Bioinformatic approaches to regulatory genomics and epigenomics

376-1347-00L | week 07

Pierre-Luc Germain

ETH Zürich

Today’s plan

(update epiwraps)

- Debriefing on the assignment
- DNA accessibility
- ATAC-seq analysis (practical)
- Nucleosome positioning

Debriefing on the assignments

Searching for matches: choose the `subject` carefully !

```{r, proportions overlap}
moi_chr1 <- matchMotifs(motif2, subject=peaks, genome=genome, out="positions")
moi_chr1 <- moi_chr1[[1]]

ov1 <- sum(overlapsAny(moi_chr1, peaks, type="within"))/length(moi_chr1)
ov1
```

```
[1] 1
```

Debriefing on the assignments

Searching for matches: choose the `subject` carefully !

```{r, proportions overlap}
moi_chr1 <- matchMotifs(motif2, subject=peaks, genome=genome, out="positions")
moi_chr1 <- moi_chr1[[1]]

ov1 <- sum(overlapsAny(moi_chr1, peaks, type="within"))/length(moi_chr1)
ov1
```

```
[1] 1
```

=> this will only search for motif matches within peaks. So naturally all of these motif matches will be within peaks

Debriefing on the assignments

Searching for matches: choose the `subject` carefully !

```{r, proportions overlap}
moi_chr1 <- matchMotifs(motif2, subject=chr1_coords, genome=genome, out="positions")
moi_chr1 <- moi_chr1[[1]]

ov2 <- sum(overlapsAny(moi_chr1, peaks, type="within"))/length(moi_chr1)
ov2
```

```
[1] 0.002501695
```

=> correct

(an alternative would be to use the *sequence* of chr1 as `**subject**`)

Debriefing on the assignments

Finding motif instances **`within`** peaks

```{r, get proportion of peaks with a motif}
ov <- findOverlaps(peaks, moi_peaks, type="within")
peakOv <- unique(subjectHits(ov))

ov1 <- length(peakOv)/length(peaks)
ov1
```

```
[1] 0
```

Debriefing on the assignments

Finding motif instances **`within`** peaks

```{r, get proportion of peaks with a motif}
ov <- findOverlaps(peaks, moi_peaks, type="within")
peakOv <- unique(subjectHits(ov))

ov1 <- length(peakOv)/length(peaks)
ov1
```

```
[1] 0
```

=> this code is actually looking for peaks within motif matches…
peaks are much larger than motif matches!

Debriefing on the assignments

Finding motif instances **`within`** peaks.

Turn around the arguments and use `unique()` on the `subjectHits()`, i.e. the peaks containing a motif strictly within

```{r, get proportion of peaks with a motif}
ov <- findOverlaps(moi_peaks, peaks, type="within")
peakOv <- unique(subjectHits(ov))

ov1 <- length(peakOv)/length(peaks)
ov1
```

```
[1] 0.4661041
```

Debriefing on the assignments

Searching for matches: choose the `subject` carefully !

Find the percentage of motifs across the genome within a peak

```{r}
# when subject peak centers, motifs are only (!) searched within the peak centers
moi_peaks <- matchMotifs(motif, subject=peak_centers, genome=genome, out="positions")
sum(overlapsAny(moi_peaks, peak_centers))/length(moi_peaks)
```

```
[1] 1
```

Debriefing on the assignments

Searching for matches: choose the `subject` carefully !

Find the percentage of motifs across the genome within a peak

```{r}
# when subject peak centers, motifs are only (!) searched within the peak centers
moi_peaks <- matchMotifs(motif, subject=peak_centers, genome=genome, out="positions")
sum(overlapsAny(moi_peaks, peak_centers))/length(moi_peaks)
```

```
[1] 1
```

correct:

Find the fraction of motifs across the genome within a peak

```{r}
ah <- AnnotationHub(localHub=FALSE)
genome <- ah[["AH68356"]]
genome_seq <- rtracklayer::import(genome)
genome_coords <-  GRanges(seqnames=Rle(c("chr1")),
                          ranges = IRanges(1, end=200*1e6)) # here for one chromosome

# with findmotifInstances (epiwraps)
moi_genome <- findMotifInstances(genome_seq, motif) #epiwraps

# With matchMotifs (motifmatchr)
moi_genome2 <- matchMotifs(motif2, subject=genome_coords, genome=BSgenome.Mmusculus.UCSC.mm10, out="positions")
```

```{r, get overlaps}
sum(overlapsAny(moi_genome, peak_centers))/length(moi_genome)
```

Of all instances of that motif in the genome, what proportion is bound by the factor (i.e. has a peak)?

```
mmusculus <- import(genome, "2bit", which = as(seqinfo(genome), "GenomicRanges"))
motif_instances_genome <- findMotifInstances(mmusculus, motif, mc.cores=2)
```

```
## Note: motif [motif] has an empty nsites slot, using 100.
```

```
length(motif_instances_genome)
```

```
## [1] 6544422
```

```
motif_with_peaks = overlapsAny(motif_instances_genome, peaks)
sum(motif_with_peaks)
```

```
## [1] 16856
```

```
percentage2 <- sum(motif_with_peaks)/length(motif_instances_genome)*100
percentage2
```

```
## [1] 0.2575629
```

Of the 6544422 motif instances, 16856 (0.2575629%) overlap a peak.

TF: GATA1

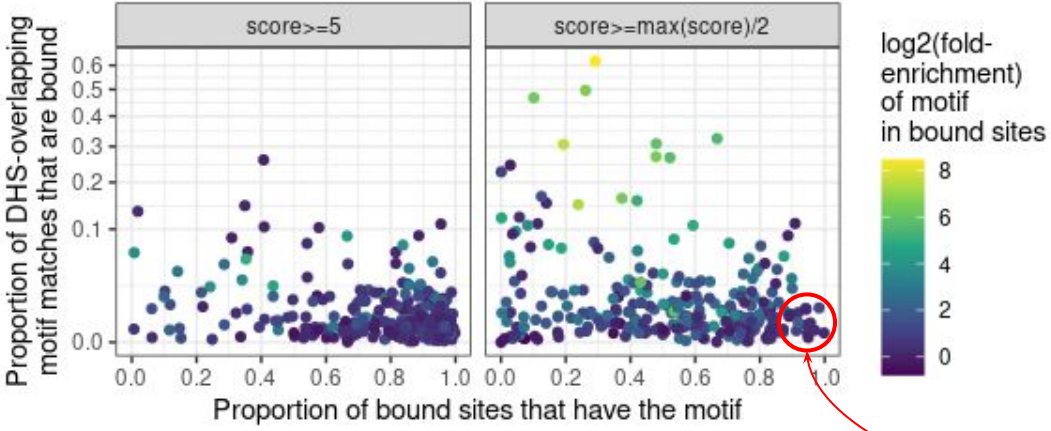
Of the 9675 GATA1 peaks, 7277 (~75%) contain a GATA1 motif, but…

Debriefing on the assignment – wrapping up

Relationship between motif and ChIP-peaks across 260 TFs
(restricting the genome to regulatory elements)

(very lenient motif matching)

(more reasonable motif matching)



Example low-specificity motif (Mdx4) :

Example high-specificity motif (BCL6) :

Most motif matches are not bound

For a large fraction of factors, most binding sites don't show a strong motif

Those TFs for which most of the peaks have a motif also tend to have a very low fraction of the motif matches being bound → low-specificity motif

DNA accessibility, which is associated to lower nucleosome density, **reflects activity**

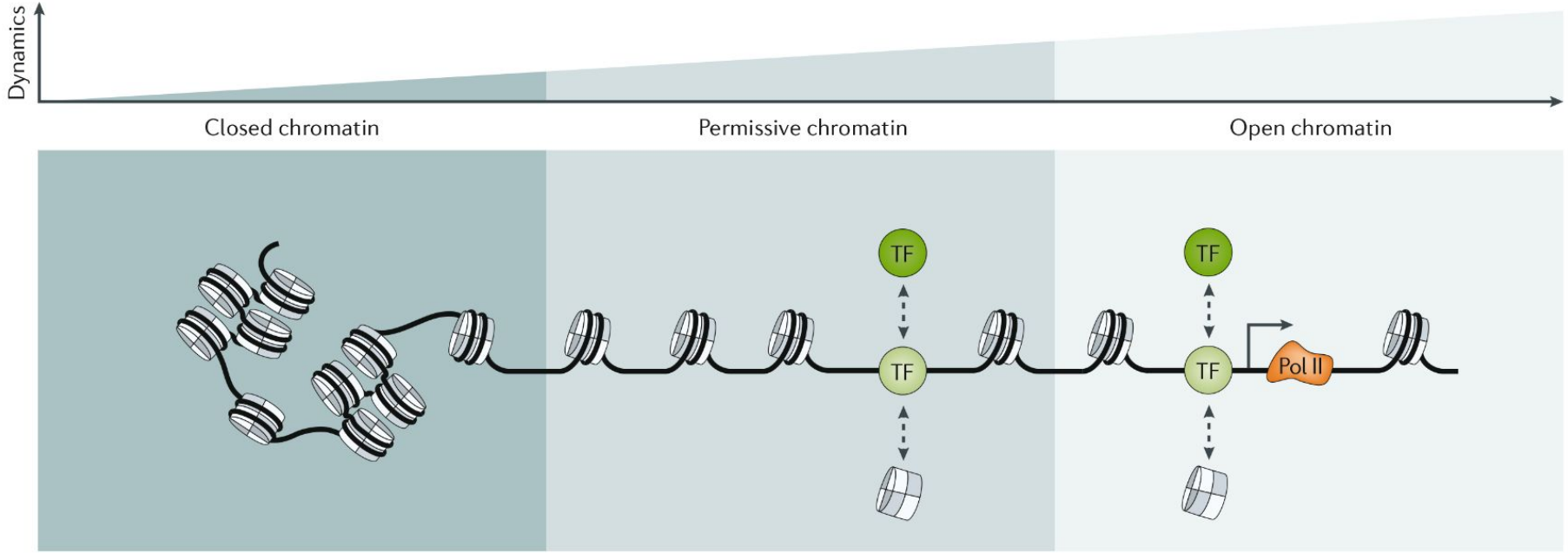
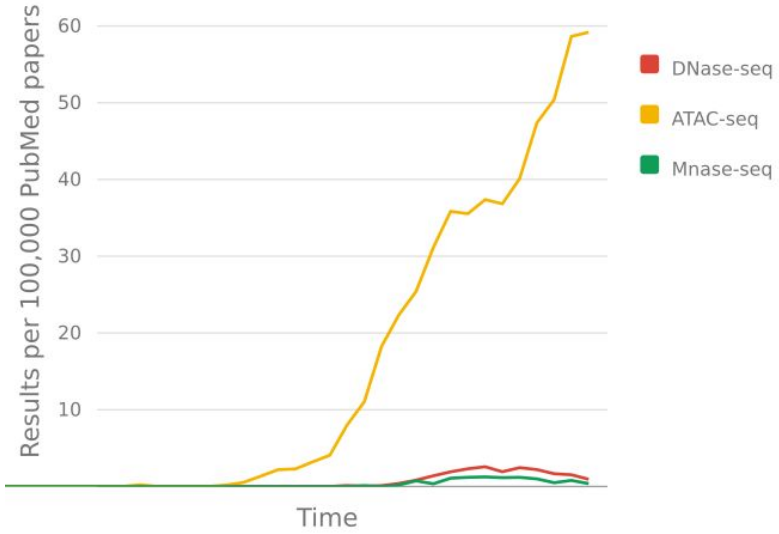


Fig. 1 | **A continuum of accessibility states broadly reflects the distribution of chromatin dynamics across the genome.** In contrast to closed chromatin, permissive chromatin is sufficiently dynamic for transcription factors to initiate sequence-specific accessibility remodelling and establish an open chromatin conformation (illustrated here for an active gene locus). Pol II, RNA polymerase II; TF, transcription factor.

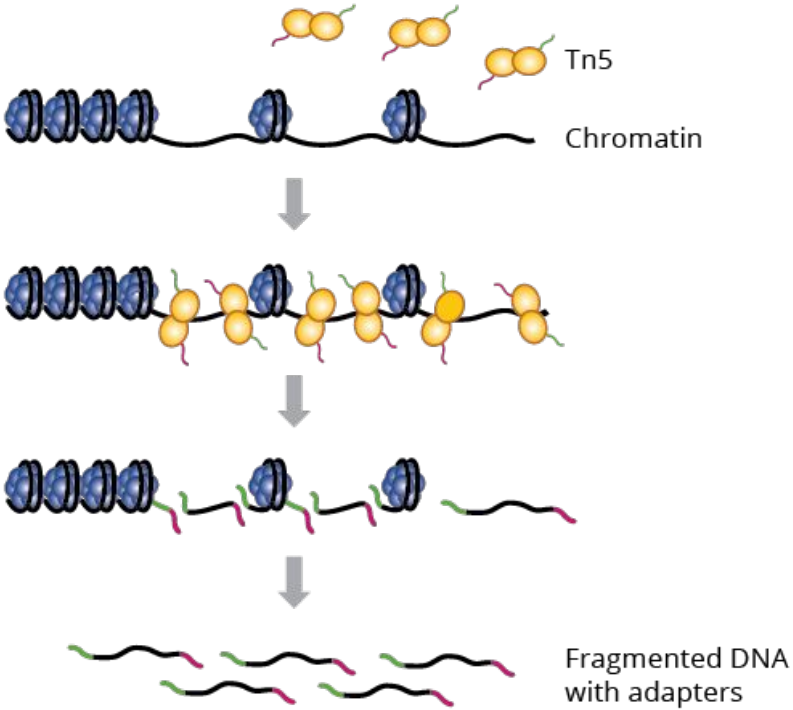
(Klemm, Shipony and Greenleaf, 2019)

Returning to our very brief history of genetics & genomics

…
1900 - Rediscovery of Mendel's work (1860s)
1913 - Chromosomes are linear arrays of genes
1941 - the one-gene-one-enzyme hypothesis
1944 - DNA is the genetic material
1951 - First protein sequenced
1977 - DNA sequencing
1977 - Eukaryotic genes are spliced
1995 - First bacterial genomes sequenced
2000 - Next Generation Sequencing (NGS)
2001 - Draft of the human genome
2003 - RNA-seq
2006 - ChIP-seq
2008 - DNAse-seq, MNase-seq
2012 - ATAC-seq
} Accessibility assays
…

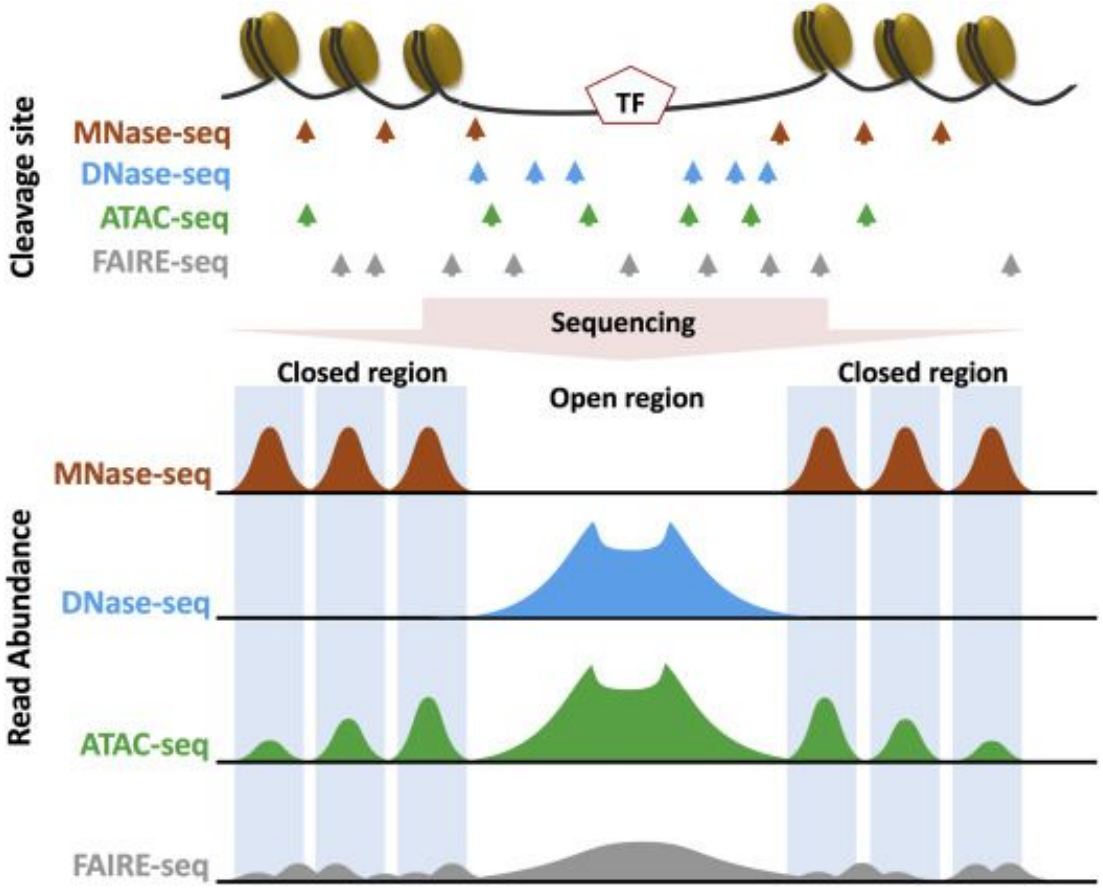


ATAC-seq



ATAC-seq recently became extremely popular due to its information content and low material requirement (i.e. # cells)

Chromatin accessibility assays

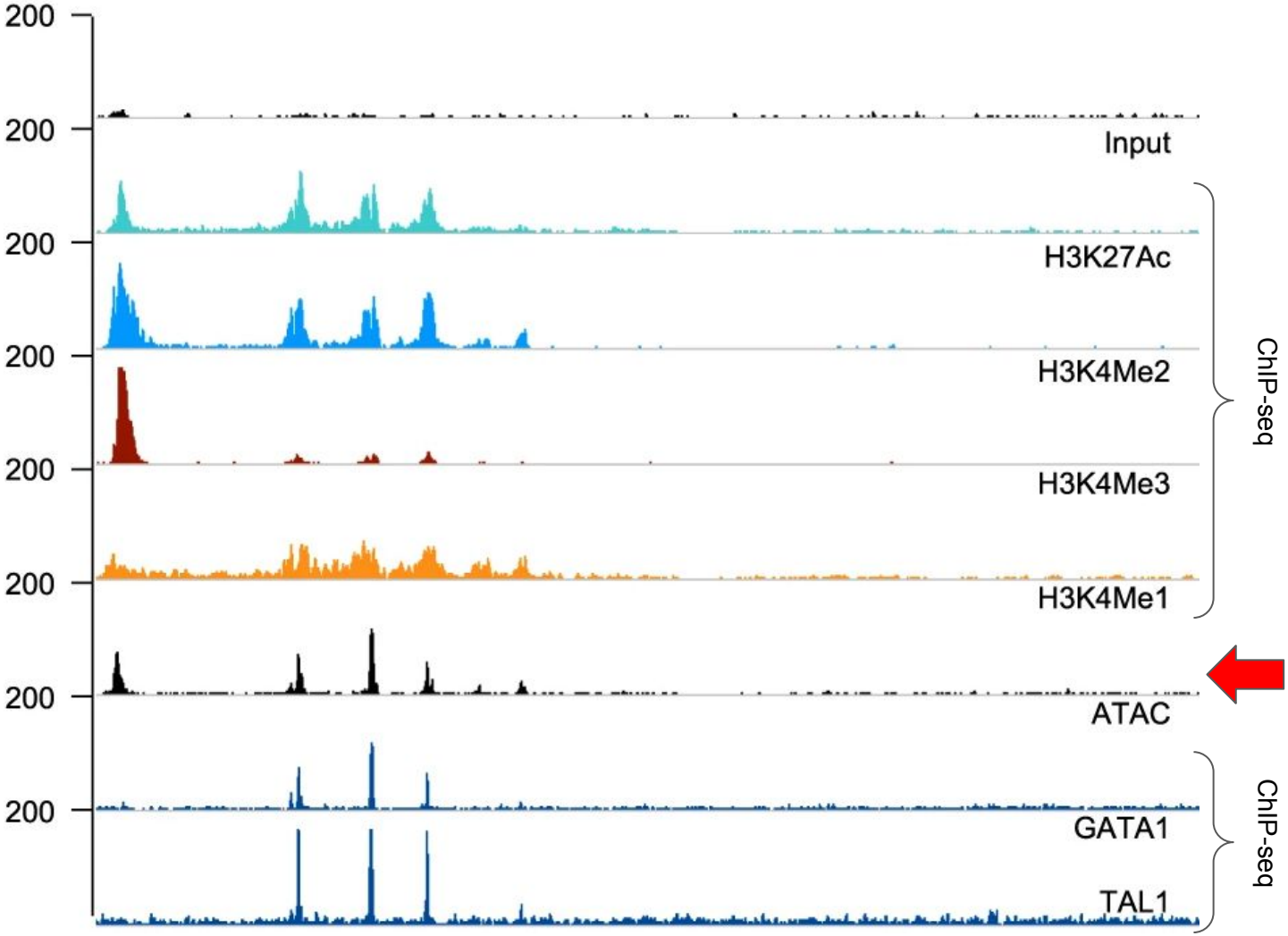


→ Nucleosome positions

→ DNase1-hypersensitive sites (DHS)
→ regulatory elements

Both

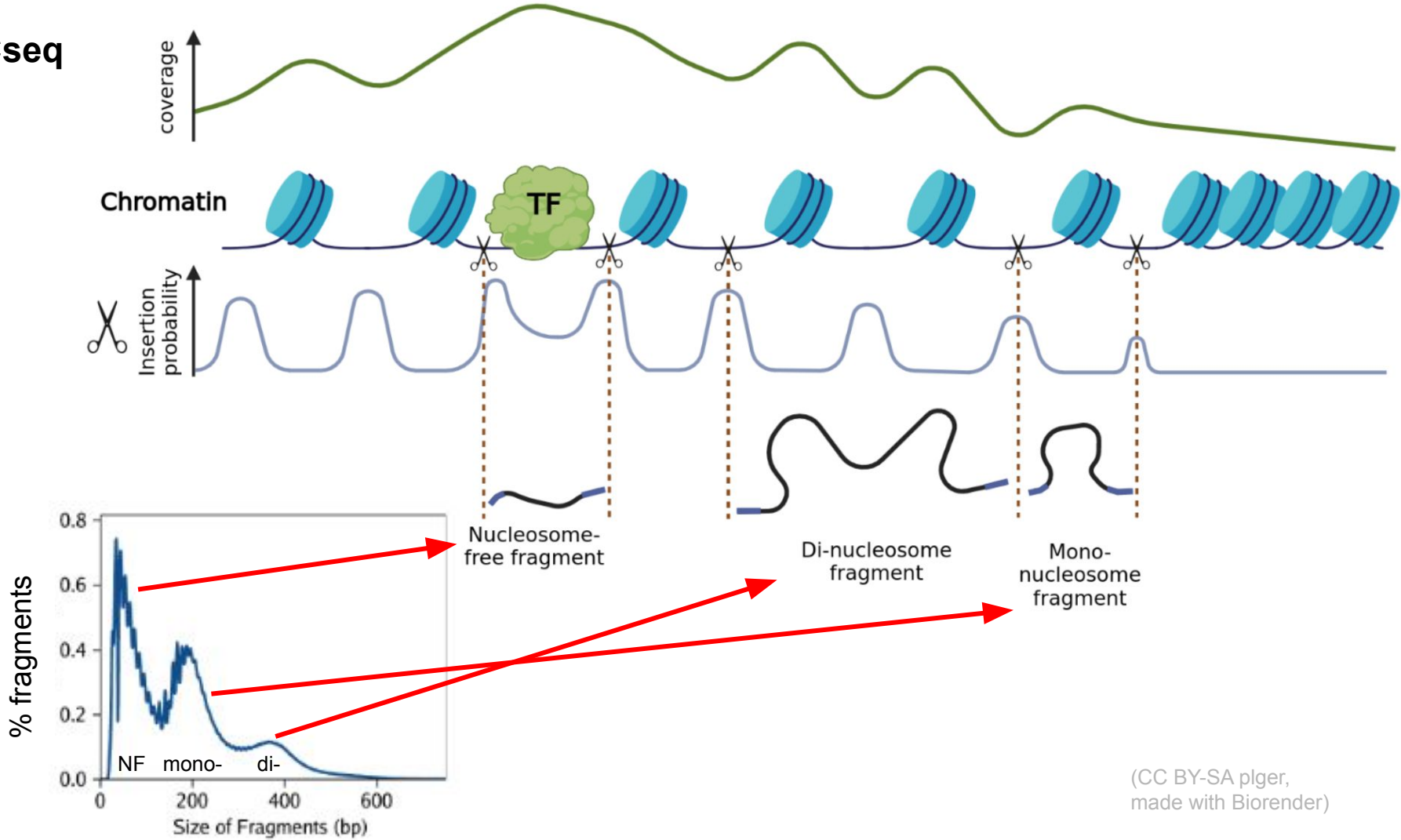
(Hsu et al., Epigenetics in Human Disease, 2018)



ATAC signal tells us that something is happening, it just doesn't tell us what exactly…

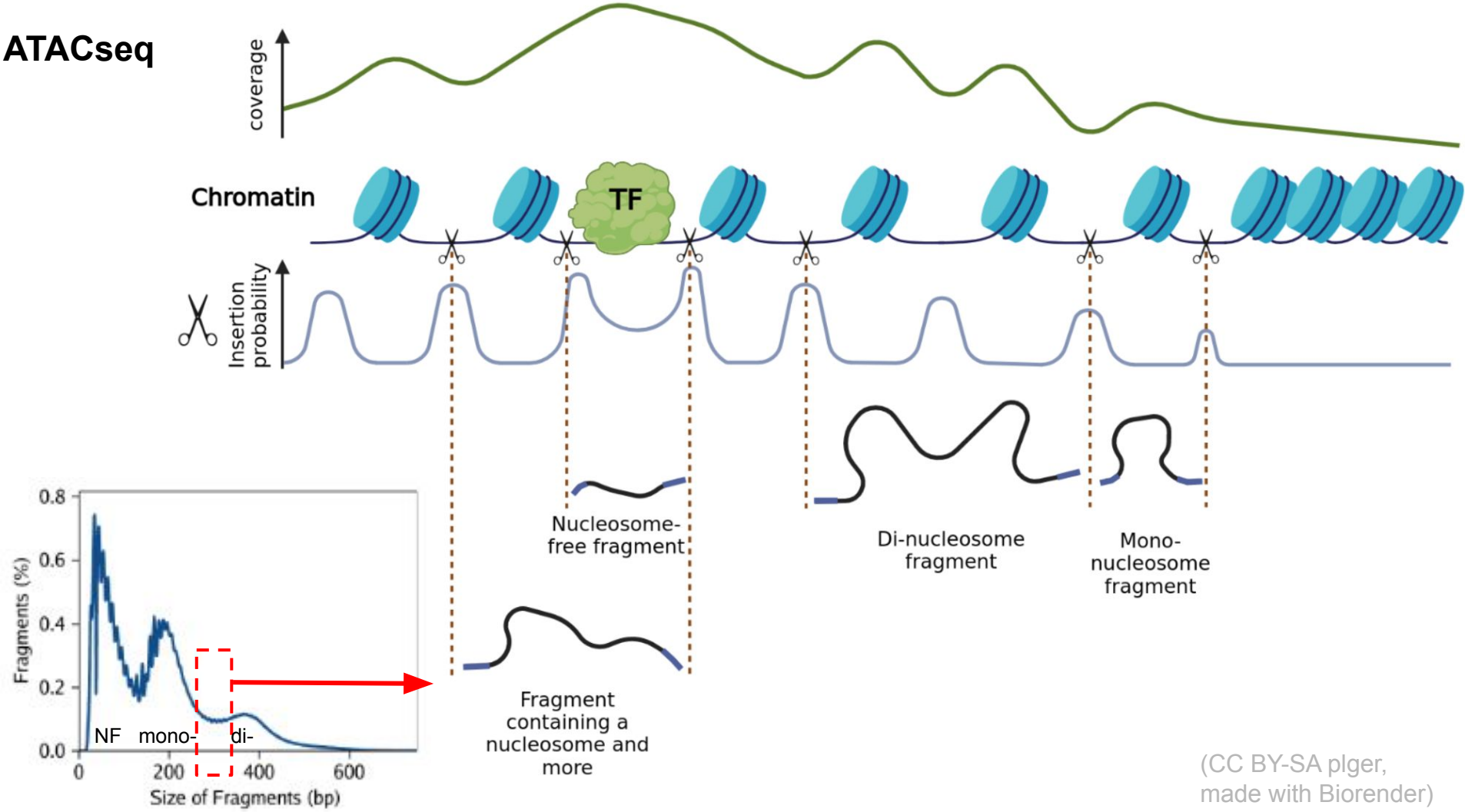
(Adapted from Fox et al., Nat Comm 2020)

ATACseq



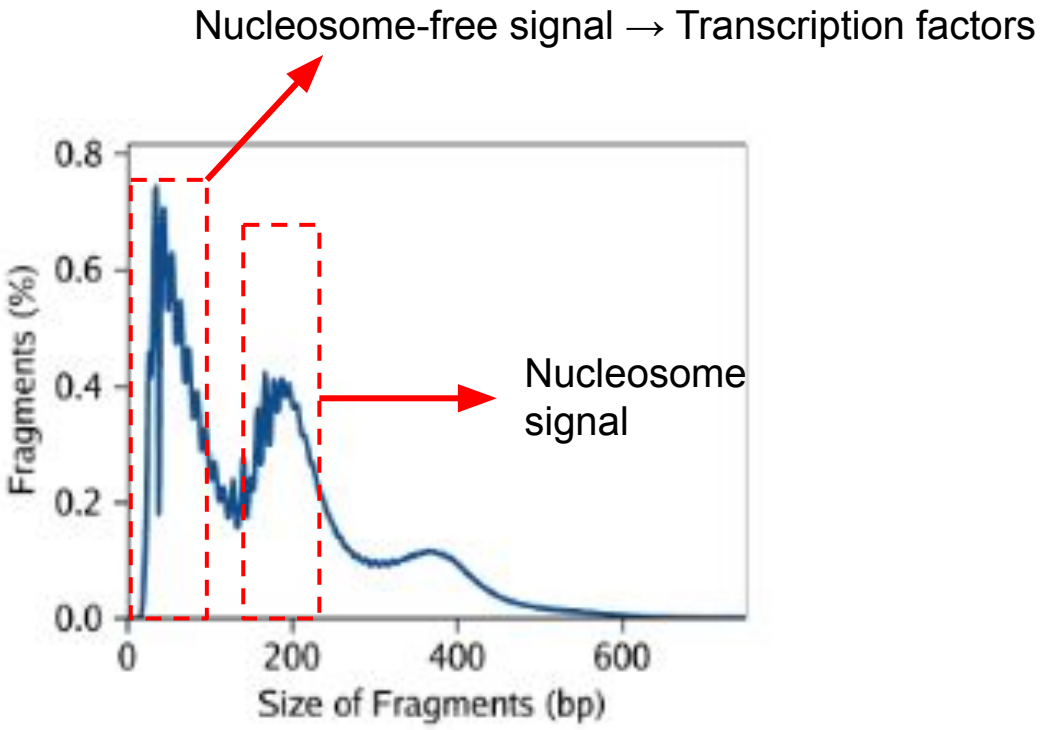
(CC BY-SA plger, made with Biorender)

ATACseq



(CC BY-SA plger, made with Biorender)

This means that once we have the data, we can split the fragments according to size in order to obtain specific information about different kinds of chromatin signals



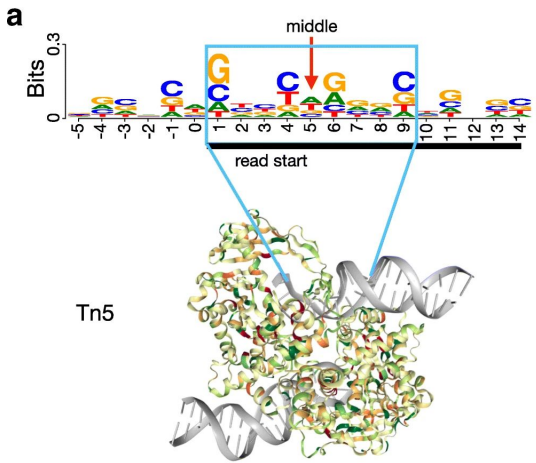
Practical

“Shifting” ATAC-seq alignments

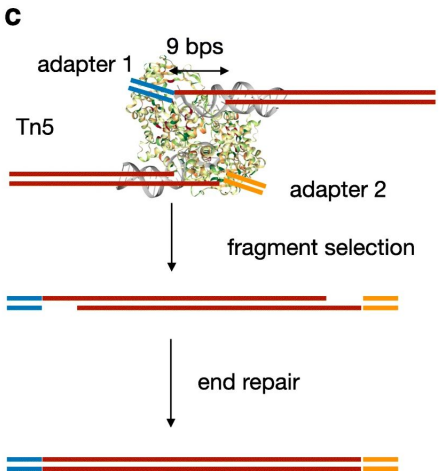
From a given ATAC-seq insertion site, the exact region that is accessible is a few nucleotides from the start of the read

When doing high-resolution things like footprinting, one therefore typically shifts the cut sites by +4/-5nt, so that it is placed in the middle of where the Tn5 was binding

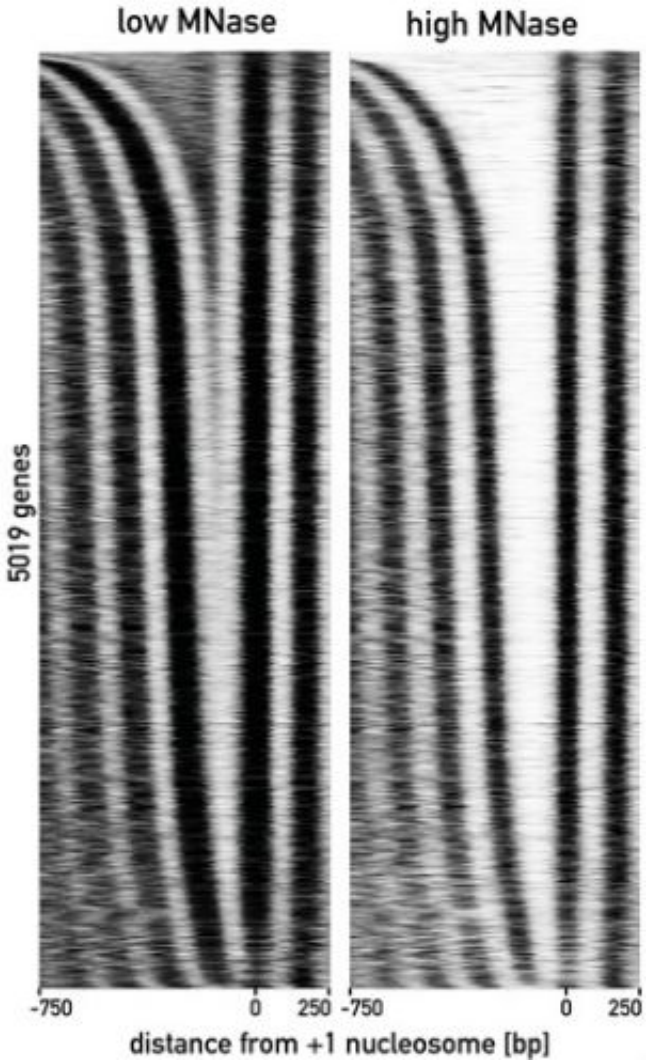
(For most other purposes, this is too fine-grained to make a difference)

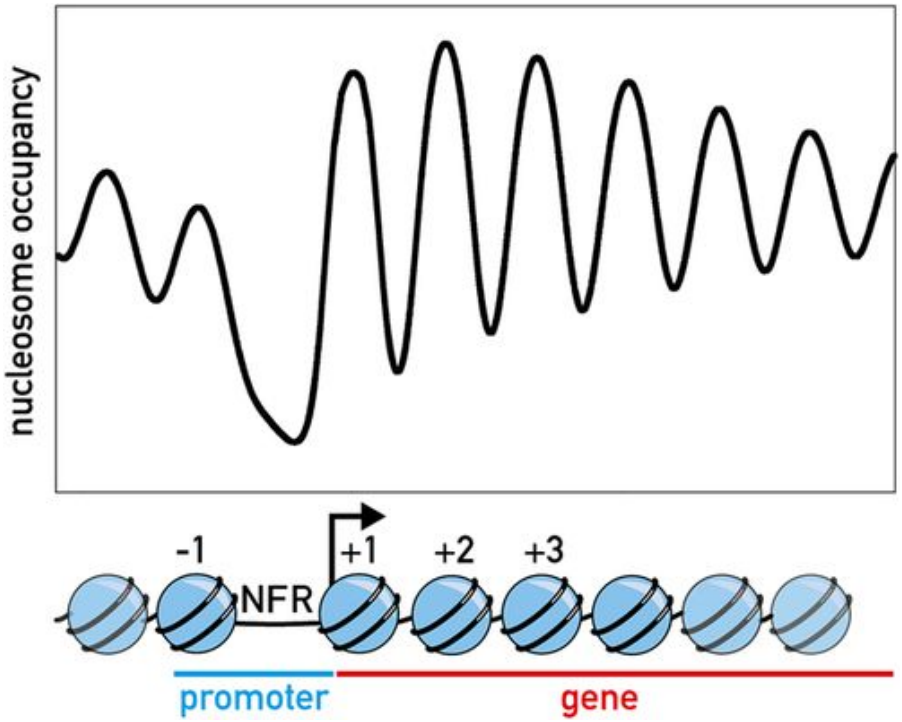


(adapted from Zhijian et al., Genome Biology 2019)

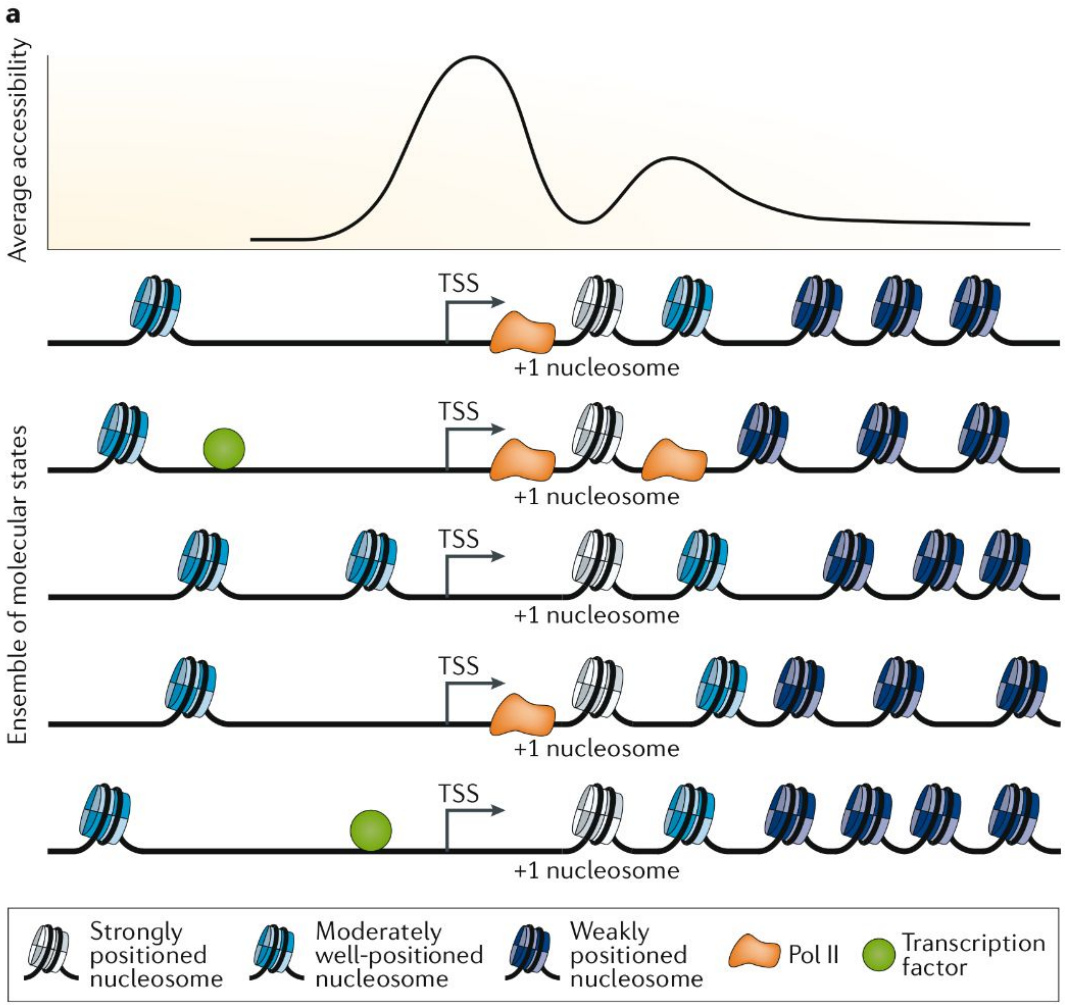


Nucleosome positioning





(Kubik, Bruzzone and Shore, 2017)



(Klemm, Shipony and Greenleaf, 2019)

Assignment

In the same dataset of ATAC on chr19, plot 1) the insertion (i.e. ‘cuts’) profile of nucleosome-free fragments and 2) the centers of nucleosome-containing fragments, around the high-confidence motifs of two factors.
You can choose your own factors of interest, or for instance use KLF4, MAZ and/or FOXD3

Expected form of the answer: 2 figures (one for each factor),
each containing the two signals around the motifs

Don't forget to render your markdown and push it as `assignment.html` !

Next week: Motif accessibility analysis

What if, rather than looking at one motif at a time, we could simply quantify the accessibility/activity of every motif, and compare that across samples/conditions?