ATAC-seq Data Analysis

Leila Taher

WS 2022/2023

Contents

1 The open chromatin landscape of human lymphocytes

Many factors, such as the chromatin structure, the position of the nucleosomes, and histone modifications, play an important role in the organization and accessibility of DNA which in turn influence activation and inactivation of genes. Assay for Transposable-Accessible Chromatin (ATAC) seq is used to asses chromatin accessibility by letting Tn5 transposase insert DNA sequences corresponding to truncated adapters into open regions of the genome and concurrently, the DNA is sheared by the transposase activity.

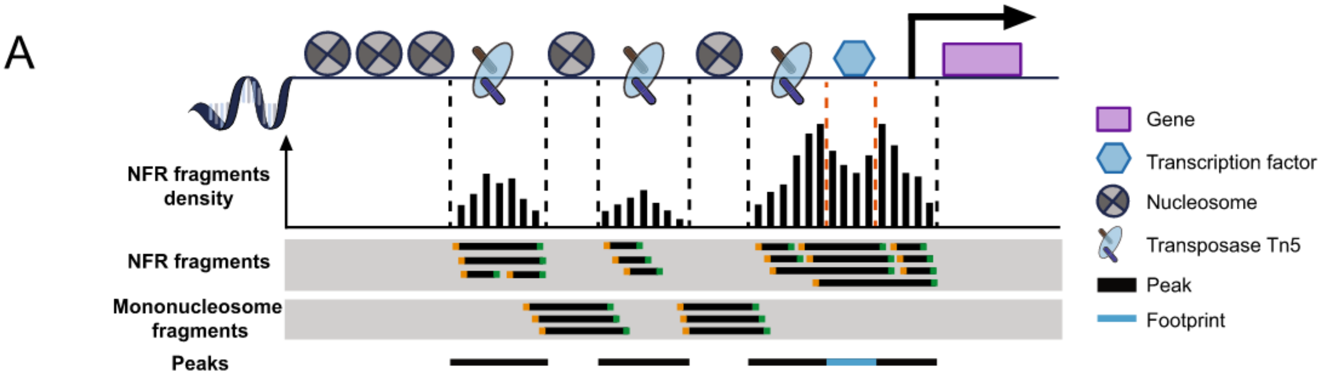


Figure 1: **ATAC-seq experiment:** Tn5 binds and cuts open chromatin and simultaneously ligates adapters. The fragments are sequenced to identify open chromatin regions (black) and footprints (blue). Nucleosome free region (NFR) fragments represent open chromatin, while nucleosome-bound fragments reflect nucleosome positions (gray shaded tracks).Yan et al, Genome Biology, 2020.

You will be working with data from the ENCODE project. Specifically, these data were generated by performing ATAC-seq on human cell line GM12878 produced in the laboratory of Michael Snyder. GM12878 is a lymphoblastoid cell line derived from the blood of a female donor with northern and western European ancestry. Lymphoblastoid cell lines are established by *in vitro* transformation of human lymphocytes by Epstein Barr Virus (EBP).

2 Preprocessing ATAC-seq data

The BAM-formatted files shipped with this tutorial contain the filtered mappings of the (paired-end) sequencing reads from each of three samples derived from the same donor. The data were preprocessed following the ENCODE ATAC-seq pipeline v.1.8.0. `Bowtie2` was used to map the sequencing reads to the GRCh38/hg38 assembly of the human genome. To simplify the analysis, only the reads mapping to chromosome 22 are included in the files.

File name	Accession	Source
ENCFF815ABY_chr22.bam	ENCFF815ABY	`https://www.encodeproject.org/files/ENCFF815ABY/`
ENCFF691PRG_chr22.bam	ENCFF691PRG	`https://www.encodeproject.org/files/ENCFF691PRG/`
ENCFF726MIS_chr22.bam	ENCFF726MIS	`https://www.encodeproject.org/files/ENCFF726MIS/`

Table 1: Datasets analyzed in this tutorial.

- Write all commands needed to complete the tasks in a shell script, so that you can easily repeat the analysis if necessary.
- Reminder: when running functions on multiple files it may be useful to use `for` loops. `sed` is useful for generating new output file names from the input. For example:

```
for INPUT in data/*chr22.bam; do
  OUTPUT=$(echo $INPUT | sed "s/\.bam/\.newname\.bam/" |
    sed "s/data\//new_directory\//" )
  echo -e "input:   $INPUT"
  echo -e "output:  $OUTPUT\n"
done
```

results in:

```
input:   data/ENCFF691PRG_chr22.bam
output:  new_directory/ENCFF691PRG_chr22.newname.bam

input:   data/ENCFF726MIS_chr22.bam
output:  new_directory/ENCFF726MIS_chr22.newname.bam

input:   data/ENCFF815ABY_chr22.bam
output:  new_directory/ENCFF815ABY_chr22.newname.bam
```

2.1 Indexing BAM-formatted files

Many of the tools that use BAM-formatted files require indexed files. Indexing aims to achieve a fast retrieval of mappings overlapping a specified region without going through the whole file. BAM-formatted files must be sorted by the name of the reference sequence (e.g., the chromosome) and then the leftmost coordinate before indexing. You can do this with SAMtools:

```
# Sorting:
samtools sort input_bam_file -o output_bam_file
# Indexing:
samtools index output_bam_file
```

Download and install the tool if necessary.

1. Run the command for each of the BAM files in Table 1.

Use SAMtools view to count the number of mapped reads (not the number of mappings!):

```
samtools view -c -F 260 input_bam_file
```

- `-c`: instead of printing the alignments, only count them and print the total number.
- `-F INT`: do not output alignments with any bits set in `INT` present in the `FLAG` field.

2. What does `-F 260` mean? How many reads are in the files?

2.2 Filtering uninformative reads

Next, you will use SAMtools to apply some filters on the mapped reads. Because the mitochondrial genome is nucleosome-free and thus very accessible to Tn5 insertion, ATAC-Seq datasets often have many reads mapping to it. The data have already been filtered to remove these reads. Otherwise, you would have to do it. In addition, you should also remove the reads with low mapping quality and those that are not *properly paired*. A pair of reads (R_1,R_2) arising from the same fragment are properly paired if and only if:

- R_1 is on the forward strand and R_2 is on the reverse strand, or
- R_2 is on the forward strand and R_1 is on the reverse strand.

Given a BAM-formatted file called "`input_bam_file`", the command:

```
samtools view -q 30 -f 2 -b -h input_bam_file > output_bam_file
```

extracts read pairs that map with a mapping quality equal to or greater than 30 (`mapQ` $\geq$ 30) and that are properly paired (`-f 2`).

- `-b` indicates that the output should be a BAM-formatted file.
- `-h` preserves the header.

The output is written to the standard output and redirected to a file called "`output_bam_file`".

3. Run the command for each of the output BAM files from 1. How many mapped read pairs are left in each case?

2.3 Filtering duplicate reads

Because of the PCR amplification, there might be read duplicates (different reads mapping to exactly the same genomic region) from over-amplification of some regions. Tn5 insertion within accessible regions is random. Therefore, only few fragments are expected to originate from the exact same genomic location. We will consider such fragments to be PCR duplicates and remove them using Picard `MarkDuplicates`:

```
java -jar /path/to/picard.jar MarkDuplicates \
      REMOVE_DUPLICATE=true \
      I=input_bam_file \
      O=filtered_duplicates.bam \
      M=dup_metrics.txt
```

Picard is a collection of command-line tools for handling high-throughput sequencing data. Download and install the tool if necessary.

4. Run the command for each of the output BAM files from 3. Index the resulting BAM files.

   `MarkDuplicates` also produces a metrics file (`dup_metrics.txt`) indicating the numbers of duplicates for both single- and paired-end reads. See the manual for a detailed explanation.

- How many read pairs are duplicates?

2.4 Checking the size of the fragments

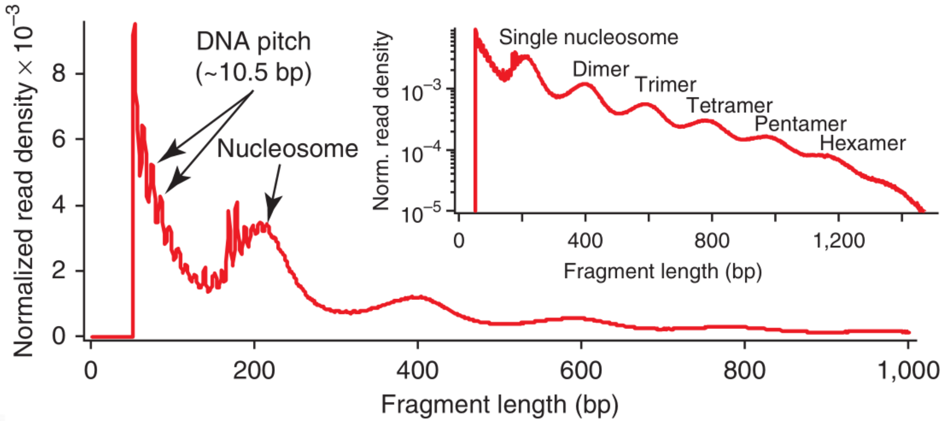
The fragment length distribution of an ATACSeq library provides a good indication of its quality. You can check the insert sizes of the read pairs using Picard `CollectInsertSizeMetrics`. The **insert size** is the distance between two reads arising from the same fragment.

```
java -jar /path/to/picard.jar CollectInsertSizeMetrics \
      I=input_bam_file \
      O=insert_size_metrics.txt \
      H=insert_size_histogram.pdf
```

5. Run the command for each of the output BAM files from 4.

   Examine the resulting histogram in “insert_size_histogram.pdf”. You expect something similar to Figure:

   - A peak at < 100bp, where Tn5 inserted into NFRs.
   - A peak at ~ 200bp, where Tn5 inserted around a single nucleosome.
   - A peak at ~ 400bp, where Tn5 inserted around two adjacent nucleosomes (dimer).
   - A peak at ~ 600bp, where Tn5 inserted around three adjacent nucleosomes (trimer).



Source: *Buenrostro et al, Nat. Methods, 2013*

6. Why do the reads peak around nucleosomes, are there no fragments with lengths in between?

3 Peak calling

In order to find potentially open chromatin regions in the genome, we will identify regions where reads are statistically enriched. This can be done using MACS2, as follows:

```
macs2 callpeak -t input_bam_file \
     -n input_bam_file \
     --outdir output_folder \
     -f BAMPE \
     -g 2.9e9 \
     --nomodel \
     --call-summits
```

- `-t/--treatment FILENAME(S)`: the file with the read mappings. If multiple files are specified, MACS will pool up all these files together.
- `-n/--name NAME`: MACS will use `NAME` as prefix for the output files.
- `--outdir FOLDER`: MACS2 will save all output files into `FOLDER`. A new folder will be created if necessary.
- `-f BAMPE`: input format is (paired-end) BAM.
- `-g INT`: the mappable size of the human genome.
- `--nomodel`: MACS2 will not build a model for the shift. There is no need to build a model because MACS2 will use the fragment information from the paired-end reads.
- `-call-summits`: MACS will reanalyze the shape of the called peaks to identify subpeaks.

7. Call peaks on each of the output BAM files from 4.

8. How many peaks do you obtain for each replicate? Visualize (e.g., with R) using a bar chart.

3.1 Finding peaks that overlap across replicates

Motifs are short, reoccurring patterns in DNA that are presumed to have biological function such as transcription factor binding sites. HOMER (Hypergeometric Optimization of Motif EnRichment) is a suite of tools for Motif Discovery and next-generation sequencing analysis. HOMER contains many useful tools for analyzing different types of functional genomics sequencing data, including ATAC-seq. Download and install the tool if necessary.

Among others, HOMER provides a utility for comparing sets of peaks called `mergePeaks`. By default, the tool takes two or more peak sets and returns a single peak set, which is output to the standard output:
`mergePeaks` now now looks for specific, literal overlaps by default (`-d` given)

```
mergePeaks -d <maximum distance to merge> <peak_file1> <peak_file2> <peak_file3> >
MergedPeakFile.txt
```

Peaks within the distance in `-d <#>`bp of eachother will be reported as the average position between the peaks found within the common region. Peaks outside this range are also retained.

(i) merged peak name (will start with "Merged-")

(ii) chromosome

(iii) start (average from merged peaks)

(iv) end (average from merged peaks)

(v) strand

(vi) average peak score (actually, the average of the original values in column 6 of the peak files - or column 5 of BED files)

(vii) original peak files contributing to the merged peak

(viii) number of peaks merged (occasionally more than one peak from a single file will be merged if the peaks are within the specify distance or two or more peaks from one file overlap with the same single peak(s) from another file)

Merging multiple peak sets into a single peak set is useful for certain types of analysis, such as comparing the read-densities between peaks from separate experiments.

9. Merge the peak sets resulting from 7 for a distance of `d=1,25,50,100`bp.
   Save the outputs as `MergedPeakFile_d.txt`. Count the number of resulting peaks.

10. Why do the number of peaks reduce as `d` increases?

11. Generate two `bed` files from the merged file for `d=50`. This file will contain three columns with values: `chrNr start end`. One file will consist of the peaks present in only one of the original peak sets; the other will consist of peaks present in all three of the original peak sets. Hint: Refer back to our first Linux exercise. You can achieve this task with `bash` commands: `cut`, `awk` and `cat`. With `awk`, you can extract a conditional as follows:

```
awk '{if ($N>=3) {print } }'
```

    where `$N` is the column number

    - How many peaks do the each of the resulting consensus peak sets consist of?

3.2 Visualization with deepTools

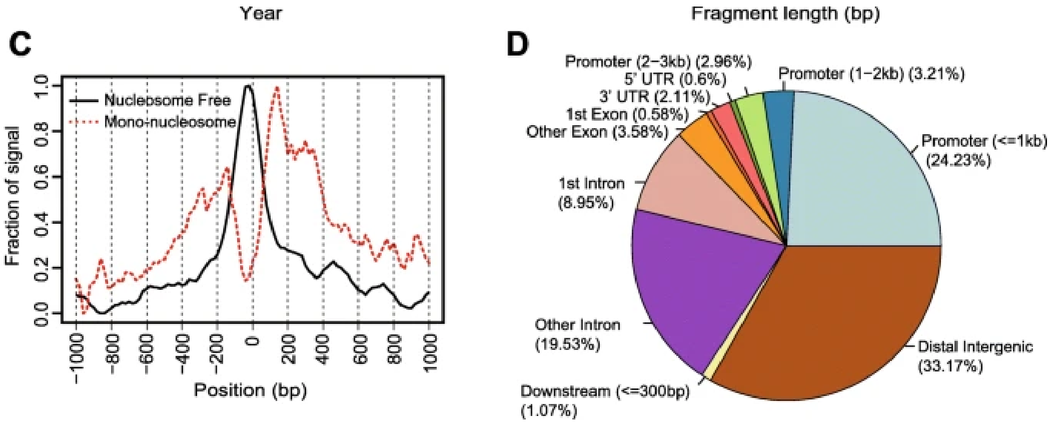


Figure 2: **C** Typical TSS enrichment plot shows that nucleosome-free fragments are enriched at TSS, while mono-nucleosome fragments are depleted at TSS but enriched at flanking regions. **D** Typical peak annotation pie chart shows that more than half of the peaks fall into enhancer regions (distal intergenic and intronic regions), and only around 25% of the peaks are in promoter regions. Yan et al, Genome Biology, 2020.

deepTools is a suite of python tools for efficient analysis of high-throughput sequencing data. Download and install the tool if necessary.
The `bamCoverage` tool takes a BAM-formatted file as input and generates a *coverage track*, in bigWig or bedGraph format, as output. The coverage is calculated as the number of reads per *bin*, where the bins are short consecutive counting windows of a defined size. `BigWig` files are much smaller than `BAM` files, especially as your bin size increases. The coverage can be normalized, which is important for comparing different samples to each other (because they will most likely vary in terms of sequencing depth).

12. Execute the following command for each of the `BAM` files from 4:

```
bamCoverage \
--binSize 10 \
--extendReads \
--ignoreDuplicates \
--normalizeUsing RPKM \
--bam input_bam_file \
--outFileFormat bigwig \
--outFileName output_bigwig_file \
--numberOfProcessors 2
```

- `--binSize/-bs INT`: size of the bins, in base pairs (Default: 50).

- `--extendReads/-e`: extend the reads to the size of the fragments.

- `--ignoreDuplicates`: consider only once (pairs of) reads that have the same orientation and start position.

- `--normalizeUsing RPKM`: normalize the number of reads per bin using “Reads Per Kilobase per Million mapped reads” (RPKM); other possible choices are `CPM` (Counts Per Million mapped reads); `BPM` (Bins Per Million mapped reads); `RPGC` (reads per genomic content, i.e., mapped reads are considered after blacklist filtering, if applied).

- `--outFileFormat/-of bigwig`. bigWig files are compressed, binary files. If you would like to see the coverage values, choose “`bedGraph`” as output format.

Given one or more bigWig-formatted files and a set of genomic coordinates, the `computeMatrix` tool constructs a matrix that can be used for visualization. We will try it on the bigWig files generated in 12 and the consensus peaks, as follows:

```
computeMatrix reference-point -S bigWig_file1 bigWig_file2 ... \
-R consensus_peaks \
--beforeRegionStartLength=1500 --afterRegionStartLength=1500 \
--referencePoint=center \
-o matrix_file_name.gz
```

`computeMatrix` will then:

(i) Subtract the value indicated in `--beforeRegionStartLength/-b` from the center of the regions in the BED-formatted file.

(ii) Add the value indicated in `--afterRegionStartLength/-a` to the center of the regions in the BED-formatted file.

(iii) Split the resulting region into 50 bp bins (or the value specified by `--binSize`).

(iv) Calculate the mean of the values in the bigWig files for each of the bins.

(v) Write out a matrix where each row corresponds to a region in the BED file and each column to a bigWig file.

Finally, the `plotHeatmap` command can be used to visualize the matrix:

```
plotHeatmap -matrixFile matrix_file_name.gz -heatmapWidth 10 -outFileName output_file.png
```

13. Generate a heatmap visualizing the ATAC-seq signal for both of the consensus peaks. plotHeatmap has several parameters that you can adjust to beautify the image (see `https://deeptools.readthedocs.io/en/develop/content/tools/plotHeatmap.html`).

14. What do the rows and columns in the heatmap represent?

15. What do the colors in the heatmap correspond to?

16. Do you expect or see a difference in the samples?

17. What can you say about the differences in the overall png files?

3.3 Motif analysis

HOMER implements a *differential* motif discovery algorithm. It takes two sets of sequences and identifies the motifs (regulatory elements) that are enriched in on set relative to the other.Specifically, it uses ZOOPS scoring (zero or one occurrence per sequence) coupled with a hypergeometric (or binomial) test to determine significance. See `http://homer.ucsd.edu/homer/introduction/basics.html` and `http://homer.ucsd.edu/homer/motif/` for more details.

18. Run a motif analysis of the consensus peaks supported by at least three peaks in the original peak sets:

```
findMotifsGenome.pl <peak_file> hg38 <output_dir> -size 200 -mask
```

19. Explore the generated html file

4 Version information

4.1 samtools

```
samtools --version
samtools 1.10
Using htslib 1.10.2
Copyright (C) 2019 Genome Research Ltd.
```

4.2 java

```
java -version
openjdk version "11.0.17" 2022-10-18
OpenJDK Runtime Environment (build 11.0.17+8-post-Ubuntu-1ubuntu220.04)
OpenJDK 64-Bit Server VM (build 11.0.17+8-post-Ubuntu-1ubuntu220.04, mixed mode,
    sharing)
```

4.3 Picard

```
java -jar $PICARD_PATH/picard.jar MarkDuplicates -version
Version:2.23.8
```

4.4 macs2

```
macs2 --version
macs2 2.2.7.1
```

4.5 deepTools

```
deeptools --version
deeptools 3.5.1
```

4.6 Homer

```
v4.11, 10-24-2019
```