UNIVERSITY OF OXFORD

Bioinformatics analysis of ChIP-Seq data

Silvia Salatino, PhD

High-throughput Bioinformatician
Wellcome Centre for Human Genetics, Oxford

ChIP-Seq module for the DPhil programme
Genomic Medicine and Statistics

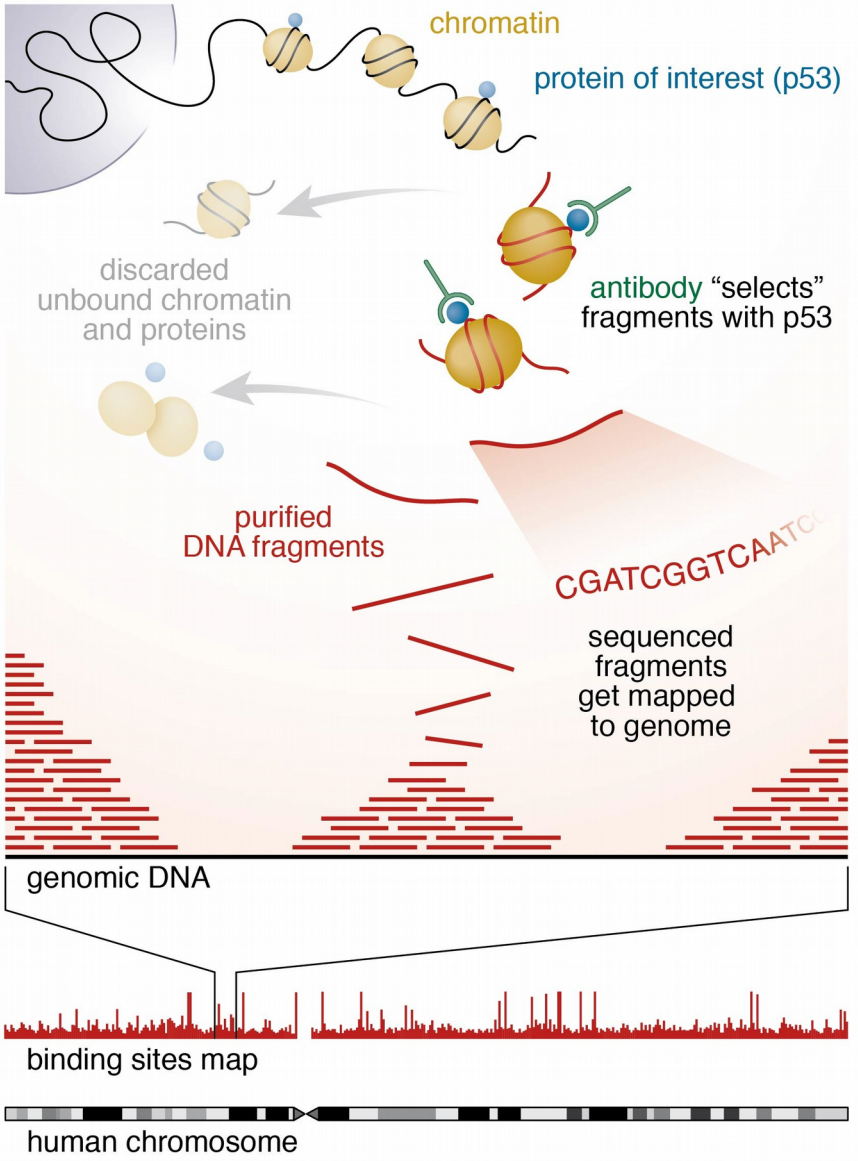
30.11.2018

Table of contents

ChIP-Seq technique overview

ChIP-Seq technique overview: aim

- Aim:
  - identify all the genome-wide binding sites of a protein of interest (POI)



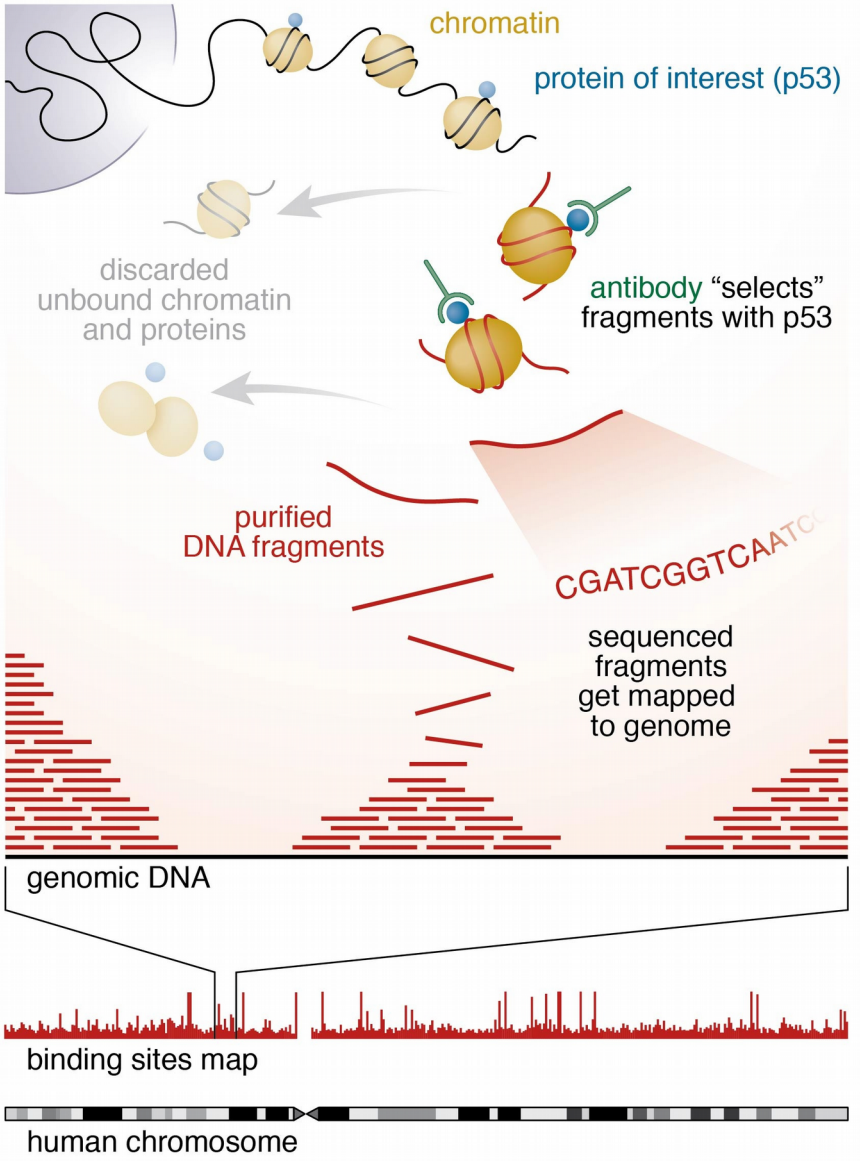
Park et al. 2009, Nature Review Genetics

ChIP-Seq technique overview: requirements

- Aim:
  - identify all the genome-wide binding sites of a protein of interest (POI)

- Requirements:
  - antibody with a good affinity for the POI
  - known reference genome
  - sufficient sequencing depth

  *(e.g. human: 3 Gb / 300 bp = 10^7 fragments; assuming 3000 binding sites and an enrichment of 10x (from qPCR), you'll have 10x * 3000 locations = $3*10^4$, therefore you need $1*10^7 + 3*10^4 \sim= 1*10^7$ reads per sample)*



Park et al. 2009, Nature Review Genetics

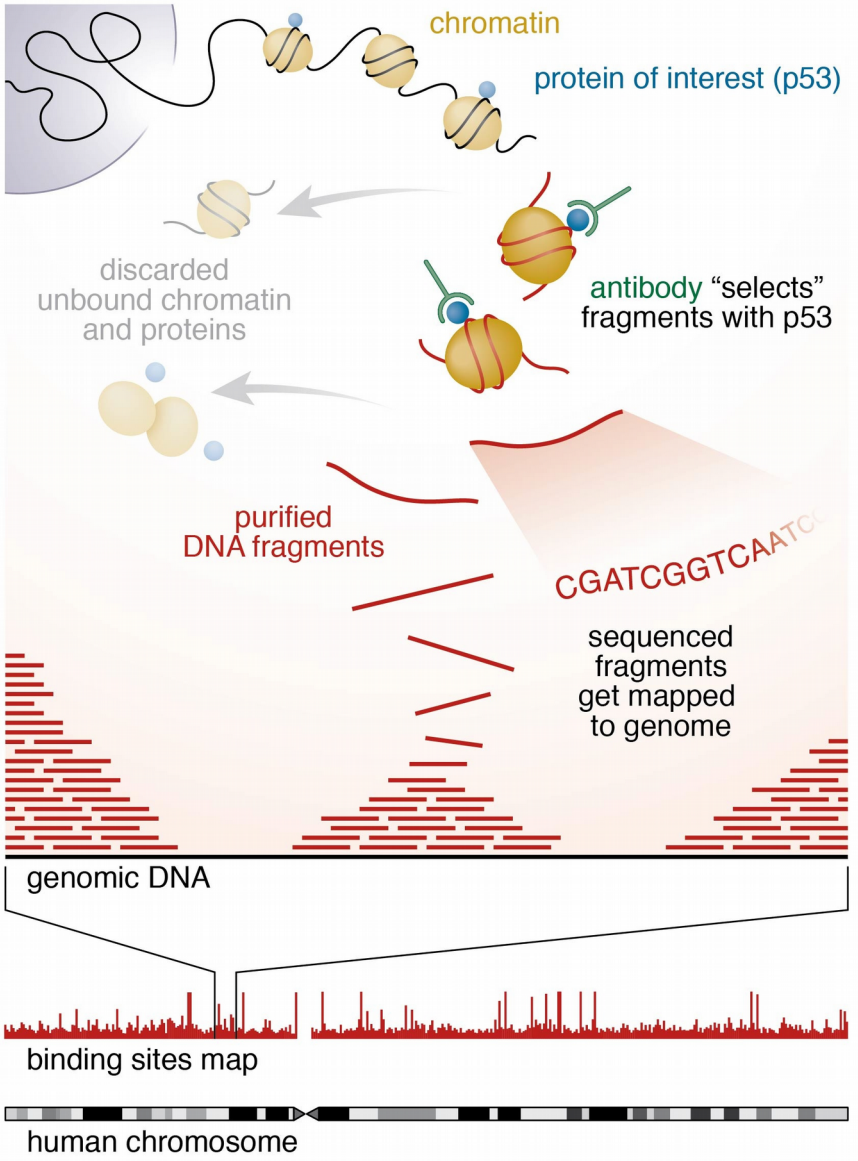
ChIP-Seq technique overview: steps

- Aim:
  - identify all the genome-wide binding sites of a protein of interest (POI)

- Requirements:
  - antibody with a good affinity for the POI
  - known reference genome
  - sufficient sequencing depth

  *(e.g. human: 3 Gb / 300 bp = 10^7 fragments; assuming 3000 binding sites and an enrichment of 10x (from qPCR), you'll have 10x * 3000 locations = $3*10^4$, therefore you need $1*10^7 + 3*10^4 \sim= 1*10^7$ reads per sample)*

- Steps:
  1. cross-link DNA and POI
  2. fragment DNA
  3. chromatin immunoprecipitation
  4. reverse cross-links and purify DNA
  5. add adapters and sequence
  6. ***bioinformatics data analysis***



Park et al. 2009, Nature Review Genetics

ChIP-Seq technique overview: study design

- It is recommended to perform the experiment at least in **duplicate**, to reduce false positives.

  There could be *biological* replicates (individuals, strains, gender, et c.), or *technical* replicates (library construction protocols, sequencing pools, sequencing runs, et c.).

- Due to imperfect antibodies and other factors, many sequenced reads do not originate from the chipped protein and are referred as background reads. Since background reads are not uniformly distributed, it is crucial to also sequence a **control sample**, whose signal can be subtracted from the treatment sample to further reduce false positive peaks.

  There are different types of controls for ChIP-Seq:

  - "input" DNA (or Whole-cell extract): DNA isolated from cells that have been cross-linked and fragmented under the same conditions as the IP DNA, but without using an antibody
  - "mock" ChIP (or IgG control): use an antibody that reacts with a non-nuclear antigen
  - use the same antibody, but with a "knock-out" (KO) or "knock-down" (KD) of the protein of interest

- Another factor likely to influence the results of the experiment is the **fragmentation**. Some researchers use specific *endonucleases* to cut DNA, although the most commonly used method is probably *sonication*. The length and intensity of the ultrasounds determines the size of the DNA fragments obtained.

Initial QC metrics and data filtering

Initial QC metrics and data filtering: the FASTQ format

The **FASTQ** format is a text-based file format to store both a biological sequence (DNA or RNA) and its corresponding quality scores. Both the sequence letter and quality score are each encoded with a single ASCII character for brevity. A FASTQ file normally uses four lines per sequence:

- Line 1 begins with a '@' character and is followed by a sequence identifier and an optional description
- Line 2 is the raw sequence letters
- Line 3 begins with a '+' character and is optionally followed by the same sequence identifier (and any description)
- Line 4 encodes the quality values for the sequence in Line 2 (therefore must be as long as Line 2).

For example, a FASTQ file containing a single sequence might look like this:

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Here are the quality value characters in left-to-right increasing order of quality (ASCII):

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

E.g. Python code to convert ASCII to number and viceversa:

```
>>> ord("%")
37
>>> chr(37)
'%'
```

Source: Wikipedia

Initial QC metrics and data filtering: the FASTQ format

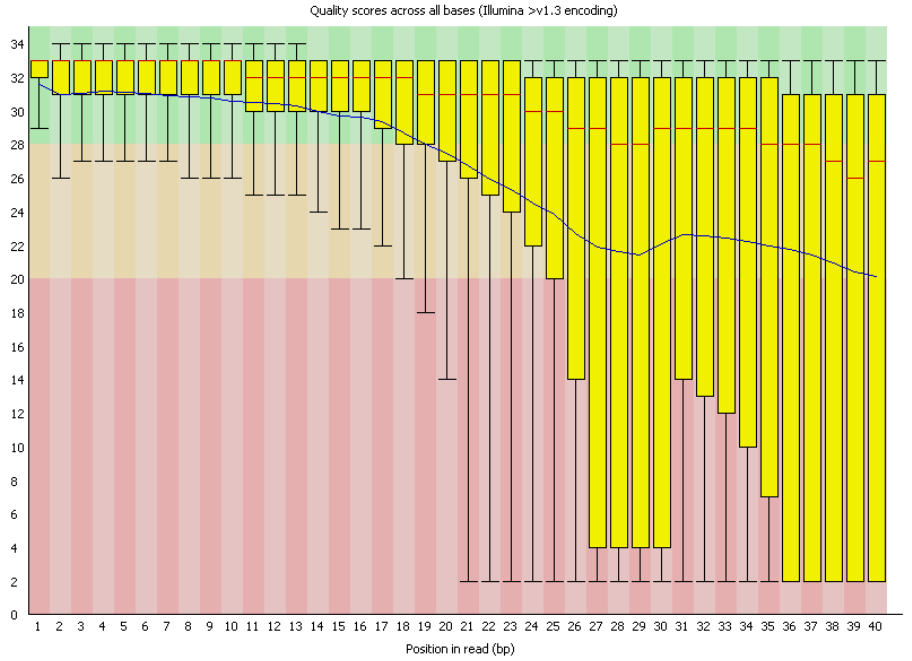
Each quality value is an integer mapping of ***p*** (the probability that the corresponding base call is incorrect). It is used to assess the reliability of base calls and it is called “Phred quality score”:

$$Q_{sanger} = -10\ log_{10}\ p$$

The currently used format can encode a Phred quality score from 0 to 62 using ASCII 64 to 126 (although in raw read data Phred scores from 0 to 40 only are expected).

```
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.....................
..............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII.....................
.................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ.....................
LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL.....................................................
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
|                         |    |        |                              |                     |
33                        59   64       73                            104                   126
 0........................26...31.......40
                          -5....0........9.............................40
                               0........9.............................40
                                 3.....9.............................40
 0.2......................26...31........41

S - Sanger        Phred+33,  raw reads typically (0, 40)
X - Solexa        Solexa+64, raw reads typically (-5, 40)
I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
J - Illumina 1.5+ Phred+64,  raw reads typically (3, 40)
    with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
    (Note: See discussion above).
L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

Quality scores tend to decrease along the read



Therefore, it is often necessary to:

- trim low-quality ends off reads
- delete poor quality reads overall

Source: Wikipedia

Initial QC metrics and data filtering: adapter trimming

- Adapter trimming:
  - may increase the fraction of successfully mapped reads
  - helps reducing the mapping computing time
  - troublesome if you don't know the exact adapter sequence
  - multiple configurations possible: 5' (leading), 3' (trailing), SE, PE, full, partial, >1 adapter, ...

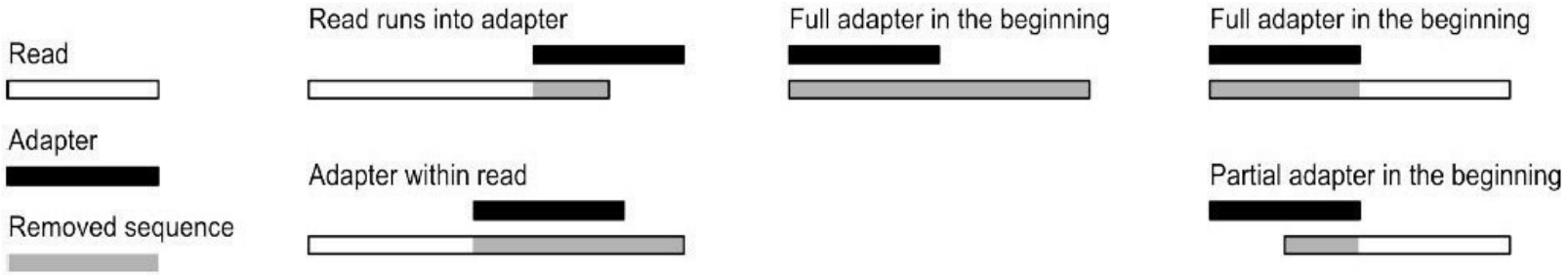
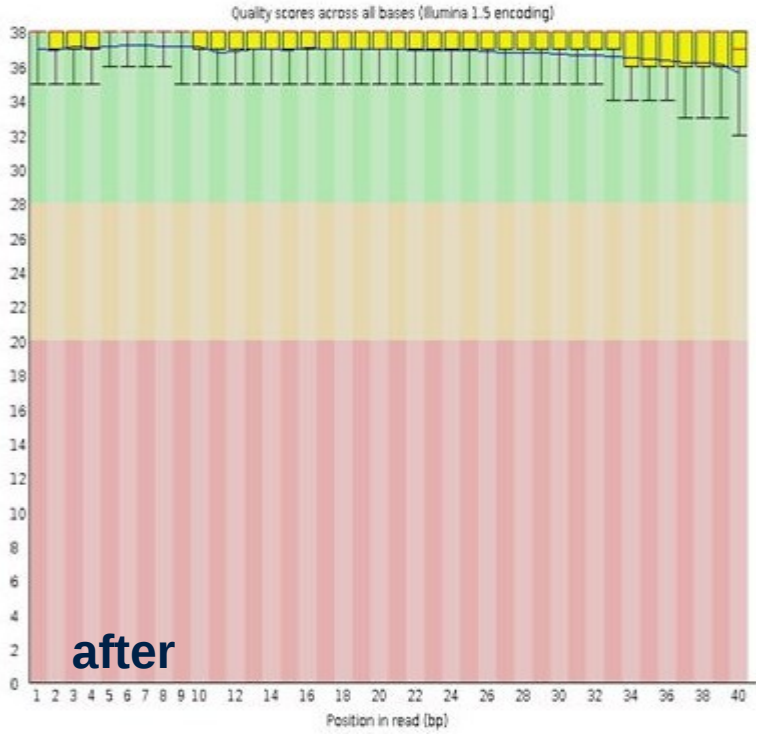


Figure 1. This illustration shows all possible alignment configurations between the read and adapter sequence.

Initial QC metrics and data filtering: quality trimming

- Quality trimming:
  - may increase the fraction of successfully mapped reads
  - leads to loss of information
  - different approaches possible:

    a) remove leading low-quality or N bases
    b) remove trailing low-quality or N bases
    c) scan the read base-by-base and cut when the average (or the median) drop below a certain cutoff
    d) drop reads below a certain length cutoff
    e) ...

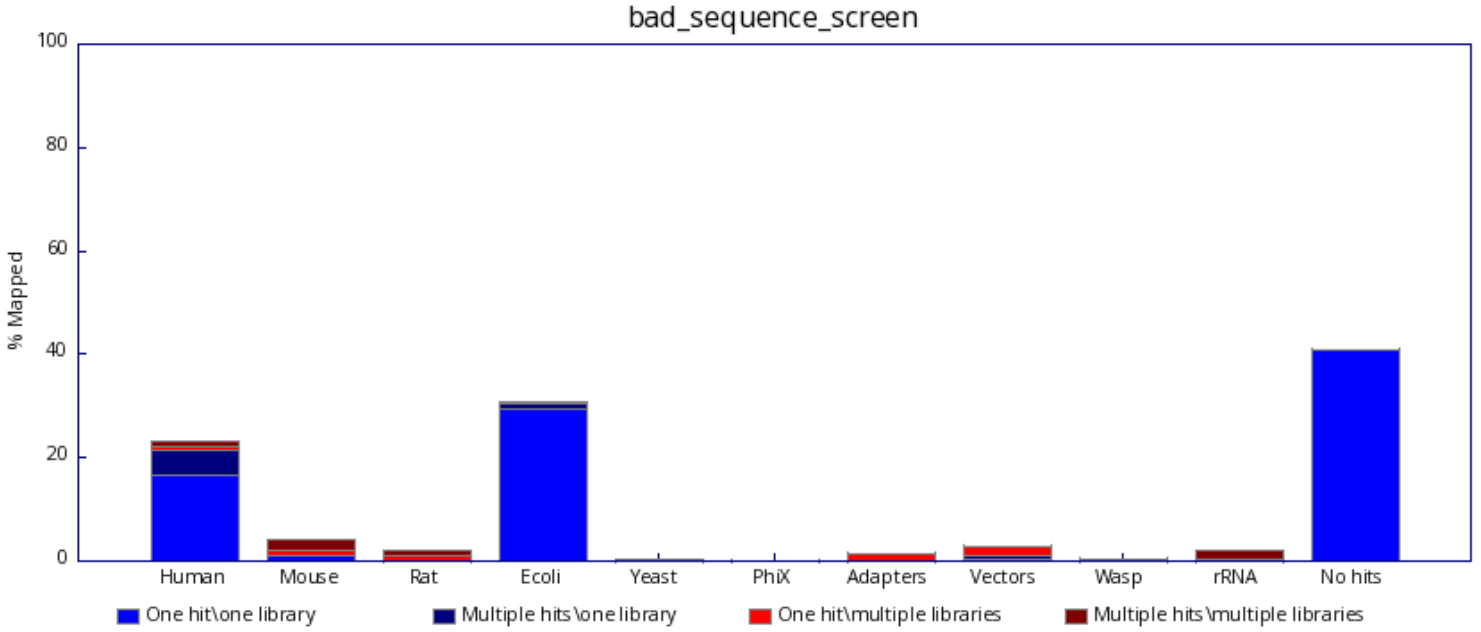


Initial QC metrics and data filtering: trimming (summary)

There are lots of trimming tools doing either or both types of data processing: **Trimmomatic**, **Skewer**, **FASTX-Toolkit**, **Trim Galore**, **Cutadapt**, etc.

It is useful to have a look at the overall quality of the reads before and after trimming (e.g. using **FastQC** or a simple script to scan all the reads and produce plots as in the previous few slides).

To exclude the possibility of contaminants in the data, one can also use **FastQ screen**, a wrapper around Bowtie that produces plots like this:



Mapping to the reference genome

Mapping to the reference genome: software

Lots of mappers available:

- *Bowtie2*
- *BWA and BWA-mem*
- *Novoalign*
- *Isaac*
- *Stampy*
- *STAR*
- *…*

Each of these tools has several parameters that can be defined for the alignment process, including:

- *# of differences allowed between reference and query*
- *seed length and # of differences allowed in the seed*
- *penalty for gap openings and gap extensions*
- *allow / disallow indels*
- *...*

Changing these parameters will affect the number and quality of reads that map to reference and the time it takes to complete mapping a sample:

- *Too stringent (e.g. no mismatches, unique mappers) → loss of low-enriched regions*
- *Too relaxed (e.g. many mismatches, multi-mappers) → increased FDR*

aln bwa aln [-n maxDiff] [-o maxGapO] [-e maxGapE] [-d nDelTail] [-i nIndelEnd] [-k maxSeedDiff] [-l seedLen] [-t nThrds] [-cRN] [-M misMsc] [-O gapOsc] [-E gapEsc] [-q trimQual] <in.db.fasta> <in.query.fq> > <out.sai>

Find the SA coordinates of the input reads. Maximum *maxSeedDiff* differences are allowed in the first *seedLen* subsequence and maximum *maxDiff* differences are allowed in the whole sequence.

OPTIONS:

-n *NUM* Maximum edit distance if the value is INT, or the fraction of missing alignments given 2% uniform base error rate if FLOAT. In the latter case, the maximum edit distance is automatically chosen for different read lengths. [0.04]

-o *INT* Maximum number of gap opens [1]

-e *INT* Maximum number of gap extensions, -1 for k-difference mode (disallowing long gaps) [-1]

-d *INT* Disallow a long deletion within INT bp towards the 3'-end [16]

-i *INT* Disallow an indel within INT bp towards the ends [5]

-l *INT* Take the first INT subsequence as seed. If INT is larger than the query sequence, seeding will be disabled. For long reads, this option is typically ranged from 25 to 35 for '-k 2'. [inf]

-k *INT* Maximum edit distance in the seed [2]

-t *INT* Number of threads (multi-threading mode) [1]

-M *INT* Mismatch penalty. BWA will not search for suboptimal hits with a score lower than (bestScore-misMsc). [3]

-O *INT* Gap open penalty [11]

-E *INT* Gap extension penalty [4]

-R *INT* Proceed with suboptimal alignments if there are no more than INT equally best hits. This option only affects paired-end mapping. Increasing this threshold helps to improve the pairing accuracy at the cost of speed, especially for short reads (~32bp).

-c Reverse query but not complement it, which is required for alignment in the color space. (Disabled since 0.6.x)

-N Disable iterative search. All hits with no more than *maxDiff* differences will be found. This mode is much slower than the default.

-q *INT* Parameter for read trimming. BWA trims a read down to argmax_x{\sum_{i=x+1}^l(INT-q_i)} if q_l<INT where l is the original read length. [0]

-I The input is in the Illumina 1.3+ read format (quality equals ASCII-64).

-B *INT* Length of barcode starting from the 5'-end. When *INT* is positive, the barcode of each read will be trimmed before mapping and will be written at the **BC** SAM tag. For paired-end reads, the barcode from both ends are concatenated. [0]

Screenshot from the BWA manual

Mapping to the reference genome:SAM/BAM format

The standard output format of short-read alignment programs is a Sequence/Alignment Map (**SAM**) file for each sample, often used in its binary format to save disk space (**BAM**):

header lines

read alignments

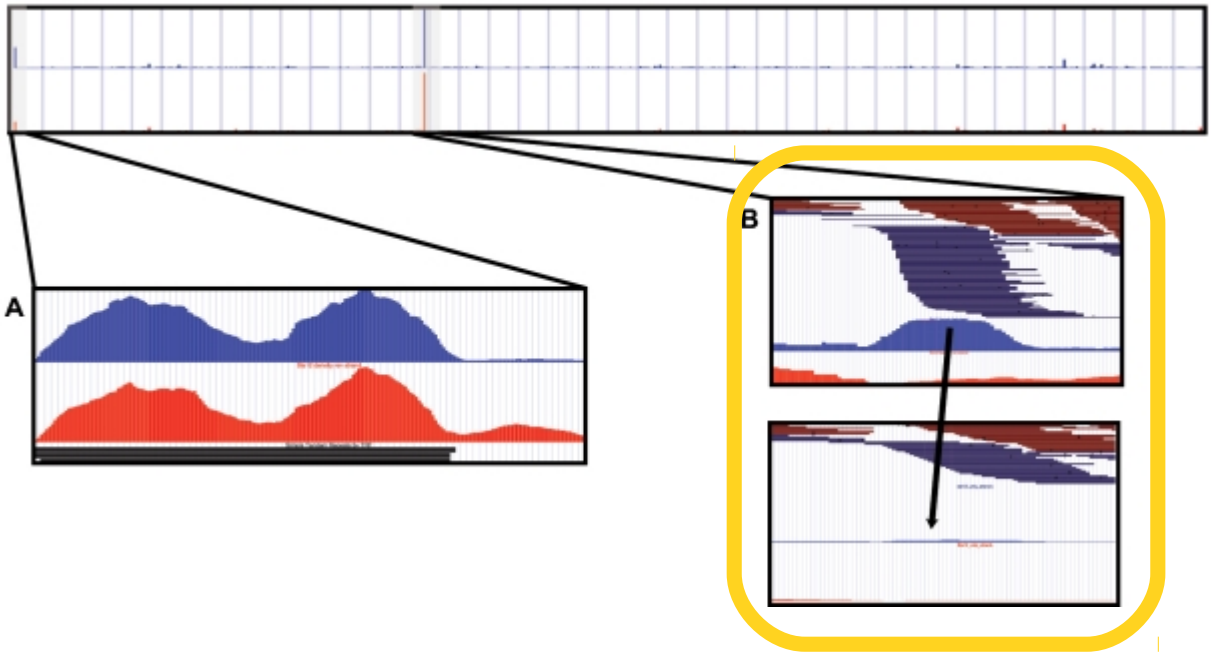
```
@HD     VN:1.4  GO:none SO:coordinate
@SQ     SN:1    LN:249250621
@SQ     SN:2    LN:243199373
@SQ     SN:X    LN:155270560
@SQ     SN:Y    LN:59373566
@RG     ID:WTCHG_179849_265     PL:ILLUMINA     PU:150425_D00529_0179_BHK325ADXX_1      LB:PIA1801A1    PI:220
DT:2015-04-26T01:00:00+0100     SM:RM:GST 1     CN:WTCHG
@PG     ID:stampy       PN:stampy       VN:1.0.23_(r2059)       CL:-t 8 -g /tmp/hs37d5 -h /tmp/hs37d5 --bamkeepg
oodreads --readgroup=ID:WTCHG_179849_265,SM:RM:GST 1,PL:ILLUMINA,PU:150425_D00529_0179_BHK325ADXX_1,LB:VAG1801A1
,DT:2015-04-26,CN:WTCHG,PI:220 --comment=@MISC/WTCHG_179849_265.comments.txt -M /tmp/WTCHG_179849_265.bwa.bam
@CO     RG:WTCHG_179850_265     FC:HK325ADXX    SN:PIA1801A1    PO:2-343008/15_MPX_10nM MT:265  IX:TCTGGCTG
PR:P150217      GE:hs37d5       CT:false        LT:Our indexes ChIP     LN:2    SV:RunMode=RapidRun,ApplicationV
ersion=2.2.58,FPGAVersion=7.9.7,CPLDVersion=3.0.0,RTAVersion=1.18.64,ChemistryVersion=Illumina_Bruno Fluidics Co
ntroller_0_v2.0340,RecipeFragmentVersion=1.5.21.0,bclToFastq=1.8.4,startPipeline=2.02,FCdetails.pm=2.0  TI:64
RW:2578199      PF:2471469      RC:65782424
HISEQ2500-10:179:HK325ADXX:2:1111:6101:9242     113     1       9957    0       51M     5       58245901
0       GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGTGACCCA     @@B<7)))&))006909673-0006BDDDDBCE@F?0FD?@<=BDDFD
C@@     PQ:i:235        SM:i:2  UQ:i:66 MQ:i:96 XQ:i:41 NM:i:2  RG:Z:WTCHG_179850_265
HISEQ2500-10:179:HK325ADXX:1:2211:16720:24687   163     1       10004   0       2I49M   =       10197   244
AGCCCTAGCCCTAGCCCTAGCCCTAGCCCTAGCCCTAGCCCTAAC     ???DD>D3CBCD:CDDE@CEB@F@+<CEEA?C>DDDDCD9?@D3B??8D?@
PQ:i:0  SM:i:0  UQ:i:201        MQ:i:0  XQ:i:9  NM:i:9  RG:Z:WTCHG_179849_265
HISEQ2500-10:179:HK325ADXX:2:1106:6422:47604    133     1       10071   0       *       =       10071   0
TGGTTTAGGTTTTGTGTTTGGGATCAGGTAGGGGTAGGGGTAGGTTTAGGG     8?@DDDDDF=CAF?<CA9A:A@<EFA@D1*:CC<:::B;F07/B88=BBA@
MQ:i:1  XQ:i:248        RG:Z:WTCHG_179850_265

sample_file.bam (END)
```

A very useful suite of tools to handle these (usually 10s of Gigabytes!) files is “**samtools**”, which allow to visualise, filter and calculate a number of metrics (e.g. coverage) on both SAM and BAM files in an easy and efficient manner.

Peak calling

Peak calling: de-duplication

The main purpose of read de-duplication (= keep 1 read per position) is to reduce the effects of a possible (and common) **PCR amplification bias** that might have happened during library construction.



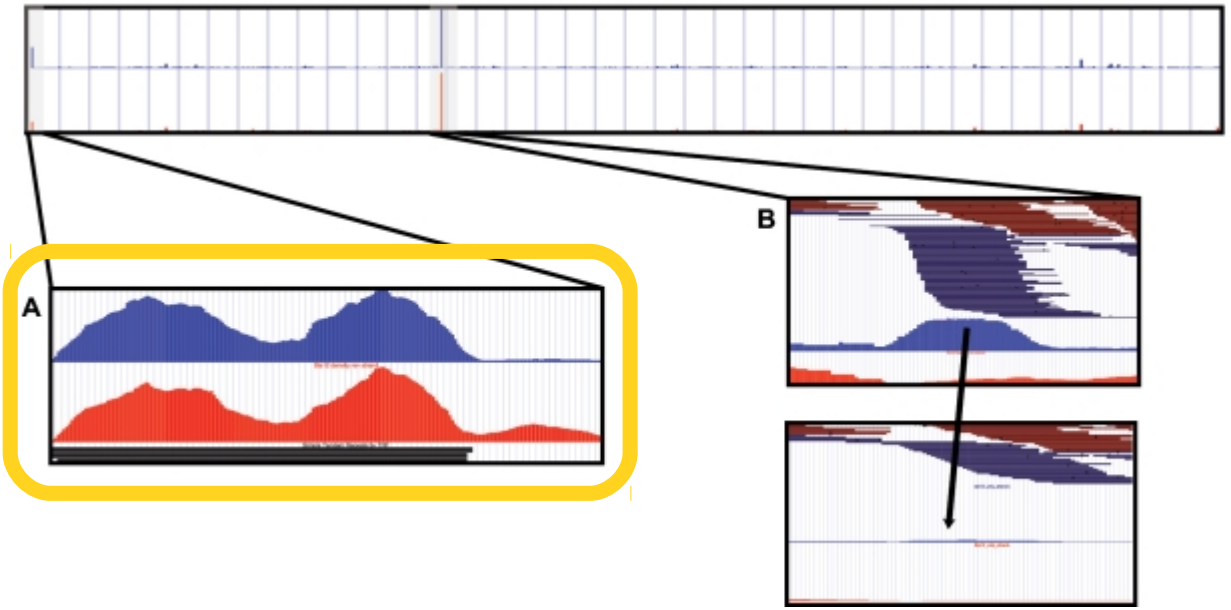
This can be done, for example, by using Picard's MarkDuplicate tool (or, alternatively, samtools).

Gain the computational benefit of reducing the reads to process downstream

Risk of setting a hard cap to the dynamic range of measurements and losing potentially true signal

Peak calling: repeats filtering

Low-complexity or repeated regions can generate high peaks with no shift between plus and minus strands.



These can be removed after peak calling by removing any peak that overlaps a repeat or low-complexity region. ENCODE provides a “blacklist” of these regions that can be used to filter the peaks.

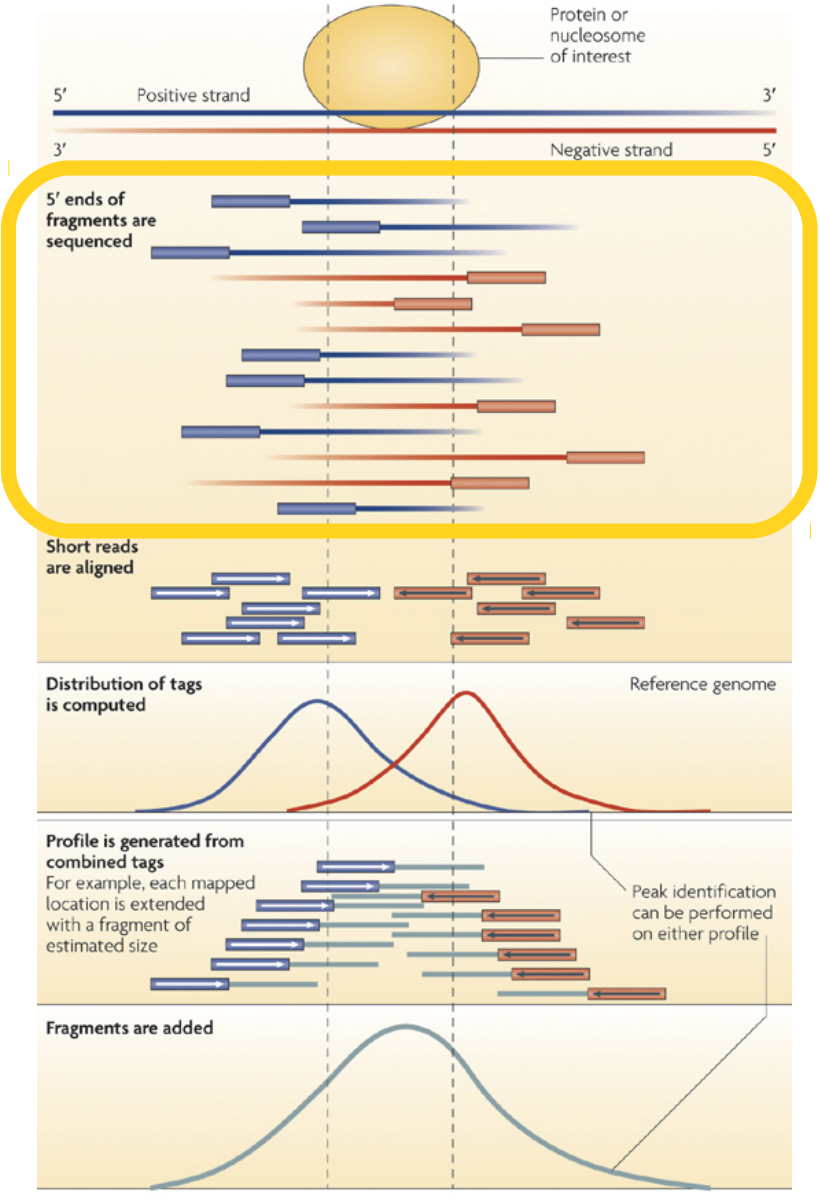
Reduce the number of false positive peaks

Might lose true peaks if the protein of interest is a TF that binds within repeats (e.g. YY1)

Peak calling: reads shift

Since only the ends of each DNA fragment is sequenced, reads will map at the left and right sides of the original binding site.

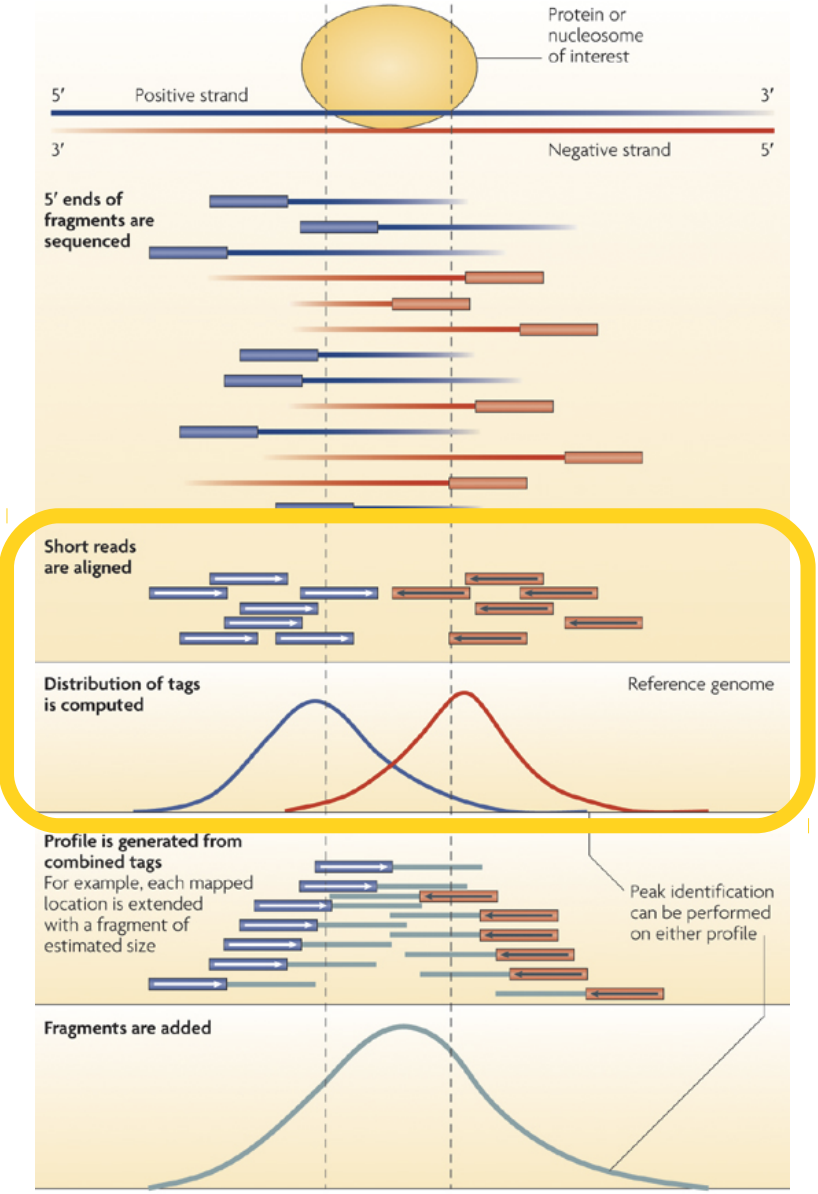
It is therefore necessary to shift all the reads towards 3' by half the estimated fragment size to better represent the precise DNA-protein interaction site.



Park et al. 2009, Nature Review Genetics

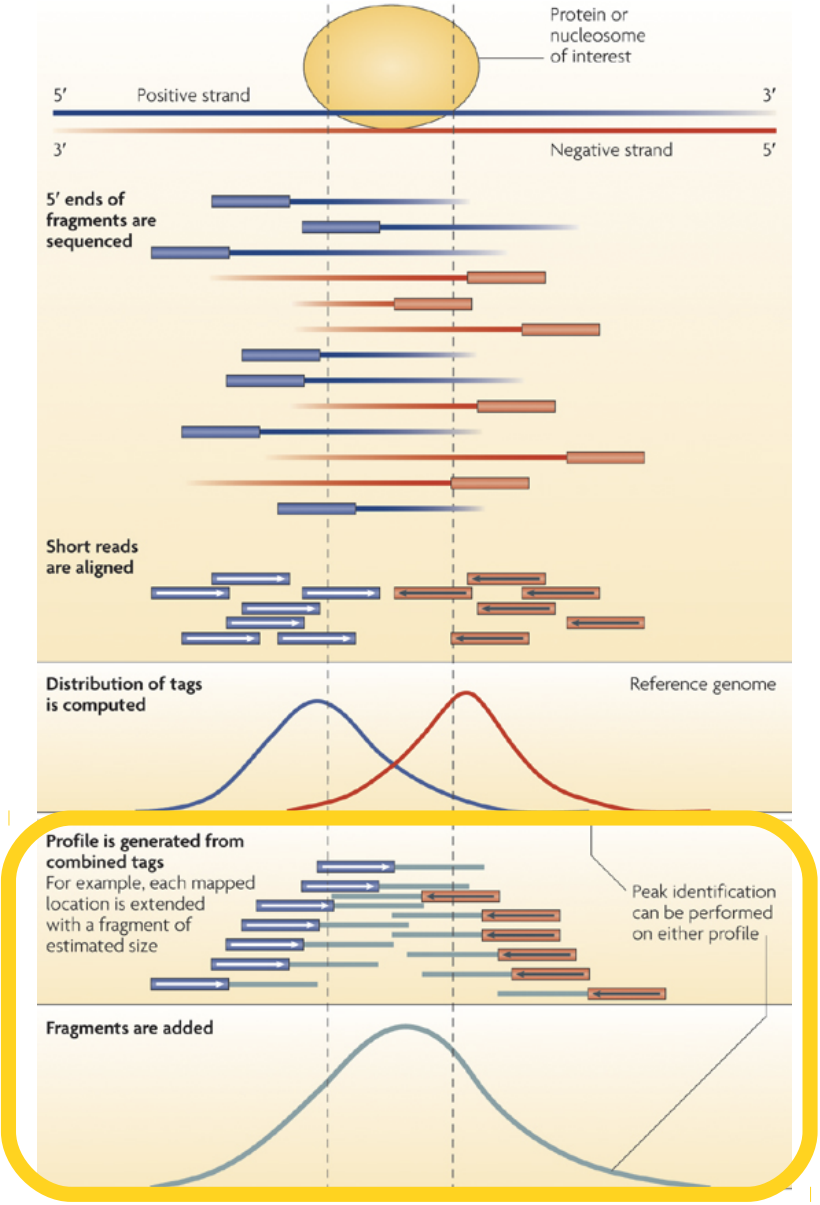
Peak calling: reads shift

Plotting the tag density around expected binding sites should show a bimodal enrichment pattern (with + reads upstream of the site and - reads downstream) and provide an indication of the required shift size.



Park et al. 2009, Nature Review Genetics

Peak calling: reads shift



Reads can be either shifted through a shell script, or by a peak caller.

For example, MACS2 slides a window along the genome to find regions enriched with respect to a random read distribution, then samples 1000 of these peaks, separates their + and – reads, and aligns them by the midpoint between their reads centres. The distance 'd' between the modes of the + and - peaks is used by MACS2 to shifts all the reads by d/2 toward the 3' ends.

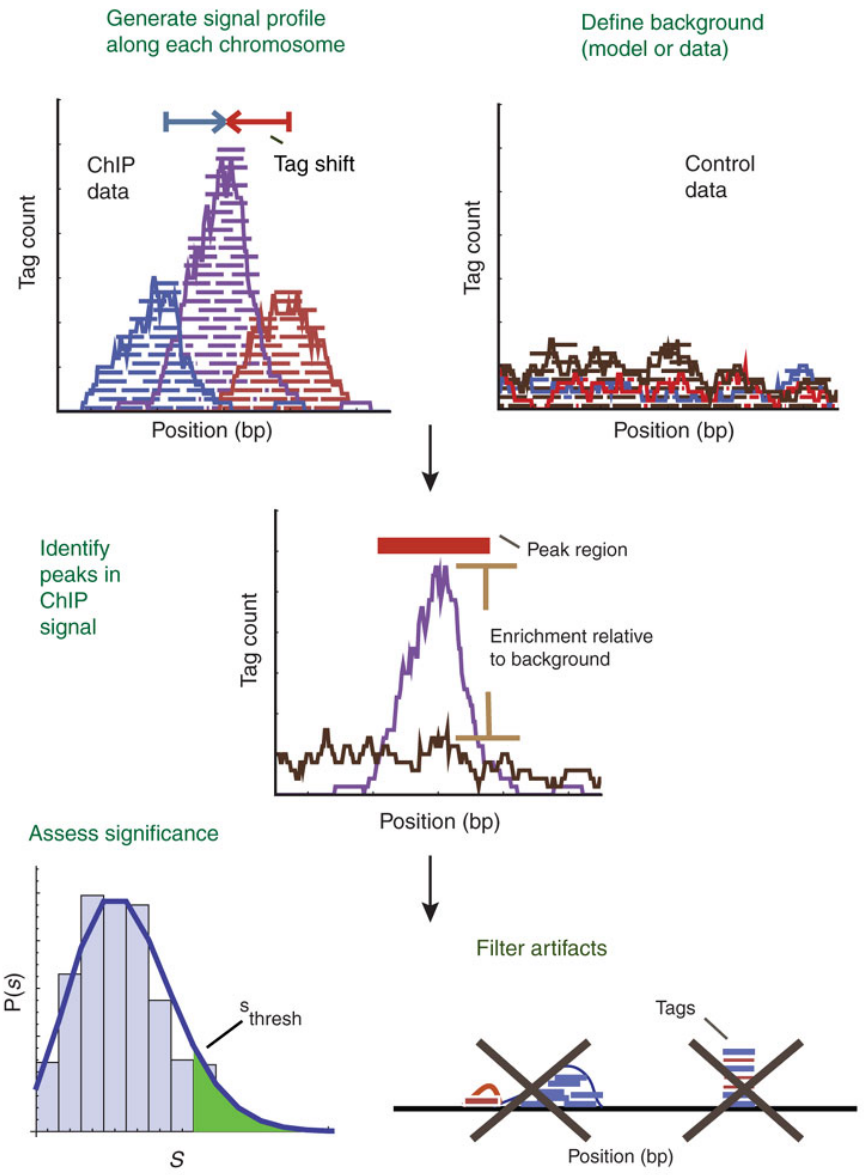
Park et al. 2009, Nature Review Genetics

Peak calling: peak identification

If no control sample is available, generally a random genomic background distribution (usually negative-binomial, Poisson, or Gamma) is assumed.

If a control sample is available, the peak calling algorithm will calculate the enrichment of the ChIP relative to the background and the regions of enrichment are identified.

Finally, peaks are filtered to reduce false positives and ranked according to relative strength or statistical significance. A distribution fit is used to indicate the cutoff above which a ChIP-seq peak might be considered significant.



Pepke et al. 2009, Nature Methods

Peak calling: peak finders

Several peak finders are freely available:

- MACS2
- QuEST
- FindPeaks
- PeakSeq
- BayesPeak
- SISSR
- F-Seq
- CisGenome
- SICER
- SPP
- …

Enriched regions (or “peaks”) are reported with a significance score (usually fold-change, p-value or q-value), whereas the FDR is either estimated during the peak calling, or calculated a posteriori (e.g. with an empirical FDR computed as the ratio between the number of peaks found in the control and the total number of identified peaks with the same parameters).

Certain software have been specifically designed for analysis of histone modifications (e.g. ChIPDiff, ChromaSig, MACS2 using the `--broad` option).

Criteria for assessing the experiment quality

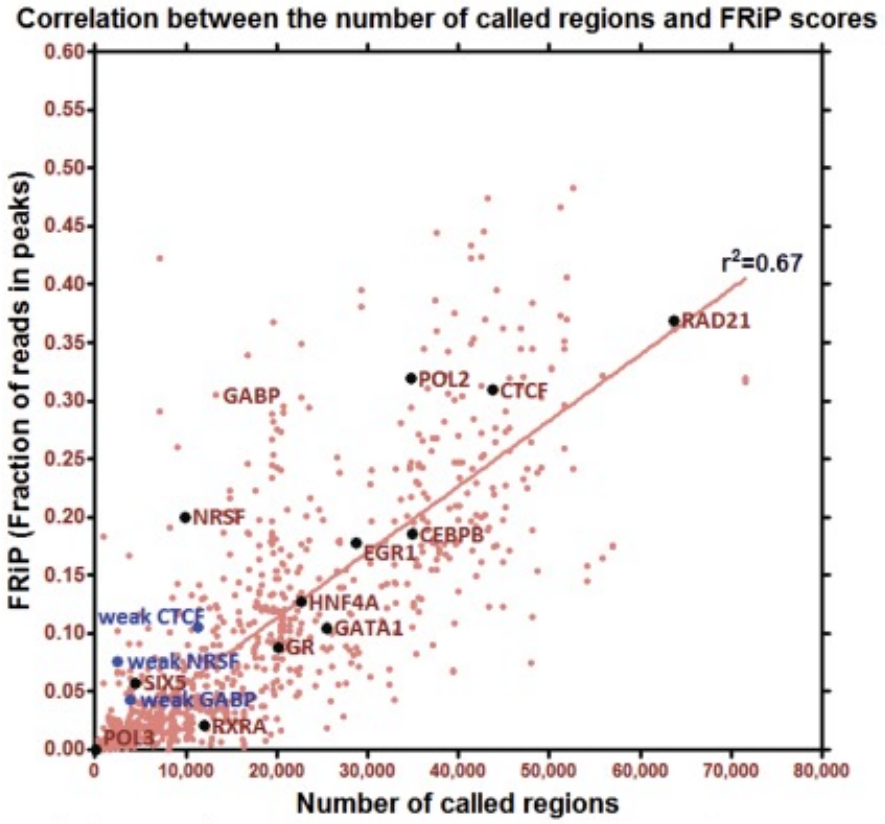
Criteria for assessing the experiment quality: FRiP

It is often useful to check the **Fraction of Reads in Peaks (FRiP)**, i.e. the fraction of all mapped reads that fall into peak regions identified by the peak calling algorithm. In general, only a minority of the ChIP-Seq reads occur in significantly enriched genomic regions, whereas the remainder represents background.

FRiP values correlate positively and linearly with the number of called regions.

Most of the ENCODE data sets have a FRiP enrichment of 1% or more using MACS. However, passing this threshold does not automatically mean that the experiment was successful, and a FRiP below the cutoff does not automatically mean a failure.

The best use of FRiP is to compare results obtained with the same antibody across cell lines or with different antibodies against the same factor (provided that peaks were called with the same algorithm and parameter set!).



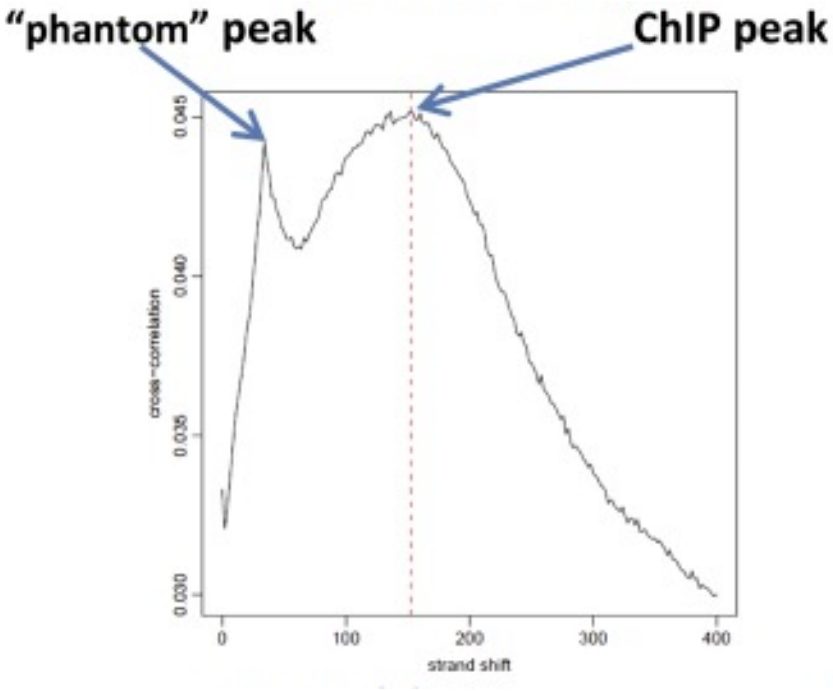
Landt et al. 2012 Genome Research

Criteria for assessing the experiment quality: cross-correlation analysis

Strand cross-correlation is a quality metric – independent of peak calling – which is based on the fact that high-quality ChIP-Seq data show a significant enrichment of reads at locations bound by the protein of interest and that forward and reverse strand read densities are centred around the binding sites.

The correlation between genome-wide read densities is computed as the Pearson linear correlation between the forward and the reverse strand read density profiles, after shifting the reverse strand reads by *k* base pairs.

This typically produces two peaks when cross-correlation is plotted against the shift value: a peak corresponding to the predominant fragment length and a peak corresponding to the read length (“phantom” peak).

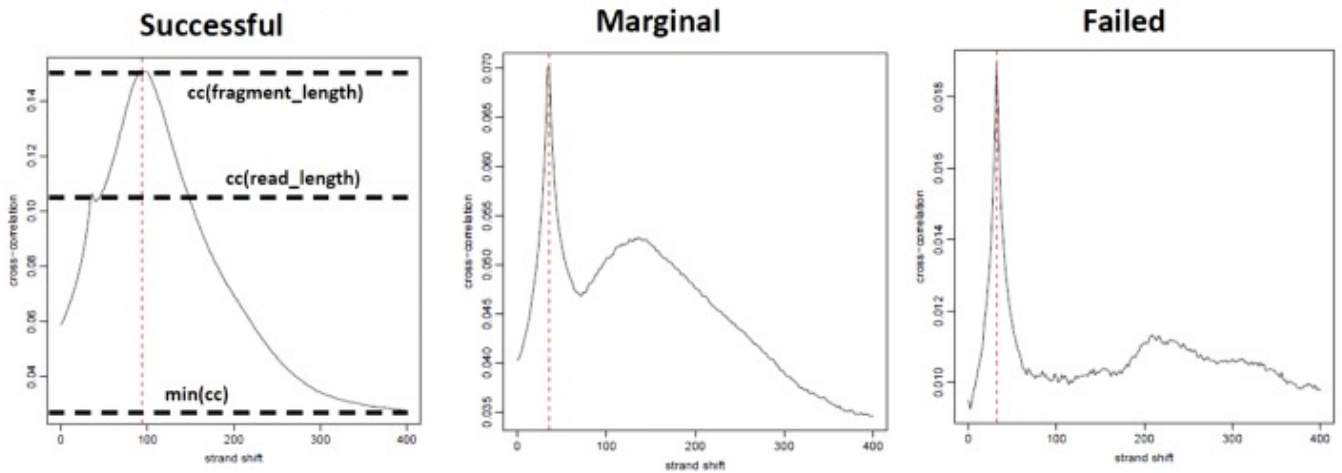


Landt et al. 2012 Genome Research

Criteria for assessing the experiment quality: NSC and RSC

This approach is used by two metrics for assessing the signal-to-noise ratio in a ChIP-Seq experiment:

- **Normalised Strand Coefficient (NSC)** → normalised ratio between the fragment-length cross-correlation peak and the background cross-correlation
- **Relative Strand Correlation (RSC)** → ratio between the fragment-length peak and the read-length peak



$$NSC = \frac{cc(fragment\ length)}{min(cc)} \qquad RSC = \frac{cc(fragment\ length) - min(cc)}{cc(read\ length) - min(cc)}$$

Landt et al. 2012 Genome Research

High-quality ChIP-seq data sets tend to have a larger fragment-length peak compared with the read-length peak, whereas failed ones and inputs have little or no such peak.

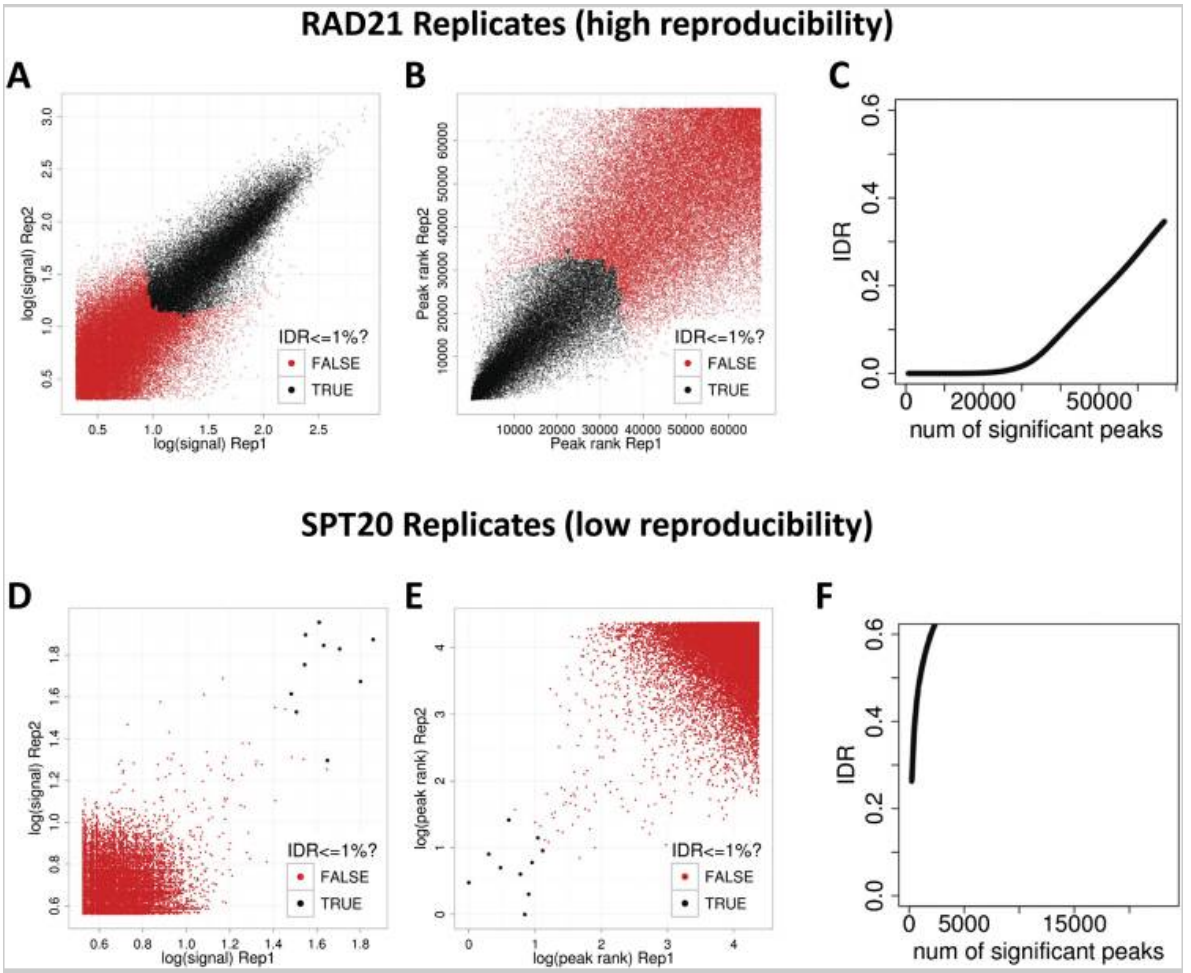
ENCODE guidelines recommend to repeat replicates when NSC <1.05 and RSC <0.8 .

Criteria for assessing the experiment quality: IDR analysis

The ENCODE guidelines recommend to perform experiments at least twice to ensure reproducibility; if the standard is not reached, a third experiment must be performed.

Highly reproducible peaks can be determined by the **Irreproducibility Discovery Rate (IDR)**, typically using a 1% threshold.

NOTE: ENCODE developers do NOT recommend using as it is for broad chromatin marks ChIP-seq!



(A,D) Scatter plots of signal scores of peaks that overlap in each pair of replicates.

(B,E) Scatter plots of ranks of peaks that overlap in each pair of replicates. Note that low ranks correspond to high signal and vice versa.

(C,F) The estimated IDR as a function of different rank thresholds.

Landt et al. 2012 Genome Research

Differential binding analysis

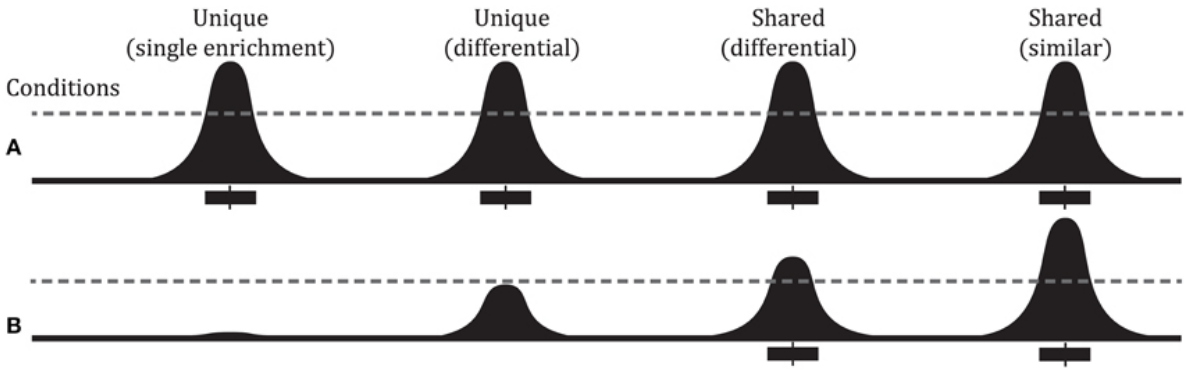
Differential binding analysis: motivation

Aim: identify changes in protein binding between different treatment conditions.

There are different ways to perform **pairwise comparisons** between ChIP-seq experiments to identify differential binding sites:

- overlap peak regions between conditions and classify peaks as unique or shared (***simplest approach***; *severely biased if samples were sequenced at different depths, as the number of peaks depends on that*)
- quantitatively compare conditions in terms of the number of reads overlapping a peak (***much better!***)

In fact, there are varying degrees of binding between a pair of conditions:



Wu et al. 2015, Front. Genet.

Differential binding analysis: software

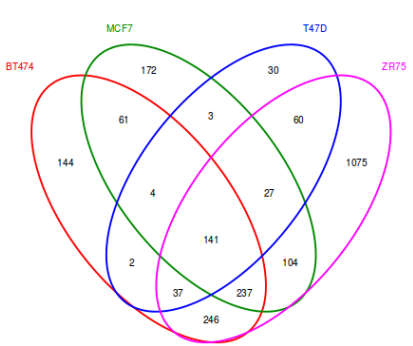
Commonly-used tools for this type of analysis are “DiffBind”, “DBChIP”, “PePr” and “csaw”.

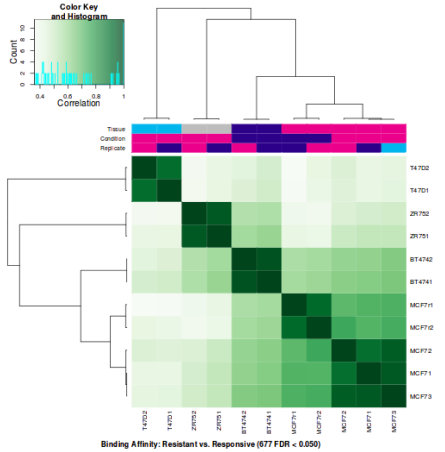
The usual steps of the analysis include:

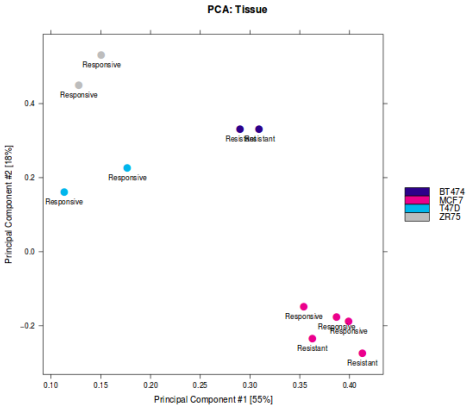
1) **defining a consensus set of peaks** to count over
2) **creating a binding counts matrix** (reads overlapping each peak per sample); with this matrix, the samples can be re-clustered using affinity, rather than occupancy, data
3) using RNA-seq methods (e.g. edgeR, DESeq2) with peak counts to **identify peaks with significant differential binding**

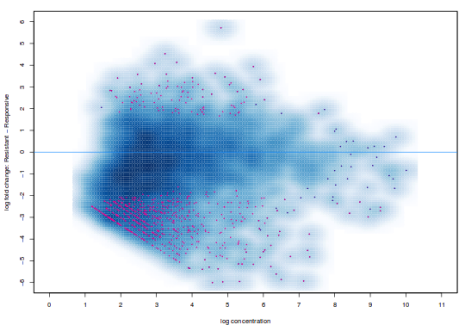
The advantage of these tools is to identify statistically significantly differentially bound sites based on evidence of binding **affinity** (measured by differences in read densities).

Additionally, packages like DiffBind provide a set of plots, including the results of PCA and hierarchical clustering.







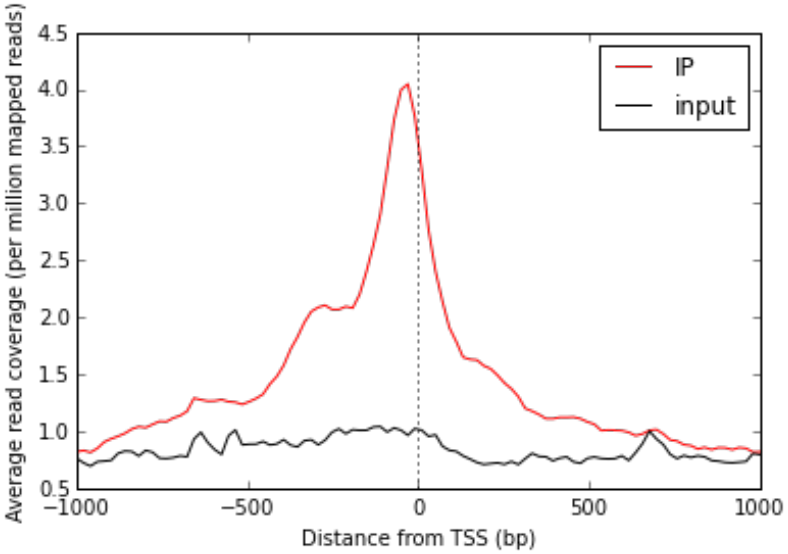


Downstream analysis

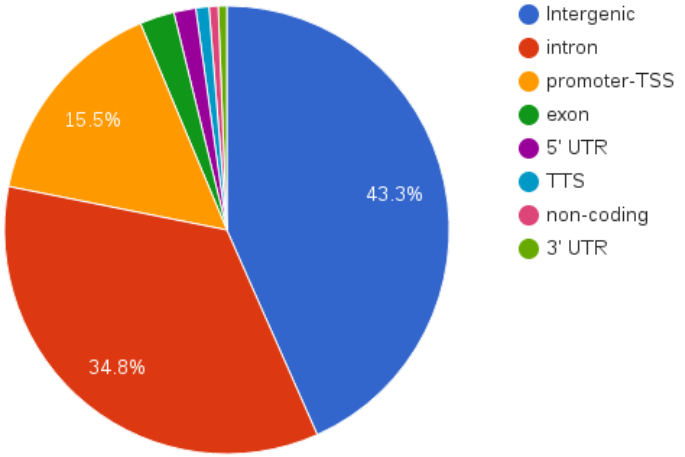
Downstream analysis: gene annotation and enrichment

A purely descriptive analysis is to check the distribution of peaks with respect to genomic features:

- **“protein-centric” approach**: centre binding sites and associate them with known genomic features around them to identify preferential targets of the protein (e.g. highly conserved regions, exonic regions, TSSs, etc.)



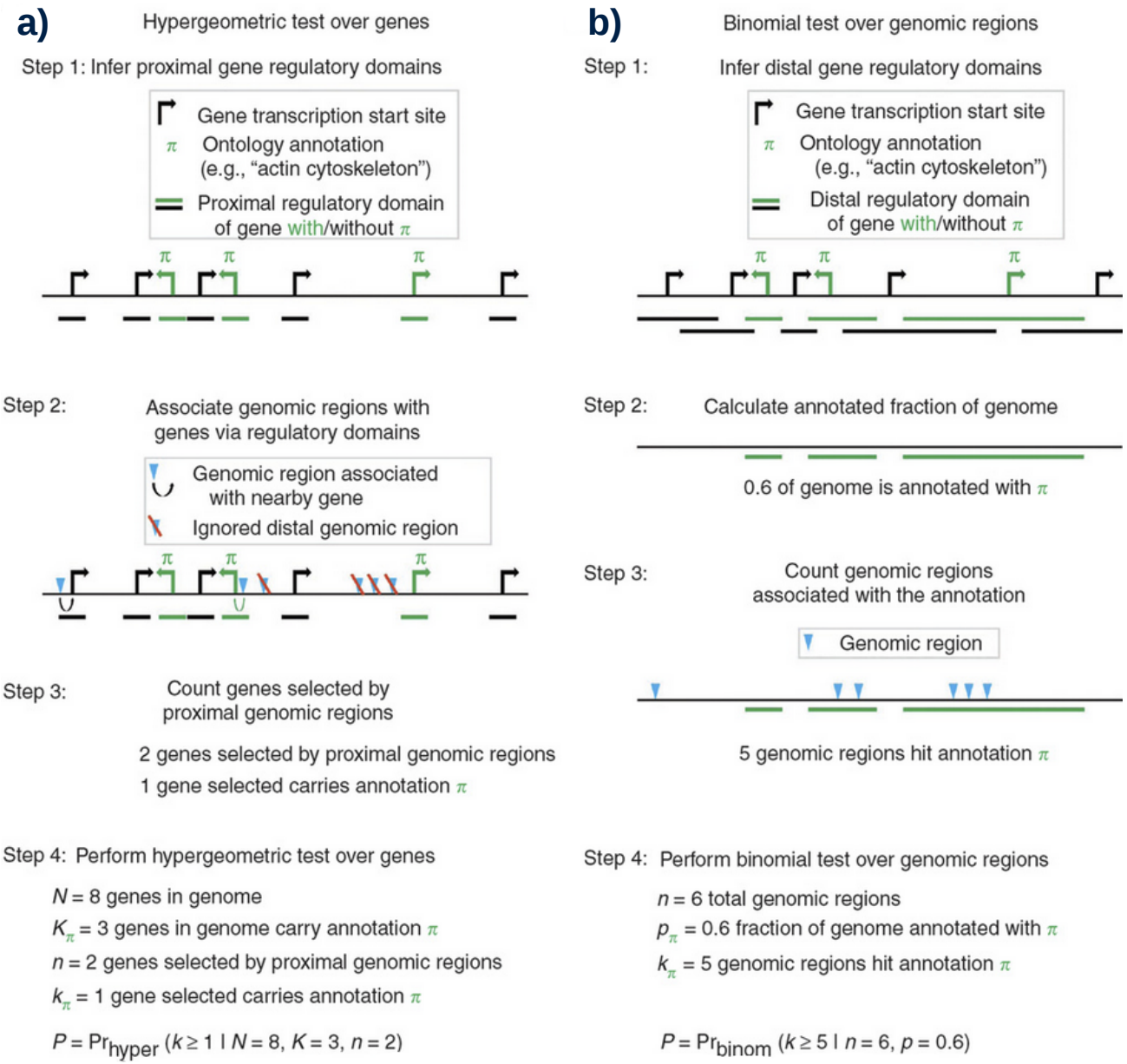
- **“feature-centric” approach**: classify binding profiles by their location with respect to pre-selected locations on the genome (e.g. promoters, introns, etc. of a set of genes of interest), or independently identified binding sites of a partner protein



Downstream analysis: gene annotation and enrichment

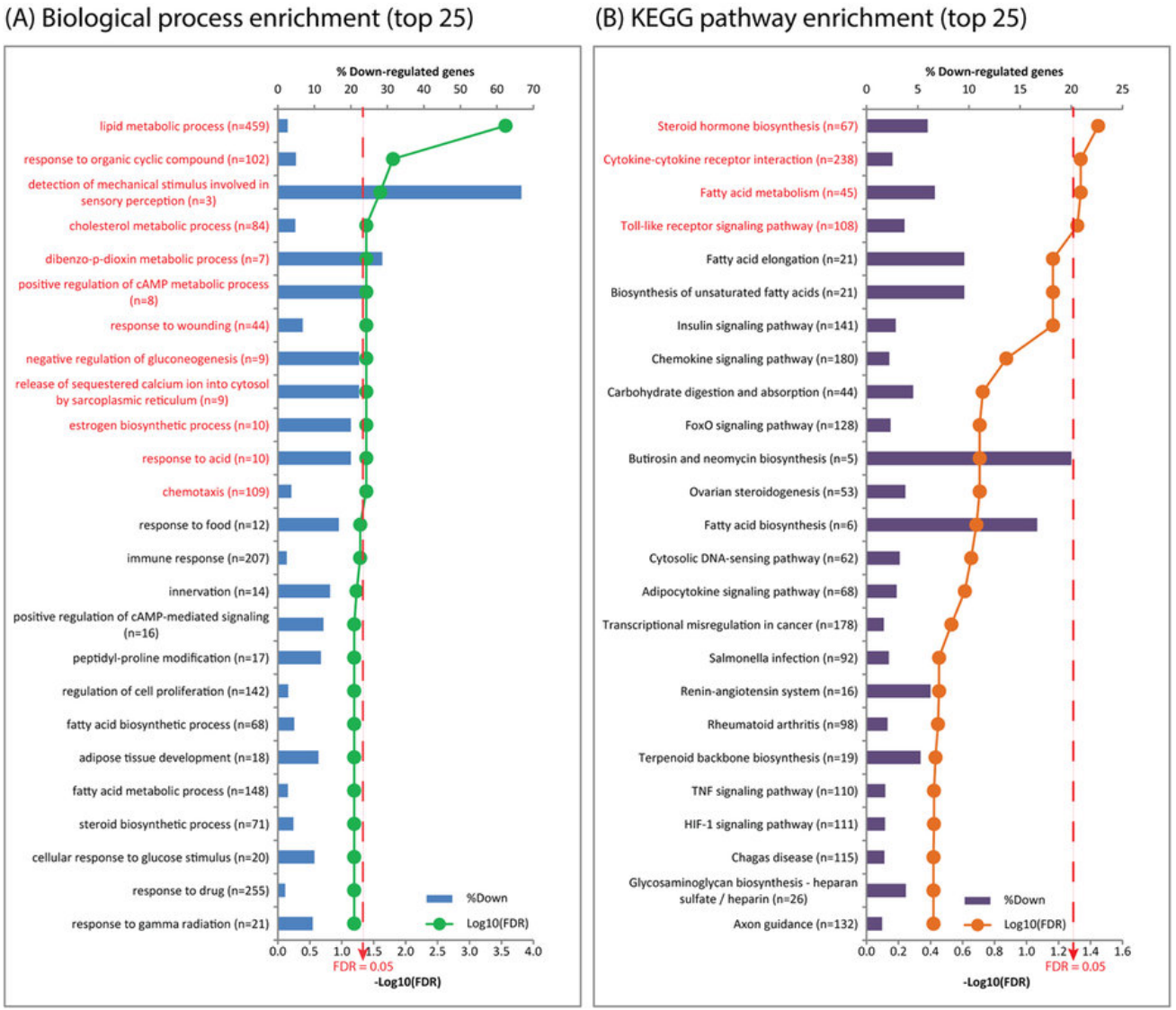
a) The most common approach used to annotate peaks approach is **proximity** (i.e. to assign each peak to the closest gene/transcript), although this ignores long-distance protein-DNA interactions!

b) Other tools (e.g. GREAT) try to take into account **distal binding events** by calculating the statistical enrichment of peaks associated with a given ontology term (using as background the percentage of genome annotated with that term and the total number of peaks).



Downstream analysis: gene annotation and enrichment

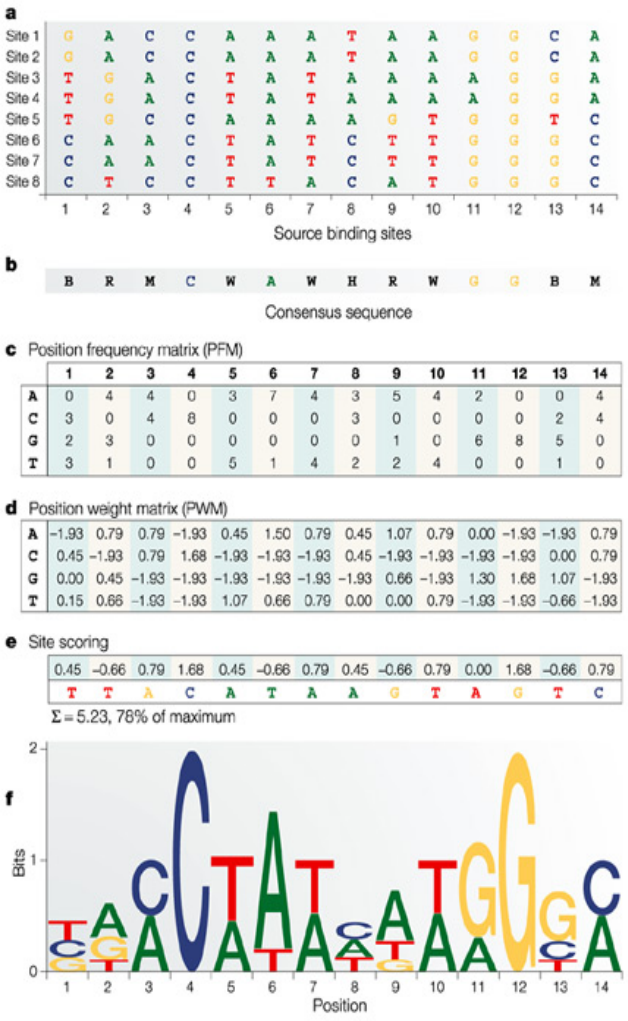
An example of **Gene Ontology** and **KEGG pathways** enrichment analysis output:



Miao et al. 2015 Scientific Reports

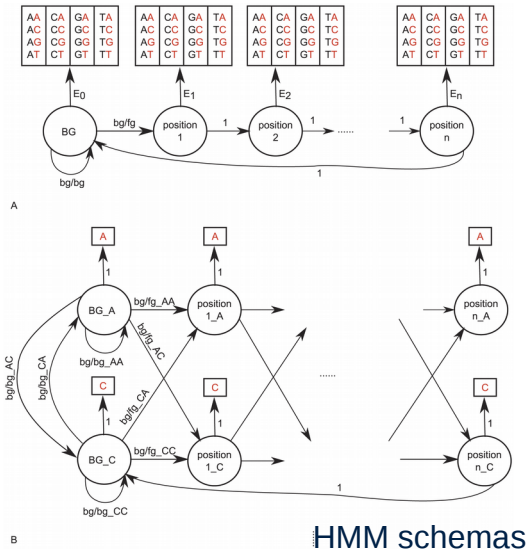
Downstream analysis: motif search

The most commonly used mathematical model to describe the binding affinity of a protein to the DNA is a **position weight matrix (PWM)**:

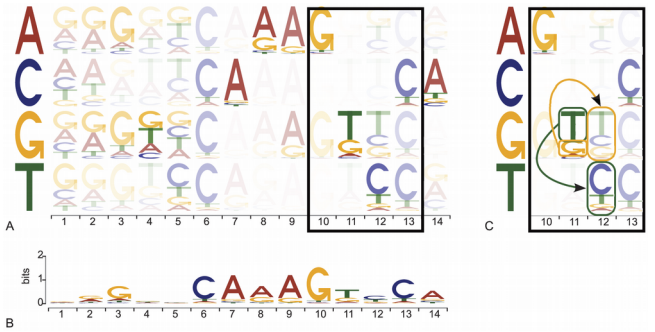


Wasserman & Sandelin, 2004 Nature Rev. Genet.

Alternatively, a **transcription factor flexible model (TFFM)** can be used to take into account dependencies between nucleotide positions:



HMM schemas



Mathelier & Wasserman, 2013 Plos Comp Biol

Downstream analysis: motif search

Two types of motif analysis:

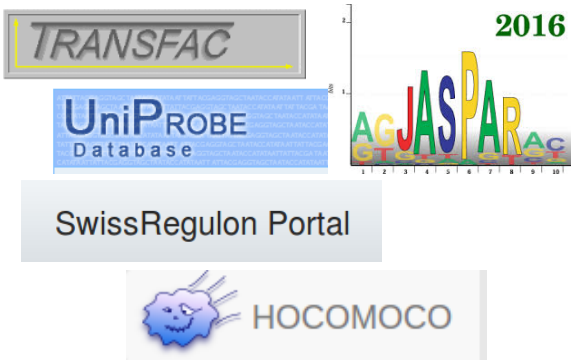
- **“known” motif scan**: when the PWM for one or more motifs are available
- **“*de novo*” motif discovery**: i) when the actual binding site is not known (e.g. newly discovered protein), or ii) your goal is to identify co-operative binding, or iii) you want to use ChIP-Seq data to refine the PWM of a known motif

Lots of programs available:

- FIMO
- MATCH
- MotifScanner
- MAST
- MSCAN
- TFBSTools (Bioconductor)
- …

} known motif scan

← User-defined PWMs or databases (TRANSFAC, JASPAR, HOCOMOCO, UniPROBE, SwissRegulon, etc.)



- Meme and Meme-ChIP
- CisFinder
- Gibbs Motif Sampler
- PhyME
- PhyloGibbs
- ChIPMunk
- Weeder
- Mprofiler
- …

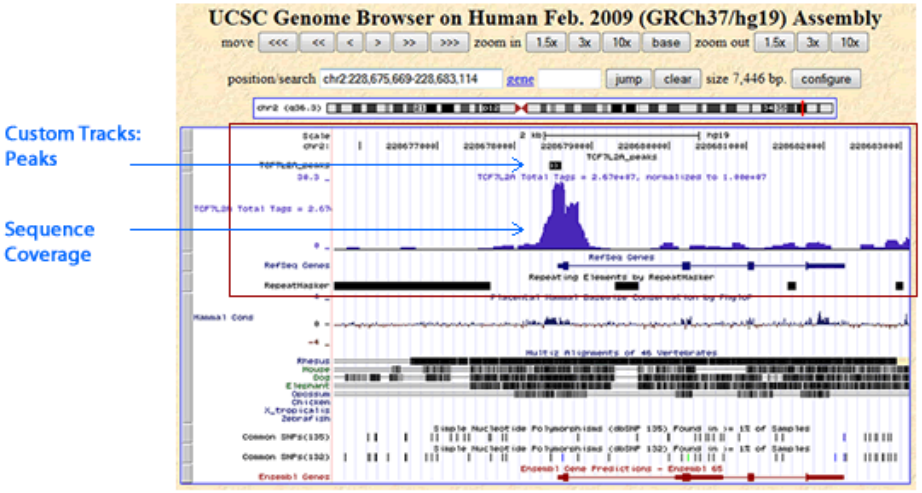
} *de novo* motif discovery

Visualization tools and formats

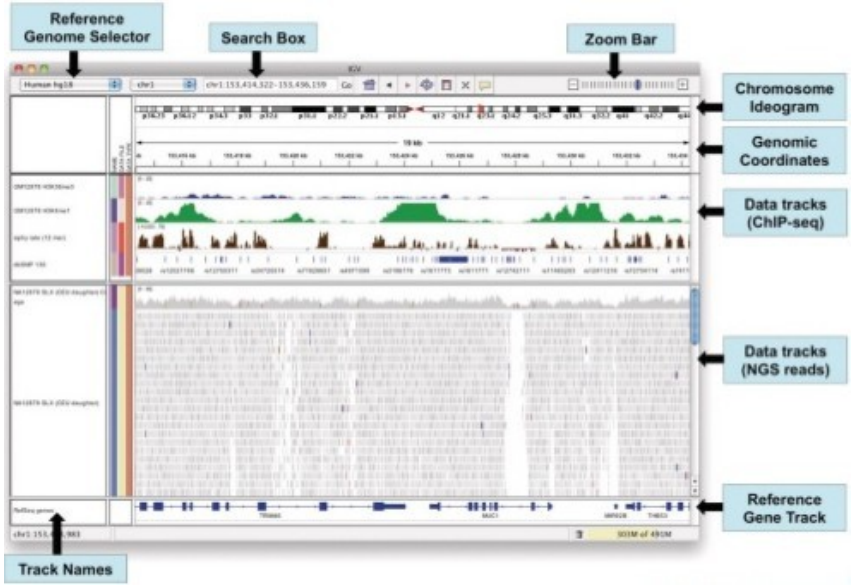
Visualisation tools and formats

Visualisation tools could help to get a first impression of how the experiment has worked, as well as to look closer at the overlap of specific peaks with other ChIP-Seq or annotation tracks.

There are several genome browsers publicly available to visualise ChIP-Seq data. The two most commonly used ones are:



UCSC Genome Browser



Integrative Genome Viewer (IGV)

Hosts lots of annotations and tracks (e.g. ENCODE)

Can be used both online and locally

Easier user interface

More customisable (read strands, read pairs, etc.)

Runs locally or using files located on DAS servers

Visualization tools and formats

Apart from SAM and its binary version BAM, other formats can be used to visualise ChIP-Seq raw and processed data on a genome browser:

- **BED**: tab-separated columns; 3 are mandatory (chrom, start, end), the others are optional

```
chr1    125872201    125872506
chr1    125893004    125893307
...
```

- **WIG**: to display continuous-values data, either spaced by a fixed or variable step

```
fixedStep chrom=chr3 start=400601 step=100
11
22
...
```

- **bigBed**: to store annotation items. BigBed files are created initially from BED type files, using the program “bedToBigBed” and are in an indexed binary format. The main advantage of the bigBed files is that only the portions of the files needed to display a particular region are transferred to the browser, so for large data sets bigBed is considerably faster than regular BED files.

- **bigWig**: to display dense, continuous data as a graph. BigWig files are created from WIG type files, using the program “wigToBigWig” and are in an indexed binary format. The main advantage is that only the portions of the files needed to display a particular region are transferred to the browser, so for large data sets bigWig is considerably faster than regular WIG files.

Questions?

silvia@well.ox.ac.uk