequence conservation (Section 6.9.2), motif enrichment (Section 6.9.3), and Tn5 cut size (Section 6.9.4).

BAM files generated in section 6.7, the unfractionated and the high-resolution scATAC-seq data, were used to create bigwig files using `deepTools BamCoverage` with the following parameters:

- `-bs 1`;
- `--extendReads`;
- `--normalizeUsing RPKM`.

`Lanceotron` was used to call peaks on the created bigwig files of unfractionated and size-fractionated data. Then, peaks were filtered by peak score `(> 0.5)` and blacklist regions were removed. A fixed position (centre of each peak region) in the peak files is used to pile up data. Each region of the peak files was extended by 1kb on each side from the fixed position.

6.9.1 Distribution of high-resolution peaks in scATAC-seq data

Bigwig and peak files created in section 6.9 were used to generate a matrix using `computeMatrix` from `deepTools` using the following parameters:

- `–referencePoint=center`;
- `--beforeRegionStartLength=1000`;
- `--afterRegionStartLength=1000`;
- `–skipZeros`.

`deepTools plotHeatmap` was performed on the generated matrix to plot a heatmap and a line graph (Figure 3.10).

6.9.2 The correlation between high-resolution peaks and conservation data

Evolutionary conservation scores were downloaded from: http://hgdownload.soe.ucsc.edu/goldenPath/hg19/phyloP100way/. `computeMatrix`, with the same parameters used in section 6.9.1, was performed on the conservation bigwig file and the same peak files used in section 6.9.1 in order to create the matrix. Line plot was obtained using `pandas`, `matplotlib` and `seaborn` python libraries (Figure 3.11).

6.9.3 Distribution of high-resolution peaks in motif enrichment

Position Frequency Matrix (PFM) for vertebrates was downloaded as `MEME` files from the JASPAR website (https://jaspar.genereg.net/downloads/). The low-resolution peak file was converted to `fasta format` using `bedtools getfasta`. Fimo analysis was performed on each motif of interest and a low-resolution 2kb extended scATAC-seq peak file. The following steps were applied to generate the pseudo motif bigwig files:

- The Fimo result showed genomic coordinates enriched for a given motif. Other regions that are not enriched in a given motif were found using `bedtools subtract` from the enriched regions;
- Read count was assigned to motif-enriched regions as 10 and non-motif regions as zero;
- After creating a pseudo wig file, `wigToBigWig` from **UCSC** tools was used to generate a bigwig file.

For each pseudo motif bigwig file, NFE2, KLF1, GATA1, GATA2, and GATA1_TAL1, `computeMatrix`, using the same parameters listed in section 6.9.1, was performed to create the matrix, considering peak files as regions of interest. A line plot was obtained using `pandas`, `matplotlib` and `seaborn` python libraries (Figure 3.12).

6.9.4 Distribution of Tn5 cut sizes around high-resolution peaks and standard peaks.

Reads in the unfractionated BAM file from section 6.7 were filtered with containing the following BAM flags 99,147, 83 and 163, representing only reads that are mapped in proper pairs. To overcome the Tn5 bias, 4bp was added to reads with + (plus/positive) strand, and 5 bp was subtracted from reads with – (minus/negative) strand. For the Tn5 bias-corrected regions, count values were acquired from the map file using `bedops` tools resulting in a wig file. Consequently, the wig file was converted into a bigwig file using `UCSC` tools. Using the bigwig file on unfractionated (low-resolution) and high-resolution peak files, a count matrix was created via `computeMatrix` from

`deepTools` with the same parameters used in section 6.9.1. Line plot was obtained using `pandas`, `matplotlib` and `seaborn` python libraries (Figure 3.13).

6.10 Avocato

Here, we describe Avocato, **A**nalysis and **V**isualization **O**f single-**C**ell **AT**AC-seq **O**bservations, for prioritising regulatory variants in scATAC-seq data after analysing and visualising the data. Avocato workflow includes two stages which can also be run independently. Stage 1 consists of three steps; these include (i) pre-processing of raw fastq files or BAM files; (ii) forming clusters using ArchR as a downstream engine; and (iii) finding the most descriptive genes per cluster, while Stage 2 contains the following steps: (i) obtaining high-resolution cluster specific scATAC-seq data; (ii) performing peak calling on the high-resolution data; and (iii) prioritising regulatory variants.

See Chapter 4 for detailed explanations of how Avocato was developed. Figure 6.1 shows the analysis steps of Avocato in Snakemake format.

6.10.1 Iterative decision tree model

This model was developed as part of Avocato in collaboration and was based on Liangti Dai work. In order to extract a subset of representative genes for each cell cluster, we take the cell clustering assignment and the imputed gene score matrix from Avocato stage 1 output and train an iterative decision tree classifier in a one-vs-rest manner for each cluster, respectively. The detailed algorithm is as follows:

Require: X - imputed gene score matrix (in which X_{ij} refers to the imputed gene score of gene i in cell j); N - number of clusters; $MaxG$ - maximum number of representative genes allowed for each cluster; $MinAUC$ - minimum AUROC to terminate the iteration of growing new decision tree; r - train/test split ratio

1: **for** cluster $n \in [1, N]$ **do**
2: Randomly split X into training and test data X_{train}, X_{test} (split ratio: r; stratefied for cluster n)
3: Initialize: $TrainAUC \leftarrow 1$, $TestAUC \leftarrow 1$, $NumOfGenes_n \leftarrow 0$, $GeneSet_n \leftarrow \{\}$
4: **while** $TrainAUC_n \geq MinAUC \ and \ TestAUC_n \geq MinAUC \ and \ NumOfGenes_n \leq MaxG$ **do**
5: Grow a decision tree DT given X_{train} to classify samples into n and $\neg n$ (one-vs-others);
6: **for** gene i of the feature in all non-leaf nodes of DT **do**
7: Append i to $GeneSet_n$
8: $NumOfGenes_n \leftarrow NumOfGenes_n + 1$
9: Remove the row corresponding to gene i from X_{train} and X_{test}
10: **end for**
11: Update train and test AUROC: $TrainAUC_n$, $TestAUC_n$
12: Iterate steps 4-11
13: **end while**
14: **end for**
15: **return** $GeneSet$

The main advantages of the algorithm are:

- the nature of decision trees allows us to include features of both over- and under-expression genes;
- the iterative training strategy ensures the inclusion of all genes responsible for the cluster assignment under the given threshold.

For the decision tree algorithm, we use the default scikit-learn function `sklearn.tree.DecisionTreeClassifier` with `gini` criterion and a predefined random state. We leave all other parameters as default with no limit of the tree depth and number of features.

6.10.2 Statistical SNP prioritisation method

Enrichment was calculated as follows:

Five shuffled folds N of background genetics from hg38, considering the number of leading SNPs, were extracted.

Subsequently, for the list of lead SNPs and each obtained fold $p_i \ with \ i \in snp, f_1, f_2, f_3, f_4, f_5$, the mass function probability for binomial discrete random variable

$pmf(k,n,p)$ was calculated, where k was the number of intersecting lead variants, n was the total number of variants, and p was the total number of base-pairs within cluster-specific peaks ($peakBP$) divided by the hg38 uniquely mappable base-pairs (3,049,315,783bp, https://genomewiki.ucsc.edu/)

Then, for each $pmf_j(k,n,p)\ with\ j\ \in f_1, f_2, f_3, f_4, f_5$, $logpmf_j(k,n,p) = \frac{-loglog\left(pmf_{snp}(k,n,p)\right)}{-loglog\left(pmf_j(k,n,p)\right)}$.

Finally, the lead enrichment score for each cluster was obtained as follows:

$enrichment\text{-}lead = \frac{\Sigma logpmf_j(k,n,p)}{N}$.

For each individual cluster, the extended imputed enrichments score was performed using the formula below.

$enrichment\text{-}imputed = enrichment\text{-}lead * \frac{k'}{peakBP}$,

Where k' was the number of intersecting imputed variants.

Both enrichment-lead and enrichment-imputed were scaled between 0 and 10 for a better visualisation. `binom.pmf` from the `SciPy` python package was used to perform the enrichment prioritisation.

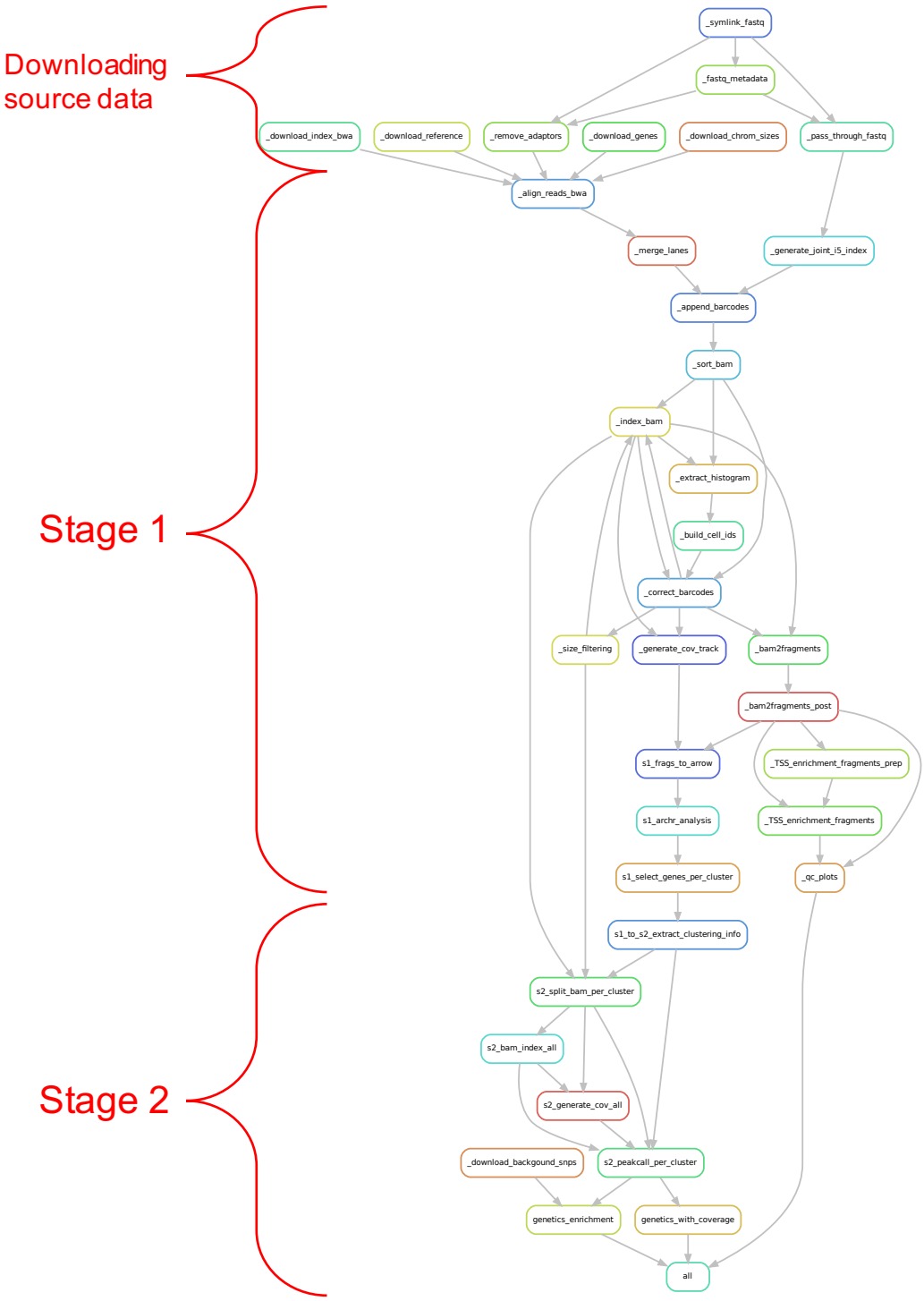


Figure 6.1: Avocato workflow in Snakemake format is shown as a directed acyclic graph (DAG).
Each step expressed in each rectangular represents each Snakemake rule (process/step) in the pipeline. Gray arrows refer to directionality. Avocato Snakemake system first starts by downloading source data used in the pipeline, followed by Stage 1 and Stage 2.
(See Snakemake website for Snakemake tutorial, https://snakemake.readthedocs.io/)

6.11 Software & Data

6.11.1 Data availability

The data used in this thesis work is listed below.

- Bulk ATAC-seq on proerythroblast,
- scATAC-seq on proerythroblast,
- scRNA-seq on proerythroblast,
- scATAC-seq on PBMC,
- scRNA-seq on PBMC,
- snATAC & snRNA sequencing on PBMC,
- ChIP-seq H3K4me1 on cells acquired at day 13 of erythropoiesis differentiation,
- ChIP-seq H3K4me3 on cells acquired at day 13 of erythropoiesis differentiation,
- ChIP-seq H3K27ac on cells acquired at day 13 of erythropoiesis differentiation,
- ChIP-seq CTCF on cells acquired at day 13 of erythropoiesis differentiation.

Proerythroblast datasets were generated as part of the collaboration [69], and their generation process can be found in Truch *et al.* [69]. For PBMC datasets, publicly available 10X datasets were downloaded from the 10X Genomic website. For ChIP-seq datasets, I used in-house datasets.

GWAS catalogue links:

- Astle red blood cell traits: https://www.ebi.ac.uk/gwas/publications/27863252
- Type-I diabetes: https://www.ebi.ac.uk/gwas/efotraits/MONDO_0005147
- Type-II diabetes: https://www.ebi.ac.uk/gwas/efotraits/MONDO_0005148
- Multiple sclerosis: https://www.ebi.ac.uk/gwas/efotraits/MONDO_0005301

6.11.2 Avocato software

Avocato uses the Conda environment to manage packages. All the packages Avocato uses (710 packages) in its Conda environment in GitHub (see section 6.11.4 for code availability).

6.11.3 Additional software

6.11.3.1 Tools

- samtools (v1.10) [35]
- deepTools (v.3.5.1) [83]
- bedtools (v2.29.2) [84]
- UCSC tools (v385) [85], [86]
- FastQC (v0.11.9) (https://www.bioinformatics.babraham.ac.uk/projects/fastqc)
- Bowtie2 (v2.4.2) [33]
- MACS2 (v2.2.7.1) [38]
- Lanceotron (v1.0.8) (https://github.com/LHentges/LanceOtron)
- 10X Genomics Cell Ranger ATAC (v1.1.0) [27]
- 10X Genomics Cell Ranger ATAC (v2.0.0) [27]
- 10X Genomics Cell Ranger ARC (v2.0.0) (https://support.10xgenomics.com/single-cell-multiome-atac-gex/software/pipelines/latest/what-is-cell-ranger-arc)
- 10X Genomics Cell Ranger (v6.1.0) [87]

6.11.3.2 R packages

- tidyverse (v1.3.2) (https://www.tidyverse.org)
- ggplot2 (v3.3.6) [88]
- GenomicRanges (v1.44.0) [89]
- ChIPseeker (v1.28.3) [90]
- Seurat (v4.1.1) [46]
- Rtracklayer (1.52.1) [91]
- Rsamtools (v2.8.0) [92]
- ArchR (v1.0.1) [40]
- cisTopic (v.0.3.0) [63]

6.11.3.3 Python libraries

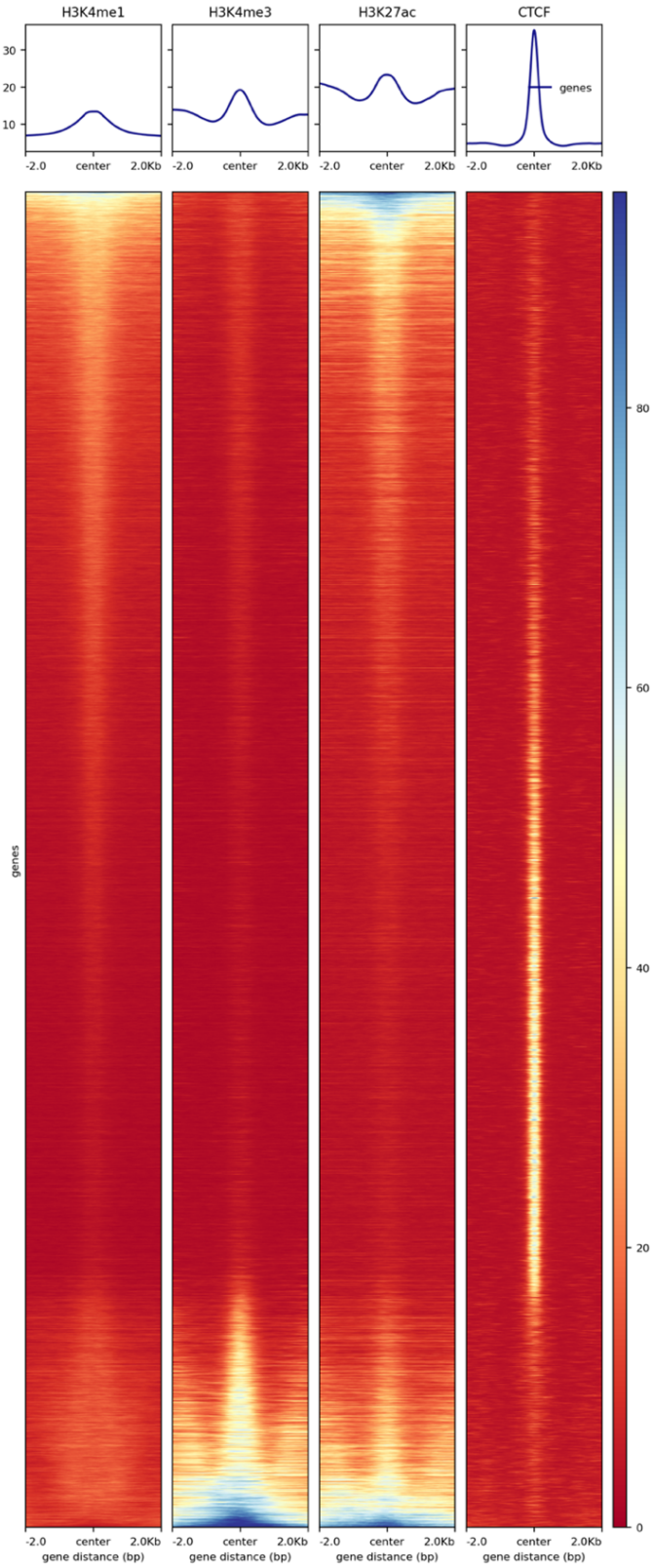
- pandas (v1.5.0) [93]
- NumPy (v1.23.5) [94]
- matplotlib (v3.4.3) [95]
- seaborn (v0.12.0) [96]
- pysam (v0.20.0) (https://github.com/pysam-developers/pysam)
- tqdm (v4.64.1) (https://github.com/tqdm/tqdm)

6.11.4 Code availability

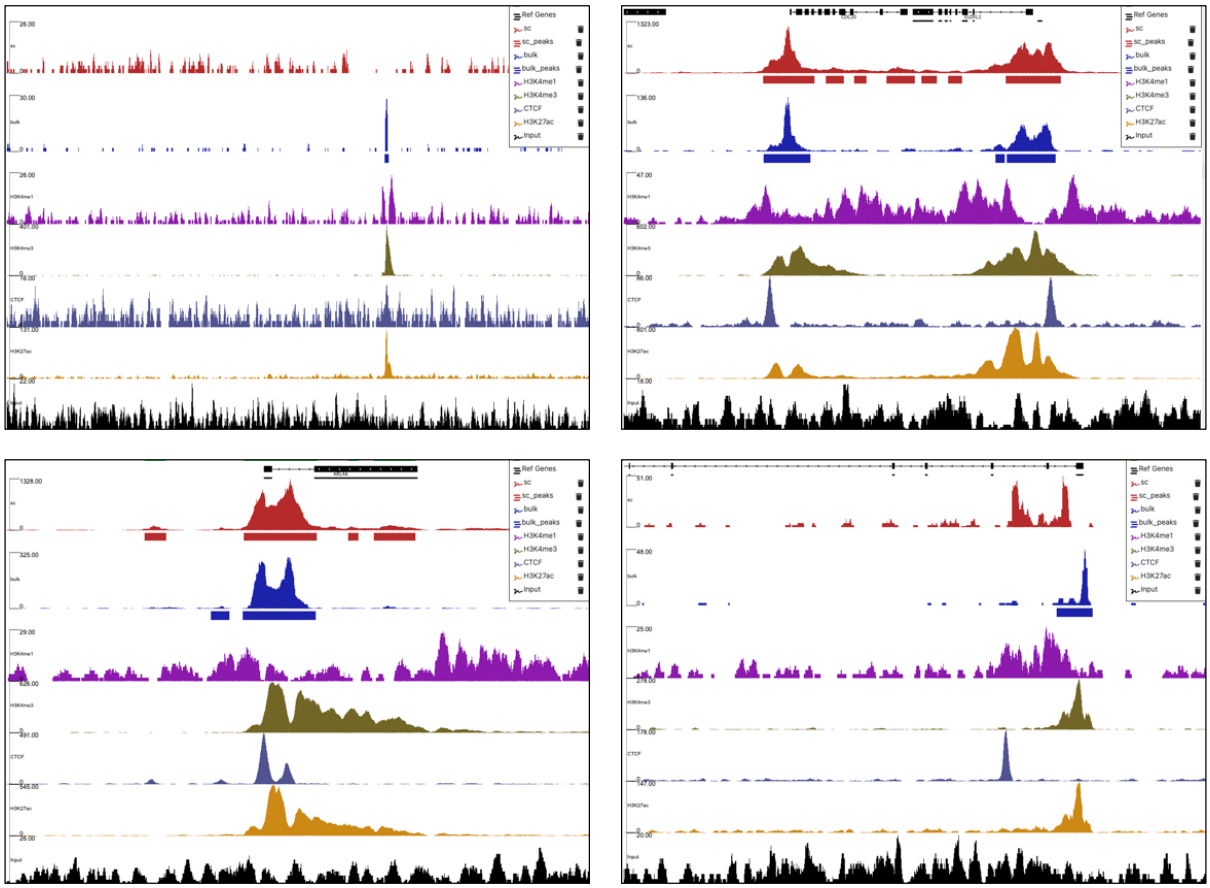
All unique codes used for data analysis are available via GitHub. https://github.com/eravza/DPhil_codes

Avocato platform is also available via GitHub. https://github.com/luntergroup/avocato

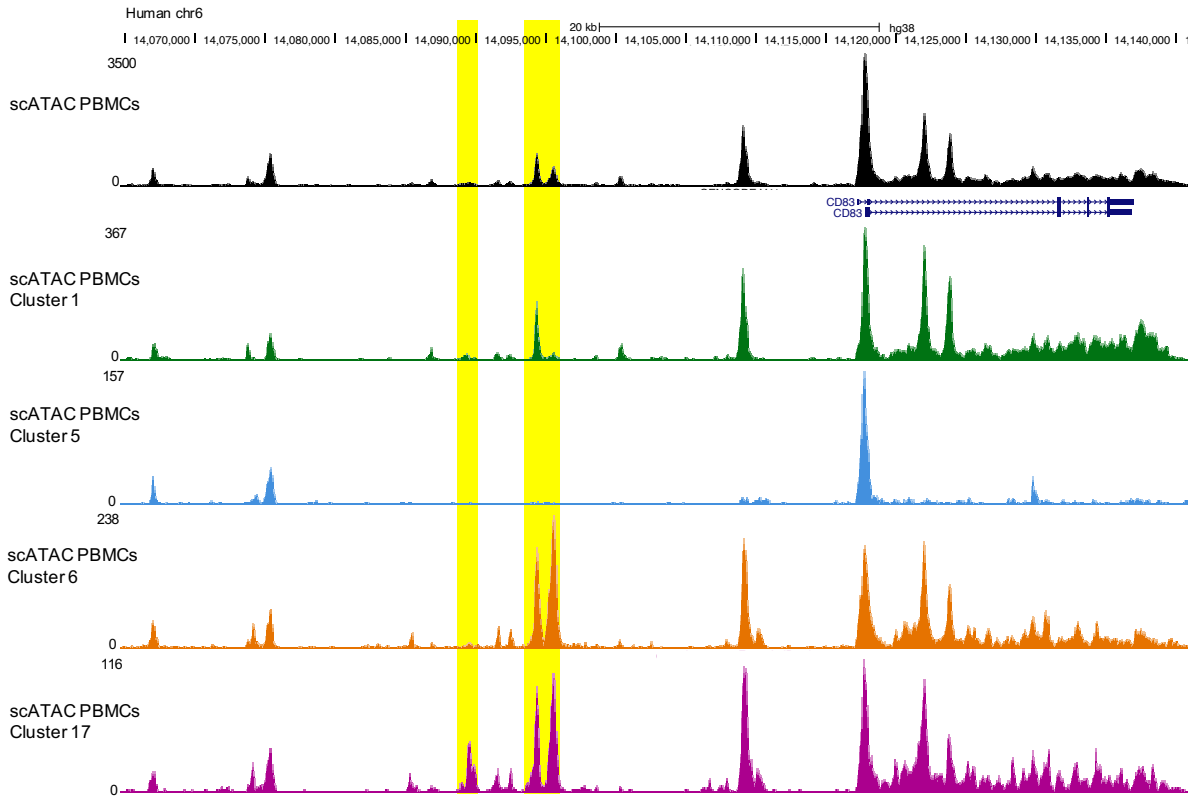
7 Supplementary Materials



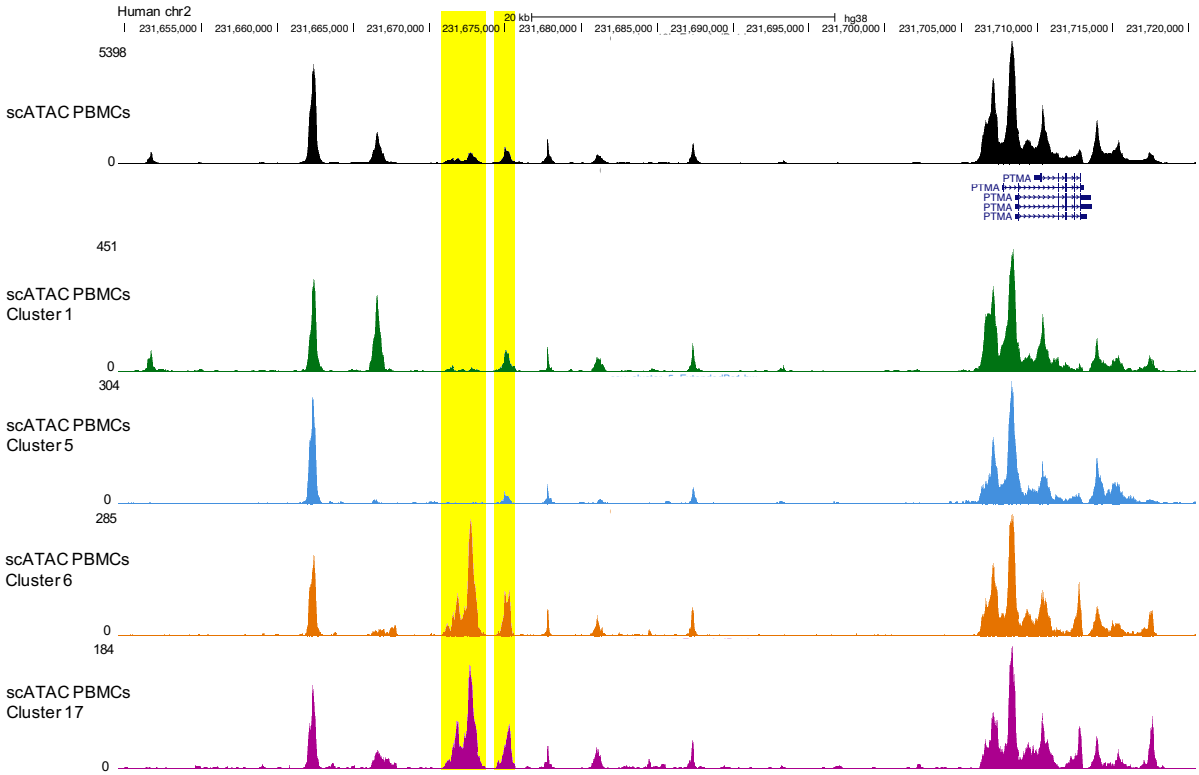
Supplementary Figure 1: Peak annotation results for peaks that are unique to scATAC-seq (57,586 unique peaks in Figure 2.3) by using in-house ChIP-seq datasets show that the majority of peaks are unique to scATAC-seq are CTCF sites, suggesting scATAC-seq is capable of identifying very sensitive regulatory elements like CTCF sites.
Peaks that are unique to scATAC-seq were centred and expanded as 2kb from each side. Read count for those fixed regions was acquired from, respectively, H3K4me1, H3K4me3, H3K27ac and CTCF. Then, peaks are categorised based on their read coverage values.



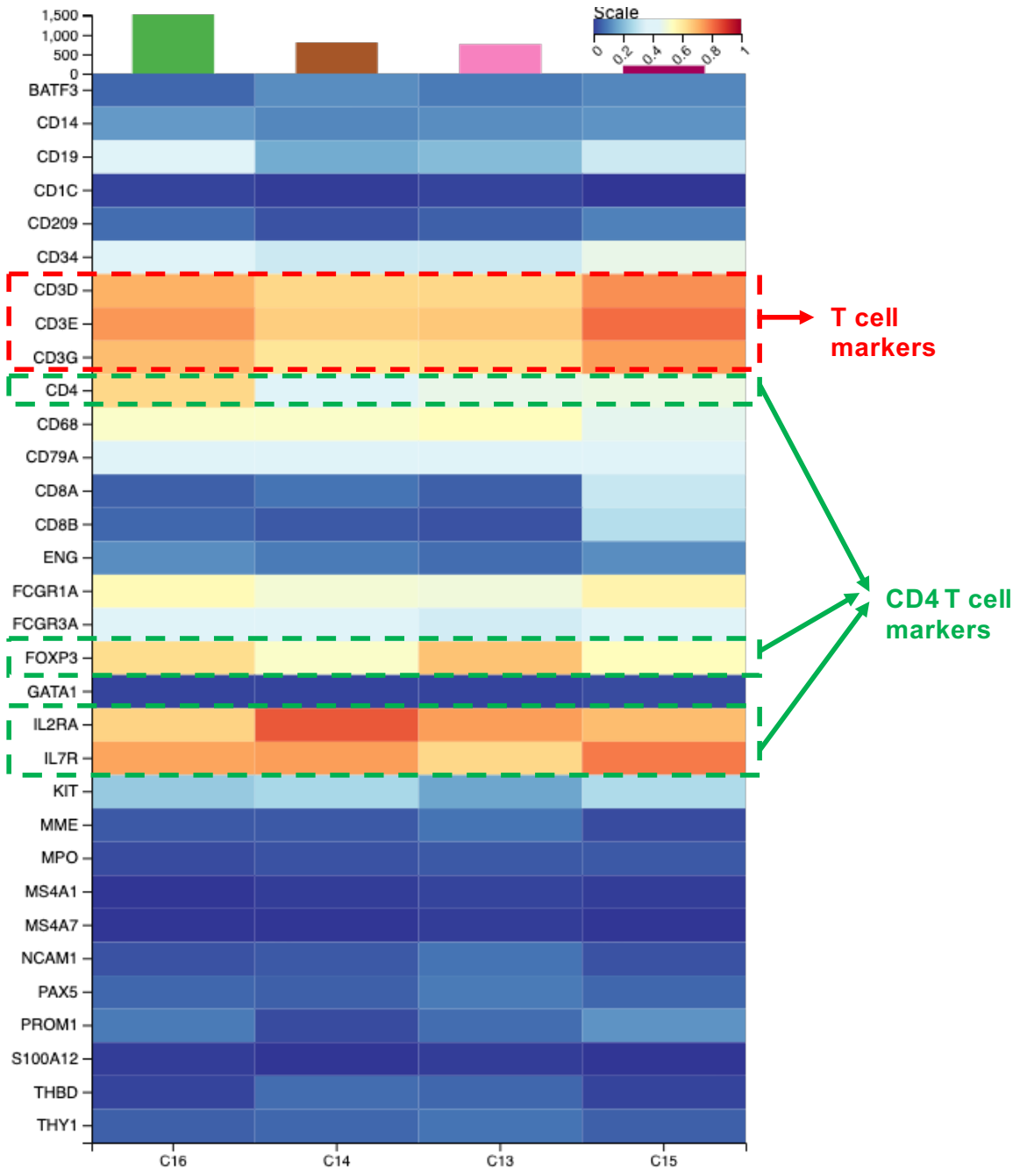
Supplementary Figure 2: A few examples of peaks that are unique only to Bulk ATAC (N=47) at different loci originated from the behaviour of peak callers to different backgrounds.
In each locus, the scATAC-seq proerythroblast data is shown at the top of the figure in red, followed by its peak set in red blocks. The Bulk ATAC-seq proerythroblast data is beneath the scATAC-seq tracks in blue, followed by its peak set in blue blocks. The ChIP markers are shown after that in order H3K4me1 (purple), H3K4me3 (light brown), CTCF (grey), H3K27ac (orange) and input (black).



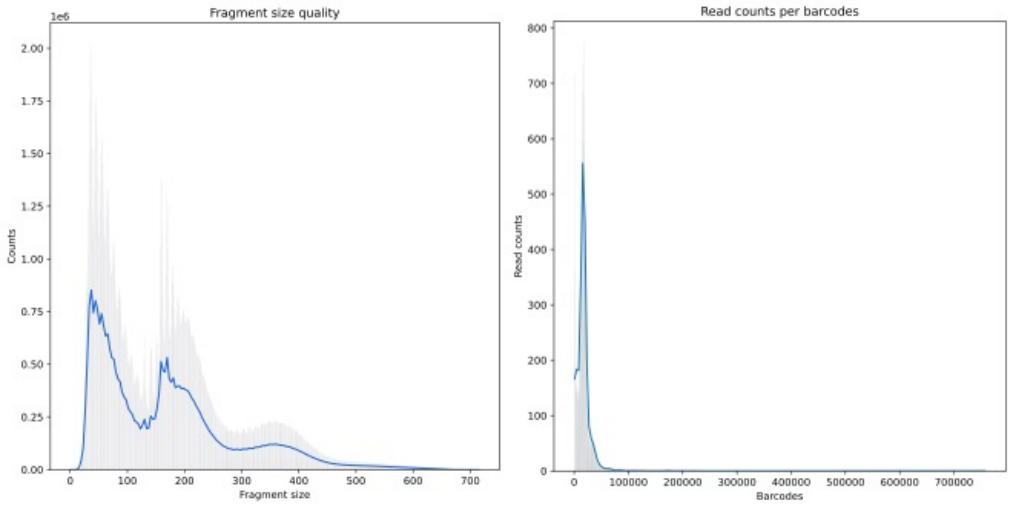
Supplementary Figure 3: Identification of the rare cell type population is very challenging as their signal contributes the least amount of total scATAC-seq data, and can be masked by larger populations of cells (The first example).
The scATAC-seq PBMC is shown at the top of the figure in black track. Their cluster-specific coverage tracks are shown below that in order cluster 1 (green), cluster 5 (blue), cluster 6 (orange) and cluster 17 (purple). The signal in the first highlighted region is very tiny in the aggregated scATAC-seq data (black track); however, this signal becomes a punctuated signal in cluster 17, suggesting indicators for rare cell populations. The same scenario is observed in the second highlighted region; two very small signals become two clear and punctuated peaks in clusters 6 and 17.



Supplementary Figure 4: Identification of the rare cell type population is very challenging as their signal contributes the least amount of total scATAC-seq data, and can be masked by larger populations of cells (The second example).
The scATAC-seq PBMC is shown at the top of the figure in black track. Their cluster-specific coverage tracks are shown below that in order cluster 1 (green), cluster 5 (blue), cluster 6 (orange) and cluster 17 (purple). The signal in the first highlighted region is small in the aggregated scATAC-seq data (black track); however, this signal becomes a punctuated signal in clusters 6 and 17, suggesting indicators for rare cell populations. The same scenario is observed in the second highlighted region; a tiny signal become a clear and punctuated peaks in clusters 6 and 17.



Supplementary Figure 5: Using only the imputed gene activity score for known gene markers is not enough for annotating clusters. C13, C14, C15 and C16 show different imputed gene activity scores for T cell markers as well as CD4 T cell markers.
This is a zoom version of the heatmap in Figure 2.16. C, Cluster.



Supplementary Figure 6: Avocato QC plots for PBMC data show a similar pattern, as seen in proerythroblast plots in Figure 4.11.
The distribution of DNA fragments and read count per barcode are shown in order.

Supplementary Table 1: Gene markers and their related cell types from Pliner *et al.* [73] were used to annotate clusters in section 2.2.6.1, Figure 2.16.

Gene Markers	Cell Types	Gene Markers	Cell Types
CD34	CD34+ cells	**CD14**	Monocytes
THY1	CD34+ cells	**FCGR1A**	Monocytes
ENG	CD34+ cells	**CD68**	Monocytes
KIT	CD34+ cells	**S100A12**	Monocytes
PROM1	CD34+ cells	**MPO**	Monocytes
NCAM1	NK cells	**MS4A7**	Monocytes
FCGR3A	NK cells	**CD3D**	T cells
CD19	B cells	**CD3E**	T cells
MS4A1	B cells	**CD3G**	T cells
CD79A	B cells	**CD4**	CD4 T cells
PAX5	B cells	**FOXP3**	CD4 T cells
MME	B cells	**IL2RA**	CD4 T cells
CD1C	DC	**IL7R**	CD4 T cells
BATF3	DC	**CD8A**	CD8 T cells
THBD	DC	**CD8B**	CD8 T cells
CD209	DC	**GATA1**	Ery

References

[1] E. Cano-Gamez and G. Trynka, “From GWAS to Function: Using Functional Genomics to Identify the Mechanisms Underlying Complex Diseases,” *Front. Genet.*, vol. 11, no. May, pp. 1–21, 2020, doi: 10.3389/fgene.2020.00424.

[2] F. Lichou and G. Trynka, “Functional studies of GWAS variants are gaining momentum,” *Nat. Commun.*, vol. 11, no. 1, pp. 2–5, 2020, doi: 10.1038/s41467-020-20188-y.

[3] E. Uffelmann *et al.*, “Genome-wide association studies,” *Nat. Rev. Methods Prim.*, vol. 1, no. 1, 2021, doi: 10.1038/s43586-021-00056-9.

[4] T. A. Myers, S. J. Chanock, and M. J. Machiela, “LDlinkR: An R Package for Rapidly Calculating Linkage Disequilibrium Statistics in Diverse Populations,” *Front. Genet.*, vol. 11, no. February, pp. 1–5, 2020, doi: 10.3389/fgene.2020.00157.

[5] T. E. Reddy, “Chapter 2 - The Functional Genome: Epigenetics and Epigenomics,” in *Genomic and Precision Medicine (Third Edition)*, Third Edit., G. S. Ginsburg and H. F. Willard, Eds. Boston: Academic Press, 2017, pp. 21–44.

[6] C. D. Allis and T. Jenuwein, “The molecular hallmarks of epigenetic control,” *Nat. Rev. Genet.*, vol. 17, no. 8, pp. 487–500, 2016, doi: 10.1038/nrg.2016.59.

[7] O. Schwartzman and A. Tanay, “Single-cell epigenomics: Techniques and emerging applications,” *Nat. Rev. Genet.*, vol. 16, no. 12, pp. 716–726, 2015, doi: 10.1038/nrg3980.

[8] S. L. Klemm, Z. Shipony, and W. J. Greenleaf, “Chromatin accessibility and the regulatory epigenome,” *Nat. Rev. Genet.*, vol. 20, no. 4, pp. 207–220, 2019, doi:

10.1038/s41576-018-0089-8.

[9] L. Minnoye *et al.*, “Chromatin accessibility profiling methods,” *Nat. Rev. Methods Prim.*, vol. 1, no. 1, pp. 1–24, 2021, doi: 10.1038/s43586-020-00008-9.

[10] A. P. Boyle *et al.*, “High-Resolution Mapping and Characterization of Open Chromatin across the Genome,” *Cell*, vol. 132, no. 2, pp. 311–322, 2008, doi: 10.1016/j.cell.2007.12.014.

[11] L. Song and G. E. Crawford, “DNase-seq: A high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells,” *Cold Spring Harb. Protoc.*, vol. 5, no. 2, pp. 1–12, 2010, doi: 10.1101/pdb.prot5384.

[12] J. D. Buenrostro, B. Wu, H. Y. Chang, and W. J. Greenleaf, “ATAC-seq: A method for assaying chromatin accessibility genome-wide,” *Curr. Protoc. Mol. Biol.*, vol. 2015, pp. 21.29.1-21.29.9, 2015, doi: 10.1002/0471142727.mb2129s109.

[13] P. G. Giresi, J. Kim, R. M. McDaniell, V. R. Iyer, and J. D. Lieb, “FAIRE (Formaldehyde-Assisted Isolation of Regulatory Elements) isolates active regulatory elements from human chromatin,” *Genome Res.*, vol. 17, no. 6, pp. 877–885, 2007, doi: 10.1101/gr.5533506.

[14] J. G. Henikoff, J. A. Belsky, K. Krassovsky, D. M. MacAlpine, and S. Henikoff, “Epigenome characterization at single base-pair resolution,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 108, no. 45, pp. 18318–18323, 2011, doi: 10.1073/pnas.1110731108.

[15] E. Shema, B. E. Bernstein, and J. D. Buenrostro, “Single-cell and single-molecule epigenomics to uncover genome regulation at unprecedented resolution,” *Nat. Genet.*, vol. 51, no. 1, pp. 19–25, 2019, doi: 10.1038/s41588-

018-0290-x.

[16] F. Yan, D. R. Powell, D. J. Curtis, and N. C. Wong, “From reads to insight: a hitchhiker’s guide to ATAC-seq data analysis,” *Genome Biol.*, vol. 21, no. 1, p. 22, 2020, doi: 10.1186/s13059-020-1929-3.

[17] H. Chen *et al.*, “Assessment of computational methods for the analysis of single-cell ATAC-seq data,” *Genome Biol.*, vol. 20, no. 1, pp. 1–25, 2019, doi: 10.1186/s13059-019-1854-5.

[18] H. Giral, U. Landmesser, and A. Kratzer, “Into the Wild: GWAS Exploration of Non-coding RNAs,” *Front. Cardiovasc. Med.*, vol. 5, no. December, 2018, doi: 10.3389/fcvm.2018.00181.

[19] D. Calderon *et al.*, “Landscape of stimulation-responsive chromatin across diverse human immune cells,” *Nat. Genet.*, vol. 51, no. 10, pp. 1494–1505, 2019, doi: 10.1038/s41588-019-0505-9.

[20] J. D. Buenrostro *et al.*, “Single-cell chromatin accessibility reveals principles of regulatory variation,” *Nature*, vol. 523, no. 7561, pp. 486–490, 2015, doi: 10.1038/nature14590.

[21] D. A. Cusanovich *et al.*, “Multiplex single-cell profiling of chromatin accessibility by combinatorial cellular indexing,” *Science (80-.).*, vol. 348, no. 6237, pp. 910–914, 2015, doi: 10.1126/science.aab1601.

[22] S. Baek and I. Lee, “Single-cell ATAC sequencing analysis: From data preprocessing to hypothesis generation,” *Comput. Struct. Biotechnol. J.*, vol. 18, pp. 1429–1439, 2020, doi: 10.1016/j.csbj.2020.06.012.

[23] S. Pott and J. D. Lieb, “Single-cell ATAC-seq: Strength in numbers,” *Genome Biol.*, vol. 16, no. 1, pp. 1–4, 2015, doi: 10.1186/s13059-015-0737-7.

[24] X. Chen, R. J. Miragaia, K. N. Natarajan, and S. A. Teichmann, “A rapid and

robust method for single cell chromatin accessibility profiling," *Nat. Commun.*, vol. 9, no. 1, pp. 1–9, 2018, doi: 10.1038/s41467-018-07771-0.

[25] S. Domcke *et al.*, "A human cell atlas of fetal chromatin accessibility," *Science (80-.).*, vol. 370, no. 6518, 2020, doi: 10.1126/science.aba7612.

[26] C. A. Lareau *et al.*, "Droplet-based combinatorial indexing for massive-scale single-cell chromatin accessibility," *Nat. Biotechnol.*, vol. 37, no. 8, pp. 916–924, 2019, doi: 10.1038/s41587-019-0147-6.

[27] A. T. Satpathy *et al.*, "Massively parallel single-cell chromatin landscapes of human immune cell development and intratumoral T cell exhaustion," *Nat. Biotechnol.*, vol. 37, no. 8, pp. 925–936, 2019, doi: 10.1038/s41587-019-0206-z.

[28] A. J. Rubin *et al.*, "Coupled Single-Cell CRISPR Screening and Epigenomic Profiling Reveals Causal Gene Regulatory Networks," *Cell*, vol. 176, no. 1–2, pp. 361-376.e17, 2019, doi: 10.1016/j.cell.2018.11.022.

[29] A. Mezger *et al.*, "High-throughput chromatin accessibility profiling at single-cell resolution," *Nat. Commun.*, vol. 9, no. 1, pp. 6–11, 2018, doi: 10.1038/s41467-018-05887-x.

[30] S. Andrews, "FastQC: A Quality Control Tool for High Throughput Sequence Data [Online]." Jun-2015.

[31] M. Martin, "Cutadapt removes adapter sequences from high-throughput sequencing reads," *EMBnet.journal; Vol 17, No 1 Next Gener. Seq. Data Anal.*, 2011, doi: 10.14806/ej.17.1.200.

[32] A. M. Bolger, M. Lohse, and B. Usadel, "Trimmomatic: a flexible trimmer for Illumina sequence data," *Bioinformatics*, vol. 30, no. 15, pp. 2114–2120, Aug. 2014, doi: 10.1093/bioinformatics/btu170.

[33] B. Langmead and S. L. Salzberg, “Fast gapped-read alignment with Bowtie 2,” *Nat. Methods*, vol. 9, no. 4, pp. 357–359, 2012, doi: 10.1038/nmeth.1923.

[34] H. Li and R. Durbin, “Fast and accurate short read alignment with Burrows–Wheeler transform,” *Bioinformatics*, vol. 25, no. 14, pp. 1754–1760, Jul. 2009, doi: 10.1093/bioinformatics/btp324.

[35] P. Danecek *et al.*, “Twelve years of SAMtools and BCFtools,” *Gigascience*, vol. 10, no. 2, p. giab008, Feb. 2021, doi: 10.1093/gigascience/giab008.

[36] Broad Institute, “Picard Tools,” *https://broadinstitute.github.io/picard/*. .

[37] P. Ewels, M. Magnusson, S. Lundin, and M. Käller, “MultiQC: summarize analysis results for multiple tools and samples in a single report,” *Bioinformatics*, vol. 32, no. 19, pp. 3047–3048, Oct. 2016, doi: 10.1093/bioinformatics/btw354.

[38] Y. Zhang *et al.*, “Model-based analysis of ChIP-Seq (MACS),” *Genome Biol.*, vol. 9, no. 9, 2008, doi: 10.1186/gb-2008-9-9-r137.

[39] E. D. Tarbell and T. Liu, “HMMRATAC: a Hidden Markov ModeleR for ATAC-seq,” *Nucleic Acids Res.*, vol. 47, no. 16, p. E91, 2019, doi: 10.1093/NAR/GKZ533.

[40] J. M. Granja *et al.*, “ArchR is a scalable software package for integrative single-cell chromatin accessibility analysis,” *Nat. Genet.*, vol. 53, no. 6, p. 935, 2021, doi: 10.1038/s41588-021-00850-x.

[41] S. Q. and R. Ali, “Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents,” *Int. J. Comput. Appl.*, vol. 181, no. 1, pp. 25–29, 2018, doi: 10.5120/ijca2018917395.

[42] S. M. Baker, C. Rogerson, A. Hayes, A. D. Sharrocks, and M. Rattray, “Classifying cells with Scasat, a single-cell ATAC-seq analysis tool,” *Nucleic Acids Res.*, vol. 47, no. 2, 2019, doi: 10.1093/nar/gky950.

[43] D. Kobak and P. Berens, “The art of using t-SNE for single-cell transcriptomics,” *Nat. Commun.*, vol. 10, no. 1, 2019, doi: 10.1038/s41467-019-13056-x.

[44] M. Z. Rodriguez *et al.*, *Clustering algorithms: A comparative approach*, vol. 14, no. 1. 2019.

[45] V. Y. Kiselev, T. S. Andrews, and M. Hemberg, “Challenges in unsupervised clustering of single-cell RNA-seq data,” *Nat. Rev. Genet.*, vol. 20, no. 5, pp. 273–282, 2019, doi: 10.1038/s41576-018-0088-9.

[46] T. Stuart *et al.*, “Comprehensive Integration of Single-Cell Data,” *Cell*, vol. 177, no. 7, pp. 1888-1902.e21, 2019, doi: 10.1016/j.cell.2019.05.031.

[47] C. Wang *et al.*, “Integrative analyses of single-cell transcriptome and regulome using MAESTRO,” *Genome Biol.*, vol. 21, no. 1, pp. 1–28, 2020, doi: 10.1186/s13059-020-02116-x.

[48] J. M. Granja *et al.*, “Single-cell multiomic analysis identifies regulatory programs in mixed-phenotype acute leukemia,” *Nat. Biotechnol.*, vol. 37, no. 12, pp. 1458–1465, 2019, doi: 10.1038/s41587-019-0332-7.

[49] Y. Hao *et al.*, “Integrated analysis of multimodal single-cell data,” *Cell*, vol. 184, no. 13, pp. 3573-3587.e29, 2021, doi: 10.1016/j.cell.2021.04.048.

[50] H. A. Pliner *et al.*, “Cicero Predicts cis-Regulatory DNA Interactions from Single-Cell Chromatin Accessibility Data,” *Mol. Cell*, vol. 71, no. 5, pp. 858-871.e8, 2018, doi: 10.1016/j.molcel.2018.06.044.

[51] H. Chen *et al.*, “Single-cell trajectories reconstruction, exploration and mapping of omics data with STREAM,” *Nat. Commun.*, vol. 10, no. 1, 2019, doi: 10.1038/s41467-019-09670-4.

[52] R. Fang *et al.*, “Comprehensive analysis of single cell ATAC-seq data with SnapATAC,” *Nat. Commun.*, vol. 12, no. 1, pp. 1–15, 2021, doi:

10.1038/s41467-021-21583-9.

[53] T. Stuart, A. Srivastava, S. Madad, C. A. Lareau, and R. Satija, “Single-cell chromatin state analysis with Signac,” *Nat. Methods*, vol. 18, no. 11, pp. 1333–1341, 2021, doi: 10.1038/s41592-021-01282-5.

[54] A. Danese, M. L. Richter, K. Chaichoompu, D. S. Fischer, F. J. Theis, and M. Colomé-Tatché, “EpiScanpy: integrated single-cell epigenomic analysis,” *Nat. Commun.*, vol. 12, no. 1, pp. 1–8, 2021, doi: 10.1038/s41467-021-25131-3.

[55] W. Yu, Y. Uzun, Q. Zhu, C. Chen, and K. Tan, “ScATAC-pro: A comprehensive workbench for single-cell chromatin accessibility sequencing data,” *Genome Biol.*, vol. 21, no. 1, pp. 1–17, 2020, doi: 10.1186/s13059-020-02008-0.

[56] E. Urrutia, L. Chen, H. Zhou, and Y. Jiang, “Destin: Toolkit for single-cell analysis of chromatin accessibility,” *Bioinformatics*, vol. 35, no. 19, pp. 3818–3820, 2019, doi: 10.1093/bioinformatics/btz141.

[57] D. A. Cusanovich *et al.*, “A Single-Cell Atlas of In Vivo Mammalian Chromatin Accessibility,” *Cell*, vol. 174, no. 5, pp. 1309-1324.e18, 2018, doi: 10.1016/j.cell.2018.06.052.

[58] C. G. de Boer and A. Regev, “BROCKMAN: Deciphering variance in epigenomic regulators by k-mer factorization,” *BMC Bioinformatics*, vol. 19, no. 1, pp. 1–13, 2018, doi: 10.1186/s12859-018-2255-6.

[59] Z. Li *et al.*, “Chromatin-accessibility estimation from single-cell ATAC-seq data with scOpen,” *Nat. Commun.*, vol. 12, no. 1, 2021, doi: 10.1038/s41467-021-26530-2.

[60] L. Xiong *et al.*, “SCALE method for single-cell ATAC-seq analysis via latent feature extraction,” *Nat. Commun.*, vol. 10, no. 1, pp. 1–10, 2019, doi: 10.1038/s41467-019-12630-7.

[61] T. Ashuach, D. A. Reidenbach, A. Gayoso, and N. Yosef, “PeakVI: A deep generative model for single-cell chromatin accessibility analysis,” *Cell Reports Methods*, vol. 2, no. 3, p. 100182, 2022, doi: 10.1016/j.crmeth.2022.100182.

[62] J. Cao *et al.*, “The single-cell transcriptional landscape of mammalian organogenesis,” *Nature*, vol. 566, no. 7745, pp. 496–502, 2019, doi: 10.1038/s41586-019-0969-x.

[63] C. Bravo González-Blas *et al.*, “cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data,” *Nat. Methods*, vol. 16, no. 5, pp. 397–400, 2019, doi: 10.1038/s41592-019-0367-1.

[64] M. Zamanighomi *et al.*, “Unsupervised clustering and epigenetic classification of single cells,” *Nat. Commun.*, vol. 9, no. 1, pp. 1–8, 2018, doi: 10.1038/s41467-018-04629-3.

[65] A. N. Schep, B. Wu, J. D. Buenrostro, and W. J. Greenleaf, “ChromVAR: Inferring transcription-factor-associated accessibility from single-cell epigenomic data,” *Nat. Methods*, vol. 14, no. 10, pp. 975–978, 2017, doi: 10.1038/nmeth.4401.

[66] C. Zhao, S. Hu, X. Huo, and Y. Zhang, “Dr.seq2: A quality control and analysis pipeline for parallel single cell transcriptome and epigenome data,” *PLoS One*, vol. 12, no. 7, pp. 1–14, 2017, doi: 10.1371/journal.pone.0180583.

[67] Z. Ji, W. Zhou, and H. Ji, “Single-cell regulome data analysis by SCRAT,” *Bioinformatics*, vol. 33, no. 18, pp. 2930–2932, 2017, doi: 10.1093/bioinformatics/btx315.

[68] J. D. Buenrostro *et al.*, “Integrated Single-Cell Analysis Maps the Continuous Regulatory Landscape of Human Hematopoietic Differentiation,” *Cell*, vol. 173, no. 6, pp. 1535-1548.e16, 2018, doi: 10.1016/j.cell.2018.03.074.

[69] J. Truch *et al.*, “The chromatin remodeller ATRX facilitates diverse nuclear processes, in a stochastic manner, in both heterochromatin and euchromatin,” *Nat. Commun.*, vol. 13, no. 1, pp. 1–16, 2022, doi: 10.1038/s41467-022-31194-7.

[70] M. Moras, S. D. Lefevre, and M. A. Ostuni, “From erythroblasts to mature red blood cells: Organelle clearance in mammals,” *Front. Physiol.*, vol. 8, no. DEC, pp. 1–9, 2017, doi: 10.3389/fphys.2017.01076.

[71] P. Ji, M. Murata-Hori, and H. F. Lodish, “Formation of mammalian erythrocytes: Chromatin condensation and enucleation,” *Trends Cell Biol.*, vol. 21, no. 7, pp. 409–415, 2011, doi: 10.1016/j.tcb.2011.04.003.

[72] J. M. Cliff *et al.*, “Cellular immune function in myalgic encephalomyelitis/chronic fatigue syndrome (ME/CFS),” *Front. Immunol.*, vol. 10, no. MAR, 2019, doi: 10.3389/fimmu.2019.00796.

[73] H. A. Pliner, J. Shendure, and C. Trapnell, “Supervised classification enables rapid annotation of cell atlases,” *Nat. Methods*, vol. 16, no. 10, pp. 983–986, 2019, doi: 10.1038/s41592-019-0535-3.

[74] Kartha Vinay K., F. M. Duarte, Y. Hu, S. Ma, J. G. Chew, and C. A. Lareau, “Functional Inference of Gene Regulation using Single-Cell Multi-Omics,” *bioRxiv*, 2021, doi: https://doi.org/10.1101/2021.07.28.453784.

[75] K. Zhang *et al.*, “A single-cell atlas of chromatin accessibility in the human genome,” *Cell*, vol. 184, no. 24, pp. 5985-6001.e19, 2021, doi: 10.1016/j.cell.2021.10.024.

[76] M. T. Kassouf *et al.*, “Genome-wide identification of TAL1’s functional targets: Insights into its mechanisms of action in primary erythroid cells,” *Genome Res.*, vol. 20, no. 8, pp. 1064–1083, 2010, doi: 10.1101/gr.104935.110.

[77] B. Soskic *et al.*, “Chromatin activity at GWAS loci identifies T cell states driving complex immune diseases,” *Nat. Genet.*, vol. 51, no. 10, pp. 1486–1493, 2019, doi: 10.1038/s41588-019-0493-9.

[78] D. J. et al. Downes, “An integrated platform to systematically identify causal variants and genes for polygenic human traits,” 2019, doi: https://doi.org/10.1101/813618.

[79] W. J. Astle *et al.*, “The Allelic Landscape of Human Blood Cell Trait Variation and Links to Common Complex Disease,” *Cell*, vol. 167, no. 5, pp. 1415-1429.e19, 2016, doi: 10.1016/j.cell.2016.10.042.

[80] R. Schwessinger *et al.*, “DeepC: predicting 3D genome folding using megabase-scale transfer learning,” *Nat. Methods*, vol. 17, no. 11, pp. 1118–1124, 2020, doi: 10.1038/s41592-020-0960-3.

[81] S. Macrì *et al.*, “Immunophenotypic profiling of erythroid progenitor-derived extracellular vesicles in Diamond-Blackfan Anaemia: A new diagnostic strategy,” *PLoS One*, vol. 10, no. 9, pp. 1–12, 2015, doi: 10.1371/journal.pone.0138200.

[82] P. Hua *et al.*, *Defining genome architecture at base-pair resolution*, vol. 595, no. 7865. Springer US, 2021.

[83] F. Ramírez, F. Dündar, S. Diehl, B. A. Grüning, and T. Manke, “DeepTools: A flexible platform for exploring deep-sequencing data,” *Nucleic Acids Res.*, vol. 42, no. W1, pp. 187–191, 2014, doi: 10.1093/nar/gku365.

[84] A. R. Quinlan and I. M. Hall, “BEDTools: A flexible suite of utilities for comparing genomic features,” *Bioinformatics*, vol. 26, no. 6, pp. 841–842, 2010, doi: 10.1093/bioinformatics/btq033.

[85] W. J. Kent, A. S. Zweig, G. Barber, A. S. Hinrichs, and D. Karolchik, “BigWig

and BigBed: Enabling browsing of large distributed datasets," *Bioinformatics*, vol. 26, no. 17, pp. 2204–2207, 2010, doi: 10.1093/bioinformatics/btq351.

[86] W. J. Kent *et al.*, "The Human Genome Browser at UCSC," *Genome Res.*, vol. 12, no. 6, pp. 996–1006, 2002, doi: 10.1101/gr.229102.

[87] G. X. Y. Zheng *et al.*, "Massively parallel digital transcriptional profiling of single cells," *Nat. Commun.*, vol. 8, 2017, doi: 10.1038/ncomms14049.

[88] Hadley Wickham, *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.

[89] M. Lawrence *et al.*, "Software for Computing and Annotating Genomic Ranges," *PLoS Comput. Biol.*, vol. 9, no. 8, pp. 1–10, 2013, doi: 10.1371/journal.pcbi.1003118.

[90] G. Yu, L. G. Wang, and Q. Y. He, "ChIP seeker: An R/Bioconductor package for ChIP peak annotation, comparison and visualization," *Bioinformatics*, vol. 31, no. 14, pp. 2382–2383, 2015, doi: 10.1093/bioinformatics/btv145.

[91] M. Lawrence, R. Gentleman, and V. Carey, "rtracklayer: An R package for interfacing with genome browsers," *Bioinformatics*, vol. 25, no. 14, pp. 1841–1842, 2009, doi: 10.1093/bioinformatics/btp328.

[92] Martin Morgan and Hervé Pagès and Valerie Obenchain and Nathaniel Hayden, "Rsamtools: Binary alignment (BAM), FASTA, variant call (BCF), and tabix file import," 2021.

[93] W. McKinney, "Data Structures for Statistical Computing in Python," *Proc. 9th Python Sci. Conf.*, vol. 1, no. Scipy, pp. 56–61, 2010, doi: 10.25080/majora-92bf1922-00a.

[94] S. Van Der Walt, S. C. Colbert, and G. Varoquaux, "The NumPy array: A structure for efficient numerical computation," *Comput. Sci. Eng.*, vol. 13, no. 2,

pp. 22–30, 2011, doi: 10.1109/MCSE.2011.37.

[95] J. D. Hunter, “Matplotlib: A 2D graphics environment,” *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, 2007.

[96] Michael L. Waskom, “seaborn: statistical data visualization,” *J. Open Source Softw.*, vol. 6, p. 3021, 2021, doi: 10.21105/joss.03021.