Introduction to CHIP-Seq data analysis

PAG 2020 Bioinformatics workshop, January 09

Rafet Al-Tobasei

Outline

- Introduction to ChiP-seq experiment
  - Motivation
  - Experimental procedure
- Chip-seq analysis workflow
- Read QC
- Alignment
  - Bwa-mem
  - Bowtie/bowtie2
- Post Alignment QC
- Methods and software for ChiP-seq peak calling.
  - Histone modification
- Peak Annotation
- Differential Binding Analysis

Introduction to ChiP-seq

- ChiP: ChromatinImmunoPrecipitation
- Seq: sequencing

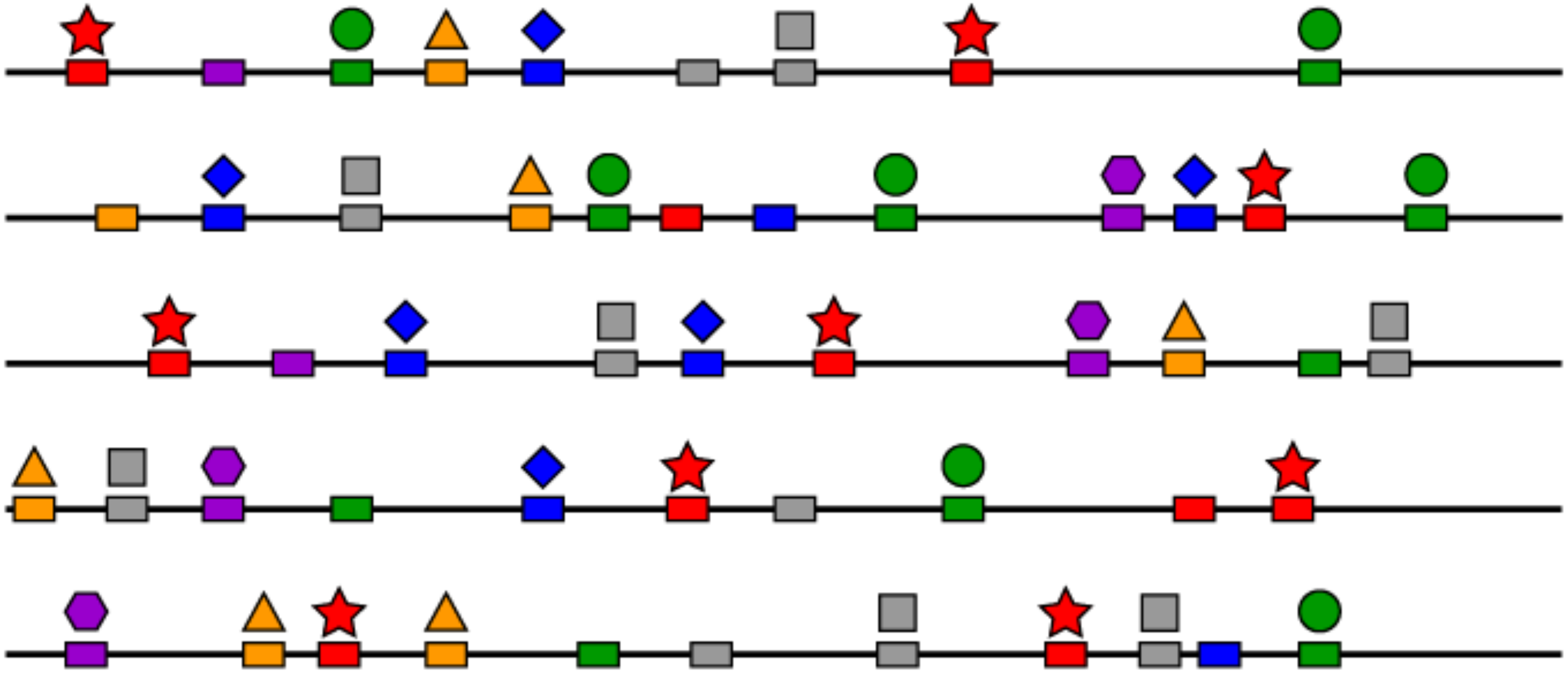
- ChiP-Seq is a method used to analyze protein interactions with DNA which help in detect essential gene-regulatory functions
    - Binding sites of DNA-binding proteins (e.g., transcription factors, DNA-polymerases 2, DNA-binding enzymes)
    - Chromatin modification (e.g., histone modifications)

Experimental Procedures

- Crosslink: bounds proteins on isolate genomics DNA
- Sonication: cut DNA in small pieces of ~200bp
- IP: use a specific antibody to capture DNA fragments with specific protein's
- Reverse crosslink: remove proteins from DNA

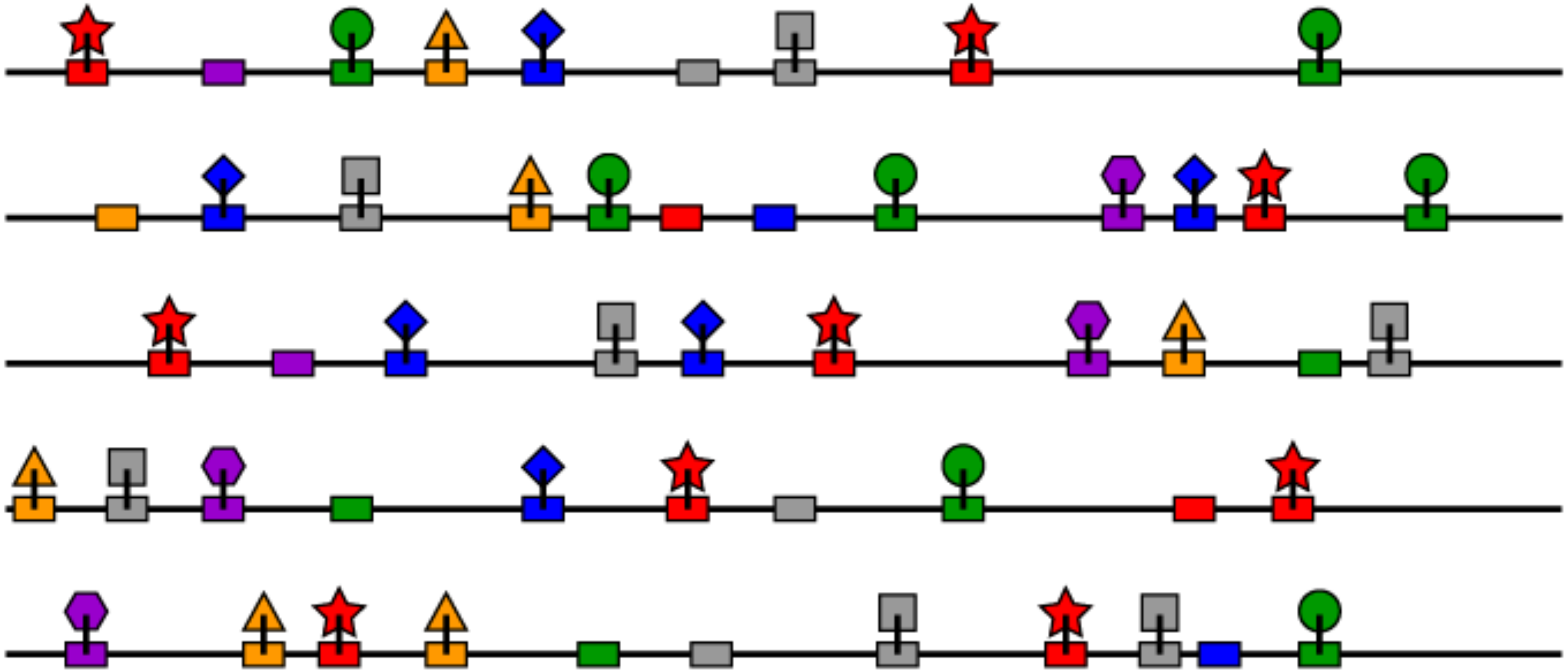
- Sequence the DNA segment

Chromatin ImmunoPrecipitation (ChIP)



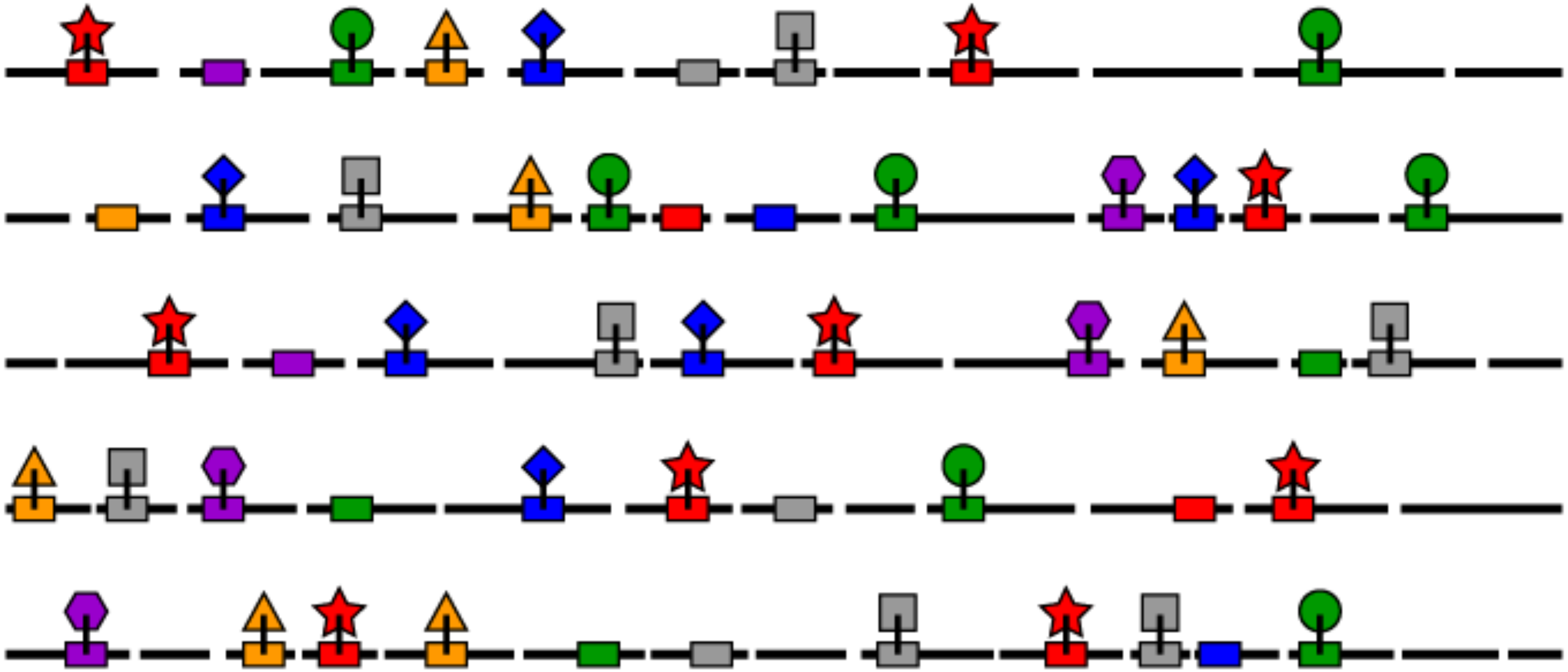
By Richard Bourgon at UC Berkley

TF/DNA Crosslinking *in vivo (formaldehyde)*



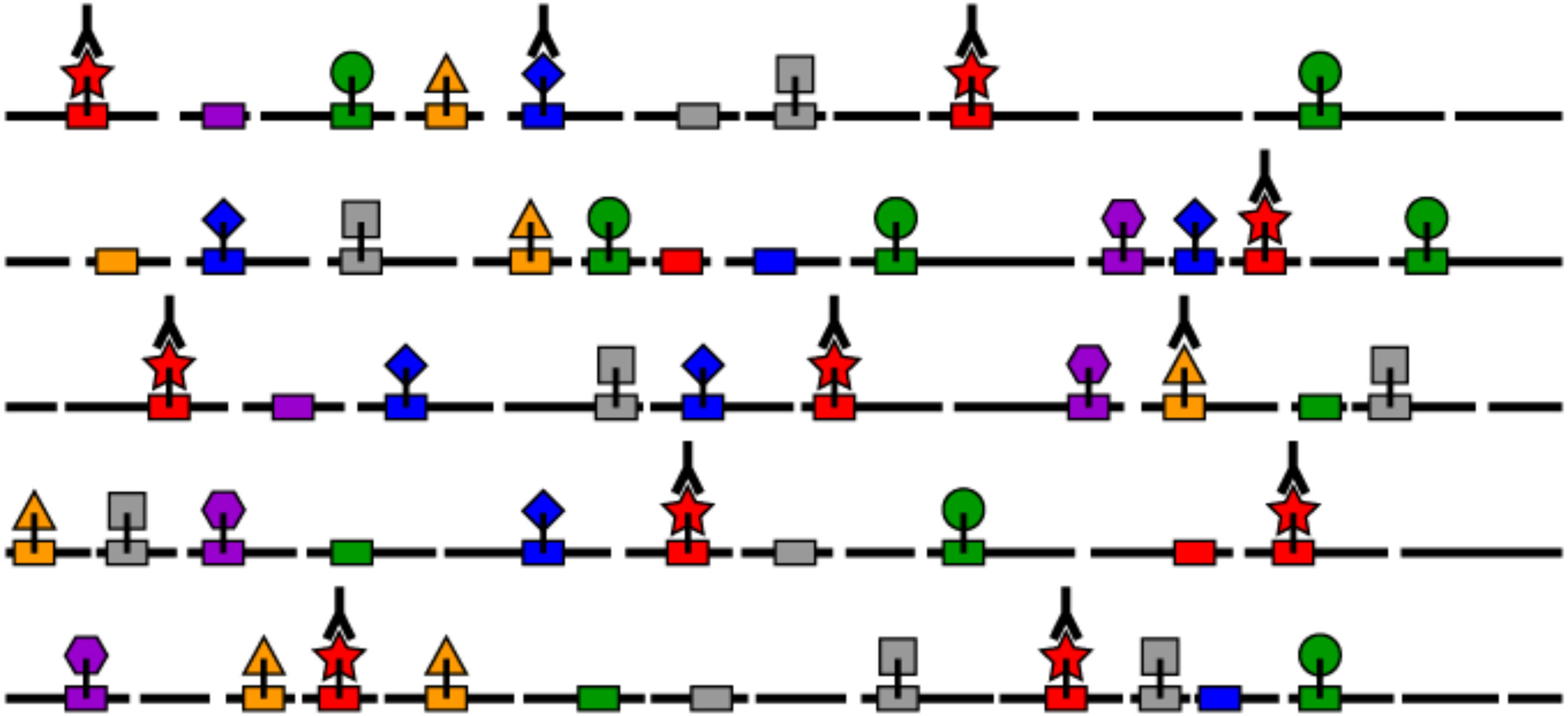
By Richard Bourgon at UC Berkley

Sonication (~200-500bp)



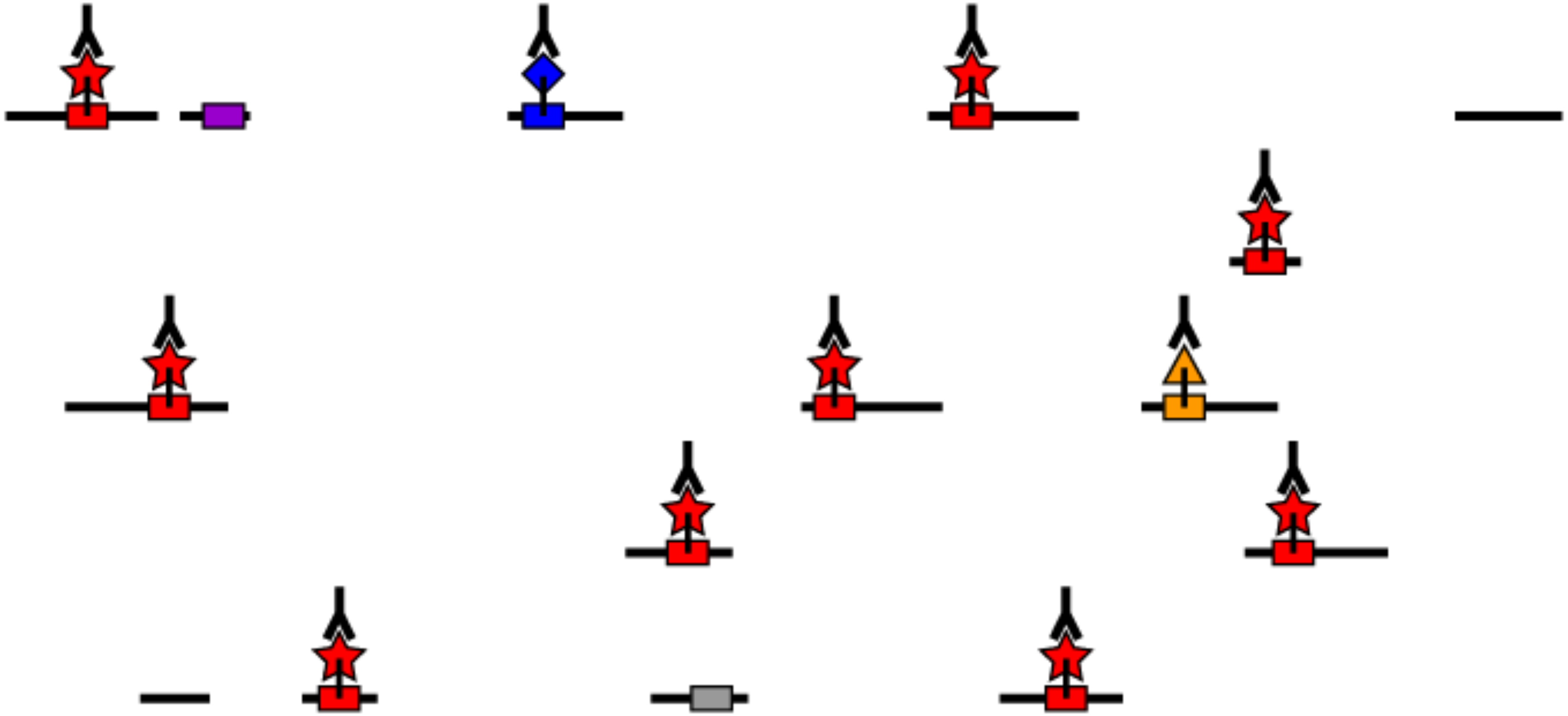
By Richard Bourgon at UC Berkley

Protein specific Antibody



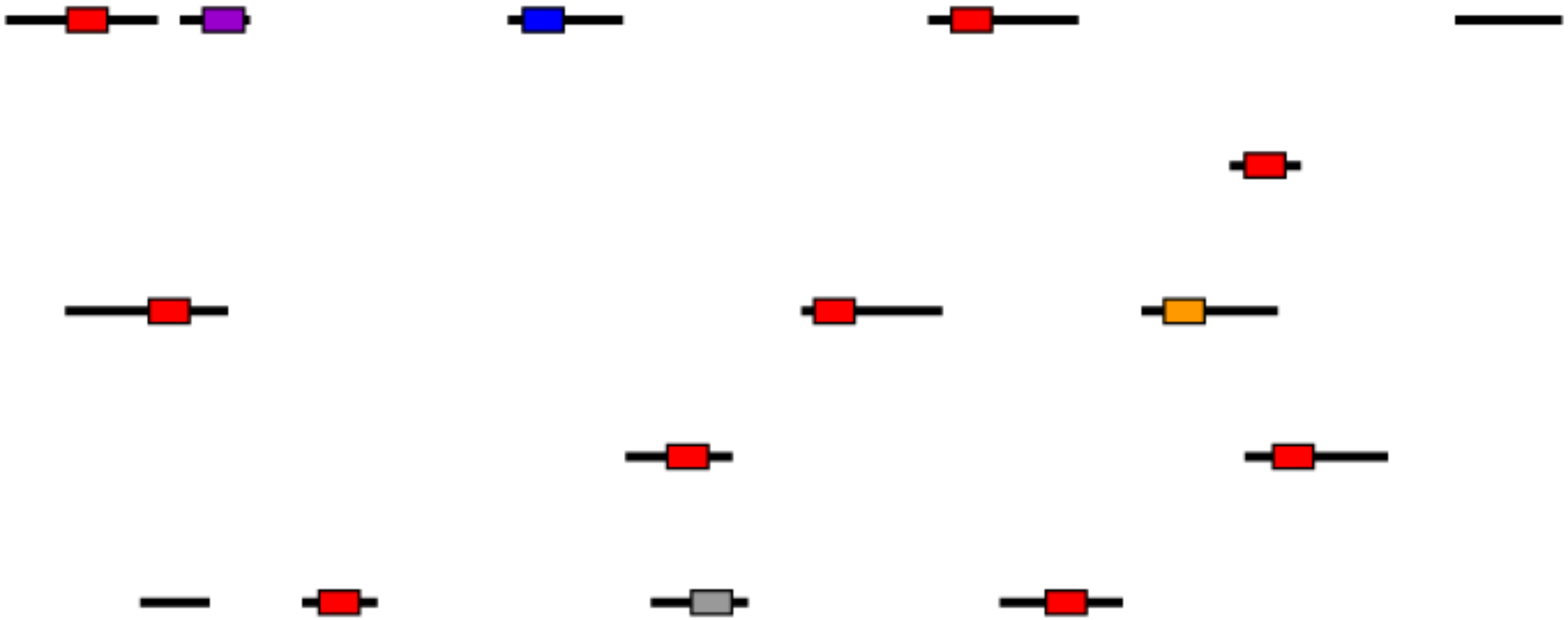
By Richard Bourgon at UC Berkley

Immunoprecipitation



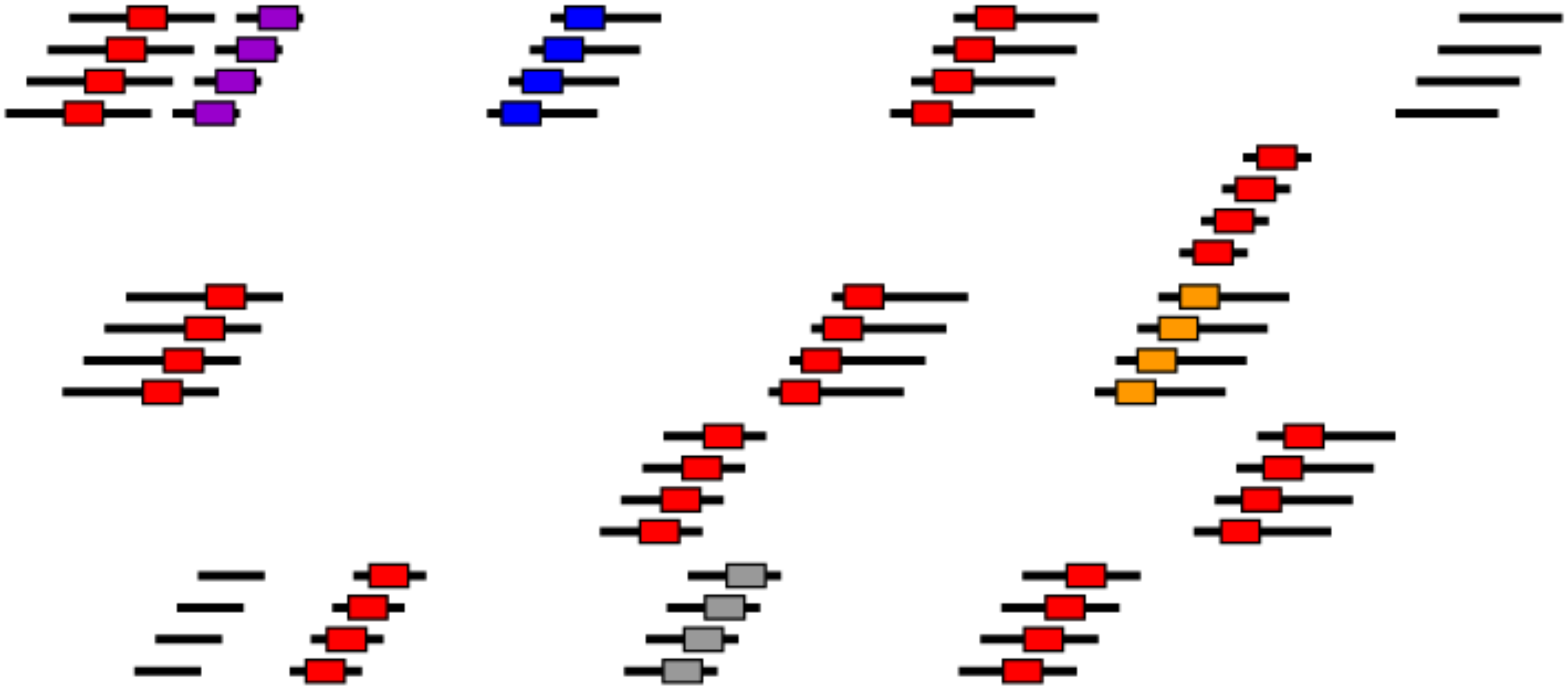
By Richard Bourgon at UC Berkley

Reverse Crosslink and DNA Purification



By Richard Bourgon at UC Berkley

Amplification



By Richard Bourgon at UC Berkley

ChIP-seq Histone Markers

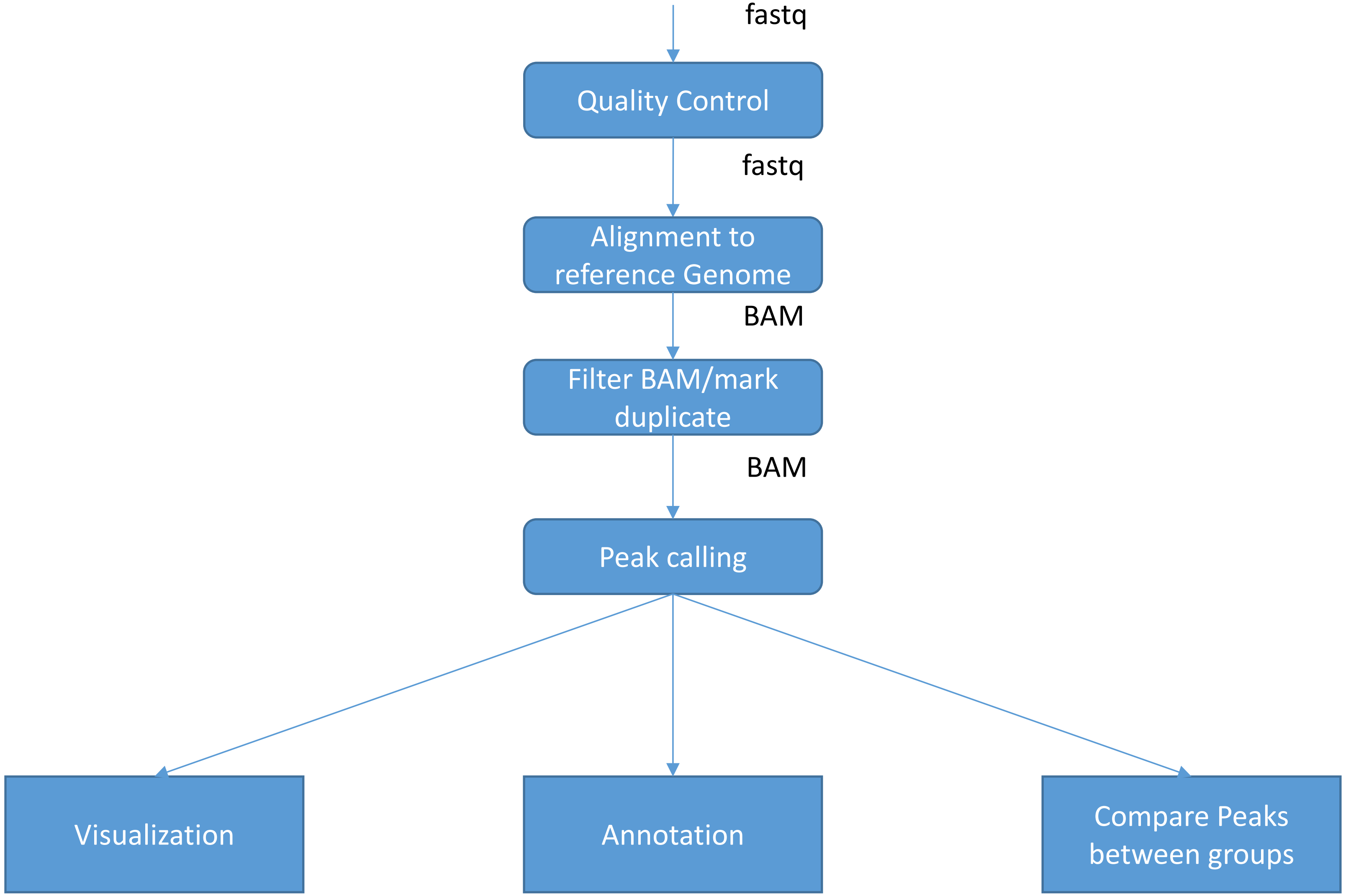
- H3K27ac active enhancers and promoters (acetylation)
- H3K4me3 promoter active genes and transcription start sites (methylation)
- H3K27me3 silenced genes (active during developmental stages)
- H3K4me1 active enhancers

Broad Marks	Narrow Marks
H3K27me3	H3ac
H3K36me3	H3K27ac
H3K4me1	H3K4me2
H3K79me2	H3K4me3
H3K79me3	H3K9ac
H3K9me1	
H3K9me2	
H4K20me1	

Control Sample

- To reduce false positive control sample used for correcting:
  - Repetitive regions.
  - DNA sequence contents affect amplification or sequencing process
  - Chromatin structure affect the DNA sonication process

Chip-seq pipeline Overview



fastq

Quality Control

fastq

Alignment to reference Genome

BAM

Filter BAM/mark duplicate

BAM

Peak calling

Visualization

Annotation

Compare Peaks between groups

Data management

- Setup directory
- chipseaq
    - Data
        - Fastq
        - map
    - genome (fasta, gff, and index)
    - Peak
    - Results
    - Optional
        - Annotation
        - diff

Read QC

- High Quality Chip-seq data needed to get good results
- Sequence Depth
    - Depend on the size of the genome
    - Mammalian:
        - >10 (20) million reads for TF
        - >20 (40) million reads for Chip-seq (narrow-peaks, broad-peaks)
- Both Single end reads and paired-end reads works
- Read length:
    - 50-150 nt
- Replicates:
    - Two replicate is enough (experiment done in two separate date)
    - Identify confident peaks

Read QC

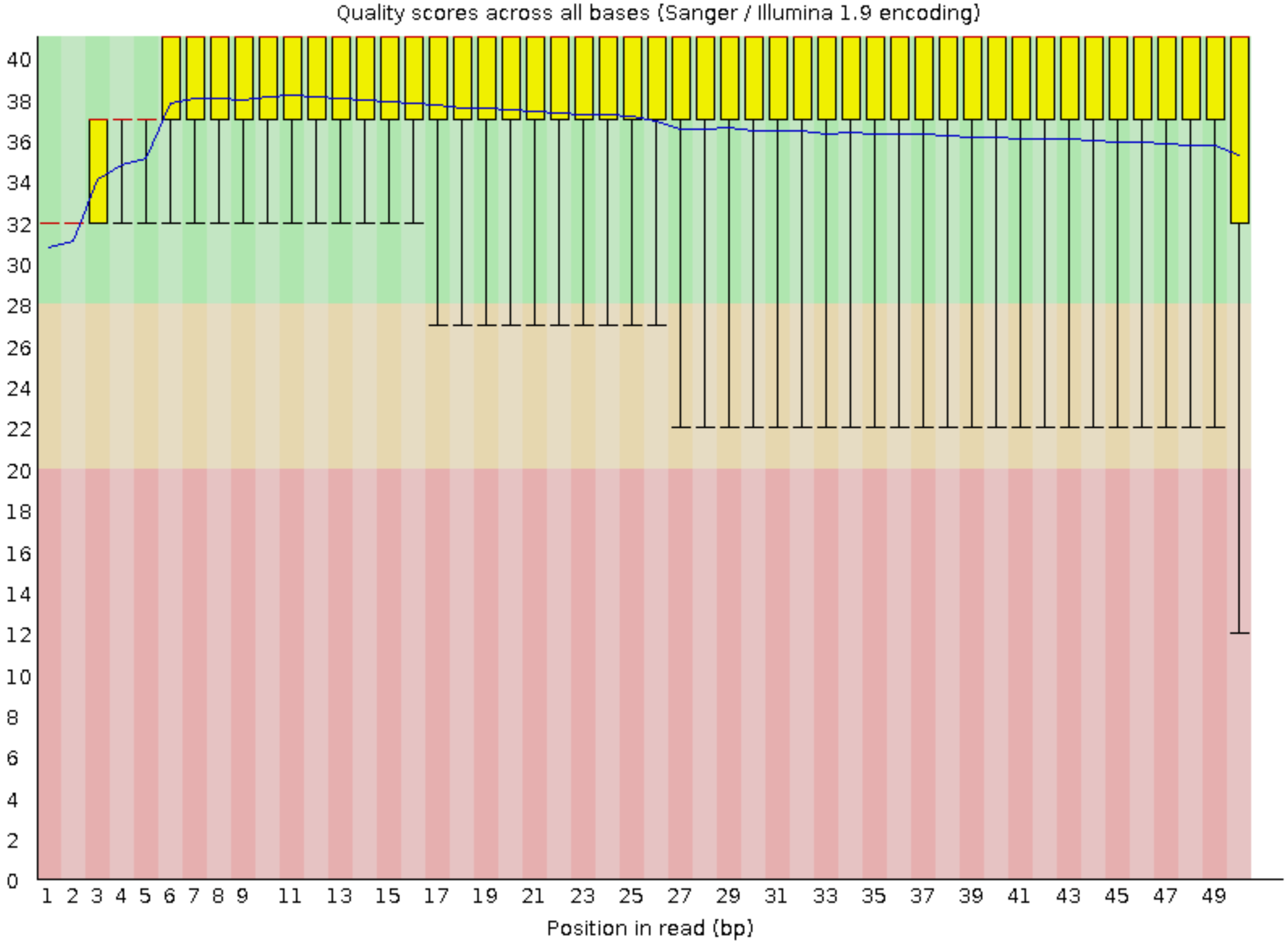
- Similar to other Sequencing data
  - Quality of the reads (sequencer problem)
  - Duplication rates (PCR amplification, short reads, not enough starting material)
  - Over represented sequence “contamination”

- FASTQC (http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help/3%20Analysis%20Modules/)
  - “FastQC is an application which takes a fastq file and runs a series of tests on it to generate a comprehensive QC report.”

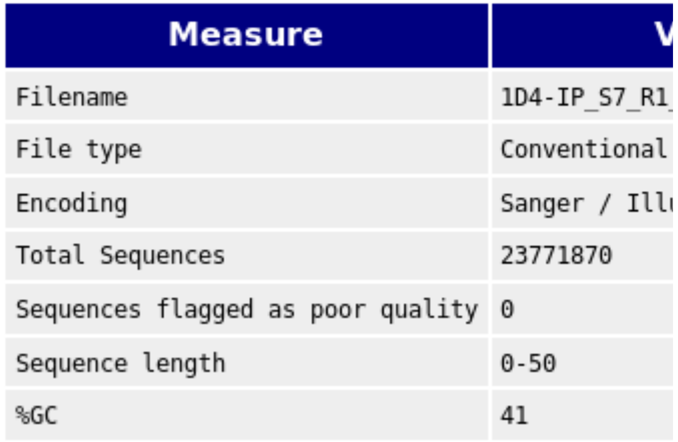
  - fastqc read_file.fastq
  - fastqc *.fastq
  - fastqc -t 8 *.fastq

- Output HTML report

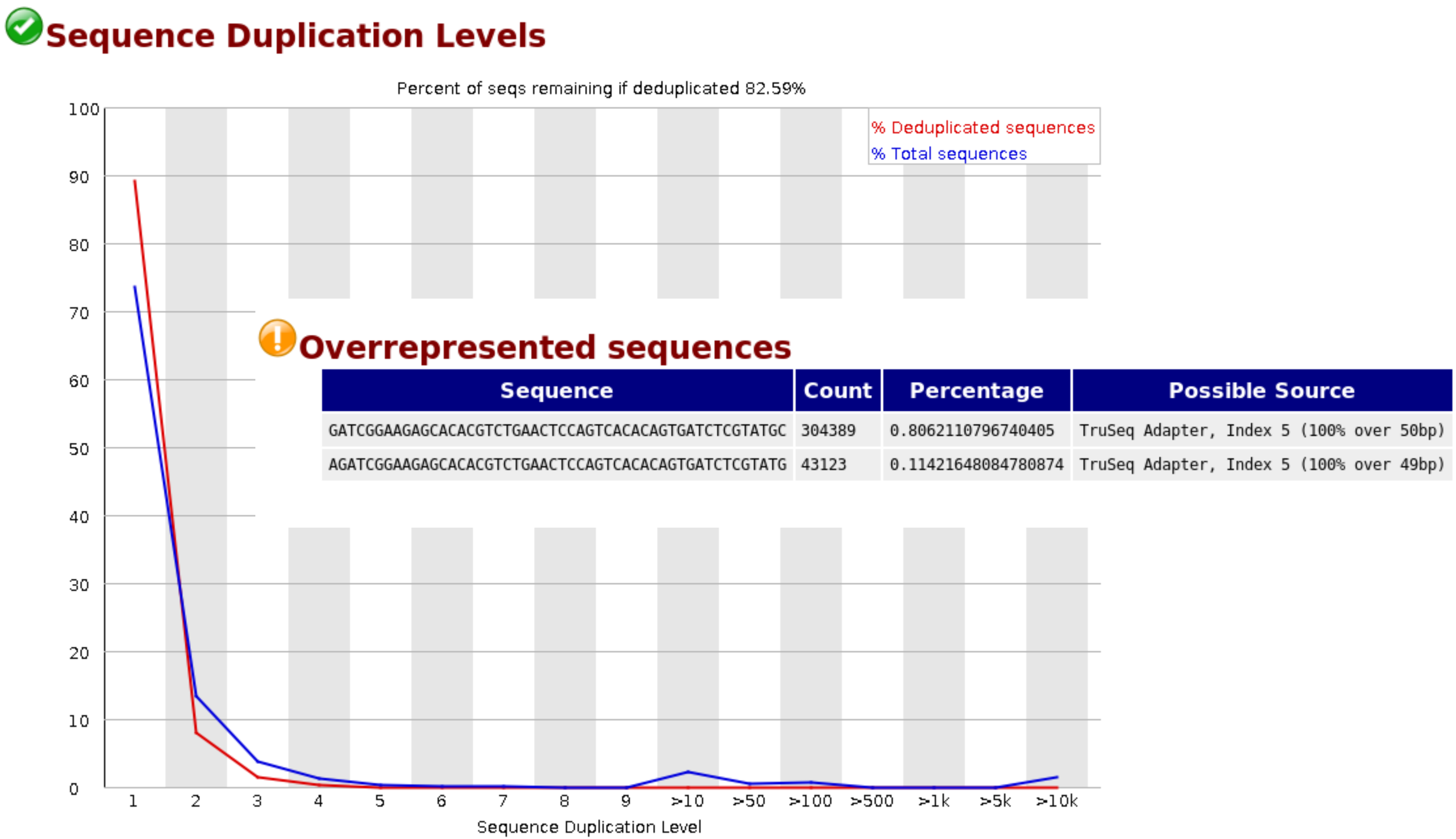
✓ Per base sequence quality



✓ Basic Statistics



Measure	V
Filename	1D4-IP_S7_R1_
File type	Conventional
Encoding	Sanger / Illu
Total Sequences	23771870
Sequences flagged as poor quality	0
Sequence length	0-50
%GC	41



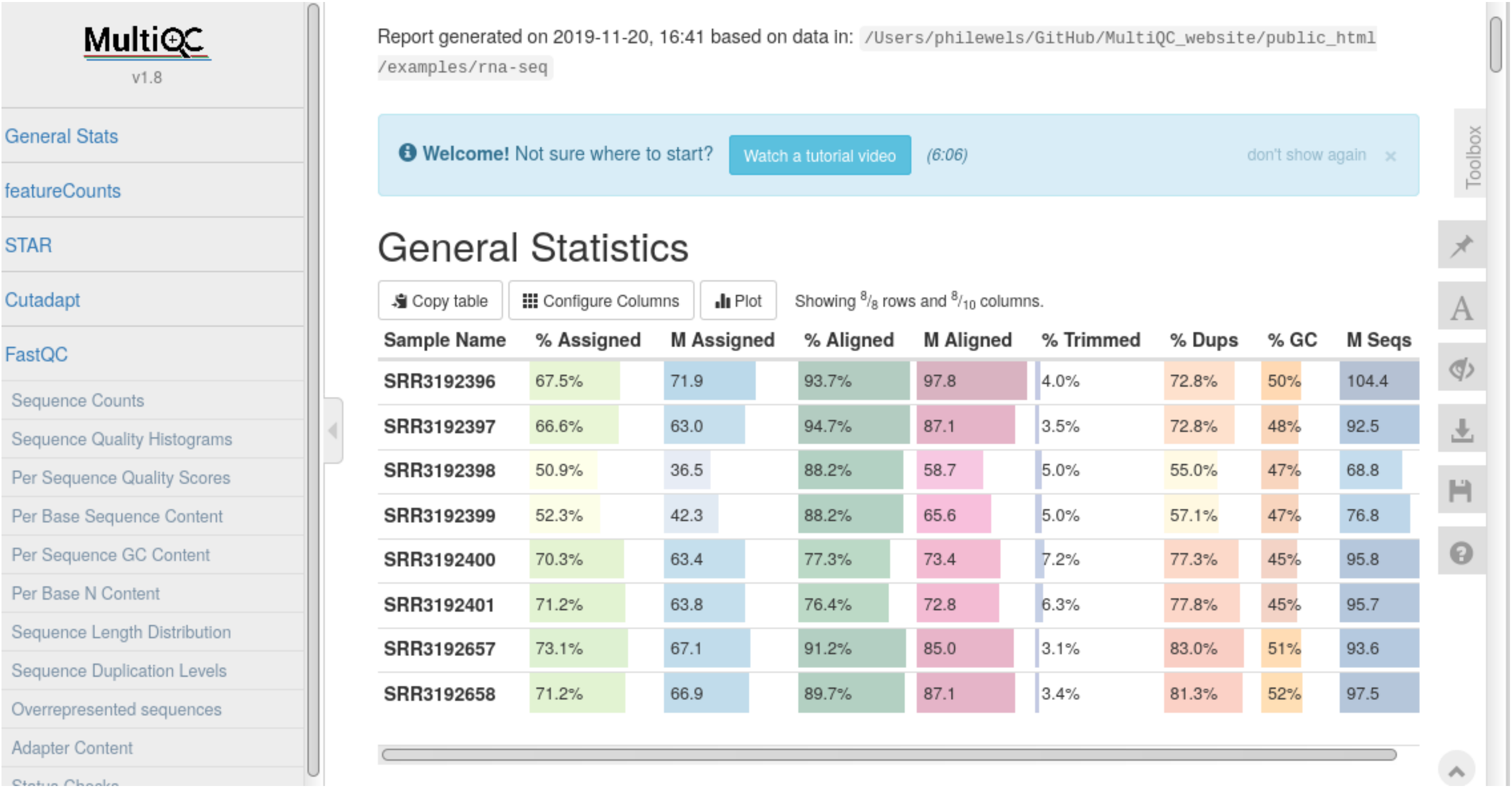
Sequence Duplication Levels

Overrepresented sequences

Sequence	Count	Percentage	Possible Source
GATCGGAAGAGCACACGTCTGAACTCCAGTCACACAGTGATCTCGTATGC	304389	0.8062110796740405	TruSeq Adapter, Index 5 (100% over 50bp)
AGATCGGAAGAGCACACGTCTGAACTCCAGTCACACAGTGATCTCGTATG	43123	0.11421648084780874	TruSeq Adapter, Index 5 (100% over 49bp)

Read QC

- MultiQC “A modular tool to aggregate results from bioinformatics analyses across many samples into a single report.”
  - multiqc .
  - To rerun in same dir.
  - multiqc --force .



Report generated on 2019-11-20, 16:41 based on data in: `/Users/philewels/GitHub/MultiQC_website/public_html/examples/rna-seq`

Welcome! Not sure where to start?

General Statistics

Showing $^8/_8$ rows and $^8/_{10}$ columns.

Sample Name	% Assigned	M Assigned	% Aligned	M Aligned	% Trimmed	% Dups	% GC	M Seqs
SRR3192396	67.5%	71.9	93.7%	97.8	4.0%	72.8%	50%	104.4
SRR3192397	66.6%	63.0	94.7%	87.1	3.5%	72.8%	48%	92.5
SRR3192398	50.9%	36.5	88.2%	58.7	5.0%	55.0%	47%	68.8
SRR3192399	52.3%	42.3	88.2%	65.6	5.0%	57.1%	47%	76.8
SRR3192400	70.3%	63.4	77.3%	73.4	7.2%	77.3%	45%	95.8
SRR3192401	71.2%	63.8	76.4%	72.8	6.3%	77.8%	45%	95.7
SRR3192657	73.1%	67.1	91.2%	85.0	3.1%	83.0%	51%	93.6
SRR3192658	71.2%	66.9	89.7%	87.1	3.4%	81.3%	52%	97.5

Read QC

- Cutadapter
    - cutadapt –a AACCGGTT –o output.fastq input.fastq

Alignment

- Determine the location of each read in the reference genome. (chromosome/position)

- Bwa mem/aln long/short reads

- Bowtie2 long reads >= 50
- Bowtie1 : short reads < 50

Example Bowtie2 Mapping

- Create Genome Index
  - Performed once (slow)

```
bowtie2-build reference_genome.fa genome_index
bwa index reference_genome.fa
```

- output
  - genome_index.4.bt2
  - genome_index.3.bt2
  - genome_index.2.bt2
  - genome_index.1.bt2
  - genome_index.rev.2.bt2
  - genome_index.rev.1.bt2

Example Bowtie2 Mapping

- Map a single FastQ file

```
bowtie2 -x genome_index –-local -U input.fastq –o output1.sam
samtools view -bS -h input1.sam -o output1.bam
```

- Bowtie2 basic options for aligning reads to the genome:
  - -x: /path/to/genome_indices_directory
  - -q: reads are in FASTQ format
  - -local: local alignment feature to perform soft-clipping
  - - end-to-end alignment default
  - -U: /path/to/FASTQ_file
  - -p: number of processors / cores
  - -o output_file

Example Bowtie2 Mapping

- Map a single FastQ file

```
bowtie2 -x genome_index --local -U input.fastq | samtools view –h -bS -o
output1.bam
```

- To increase the speed use --threads p option for both samtools and bowtie2
- Multiple files *for file in *.fastq ; do command ; done*

Post alignment QC

- Percentage of uniquely mapped reads
  - >70% normal
  - <50% may be of concern
- Encode guideline <20% duplication rate for paired-end (less for single end and short reads)
- Statistics using Samtools
  - *Samtools sort –o Aligned_Sorted.bam Aligned.bam*
  - *Samtools flagstat Aligned_sorted.bam*
  - *multiqc .*

Post Alignment Processing MAPQ Filtering

- Only good quality reads should be kept.
- MAPQ provide information about how confident that the reported position is correct
- Use MAPQ value to filter out unconfident mapping reads.

```
samtools sort reads.bam -o reads.sorted

samtools index reads.sorted.bam

samtools view -q 20 -b -o filtered_sorted.bam reads_sorted.bam
```

Post alignment QC Picard Tools

- Create index for bam file
  - samtools index filtered_sorted.bam

- Picard Tools
  - java -jar MarkDuplicates.jar INPUT=filterd_sorted.bam OUTPUT=Aligned_Sorted_PCRDupes.bam ASSUME_SORTED=true METRICS_FILE=Aligned_Sorted_PCRDupes.txt VALIDATION_STRINGENCY=SILENT ;

Peak calling

- Computational method used to identify enriched (peaks)
- Count based: regions with statistically significant number of reads
- Shape based: Model the spatial distributions of reads in a regions
- Model-based Analysis of ChIP-seq (MACS) most widely used peak caller
    - Narrow-peak (TF)
    - Broad-peak (Histone modifications)
    - There are several functions available in MACS2
    - macs2 --h
    - **callpeak**
    - For full description of command line options, Type
        - **macs2 callpeak**

- Build the model
- MACS shifts every tag by d/2
- Use Poisson distribution to calculate the parameter λ, the expected number of reads in that window.
- Calculate Poisson distribution p-value based on λ < 10e-5 (can be changed)
- Estimate false discovery rate (FDR)

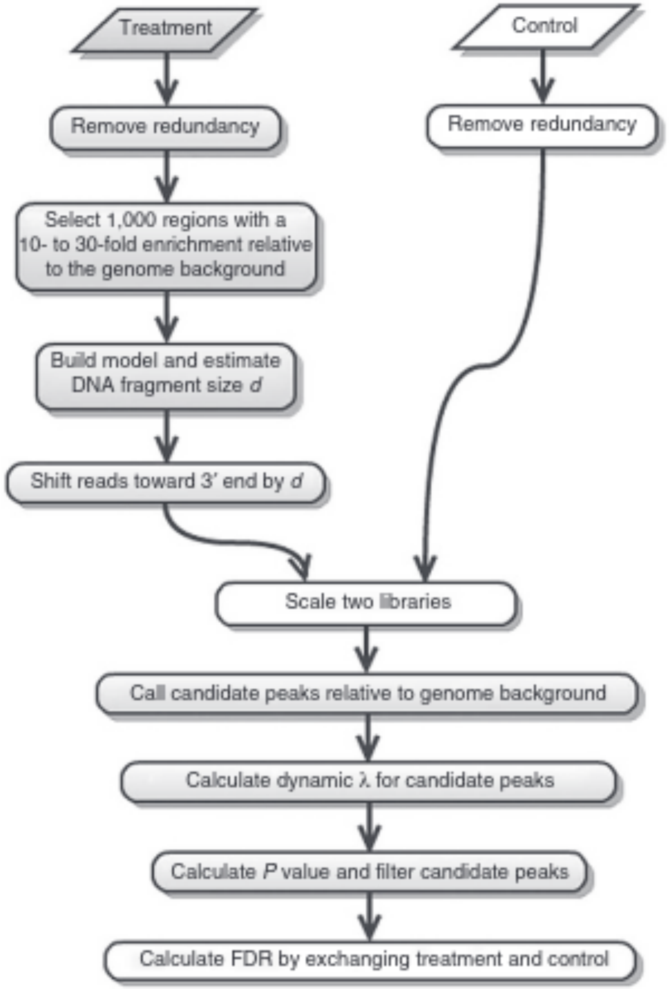

Treatment
Control
Remove redundancy
Remove redundancy
Select 1,000 regions with a 10- to 30-fold enrichment relative to the genome background
Build model and estimate DNA fragment size *d*
Shift reads toward 3′ end by *d*
Scale two libraries
Call candidate peaks relative to genome background
Calculate dynamic λ for candidate peaks
Calculate *P* value and filter candidate peaks
Calculate FDR by exchanging treatment and control

Figure 1 | Workflow of MACS 1.4.2. If the control sample is missing, then the steps shown in white boxes will be skipped (remove redundancy of the control sample, scale two libraries and calculate FDR by exchanging treatment and control).

Feng, J., Liu, T., Qin, B. et al. Identifying ChIP-seq enrichment using MACS. Nat Protoc 7, 1728–1740 (2012) doi:10.1038/nprot.2012.101

Macs2 callpeak options

- Input file options
  - -t: The IP data file (this is the only REQUIRED parameter for MACS)
  - -c: The control data file(optional)
  - -f: format of input file; Default is “AUTO” which will allow MACS to decide the format automatically.
  - -g: mappable genome size which is defined as the genome size which can be sequenced.
  - --broad: broad peak calling
- Output arguments
  - --outdir: MACS2 will save all output files into speficied folder for this option
  - -n: The prefix string for output files

Macs2 command line

```
macs2 callpeak -t Aligned_Sorted_PCRDupes.bam \
	-c control_file.bam \
	-f BAM -g 1.9e+9 \
	-n D4_K4_narrow_peak \
	--outdir peak
```

MACS2 Output files

- Narrow peaks
    - Name_peaks.xls
    - Name_summit.bed
    - Name_model.r
    - Name_peaks.narrowPeak
- For broad peaks
    - Name_peaks.xls
    - Name_peaks.broadPeak
    - Name_model.r
- For number of peaks
    - Wc –l Name_peaks.narrowPeak

.xls file

- chr
- start
- end
- length
- abs_summit
- pileup
- -log10(pvalue)
- fold_enrichment
- -log10(qvalue)
- name

Peak analysis

- Assess the quality of ChIP-seq data.
    - Good quality ChIP enrichment over background.

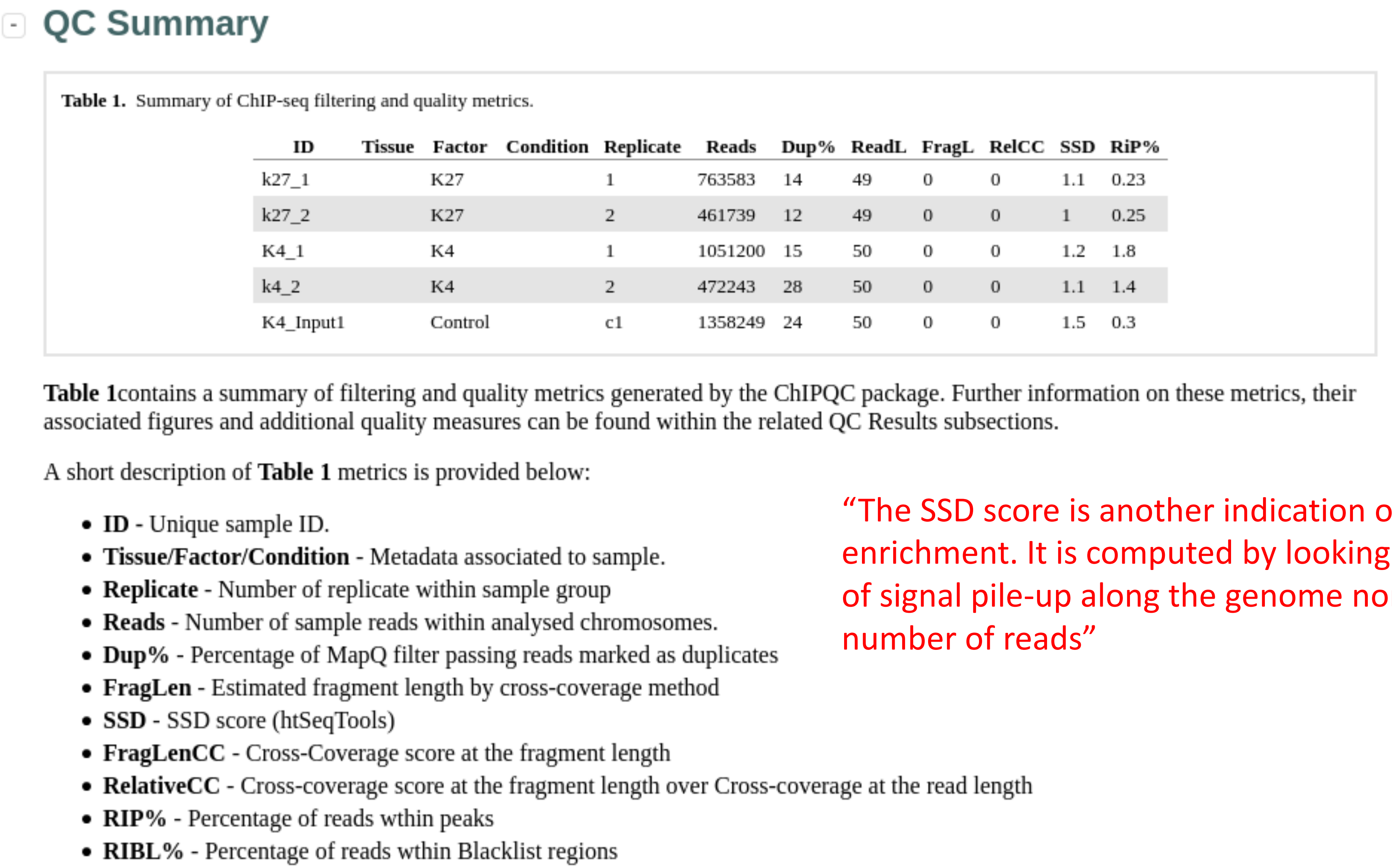
- ChipQC
- Deeptools
- SPP

Post alignment QC using ChipQC

- Bioconductor R Package
- Generate a quality report and figures
- Compute a number of quality metrics
- Need Sample sheet that contains metadata information about the dataset
  - Sample ID, Input bam file, peaks bed file, control bam file replicate and peak Caller
- Same sample file will be used in DiffBind
  - SampleID,Tissue,Replicate,bamReads,ControlID,bamControl,Peaks,PeakCaller,Tissue, Condition
  - K4_1, K4, 1, data/map/ID8-K4_S2Ch1sortedfilteredmDup.bam ,K4_Input1, data/map/1D8-IP_S8Ch1sortedfilteredmDup.bam, peak/ID8-K4_S2Ch1_peaks.narrowPeak, macs, NA, NA
- http://chipqc.starkhome.com/Reports/tamoxifen/ChIPQC.html

ChipQC R codes

```
library(ChIPQC)
## Load sample data
samples <- read.csv('sample.csv')

## Create ChIPQC object
chipObj <- ChIPQC(samples)  # you could add the genome of your species , TxDb=txdb  , consensus=TRUE
ChIPQCreport(chipObj, reportName="ChIP QC report: K4 and K27", reportFolder="ChIPQCreport")

library(EnsDb.Hsapiens.v75)
library(clusterProfiler)
library(AnnotationDbi)
library(org.Hs.eg.db)

trout_txdb <-makeTxDbFromGFF("/localstorage/index_trout_genom/GCF_002163495.1_Omyk_1.0_genomic.gff")
saveDb(trout_txdb, file="Trout.sqlite")
trout_txdb <- loadDb("Trout.sqlite")
txdb <- trout_txdb
```



QC Summary

Table 1. Summary of ChIP-seq filtering and quality metrics.

ID	Tissue	Factor	Condition	Replicate	Reads	Dup%	ReadL	FragL	RelCC	SSD	RiP%
k27_1		K27		1	763583	14	49	0	0	1.1	0.23
k27_2		K27		2	461739	12	49	0	0	1	0.25
K4_1		K4		1	1051200	15	50	0	0	1.2	1.8
k4_2		K4		2	472243	28	50	0	0	1.1	1.4
K4_Input1		Control		c1	1358249	24	50	0	0	1.5	0.3

Table 1contains a summary of filtering and quality metrics generated by the ChIPQC package. Further information on these metrics, their associated figures and additional quality measures can be found within the related QC Results subsections.

A short description of **Table 1** metrics is provided below:

- **ID** - Unique sample ID.
- **Tissue/Factor/Condition** - Metadata associated to sample.
- **Replicate** - Number of replicate within sample group
- **Reads** - Number of sample reads within analysed chromosomes.
- **Dup%** - Percentage of MapQ filter passing reads marked as duplicates
- **FragLen** - Estimated fragment length by cross-coverage method
- **SSD** - SSD score (htSeqTools)
- **FragLenCC** - Cross-Coverage score at the fragment length
- **RelativeCC** - Cross-coverage score at the fragment length over Cross-coverage at the read length
- **RIP%** - Percentage of reads wthin peaks
- **RIBL%** - Percentage of reads wthin Blacklist regions

“The SSD score is another indication of evidence of enrichment. It is computed by looking at the standard deviation of signal pile-up along the genome normalized to the total number of reads”

QC Results

Mapping, Filtering and Duplication rate

This section presents the mapping quality, duplication rate and distribution of reads in known genomic features.

Table 2. Number and percantage of mapped,duplicated and MapQ filter passing reads

ID	Tissue	Factor	Condition	Replicate	Unmapped	Mapped	Pass MapQ Filter and Dup	Total Dup%	Pass MapQ Filter%	Pass MapQ Filter and Dup%
k27_1		K27_		1	0	526462	66262	13	100	13
k27_2		K27_		2	0	429947	52265	12	100	12
K4_1		K4_		1	0	948072	326051	34	100	34
k4_2		K4_		2	0	1038741	310975	30	100	30
K4_Input1	NA	NA	NA	NA	0	786459	158141	20	100	20

Table 2 shows the absolute number of total, mapped, passing MapQ filter and duplicated reads. The percent of mapped reads passing quality filter and marked as duplicates (Non-Redundant Fraction?) are also included.

Description of read filtering and flag metrics:

- **Total Dup%**-Percentage of all **mapped** reads which are marked as **duplicates.**
- **Pass MapQ Filter%**-Percentage of all **mapped** reads which**pass MapQ quality** filter
- **Pass MapQ Filter and Dup%**-Percentage of all reads which pass **MapQ filter** and are marked as**duplicates.**

Duplication rates (Dup %) are dependent on the ChIP library complexity and the number of reads sequenced Higher duplication rates maybe due to low ChIP efficiency when read counts are lower or conversely saturation of ChIP signal when sequencing large number of reads. Since this metric is dependent on both read depth and the properties of the ChIP itself, comparison between biological or technical replicates of similat total read counts can best identify problematic libraries .

Highly mappable (multimappable) positions within the genome can attract large levels of duplication and so assessment of duplication before and after MapQ quality filtering can identify contribution of these positions to the duplication rate.

Peak Profile and ChIP Enrichment

Following the identification of genome wide enrichment (peak calling), the percentage of ChIP signal within enriched regions, as well the average profile across these regions can be used to further evaluate ChIP quality

Figure 3. Plot of the average signal profile across peaks

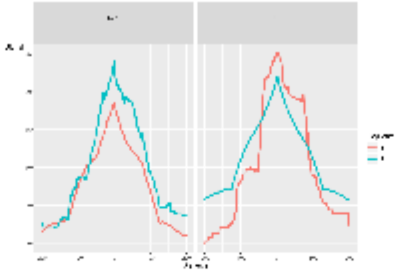


Figure5 represents the mean read depth across and around peaks. By identying the average pattern of enrichment across peaks, differences in both mean peak height and shape may be found. This not only assits in a better characterisation of ChIP enrichment but can aid in the identification of outliers.

Figure 4. Barplot of the percentage number of reads in peaks

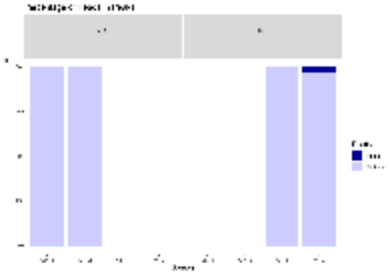


Figure6 shows the total percentage of reads contained within enriched regions or peaks. The higher efficiency ChIP-seq will show a higher percentage of reads in enriched regions/peaks and longer epigenetic marks will often have a higher ranges of efficiencies than punctate marks or transcription factors.

Figure 4. Barplot of the percentage number of reads in peaks

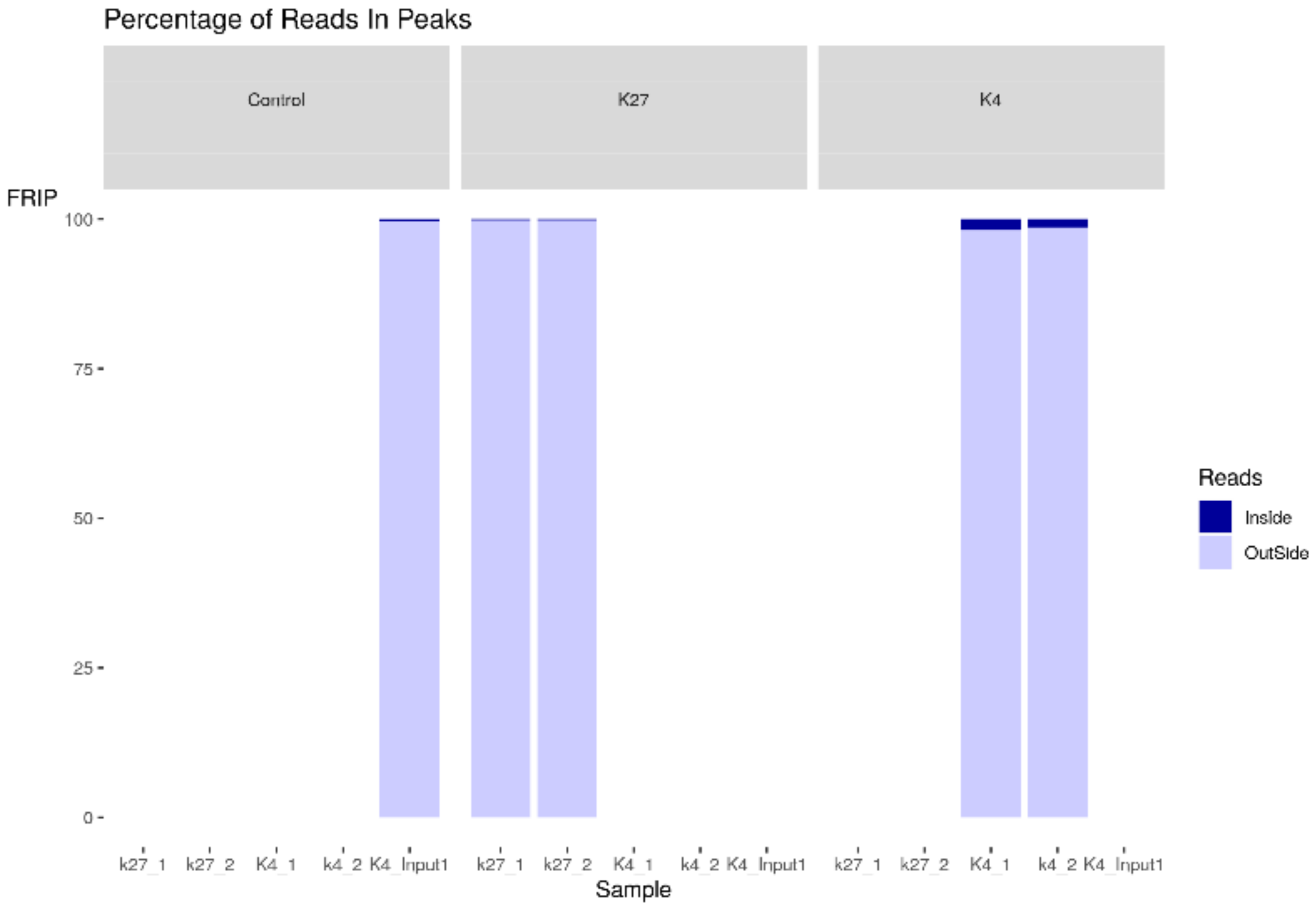


Figure6 shows the total percentage of reads contained within enriched regions or peaks. The higher efficiency ChIP-seq will show a higher percentage of reads in enriched regions/peaks and longer epigenetic marks will often have a higher ranges of efficiencies than punctate marks or transcription factors.

Handling Replicate in Chip-seq

- Keep overlapping peak calls across replicates (bedtools).

- statistical method by testing and evaluate the reproducibility between replicates (Irreproducible Discovery Rate IDR).

Handling Replicate in Chip-seq

- Keep overlapping peak calls across replicates (bedtools).

```
bedtools intersect  -a ../peak/replicate1 –b ../peak/replicate2 –wo > combin_replicate1_and_2
```

- 50% > overlap.

- You could merge the sequence reads first then find the peaks

Peak Annotation and Functional Analysis

- ChipSeeker
- GREAT

ChipSeeker Peak Annotation

- ChIPseeker “is an R package for annotating ChIP-seq data analysis. It supports annotating ChIP peaks and provides functions to visualize ChIP peaks coverage over chromosomes and profiles of peaks binding to transcription starting site (TSS) regions.”

ChipSeeker genome setup

```
# Load libraries
library(ChIPseeker)
#library(TxDb.Hsapiens.UCSC.hg19.knownGene)
library(EnsDb.Hsapiens.v75)
library(clusterProfiler)
library(AnnotationDbi)
library(org.Hs.eg.db)

# create database object using rainbow trout gff file
trout_txdb <-makeTxDbFromGFF("/path_to_genome_gff/genomic.gff") # gtf works too.
saveDb(trout_txdb, file="Trout.sqlite")
trout_txdb <- loadDb("Trout.sqlite")
txdb <- trout_txdb
```

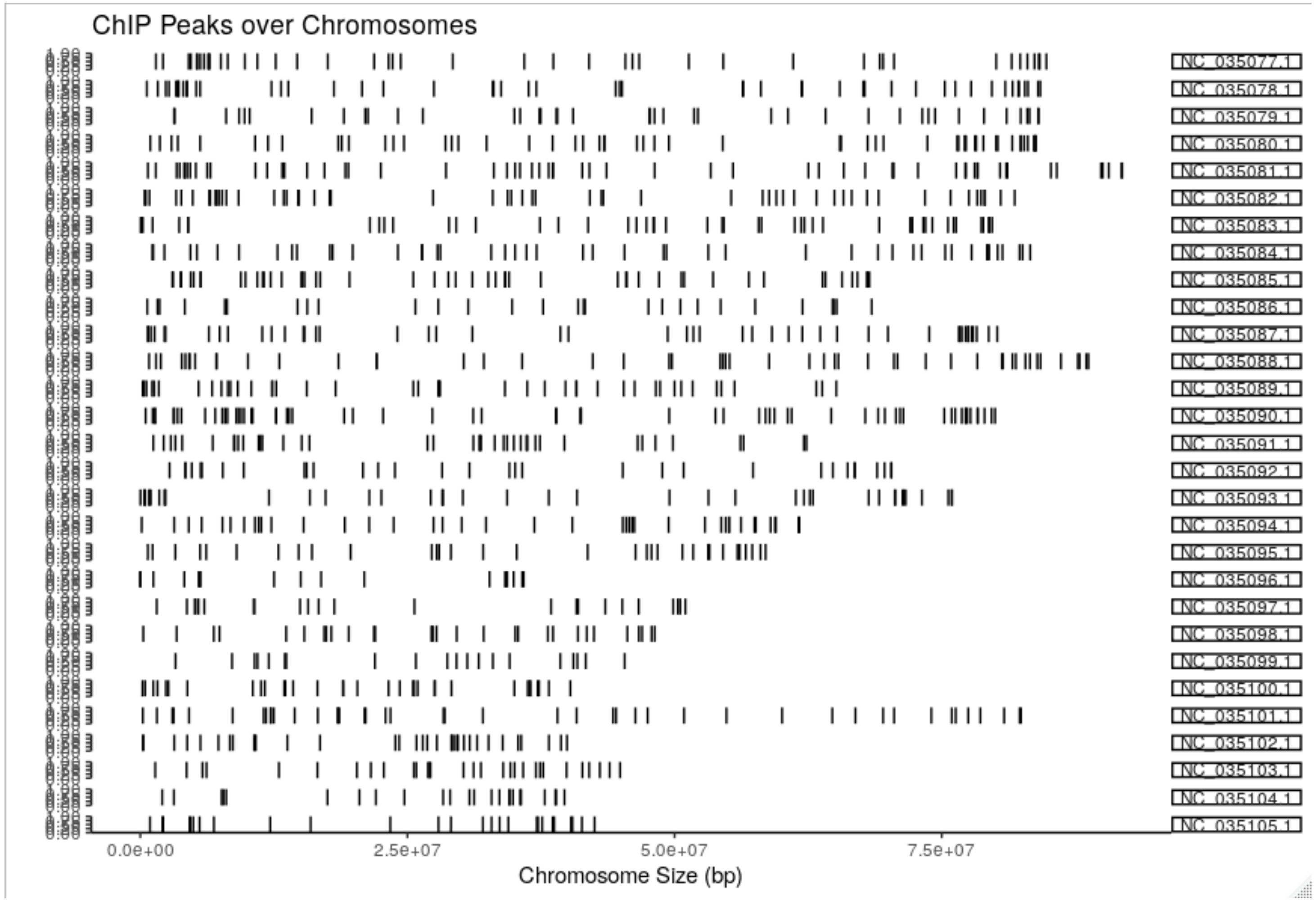
https://bioconductor.org/packages/release/bioc/vignettes/GenomicFeatures/inst/doc/GenomicFeatures.pdf

ChipSeeker

the peak locations over the whole genome,

```
samplefiles <- list.files("/localstorage/Epi_alig/peak_macs_ind/", pattern= ".broadPeak", full.names=T)
samplefiles <- as.list(samplefiles)

peak <- readPeakFile(samplefiles[[4]])
covplot(peak)
```

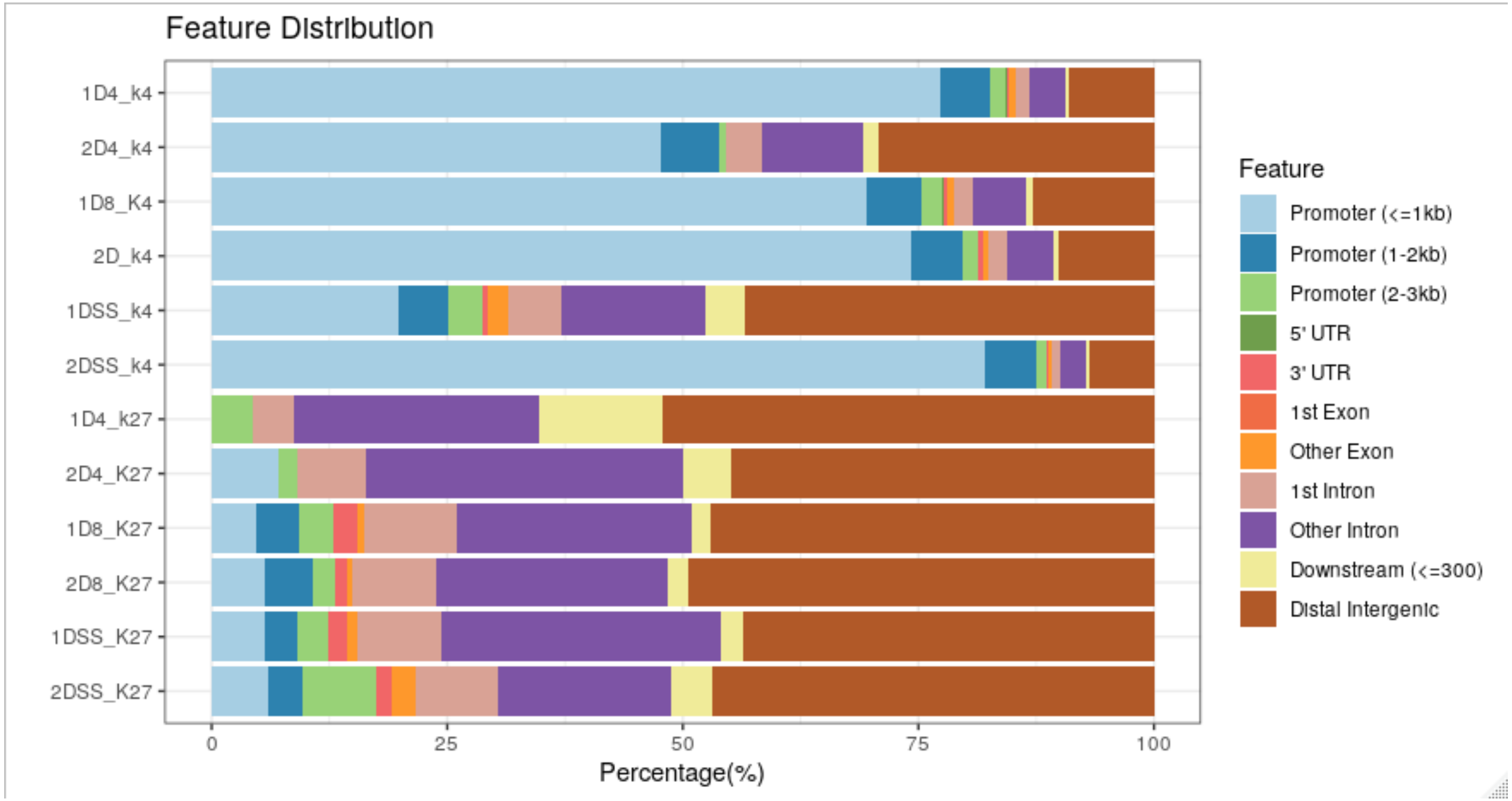
```
#Peak over specific chromosome
covplot(peak, weightCol="V5", chrs=c("chr17", "chr18"), xlim=c(4.5e7, 5e7))
```



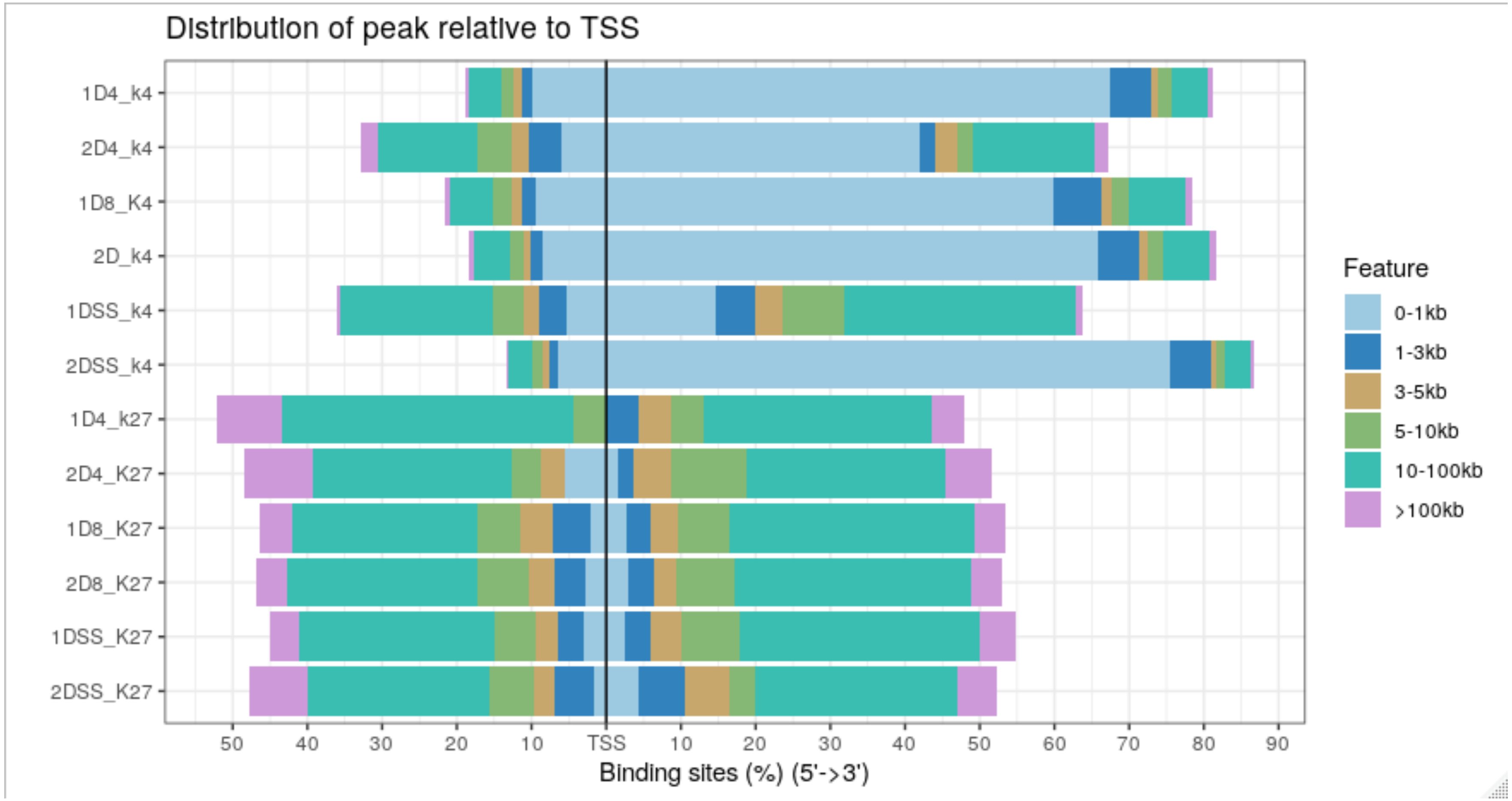
ChipSeeker

```
#Second step  is to  Load data
samplefiles <- list.files("peak", pattern= ".narrowPeak", full.names=T) # path,  file_extention
samplefiles <- as.list(samplefiles)
names(samplefiles) <- c("2D8-_K27", "2D8-_K4","1D8-_K27", "1D8-_K4") # name matching the samplefiles order
peakAnnoList <- lapply(samplefiles, annotatePeak, TxDb=txdb,  tssRegion=c(-3000, 3000), verbose=FALSE)
peakAnnoList
# plotting results
plotAnnoBar(peakAnnoList)
# Plot distance to TSS
plotDistToTSS(peakAnnoList, title="Distribution of peak relative to TSS")
```

ChipSeeker

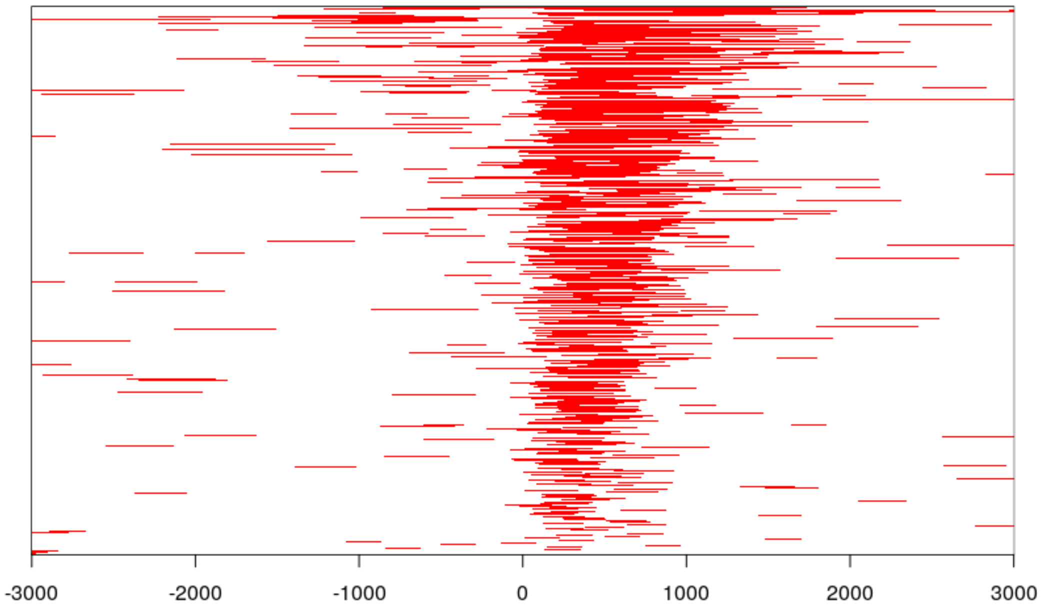

Feature Distribution
1D4_k4
2D4_k4
1D8_K4
2D_k4
1DSS_k4
2DSS_k4
1D4_k27
2D4_K27
1D8_K27
2D8_K27
1DSS_K27
2DSS_K27
0
25
50
75
100
Percentage(%)
Feature
Promoter (<=1kb)
Promoter (1-2kb)
Promoter (2-3kb)
5' UTR
3' UTR
1st Exon
Other Exon
1st Intron
Other Intron
Downstream (<=300)
Distal Intergenic

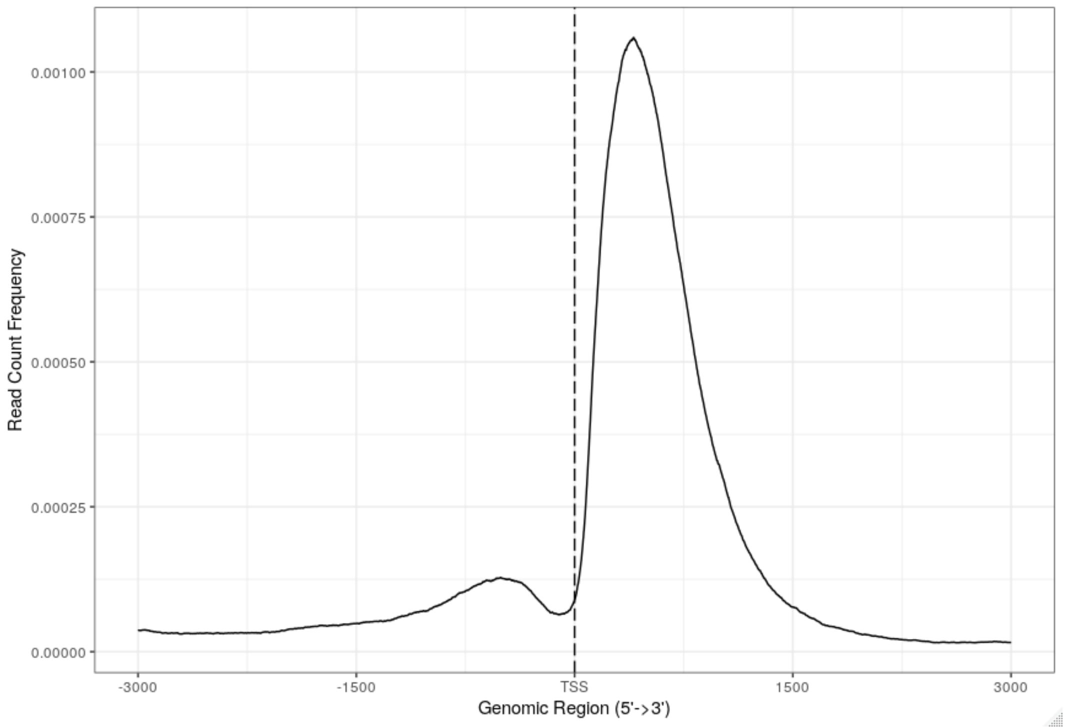
ChipSeeker


Distribution of peak relative to TSS
1D4_k4
2D4_k4
1D8_K4
2D_k4
1DSS_k4
2DSS_k4
1D4_k27
2D4_K27
1D8_K27
2D8_K27
1DSS_K27
2DSS_K27
50
40
30
20
10
TSS
10
20
30
40
50
60
70
80
90
Binding sites (%) (5'->3')
Feature
0-1kb
1-3kb
3-5kb
5-10kb
10-100kb
>100kb

ChipSeeker Heatmap of ChIP binding to TSS regions

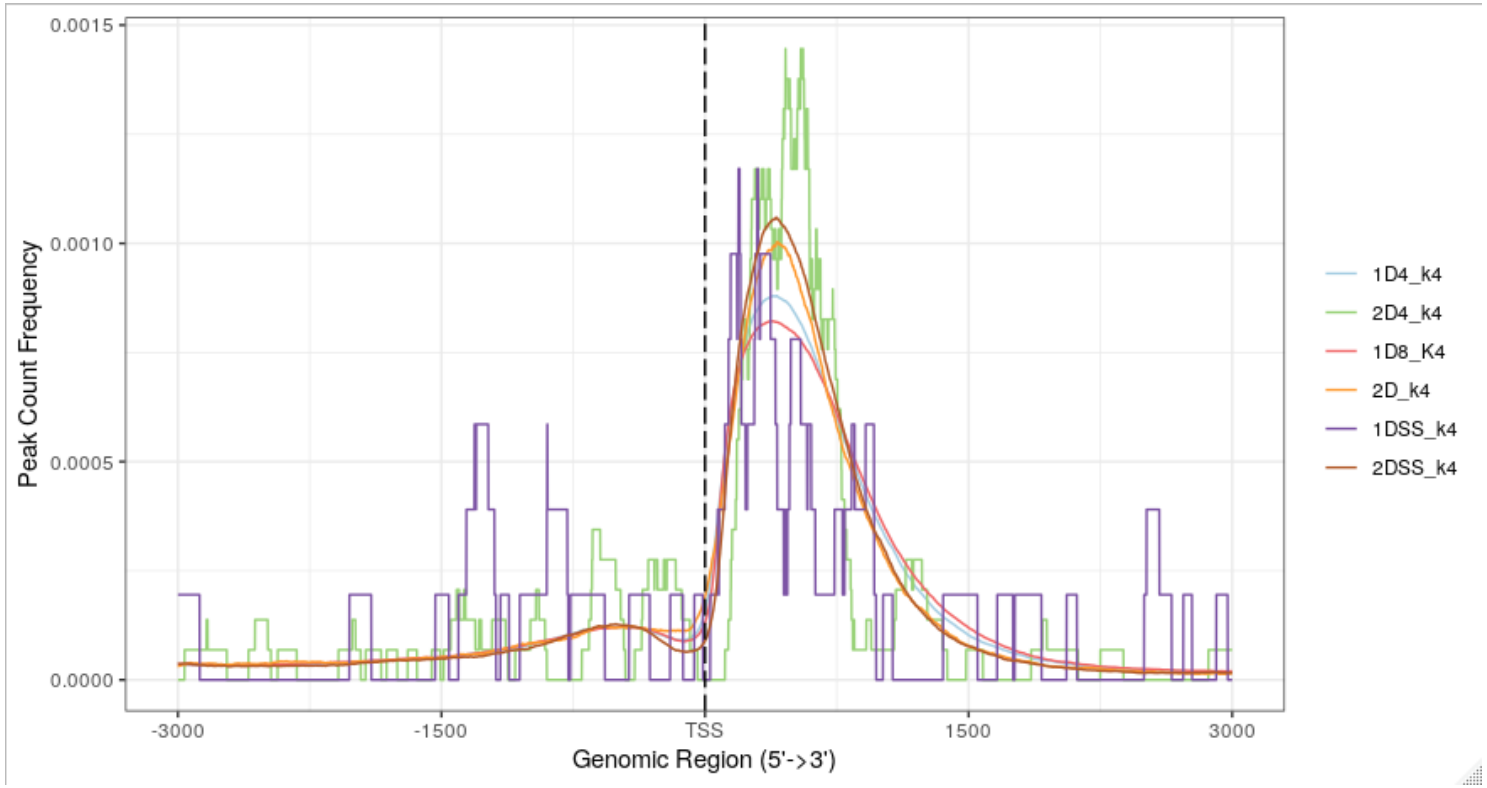
```
peak <- readPeakFile(samplefiles[[11]])
promoter <- getPromoters(TxDb=txdb, upstream=3000, downstream=3000)
tagMatrix <- getTagMatrix(peak, windows=promoter) # peak file name and path for individual
tagHeatmap(tagMatrix, xlim=c(-3000, 3000), color="red")
#peakHeatmap(samplefiles[[11]], TxDb=txdb, upstream=3000, downstream=3000, color="red")
plotAvgProf(tagMatrix, xlim=c(-3000, 3000),  xlab="Genomic Region (5'->3')", ylab = "Read Count Frequency")
```





Profile of several ChIP peak data binding to TSS region

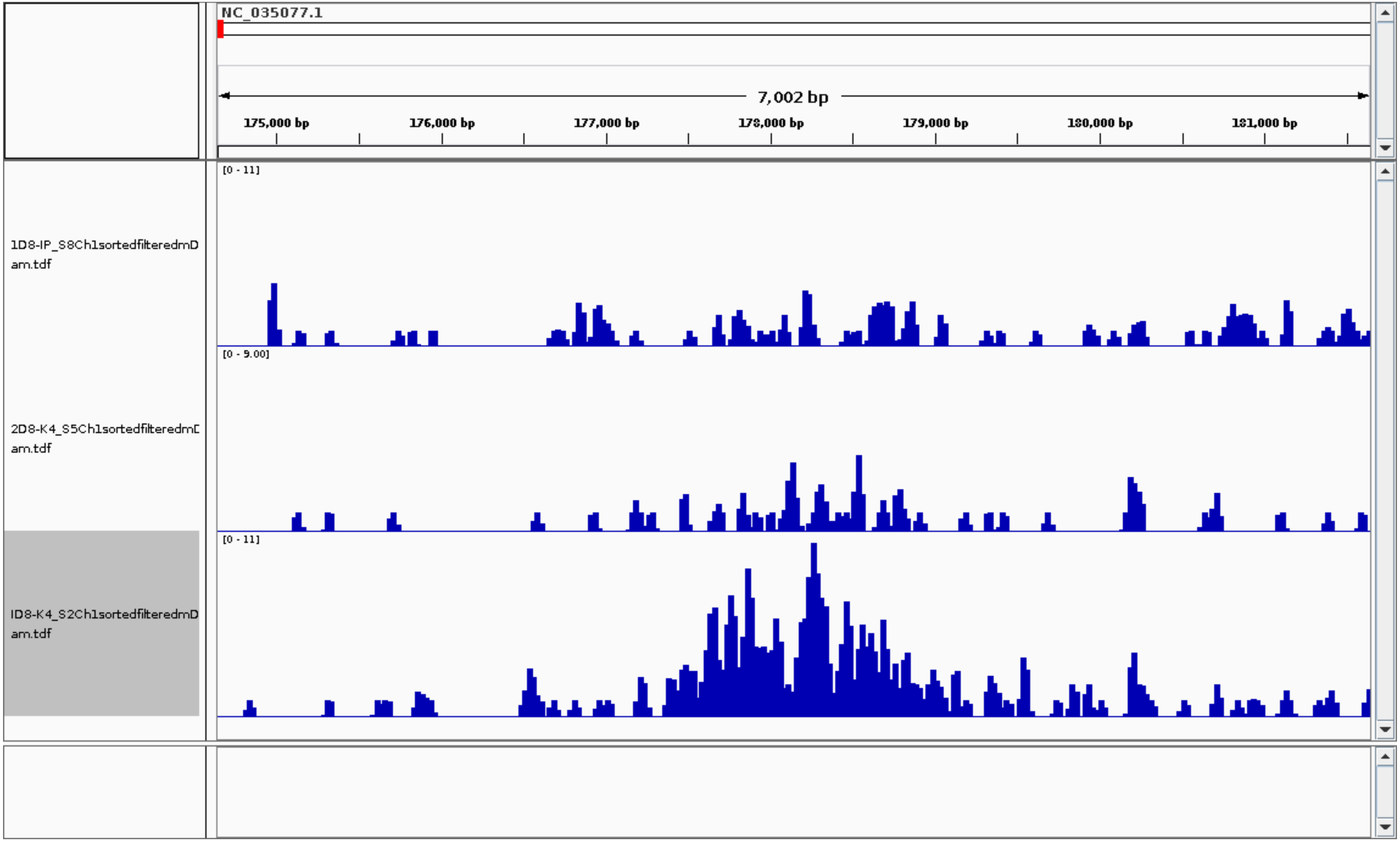
```
samplefiles <-
list.files("/localstorage/Epi_alig/peak_macs_ind",
pattern= ".broadPeak", full.names=T)

samplefiles_sortk4<- c(samplefiles[2],
samplefiles[7], samplefiles[12],
samplefiles[9],samplefiles[5], samplefiles[11])

samplefilesk4 <- as.list(samplefiles_sortk4)

names(samplefilesk4) <-
c("1D4_k4","2D4_k4","1D8_K4","2D_k4","1DSS_k
4", "2DSS_k4")


tagMatrix <- lapply(samplefilesk4, getTagMatrix,
windows=promoter)

plotAvgProf(tagMatrix, xlim=c(-3000, 3000))
```



Qualitative assessment (visualization) using IGV

- Genome browser used for assessing the quality of your alignment.
    - Use to visualize the alignment
- IGV takes .tdf files for reads counts. First, convert the bam files into tdf with following steps:
    - Select reference genome (you may need to create genome by selecting Genomes tab and the load genome from file "fasta file").
    - Select "Run igvtools" form the "Tools" menu.
    - In command, select "Count." Then select the Input file. Keep other options as default and click "Run."

- After conversion, you'll get five files with extension ".bam.tdf"
- Now click File -> Load from File and select the tdf files,
- The coverage will be shown as bar plots.


NC_035077.1
7,002 bp
175,000 bp
176,000 bp
177,000 bp
178,000 bp
179,000 bp
180,000 bp
181,000 bp
[0 - 11]
1D8-IP_S8Ch1sortedfilteredmD am.tdf
[0 - 9.00]
2D8-K4_S5Ch1sortedfilteredmD am.tdf
[0 - 11]
ID8-K4_S2Ch1sortedfilteredmD am.tdf

DiffBind: Differential binding analysis of ChIPSeq peak data

- Identifying sites that are differentially bound between two sample groups
- Works with multiple peak sets simultaneously (transcription factor ,histone marks, experimental conditions, replicates)
- Count reads in peaks in all the replicates and conditions
- Identifying statistically significantly differentially bound sites based on evidence of binding affinity
- Perform edgeR or DESeq2 analysis (assign a p-value and FDR to each candidate binding site indicating confidence that they are differentially bound)
- Provides various plotting functions

DiffBind

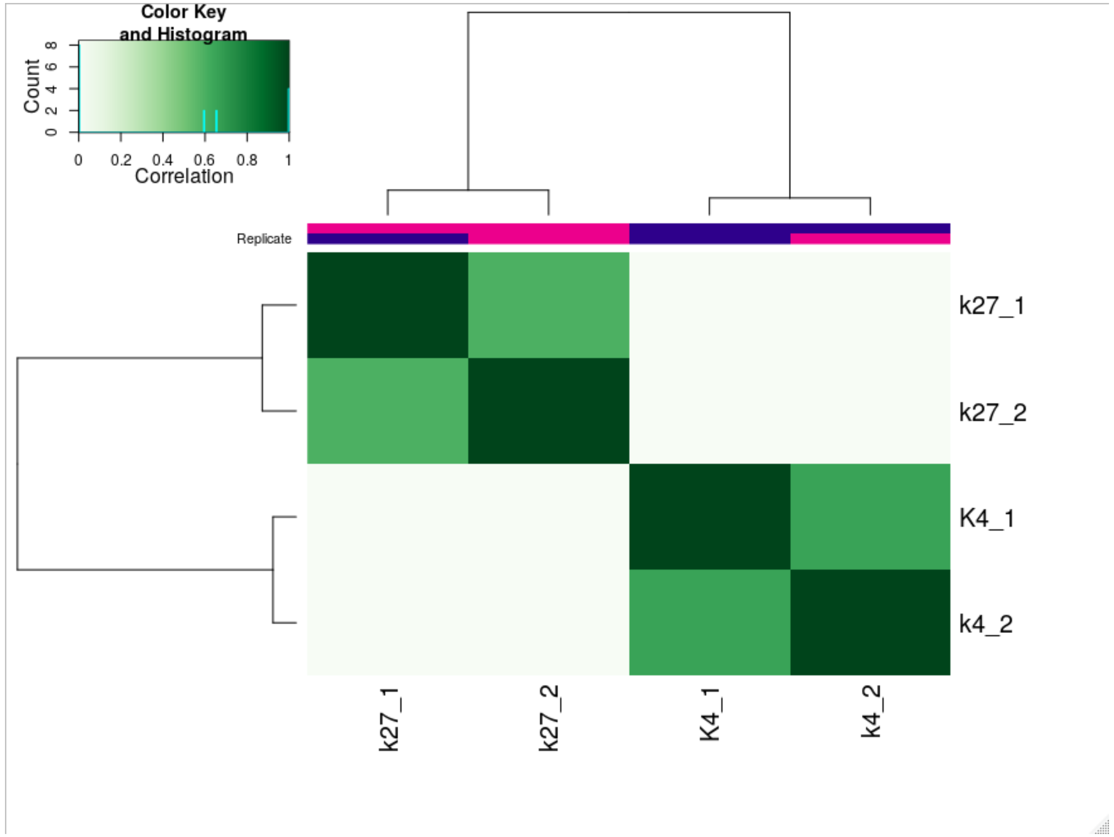
- First step is to create a sample sheet that hold the information and the location of the map and peak files
- SampleID" "Tissue" "Factor" "Condition" "Treatment“ "Replicate" "bamReads" "ControlID" "bamControl" "Peaks“ "PeakCaller“
- read sample file
    - samples <- read.csv('meta/sample.csv')
- create an object
    - dbObj <- dba(sampleSheet=samples)
- use alignment files and compute count information for each of the peaks/regions
    - dbObj <- dba.count(dbObj, bUseSummarizeOverlaps=TRUE)

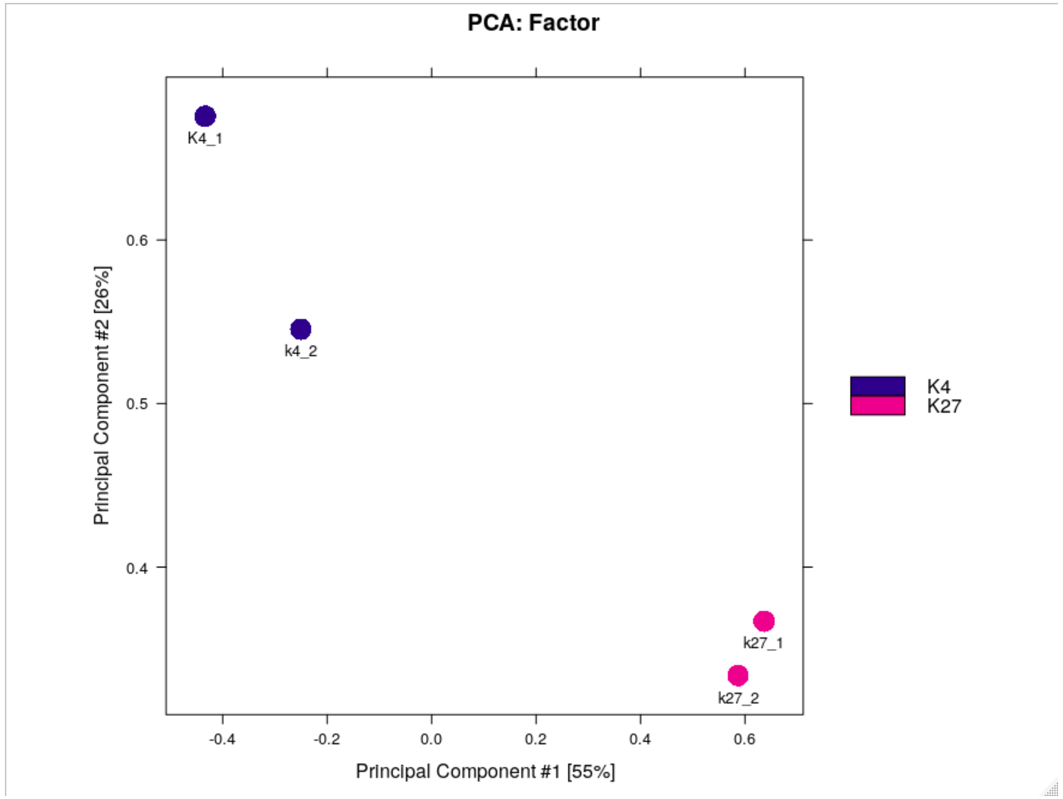
DiffBind: deeper insight into how samples are associated.

- PC plot
  - dba.plotPCA(dbObj, attributes=DBA_FACTOR, label=DBA_ID)

- plot a correlation heatmap
  - plot(dbObj)

Diffbind correlation heat map and PCA

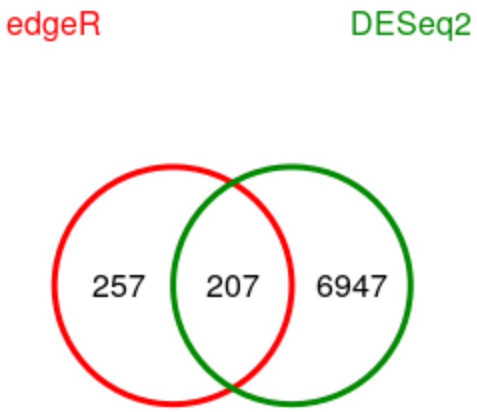


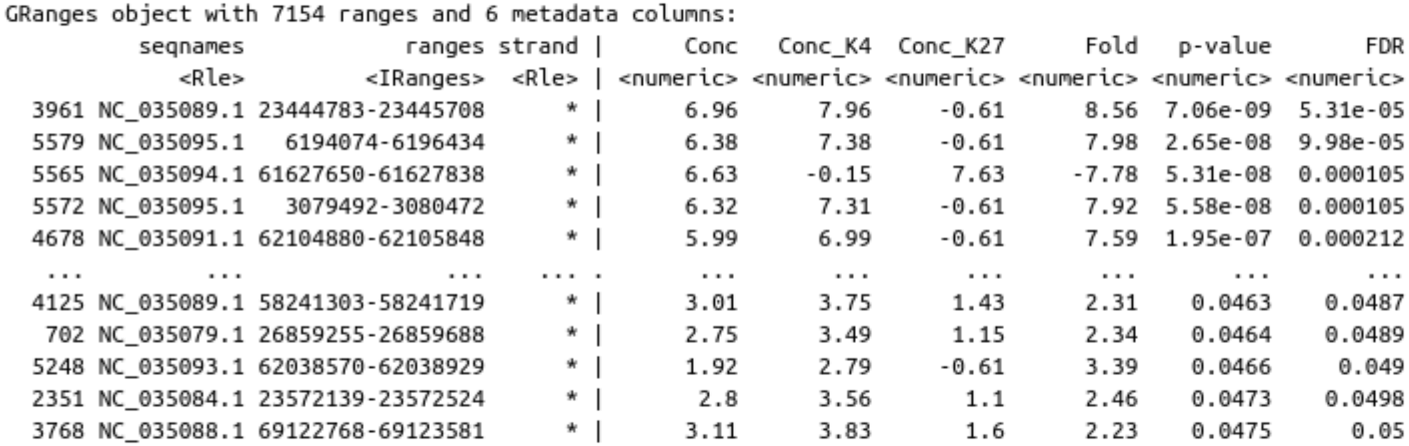


DiffBind

- Indicate which samples we want to compare to one another
  - dbObj <- dba.contrast(dbObj, categories=DBA_FACTOR, minMembers = 2)
- Performing the differential enrichment analysis using both DESeq2 and edgeR
  - dbObj <- dba.analyze(dbObj, method=DBA_ALL_METHODS)
  - dba.show(dbObj, bContrasts=T)

DiffBind differential enrichment analysis results





```
GRanges object with 7154 ranges and 6 metadata columns:
         seqnames               ranges strand |      Conc   Conc_K4  Conc_K27      Fold   p-value       FDR
            <Rle>            <IRanges>  <Rle> | <numeric> <numeric> <numeric> <numeric> <numeric> <numeric>
  3961 NC_035089.1 23444783-23445708      * |      6.96      7.96     -0.61      8.56  7.06e-09  5.31e-05
  5579 NC_035095.1   6194074-6196434      * |      6.38      7.38     -0.61      7.98  2.65e-08  9.98e-05
  5565 NC_035094.1 61627650-61627838      * |      6.63     -0.15      7.63     -7.78  5.31e-08  0.000105
  5572 NC_035095.1   3079492-3080472      * |      6.32      7.31     -0.61      7.92  5.58e-08  0.000105
  4678 NC_035091.1 62104880-62105848      * |      5.99      6.99     -0.61      7.59  1.95e-07  0.000212
   ...         ...               ...    ... .       ...       ...       ...       ...       ...       ...
  4125 NC_035089.1 58241303-58241719      * |      3.01      3.75      1.43      2.31    0.0463    0.0487
   702 NC_035079.1 26859255-26859688      * |      2.75      3.49      1.15      2.34    0.0464    0.0489
  5248 NC_035093.1 62038570-62038929      * |      1.92      2.79     -0.61      3.39    0.0466     0.049
  2351 NC_035084.1 23572139-23572524      * |       2.8      3.56       1.1      2.46    0.0473    0.0498
  3768 NC_035088.1 69122768-69123581      * |      3.11      3.83       1.6      2.23    0.0475      0.05
```

Retrieve each histone
K4_enrich <- dplyr::filter(out, FDR < 0.05 & Fold > 0)
K27_enrich = dplyr::filter(out, FDR < 0.05 & Fold < 0)

http://bioconductor.org/packages/release/bioc/vignettes/DiffBind/inst/doc/DiffBind.pdf

DiffBind

```
# Load needed library
library(DiffBind)
library(tidyverse)
#read sample file
samples <- read.csv('meta/sample.csv')
#create an object
dbObj <- dba(sampleSheet=samples)
#use alignment files and compute count information for each of the peaks/regions
dbObj <- dba.count(dbObj, bUseSummarizeOverlaps=TRUE)
#PC plot
dba.plotPCA(dbObj,  attributes=DBA_FACTOR, label=DBA_ID)
#plot a correlation heatmap
plot(dbObj)
#tell DiffBind which samples we want to compare to one another
dbObj <- dba.contrast(dbObj, categories=DBA_FACTOR, minMembers = 2)
# Performing the differential enrichment analysis using both DESeq2 with and edgeR
dbObj <- dba.analyze(dbObj, method=DBA_ALL_METHODS)
dba.show(dbObj, bContrasts=T)
# Visualizing the results
dba.plotVenn(dbObj,contrast=1,method=DBA_ALL_METHODS)
```

Hands on session

- Read QC fastqc
- Alignment bowtie
- Post alignment
- Peak calling MACS2
- Handling Replicate
- ChipQC
- Visualization igv
- Peak Annotation chipSeeker
- Diffbind

Questions??????