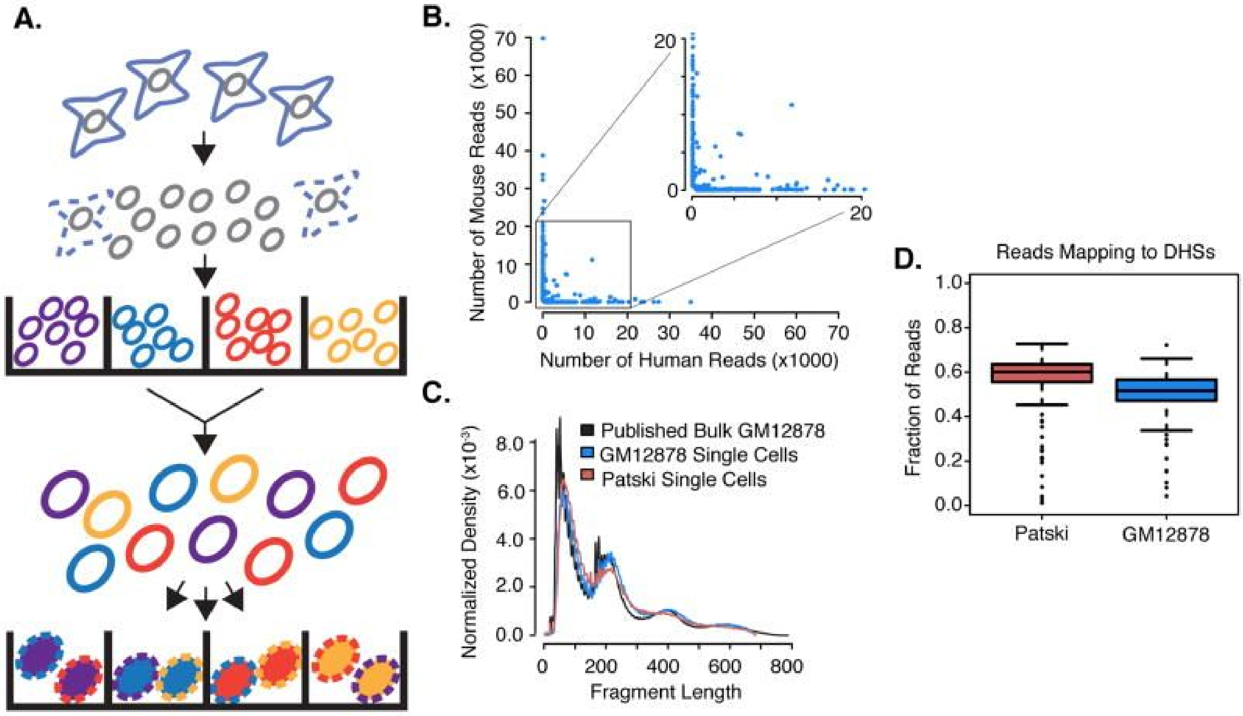
Single-cell ATAC-seq

Library preparation approaches

Cusanovich approach (sci-ATAC-seq)

Multiplex Single Cell Profiling of Chromatin Accessibility by Combinatorial Cellular Indexing, Cusanovich et al., 2015
Developed by Cusanovich (Shendure Lab) + Trapnell Lab (University of Washington) + Illumina.
Mentioned on Illumina website.
sci-ATAC-seq = single-cell combinatorial indexing ATAC-seq.

Protocol



Two-phase tagging (combinatorial cellular indexing):

1. molecularly tag nuclei in 96 wells with barcoded transposase complexes;
2. pool, dilute, and redistribute 15–25 nuclei to each of 96 wells of a second plate using a cell sorter. After lysing nuclei, a second barcode is introduced during PCR with indexed primers complementary to the transposase-introduced adapters.

Finally, all PCR products are pooled and sequenced, with the expectation that most sequence reads bearing the same combination of barcodes will be derived from a single cell (estimated collision rate of ~11% for experiments described here).

Applications

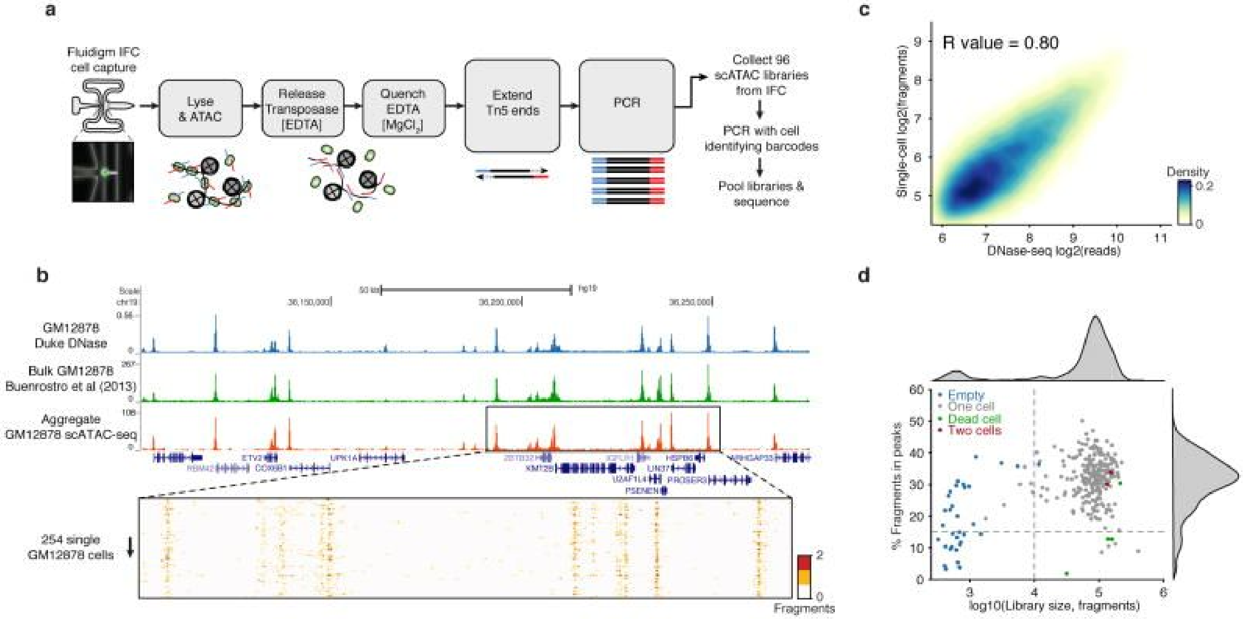
The authors reliably separated mouse cells and human cells, as well as different human cell lines. Unsupervised multidimensional scaling for a two cell-type mixture showed that the first component was strongly related to the read depth, while the second separated the cell types. The authors also detected many novel (differential) DNAse hypersensitivity sites.

Summary

"Our combinatorial cellular indexing scheme could feasibly be scaled to collect data from ~17,280 cells per experiment by using 384×384 barcoding and sorting 100 nuclei per well (assuming similar cell recovery and collision rates). [...] [I]t is worth noting that because DNA is at uniform copy number, single-cell chromatin accessibility mapping may require far fewer reads per single cell in order to define cell types, relative to single-cell RNA-seq."

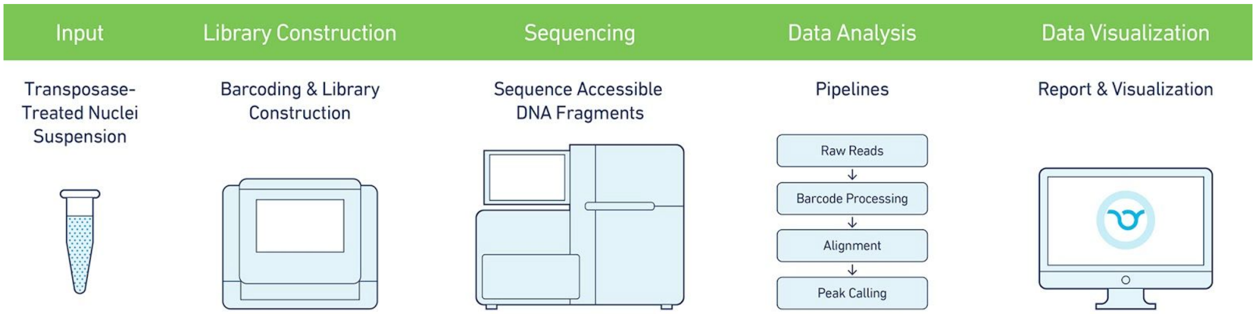
Buenrostro approach (C1-based)

Single-cell chromatin accessibility reveals principles of regulatory variation, Jason D. Buenrostro et al., 2015
Developed in Stanford University. Uses Fluidigm C1 device. Isolates individual cells. No collisions, but only 254 cells processed in the original article.



10x approach

Chromium single cell ATAC device uses barcoded GEMs (beads in gel emulsion). The resulting library can then be sequenced by any available sequencer.

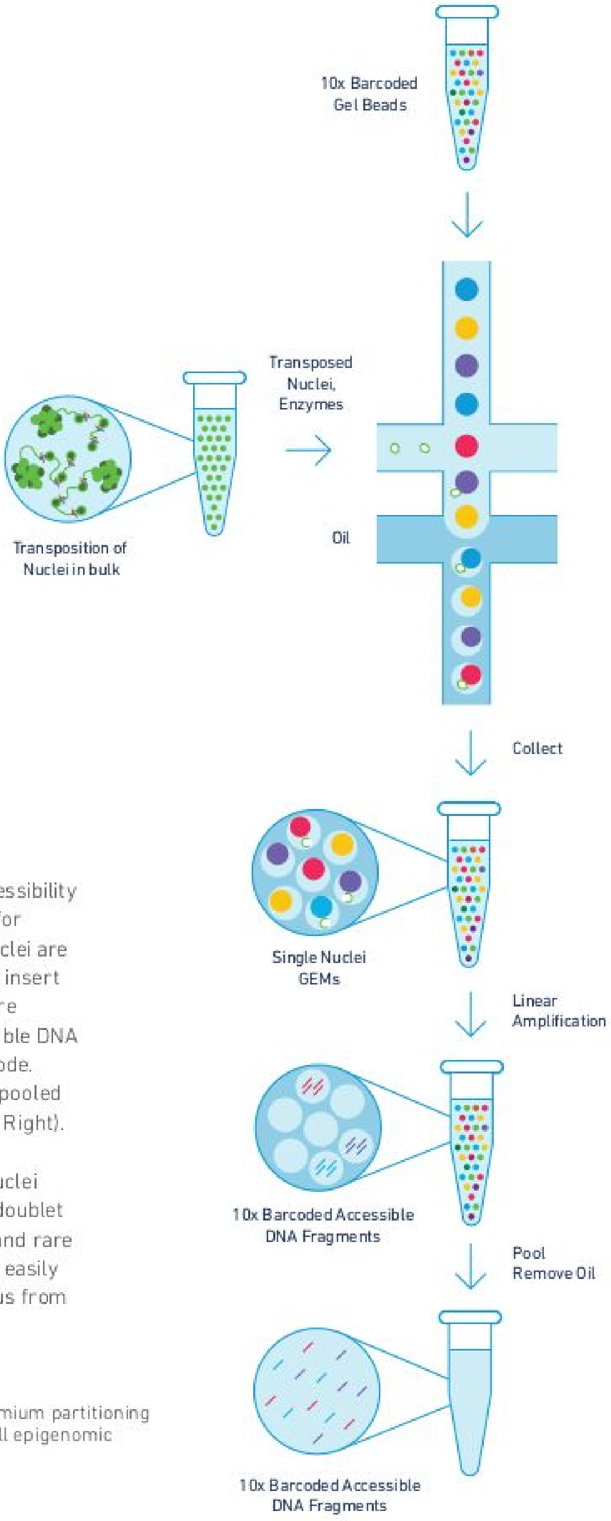


Input
Library Construction
Sequencing
Data Analysis
Data Visualization
Transposase-Treated Nuclei Suspension
Barcoding & Library Construction
Sequence Accessible DNA Fragments
Pipelines
Report & Visualization
Raw Reads
Barcode Processing
Alignment
Peak Calling

ı accessibility
ssay for
jle nuclei are
ıtially insert
ıen are
:cessible DNA
barcode.
ently pooled
gure, Right).

0+ nuclei
low doublet
jing and rare
en be easily
ıucleus from

Chromium partitioning
gle cell epigenomic



Processing pipelines

"Recently, the ATAC-seq protocol was modified to apply with single-cell resolution (Buenrostro et al., 2015b; Cusanovich et al., 2015). Buenrostro et al. (2015b) used a microfluidic approach to isolate the cells whereas Cusanovich et al. (2015) avoided physical isolation of cells by using a combinatorial indexing strategy. However, neither of the studies developed a clear bioinformatics pipeline for the processing of the data and its downstream analysis." Classifying cells with Scasat - a tool to analyse single-cell ATAC-seq, Baker et al., 2017

Scasat

Single-cell ATAC-seq Analysis Tools. Classifying cells with Scasat - a tool to analyse single-cell ATAC-seq, Baker et al., 2017
Developed in jupyter notebook + R.

Processing

Notebook.

1.
   ```
   java -jar {trimmomatic} PE -phred33 {r1_input} {r2_input}
   {Trimmomatic_Files}_r1_paired.fq
   {Trimmomatic_Files}_r1_unpaired.fq
   {Trimmomatic_Files}_r2_paired.fq
   {Trimmomatic_Files}_r2_upaired.fq
   ILLUMINACLIP:/home/baker/my-hydra-share/Packages/Trimmomatic-0.35/adapters/NexteraPE-PE.fa:2:30:10:1:true TRAILING:3
   SLIDINGWINDOW:4:15 MINLEN:25
   ```
   a. Use `trimmomatic` to trim adapters and remove low-quality reads.

2.
   ```
   {bowtie2} -p 12 -X 2000 --dovetail -x {ref_genome} -1
   {Trimmomatic_Files}_r1_paired.fq -2
   {Trimmomatic_Files}_r2_paired.fq -S {Bowtie_files}.sam 2>
   {Bowtie_files}_allignment.log
   ```
   a. `-X/--maxins <int> maximum fragment length (500)`
   b. `--dovetail concordant when mates extend past each other`

3.
   ```
   {samtools} view -SbhF 4 -f2 -q30 {Bowtie_files}.sam >
   {Bowtie_files}.bam
   ```
   a. `-S` option indicates the input is a sam file
   b. `-b` option asked samtools to write output as a bam file
   c. `-h` option asked samtools to print header, it is very important to include this "-h" when converting between sam <-> bam files.
   d. `-F 4` option removes non-mapped reads
   e. `-f2 only include reads with all bits set in INT set in FLAG [0]`

   f. `-q only include reads with mapping quality >= INT [0]`

4.
   ```
   intersectBed -v -abam {Bowtie_files}.bam -b {blackListFile} > {Blacklist_removed}_noexclu.bam
   ```
   a. Next-gen sequencing [experiments] often produce artifact signal in certain regions of the genome. [ENCODE] blacklists were empirically derived from large compendia of data using a combination of automated heuristics and manual curation. Once can download hg19 blacklist at http://hgdownload.cse.ucsc.edu/goldenPath/hg19/encodeDCC/wgEncodeMapability/wgEncodeDacMapabilityConsensusExcludable.bed.gz .

5.
   ```
   {samtools} sort {Blacklist_removed}_noexclu.bam {Blacklist_removed}_noexclu_sorted
   ```
   a. Sort the filtered reads

6.
   ```
   java -Xmx2g -jar {MarkDuplicate} INPUT={Blacklist_removed}_noexclu_sorted.bam OUTPUT={Duplicates_removed}_nodup.bam METRICS_FILE={Duplicates_removed}_nodup_stats REMOVE_DUPLICATES=True
   ```
   a. Remove duplicates with Picard tools.

7.
   ```
   {samtools} sort {Duplicates_removed}_nodup.bam {Duplicates_removed}_nodup_sorted
   ```
   a. Sort again (?!).

8.
   ```
   {samtools} view -b {Duplicates_removed}_nodup_sorted.bam chr1 chr2 chr3 chr4 chr5 chr6 chr7 chr8 chr9 chr10 chr11 chr12 chr13 chr14 chr15 chr16 chr17 chr18 chr19 chr20 chr21 chr22 chrX > {Duplicates_removed}_nodup_sorted_cleaned.bam
   ```
   a. Retain regular chromosomes only.

9.
   ```
   {samtools} index {Duplicates_removed}_nodup_sorted_cleaned.bam
   ```
   a. Index BAM.

10.
    ```
    {samtools} idxstats {Duplicates_removed}_nodup_sorted_cleaned.bam > {Duplicates_removed}_nodup_sorted_cleaned_chrom_stat.txt
    ```
    a. Samtools stats.

11.
    ```
    macs2 callpeak -t {Duplicates_removed}_nodup_sorted_cleaned.bam -n {Macs2_files} -p 0.0001 -g hs -f BAMPE --nomodel --nolambda -B --keep-dup all --call-summits
    ```
    a. Perform peak calling and generate bedGraph using 50 bp reads centred on 5 prime cutting end, within a 50-bp smoothing window for bedGraph signal track.

    b. [W]e set the p-value to 0.0001. This is due to the fact that otherwise MACS2 will not call the peaks having a single read mapped to it as it would consider those reads to be background noise (?!).

    c. `--call-summits`: MACS will now reanalyze the shape of signal profile (p or q-score depending on cutoff setting) to deconvolve subpeaks within each peak called from general procedure. It’s highly recommended to detect adjacent binding events. While used, the output subpeaks of a big peak region will have the same peak boundaries, and different scores and peak summit positions.

12.
    ```
    awk 'BEGIN {OFS = """\t"""} ; {print $1, $2 - 250, $3 + 250, $4, $5}' {Macs2_files}_summits.bed > {Macs2_files}_summits_shifted.bed
    ```
    a. We extend MACS2 summits 250bp in each direction.

13.
    ```
    {bdg2bw} {Macs2_files}_treat_pileup.bdg /home/baker/my-hydra-share/Packages/Homer_4.6/bin/hg19.chrom.sizes
    ```
    a. Create bigWig files for the UCSC genome browser visualization.
14. Filter BAM and MACS2 files with estimated library size >= 10K (tedious Picard summary parsing).
15. Generate read summary in form "`Cell_ID Number_of_input_reads Unmapped_reads_number Percent_Unmapped Uniquely_mapped_reads_number Percent_of_uniquely_mapped`"

Analysis

Single-cell ATAC-seq is essentially **binary** in nature. A specific location in a chromosome for a specific cell can **either be open or closed.** In contrast to bulk data where more reads aligning to a specific location of a chromosome would indicate more cells in the population having open chromatin at that location, in single-cells it could only be due to the multiple insertions in that region or possibly other alleles at that locus. As described by Buenrostro et al. (2015b) such reads are rare for single-cell ATAC-seq which is **overwhelmly dominated by single reads for a specific location** of a chromosome for each individual cell.

Peak accessibility matrix

The analysis workflow of Scasat starts by **merging all the single-cell BAM files** and creating a single aggregated BAM file. Peaks are called using MACS2 on this aggregated BAM file and sorted based on q-value. Peaks in this list are the ones that are **open in at least one single-cell**. Using this list of peaks we generate the **peak accessibility matrix**. The rows of this matrix represent all the peaks from the reference set and the columns represent each single cell. The pipeline calculates the accessibility of the peaks for each individual cell where it has **at least one overlapping read** and encodes it as a binary value. For each individual cell, peaks that overlap with this list of accessible regions are given the value of 1 in the table. For all the other peaks it is 0.

Bulk vs aggregate

If a **Bulk measurement** is available for the same cell-type or sample, then the pipeline can calculate Number_of_peaks vs. precision for the aggregated single-cell data against its population-based Bulk data. This demonstrates **how closely the single-cell data recapitulates its Bulk counterpart.**

Filtering Peaks

Peaks appearing in a **small number of cells are less informative** and are not always appropriate for downstream analysis. Similarly, **some cells passing through library size filtering**, might still have **a very low number of peaks.** In our downstream analysis we filter out these cells and peaks. If a cell has **open peaks below a user defined threshold** (default: 50 peaks or 0.2% of total peaks) in peak accessibility matrix we would drop that cell. Also **peaks not observed across a user defined number of valid cells** (default: 8 cells) are not considered for further downstream analysis. Choice of this threshold depends on the number of cells in the experiment, nature of those cells and other biological as well as technical factors. Users need to carefully define these thresholds for the filtering based on their experimental design.

Jaccard

Calculate **Jaccard distance** between cells.

Dimensionality reduction

MDS + t-SNE.

Clustering

k-medoids on Jaccard distance.

Differential accessibility (DA) analysis

- **Information Gain** measures the attribute that provides the best prediction of the target attribute where entropy is reduced. Peaks are [...] **sorted** based on the information gain and the user can **choose the cutoff value** for selecting the DA peaks.
- **Fisher exact test.** We run this test on **a peak-by-peak basis** by organizing the open and closed (1’s and 0’s) for each peak in a 2×2 contingency table. The p-values are then corrected using **Bonferroni** correction for multiple comparisons.

proatac

proatac is a Snakemake-based pipeline for ATAC-seq. It acknowledges C1-base single cell analysis (i.e. Buenrostro approach). Quite logical, since it's a product of the Buenrostro lab. Their GitHub issue tracker is more or less dead, though; a possible indication that no one actually uses this. It's pip-installable.

Pipeline

Much less documented, but mostly the same as Scasat from the look on Snakemake and Python files.

1. Trim (custom in-house scripts);

2. Align (`bowtie2 -X 2000`);
3. Process (samtools sort, quality filter, chromosome filter, Picard remove duplicates);
4. QC (fastqc, multiqc, in-house Picard logs parsing);
5. Analysis (macs2 for bulk samples, NucleoATAC to call nucleosomes, chromVAR to identify variable transcription factors, compare peaks to existing dataset using CistromeDB).

Cell Ranger ATAC

Cell Ranger ATAC is a custom 10x pipeline.

Preprocessing

[Cell Ranger ATAC] demultiplexes raw base call (BCL) files generated by Illumina® sequencers into FASTQ files. It is a wrapper around bcl2fastq from Illumina®, with additional useful features that are specific to 10x Genomics libraries and a simplified sample sheet format.

Processing

- Read filtering and alignment
  - Use the cutadapt tool to identify the reverse complement of the primer sequence at the end of each read, and trim it from the read prior to alignment
  - Align to a specified reference using BWA-MEM with default parameters
  - Find duplicate reads, identify the most common barcode, report it as the only occurrence
  - Filter MAPQ > 30 on both reads, not mitochondrial, and not chimerically mapped
- Barcode counting
  - Attempt to correct barcodes that aren't on the whitelist
- Identification of transposase cut sites
  - Adjust the 5' ends of the read-pair to account for transposition
  - Note the most common barcode observed for the group of read-pairs and the number of times this fragment is observed
  - Each unique interval on the genome can be associated with only one barcode
  - `samtools tabix` with default parameters
- Detection of accessible chromatin peaks
  - Count the number of transposition events at each base-pair
  - Generate a smoothed profile with a 401bp moving window
  - Fit a ZINBA-like mixture model consisting of geometric distribution to model zero-inflated count, negative binomial distribution to model noise and another negative binomial distribution to model the signal
  - Signal threshold based on an odds-ratio of ⅕
  - Peaks within 500bp of each other are merged

  - Metaparameters selected to maximize the area under ROC curve for ENCODE GM12878 data and to produce adequate clustering in PBMC libraries
  - Independent of barcodes, not able to identify rare peaks appearing only in very rare populations
- Cell calling
  - Use the number of fragments that overlap any peak regions for each barcode
  - Subtract a depth-dependent fixed count from all barcode counts to model whitelist contamination (number of fragments per barcode that originated from a different GEM), assuming a contamination rate of 0.02.
  - Fit a mixture model of two negative binomial distributions.
  - Odds ratio of 1000 to separate the barcodes that correspond to real cells from the non-cell barcodes.
  - Limited to produce < 20k cells per species.
  - In the case of mixed species sample, mask out the non-cell barcodes and fit the same mixture model to the two species distributions, or just report top N barcodes.
- Count matrix generation for peaks and transcription factors
  - Produce a count matrix consisting of the counts of fragment ends (or cut sites) within each peak region for each barcode, filtered to consist of only cell barcodes
- Dimensionality reduction
  - Latent Semantic Analysis (LSA)
  - Can be replaced by Principal Component Analysis (PCA) or Probabilistic Latent Semantic Analysis (PLSA)
  - t-SNE (Barnes-Hut-SNE algorithm).
  - Number of dimensions is fixed to 15 as it was found to sufficiently separate clusters visually and in a biologically meaningful way when tested on peripheral blood mononuclear cells (PBMCs).
- Cell clustering
  - k-means for PCA or k-medoids for [P]LSA
  - k-NN
- Peak annotation
  - Use `bedtools closest -D=a` to associate each peak with genes based on closest transcription start sites (packaged within the reference) such that the peak is within 1000 bases upstream or 100 bases downstream of the TSS.
- TF binding sites detection
  - calculate the GC% distribution of peaks and then bin the peaks into equal quantile ranges
  - Use the MOODS Python library to scan each peak for matches to motif position-weight-matrices (PWMs) for transcription factors from the JASPAR database
  - P-value threshold of 1E-7
- TF motif enrichment

  - For each TF, calculate the total cut-sites in a cell barcode for peaks that share the TF motif
  - Calculate the proportion of cut-sites for a TF within a barcode out of the total cut-sites for that barcode
  - Modified median-based z-score
- Cluster differential accessibility
  - Test, for each motif and each cluster, whether the in-cluster mean differs from the out-of-cluster mean
  - Use the quick and simple sSeq method which employs a negative binomial exact test, or, when the counts become large, switch to the fast asymptotic beta test used in edgeR.
  - Produce a list of genes that are differentially expressed in that cluster relative to the rest of the sample (that really looks like a copy-paste error from Single Cell Gene Expression Solution).