Bioinformatic approaches to regulatory genomics and epigenomics

376-1347-00L | week 03

Pierre-Luc Germain

ETH Zürich

Plan for today

- Debriefing on the assignments
- Overview of NGS technologies, ChIP-seq and its analysis
- Practical:
  - primary processing of a ChIP-seq experiment
    (to be continued next week)

Debriefing on the assignments (I/III)

use appropriate filters

```
gns <- genes(ensdb, filter=GeneBiotypeFilter("protein_coding"))
print(paste("all gene ids:", length(gns$gene_id)))
```

```
## [1] "all gene ids: 22287"
```

```
gns <- genes(ensdb, filter=TxBiotypeFilter("protein_coding"))
print(paste("all gene ids:", length(gns$gene_id)))
```

```
## [1] "all gene ids: 22233"
```

Debriefing on the assignments (II/III)

```
exs <- exonsBy(ensdb,
               column=c("tx_id","tx_biotype"),
               filter=TxBiotypeFilter("protein_coding")) # here do not use exons()!
exs[[1]]
```

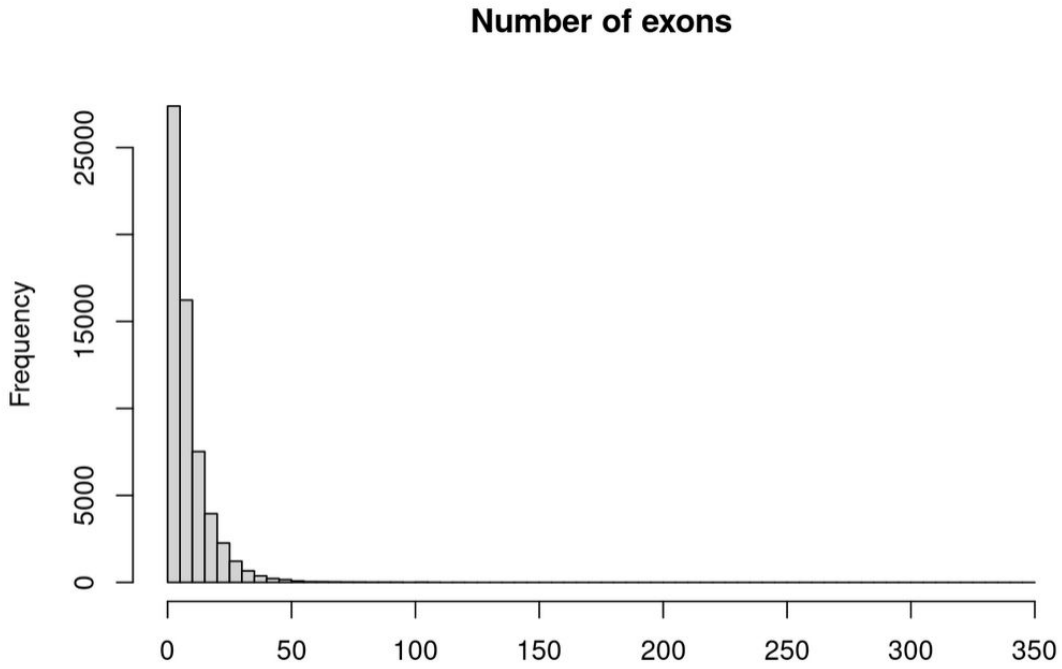
```
## GRanges object with 9 ranges and 4 metadata columns:
##       seqnames              ranges strand |              tx_id     tx_biotype
##          <Rle>           <IRanges>  <Rle> |        <character>    <character>
##   [1]        3 108145888-108146146      - | ENSMUST00000000001 protein_coding
##   [2]        3 108123795-108123837      - | ENSMUST00000000001 protein_coding
##   [3]        3 108123542-108123683      - | ENSMUST00000000001 protein_coding
##   [4]        3 108118301-108118458      - | ENSMUST00000000001 protein_coding
##   [5]        3 108115763-108115891      - | ENSMUST00000000001 protein_coding
##   [6]        3 108112473-108112602      - | ENSMUST00000000001 protein_coding
##   [7]        3 108111935-108112088      - | ENSMUST00000000001 protein_coding
##   [8]        3 108109403-108109612      - | ENSMUST00000000001 protein_coding
##   [9]        3 108107280-108109316      - | ENSMUST00000000001 protein_coding
##                   exon_id exon_rank
##               <character> <integer>
##   [1] ENSMUSE00000334714         1
##   [2] ENSMUSE00000276500         2
##   [3] ENSMUSE00000276490         3
##   [4] ENSMUSE00000276482         4
##   [5] ENSMUSE00000565003         5
##   [6] ENSMUSE00000565001         6
##   [7] ENSMUSE00000565000         7
##   [8] ENSMUSE00000404895         8
##   [9] ENSMUSE00000363317         9
##   -------
##   seqinfo: 104 sequences (1 circular) from GRCm38 genome
```

```
length(exs[[1]])
```

```
## [1] 9
```

Debriefing on the assignments (II/III)

```
nex <- lengths(exs)
hist(nex, breaks=100, main="Number of exons")
```



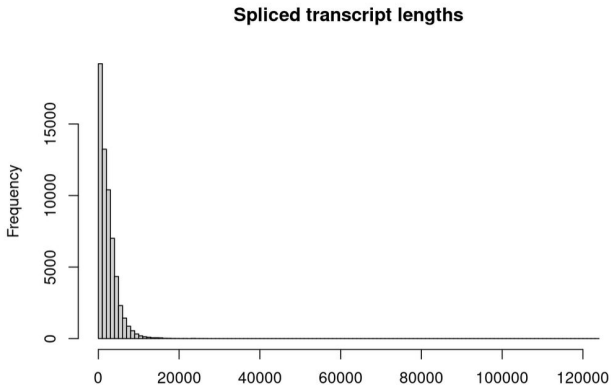
Debriefing on the assignments (II/III)

```
ew <- width(exs)
ew
```

```
## IntegerList of length 60320
## [["ENSMUST00000000001"]] 259 43 142 158 129 130 154 210 2037
## [["ENSMUST00000000003"]] 215 140 68 111 102 52 214
## [["ENSMUST00000000010"]] 602 1972
## [["ENSMUST00000000028"]] 169 195 60 93 138 144 56 ... 162 139 84 119 77 67 127
## [["ENSMUST00000000033"]] 109 163 149 3287
## [["ENSMUST00000000049"]] 115 177 97 77 189 180 198 157
## [["ENSMUST00000000058"]] 326 188 2219
## [["ENSMUST00000000080"]] 361 577 124 3163
## [["ENSMUST00000000087"]] 150 129 63 24 143 71 235 ... 201 192 195 66 143 1051
## [["ENSMUST00000000090"]] 137 117 122 124 145
## ...
## <60310 more elements>
```

```
tl <- sum(ew)

hist(tl, breaks=100, main="Spliced transcript lengths")
```



Debriefing on the assignments (III/III)

Extra: One can order query results (e.g. by date)

```
q <- query(ah, c("Mus Musculus","dna_sm", "2bit", "GRCm38"))
colnames(mcols(q))
```

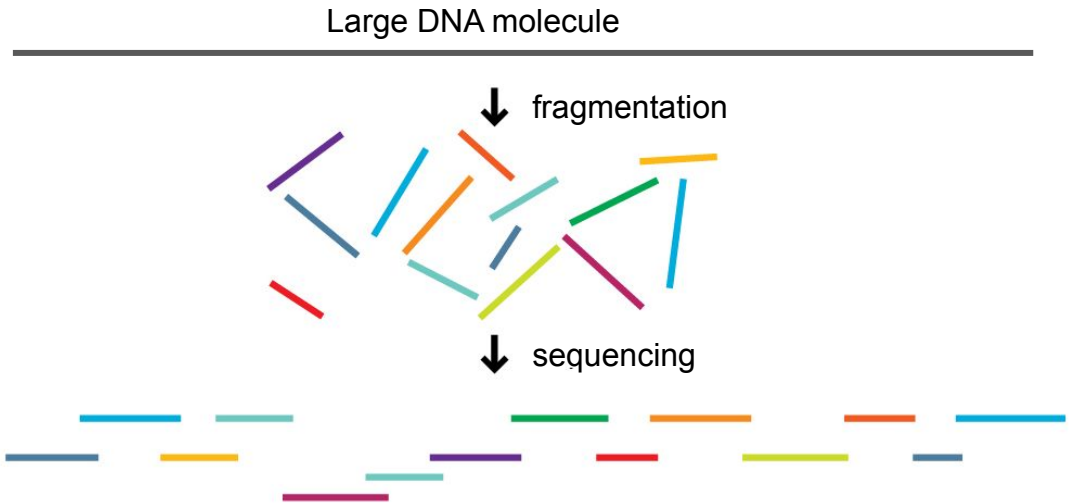
```
##  [1] "title"              "dataprovider"       "species"
##  [4] "taxonomyid"         "genome"             "description"
##  [7] "coordinate_1_based" "maintainer"         "rdatadateadded"
## [10] "preparerclass"      "tags"               "rdataclass"
## [13] "rdatapath"          "sourceurl"          "sourcetype"
```

```
date_added <- mcols(q)[,c("rdatadateadded", "genome")]
date_added[order(date_added$rdatadateadded),]
```

```
## DataFrame with 19 rows and 2 columns
##         rdatadateadded      genome
##            <character> <character>
## AH49775     2015-12-28      GRCm38
## AH50120     2015-12-29      GRCm38
## AH50611     2016-05-03      GRCm38
## AH51299     2016-08-15      GRCm38
## AH51645     2016-11-03      GRCm38
## ...                ...         ...
## AH70177     2019-04-29   GRCm38.p6
## AH77927     2019-10-29   GRCm38.p6
## AH82549     2020-04-27   GRCm38.p6
## AH84787     2020-10-26   GRCm38.p6
## AH88477     2020-10-27   GRCm38.p6
```

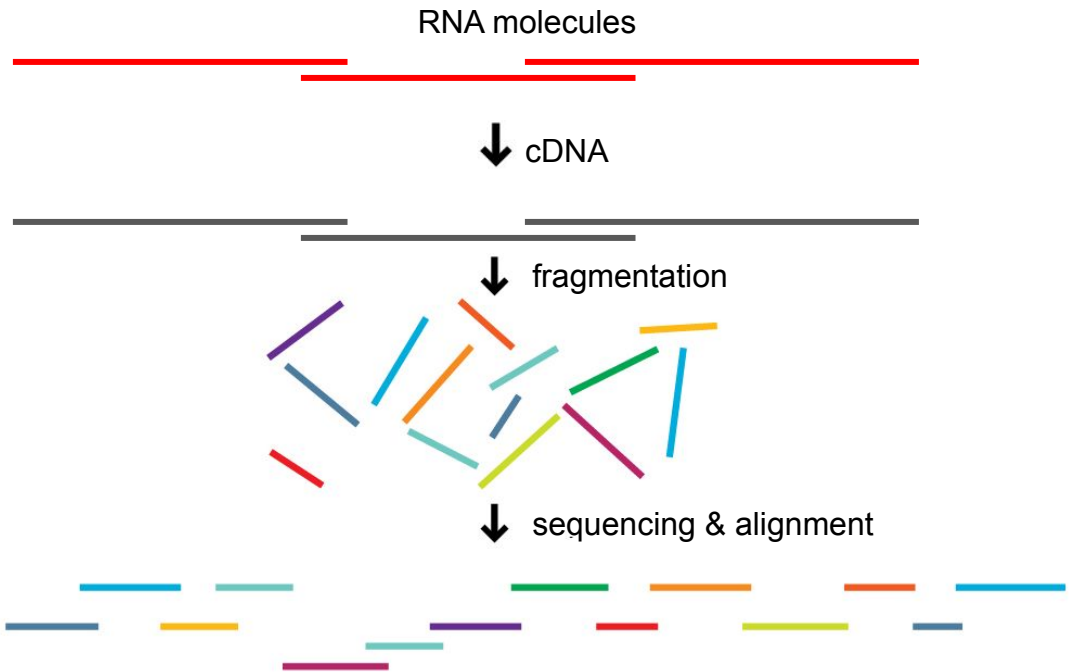
Next Generation Sequencing (NGS)

Shotgun sequencing:



Next Generation Sequencing (NGS)

RNA sequencing:



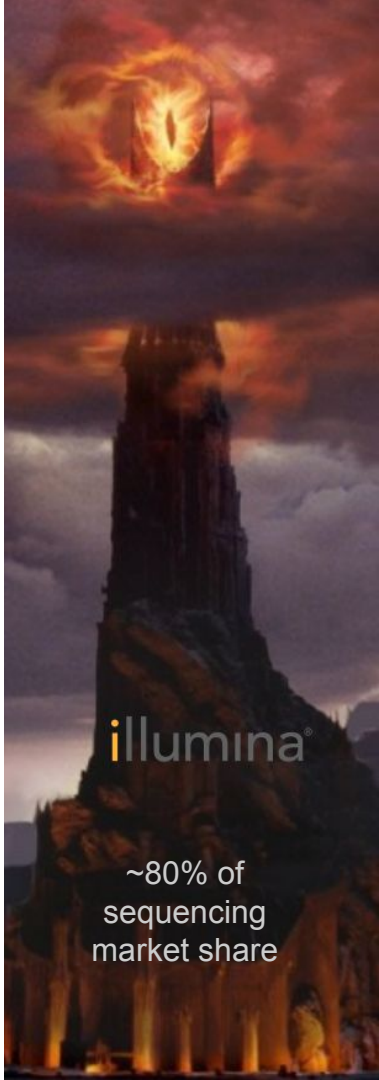
Next Generation Sequencing:
one technology to rule them all

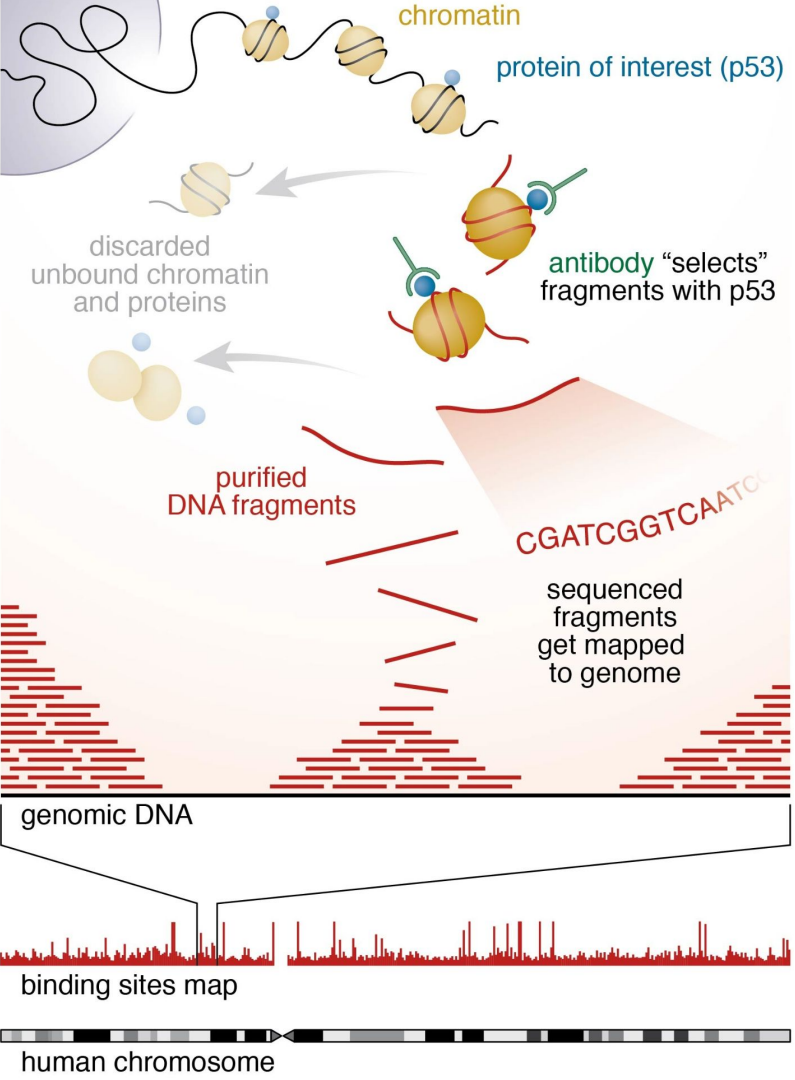
Genomics & meta-genomics

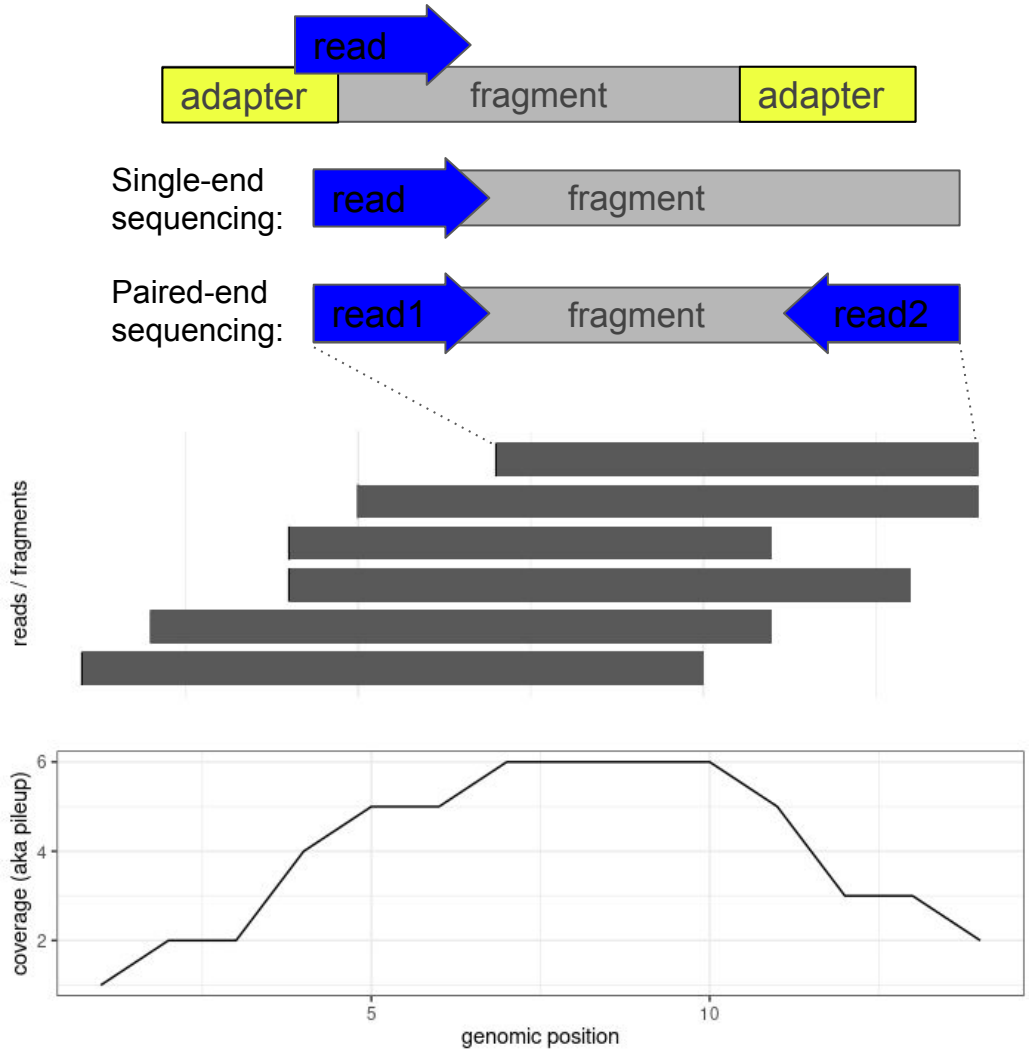
Epigenomics & chromatin conformation

Transcriptomic, translatomics, epi-transcriptomics

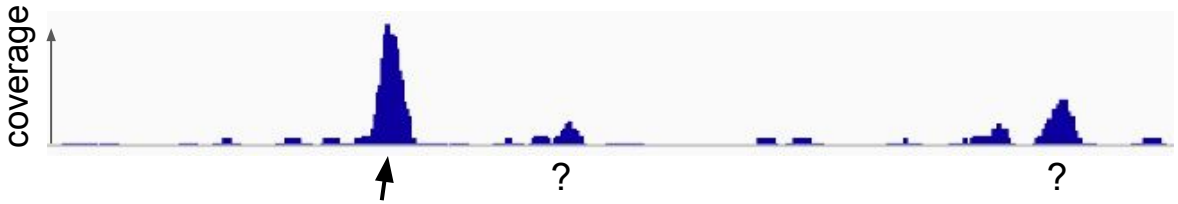
A lot of convergence in terms of analysis tools and techniques




chromatin
protein of interest (p53)
discarded
unbound chromatin
and proteins
antibody "selects"
fragments with p53
purified
DNA fragments
CGATCGGTCAATC
sequenced
fragments
get mapped
to genome
genomic DNA
binding sites map
human chromosome

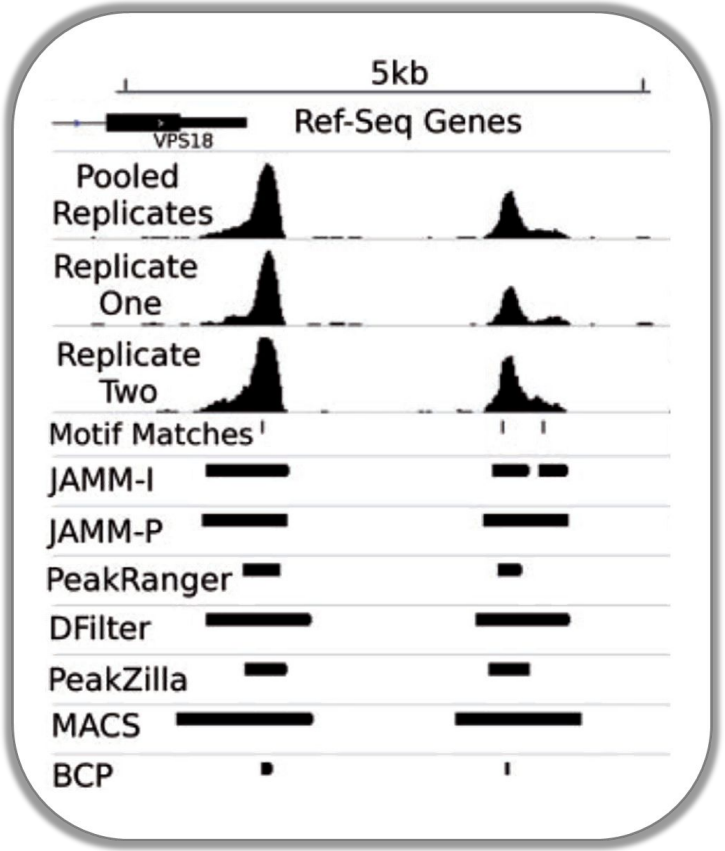

read
adapter
fragment
adapter
Single-end
sequencing:
read
fragment
Paired-end
sequencing:
read1
fragment
read2
reads / fragments
coverage (aka pileup)
6
4
2
5
10
genomic position

Peak calling

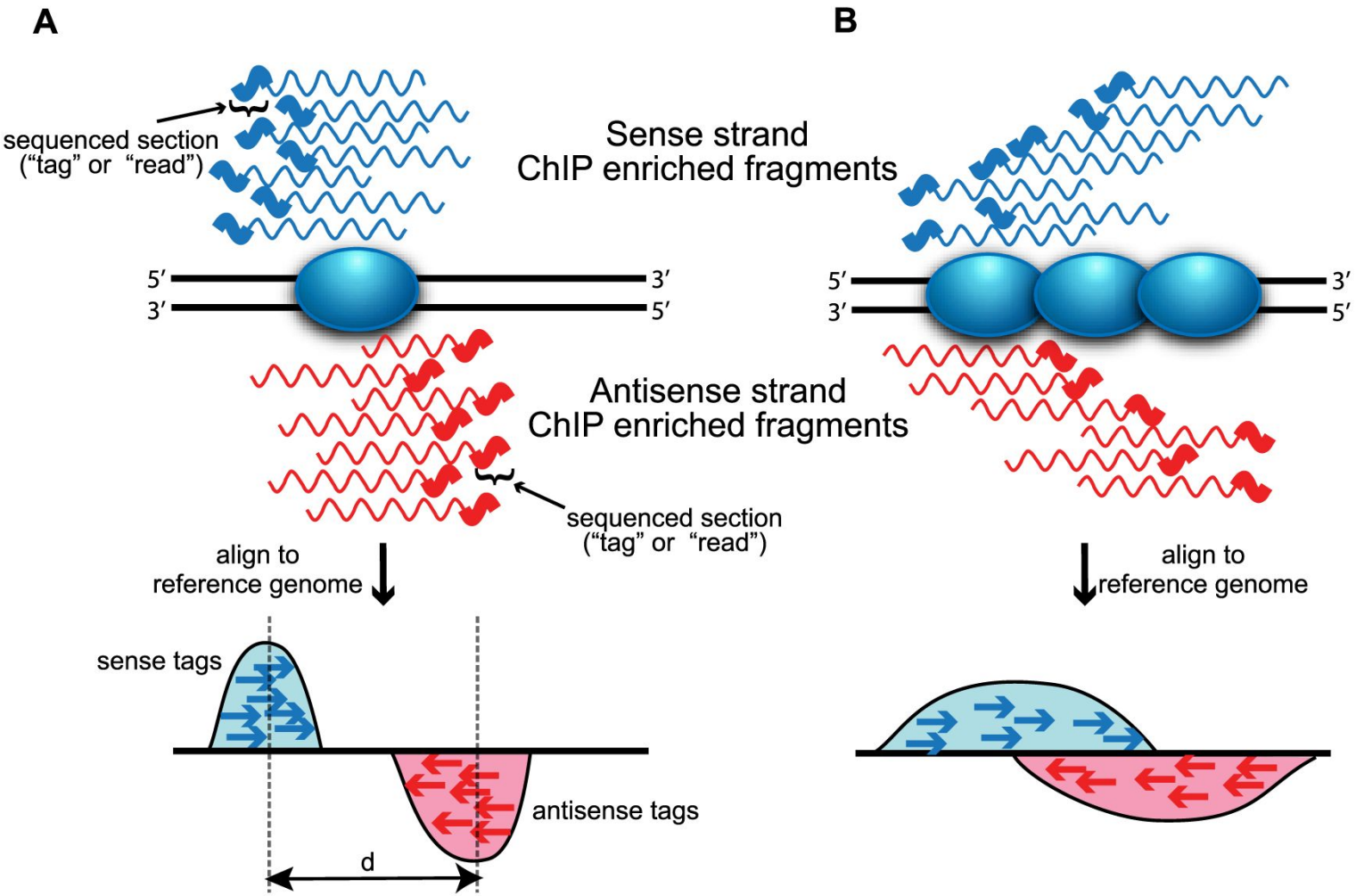


TF binding site

Enrichment over input?

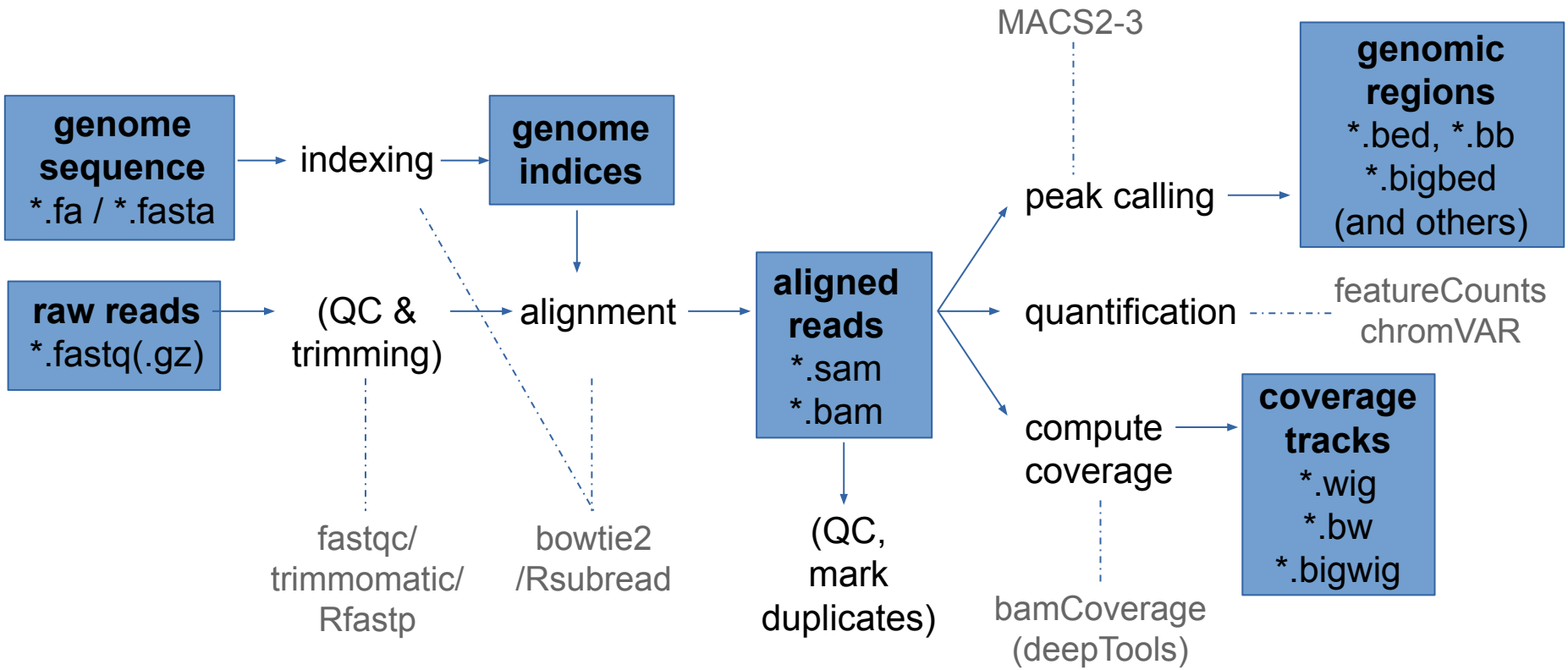


(Ibrahim et al., NAR 2014)



(Wilbanks et al., PLoS One 2010)

Overview of a primary analysis pipeline (ChIP-seq and the likes)



Alternative toolsets for (DNA) primary analysis

- The most standard one:
  - fastqc
  - trimmomatic
  - bowtie2
  - picard
  - deeptools

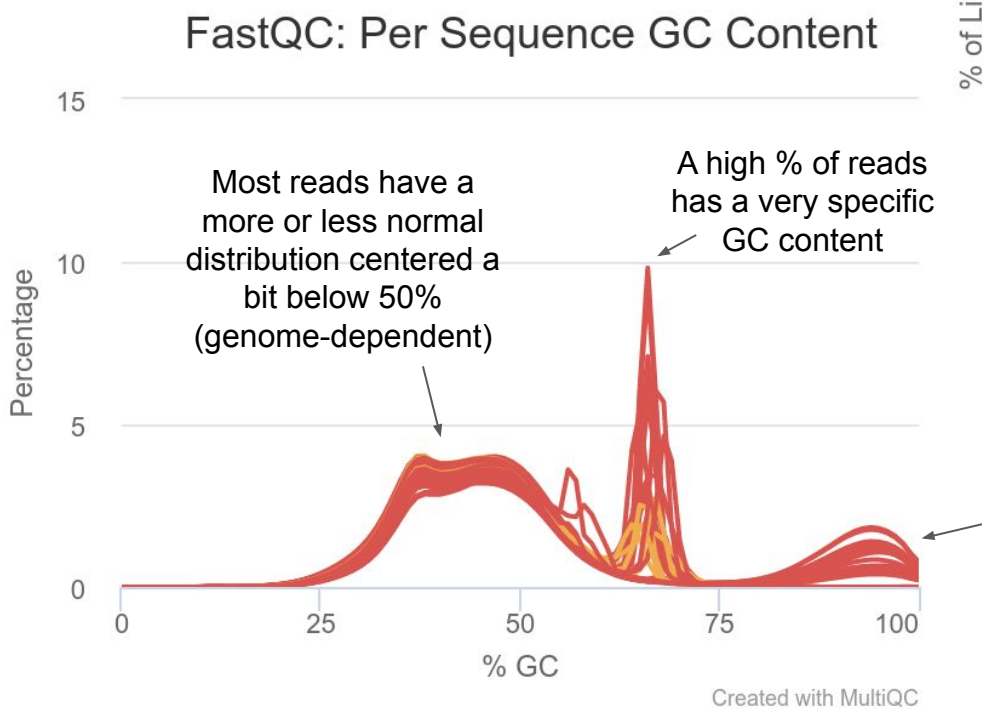
- Pure R-based
  - rfastp
  - Rsubread

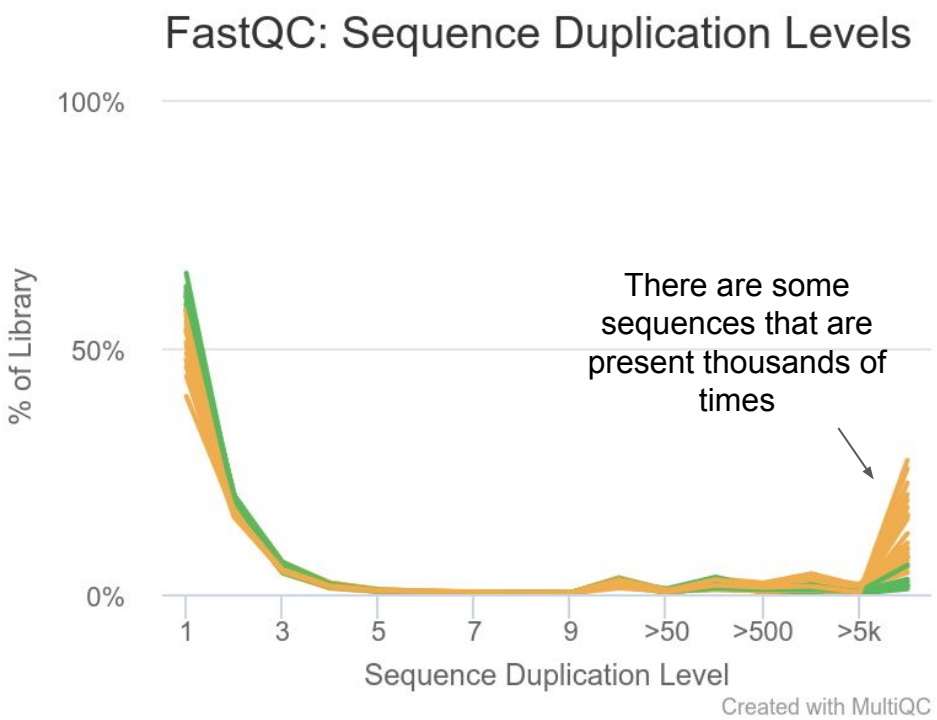
QuasR

Downstream analysis (R)

- epiwraps
- ChIPseeker
- etc…

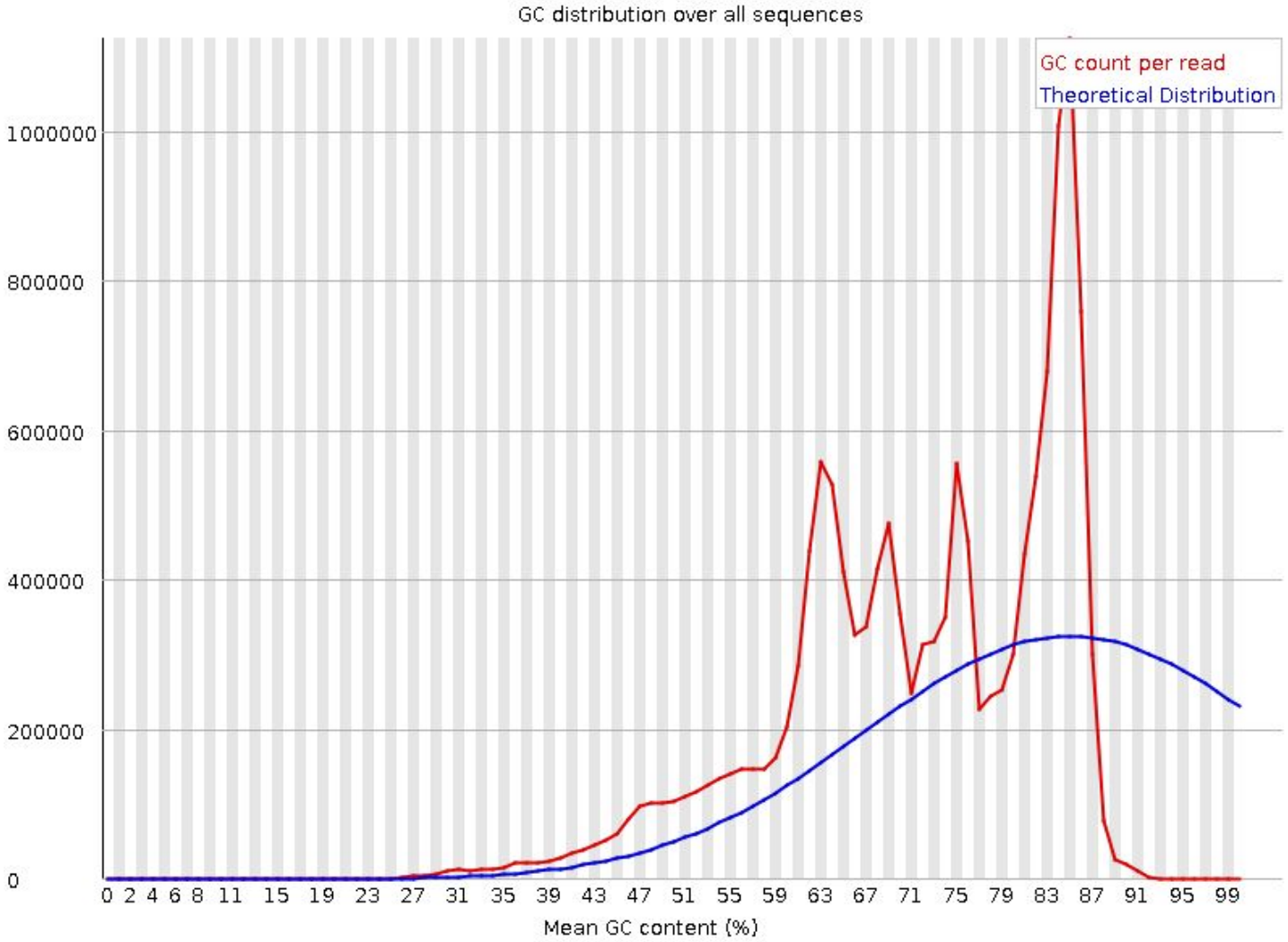
Example (rather extreme) QC problems





Example (rather extreme)
QC problems:

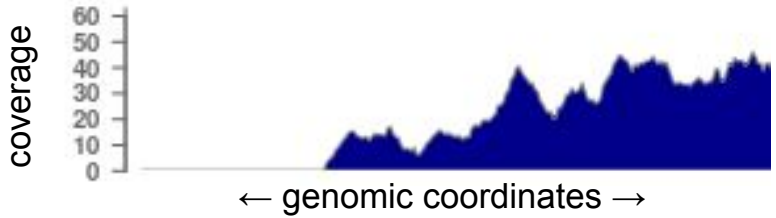
Bias from overamplification



Visualizations available in *epiwraps*

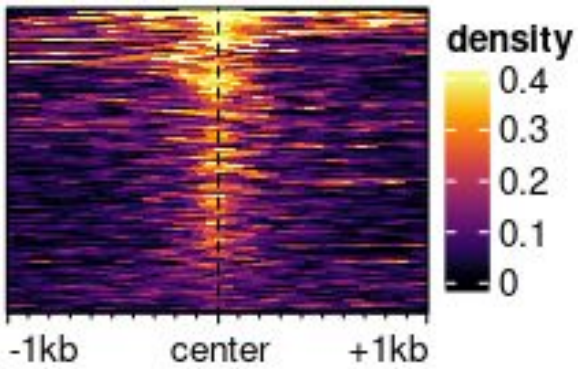
[Documentation]

- Signal across one genomic region: `plotSignalTracks`



(Based on the *Gviz* R package)

- Signal across several genomic regions: `signal2Matrix` → `plotEnrichedHeatmaps`



(Mainly based on the EnrichedHeatmap R package, itself based on ComplexHeatmap)

Assignment

- Download the following Drosophila ChIP-seq for the protein CTCF:
  - IP: https://www.encodeproject.org/files/ENCFF127RRR/@@download/ENCFF127RRR.fastq.gz

    (no input control for the purpose of this exercise)
- Process it from the raw data, obtaining:
  - bam file
  - peaks
- Report:
  - how many reads (and what percentage) were mapped
  - how many peaks were found
- Plot the signal around one of the peaks that is located *inside a gene*

- Please make sure that you name your final file **`assignment.html`** !!