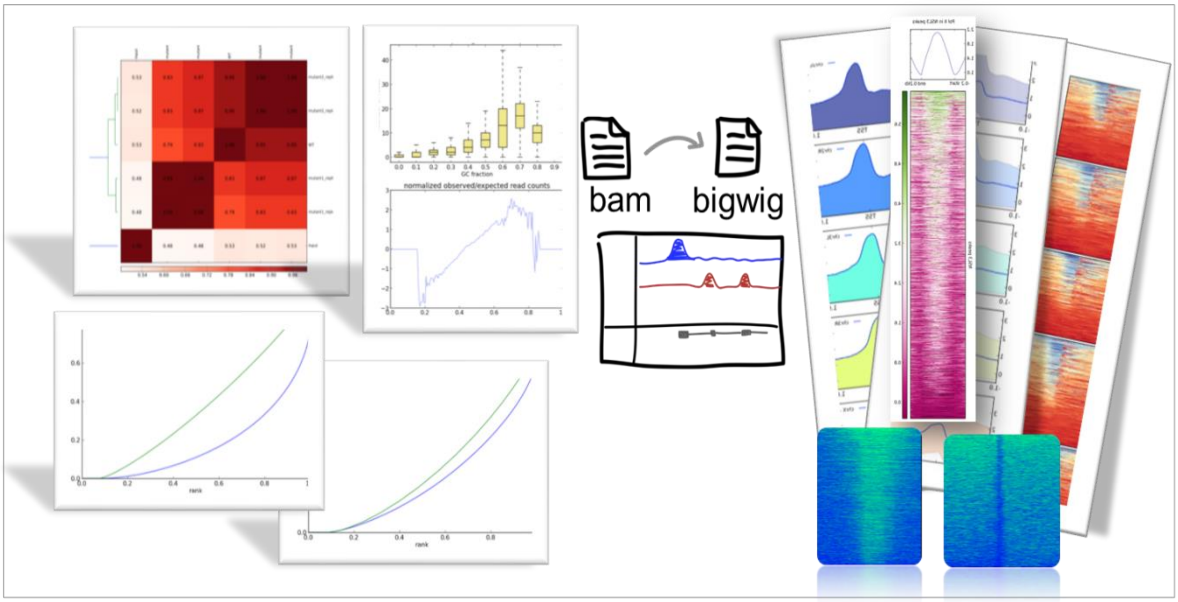
MAX PLANCK INSTITUTE
OF IMMUNOBIOLOGY AND EPIGENETICS

Analyze more, process less

Visualizing and interpreting genome-wide sequencing data using deepTools

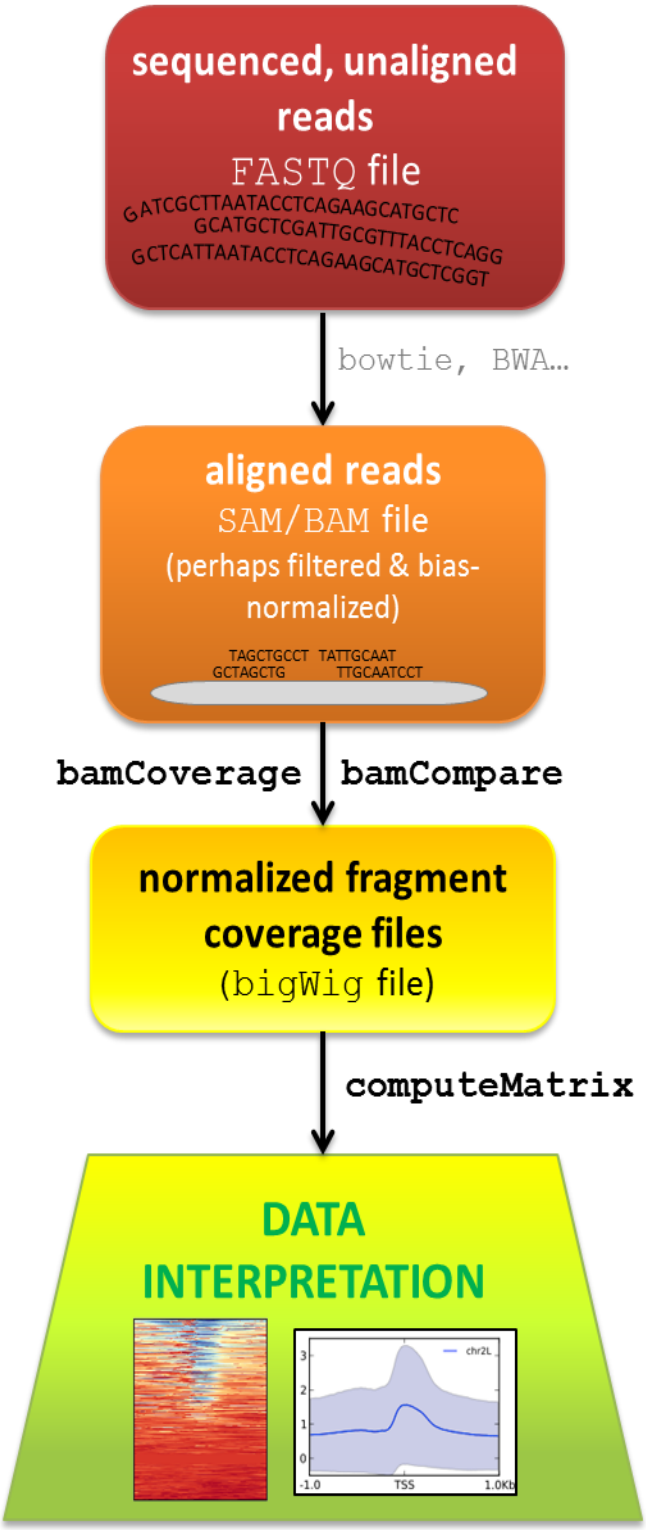


Fidel Ramírez, Friederike Dündar, Sarah Diehl, Björn A. Grüning, and Thomas Manke

November 2013

deeptools.ie-freiburg.mpg.de

Data processing workflow



sequenced, unaligned reads
FASTQ file

GATCGCTTAATACCTCAGAAGCATGCTC
GCATGCTCGATTGCGTTTACCTCAGG
GCTCATTAATACCTCAGAAGCATGCTCGGT

bowtie, BWA…

aligned reads
SAM/BAM file
(perhaps filtered & bias-normalized)

TAGCTGCCT TATTGCAAT
GCTAGCTG TTGCAATCCT

bamCoverage | bamCompare

normalized fragment coverage files
(bigWig file)

computeMatrix

DATA INTERPRETATION

chr2L
3
2
1
0
-1.0
TSS
1.0Kb

Typical analyses of high-throughput sequencing data usually begin with one or more FASTQ file(s) of deeply-sequenced samples (see the next slide for a glossary of file formats).

After a first quality control using FASTQC, the reads are aligned to the reference genome, e.g. using Bowtie (PMID: 22388286) or BWA (PMID: 20080505). deepTools can then be used to assess the quality of the aligned reads with **bamCorrelate, bamFingerprint and computeGCbias.**

Following the quality checks, most read-related information is not required for subsequent analyses. These are instead based on the coverage values along the genome.
The deepTools modules **bamCompare and bamCoverage** calculate those read coverages that will be stored in bigWig (or bedGraph) format. These files are very useful for data sharing, storage, display in Genome Browsers and efficient down-stream analyses. The tools offer multiple parameters to normalize for sequencing depth, background reads and GC bias so that different samples can be faithfully compared to each other.

Once we are satisfied by the quality checks , we use the coverage files to generate heatmaps and average profiles, analyzing and interpreting the processed data.

Overview of deepTools modules

tool name	type	output files	application
bamCorrelate	QC	clustered heatmap	calculate the correlation between read coverages
bamFingerprint	QC	xy-plot	assess the enrichment strength of a ChIP sample
computeGCBias	QC	box plot, xy-plot	calculate expected and observed GC distribution of reads
correctGCBias	norm.	aligned reads	obtain GC-corrected read file
bamCoverage	norm.	continuous profile	obtain normalized read coverage for a single sample
bamCompare	norm.	continuous profile	normalize 2 BAM files to each other with a mathematical operation of choice (fold change, log2(ratio), sum, difference)
profiler	visual.	xy-plot (``meta-profile")	generate average profiles of read coverage for genome regions
heatmapper	visual.	unclustered heatmap	display individual read coverages for genome regions of interest

The individual tools take care of the different workflow phases.

Every module can be used completely independent of the others, i.e. if a user already has downloaded a bigWig file, this can directly be used to plot heatmaps and average plots.

All tools can be used to export the data matrix underlying any figure.

Glossary of file formats

name	explanation
BAM	• compressed, binary file format (complement to SAM), not “human-readable” • the common output file format of the most popular read aligners such as bowtie2 (Langmead & Salzberg (2012) Nat Methods) • each line corresponds to one mapped read with many additional information, e.g. about its mapping quality, its sequence, its location in the genome etc. • highly recommended format for storing raw data
BED	• text file • used to store genomic intervals, e.g. genes, peak regions etc. • for `deepTools`, the first 3 columns are important: chromosome, start position of the region, end position of the genome
bedGraph	• text file • similar to a bed file, except that it is limited to 4 columns and 4th column must be a numeric value, e.g. a coverage score
bigwig	• binary version of a bedGraph file • usually contains 4 columns: chromosome, start of genomic bin, end of genomic bin, score • the score can be anything, e.g. an average read coverage
FASTA	• text file • commonly used to store DNA or protein sequences
FASTQ	• text file • common output file format of Illumina sequencers • contains raw read information (e.g. base calls, sequencing quality measures etc.), but no information about where in the genome the read originated from
SAM	• text file • same (uncompressed) content as BAM file
2bit	• compressed file format for DNA sequences

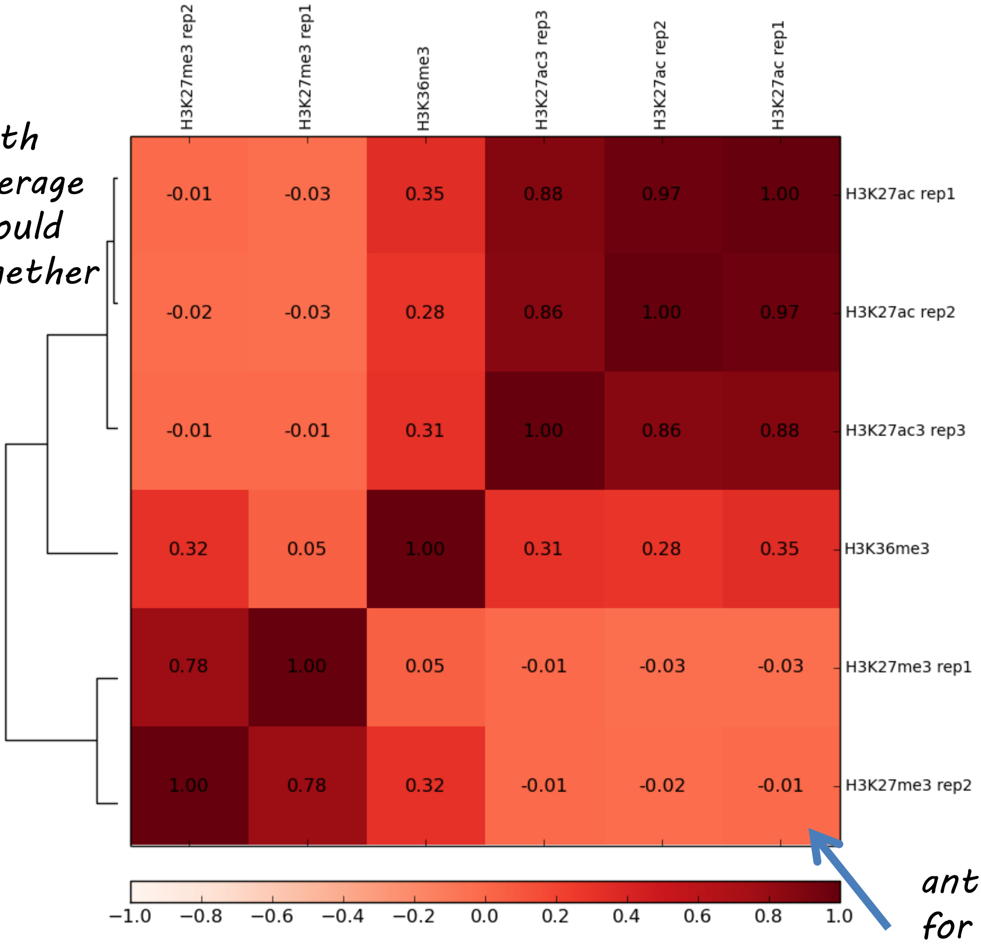
1. Visualization of data quality

diagnostic plots of aligned reads

Basic correlation of samples

- this should be the starting point of any analysis
- results of the correlation analysis can:
  - identify sample swaps
  - raise awareness for possible biases
  - be useful to assess the similarity of replicates, similarity with published data etc.

samples with similar coverage profiles should cluster together



	H3K27me3 rep2	H3K27me3 rep1	H3K36me3	H3K27ac3 rep3	H3K27ac rep2	H3K27ac rep1
H3K27ac rep1	-0.01	-0.03	0.35	0.88	0.97	1.00
H3K27ac rep2	-0.02	-0.03	0.28	0.86	1.00	0.97
H3K27ac3 rep3	-0.01	-0.01	0.31	1.00	0.86	0.88
H3K36me3	0.32	0.05	1.00	0.31	0.28	0.35
H3K27me3 rep1	0.78	1.00	0.05	-0.01	-0.03	-0.03
H3K27me3 rep2	1.00	0.78	0.32	-0.01	-0.02	-0.01

high correlation for replicates

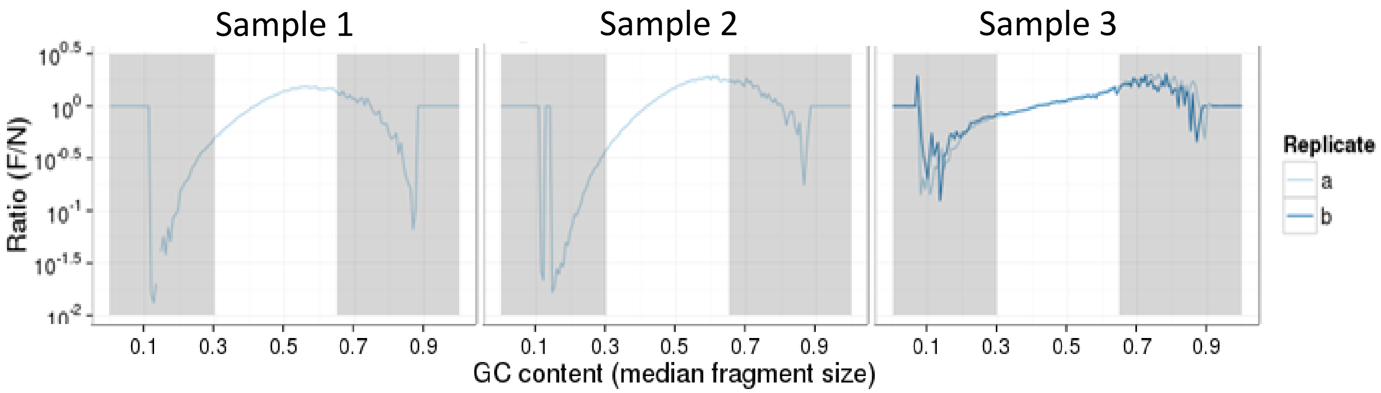
anti-correlation for repressive and active histone marks

- there is no limit on the number of files to be compared to each other
- Pearson or Spearman correlation can be computed

deepTools: bamCorrelate

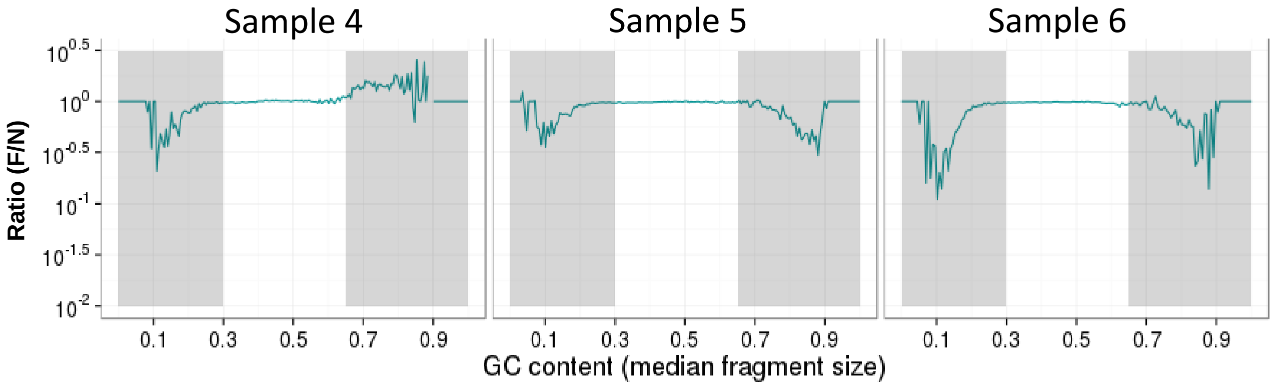
Check for GC bias

bad example



dramatic deviations from (obs/exp) = 1 → when Sample 1 should be compared with Sample 3, we strongly recommend to use deepTools to correct for GC bias

good example



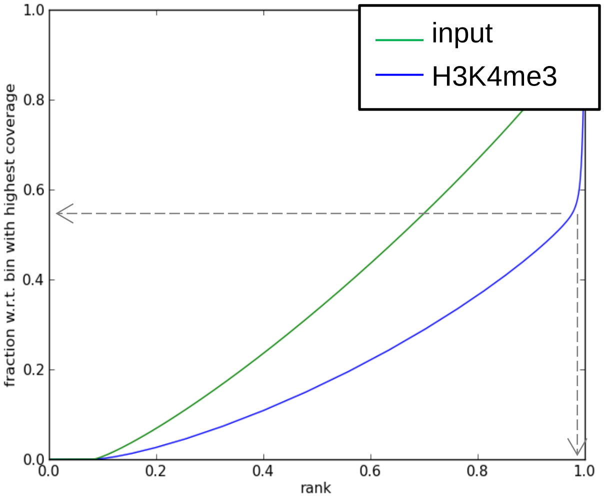
the majority of the genome is covered equally, regardless of the GC content → GC correction is not necessary

deepTools: computeGCbias (to calculate the bias)

deepTools: correctGCbias (to correct the bias)

Assessing ChIP strength

localized histone mark

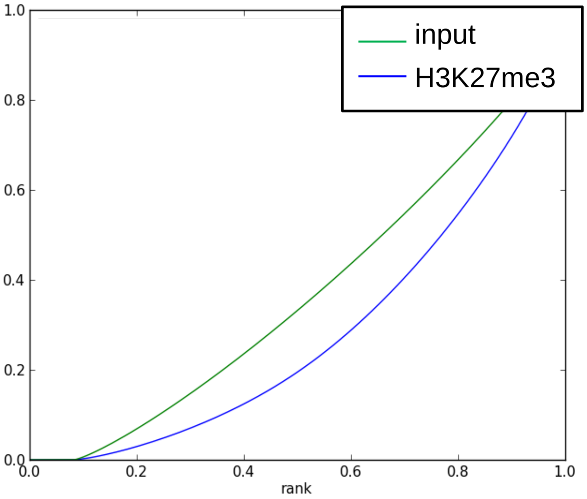


when counting the reads contained in 97% of all genomic bins, only 55% of the maximum number of reads are reached, i.e. 4% of the genome contain a very large fraction of reads

This plot is typical for narrow, strong enrichments – which indicates that the H3K4me3 profile matches the expectations.

Input and ChIP are very well separated, subsequent normalization via the SES can be applied (using bamCompare)

broad histone mark



compared to H3K4me3, input and ChIP cannot be distinguished as easily here

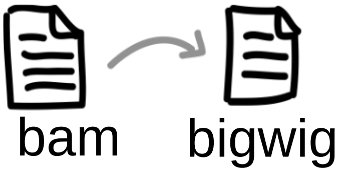
As H3K27me3 is a mark that yields broad domains instead of narrow peaks, this plot does not necessarily indicate a failure of the experiment, but it demonstrates why the SES method should not be used for normalization in this case

deepTools: bamFingerprint on input and ChIP sample

2. Visualizing and comparing different(ly) deeply-sequenced samples

how to generate normalized continuous signal profiles and use them in Genome Browsers, heatmaps and average plots

Generating signal profiles of individual samples



bam → bigwig

If the BAM file contains reads from paired-end sequencing, reads are extended to the exact fragment length. For matel-ess and single-end reads, the user must specify the average fragment length that was selected prior to deep-sequencing (usually 200 bp). In addition, the user decides about the size of the genome bins for which the fragment coverage should be determined (default is 50 bp; the smaller the bin size, the bigger the resulting file). bamCoverage first calculates all the number of fragments that overlap with each bin in the genome. Bins with zero counts are skipped, i.e. not added to the output file. The resulting read counts can be normalized using either a given scaling factor, the RPKM formula or to get a 1x depth of coverage (RPGC).

Name	Details
Reads per genomic content (RPGC)	This method will normalize a sample to 1x genome-wide coverage using the assumption: normalized bin count/1x coverage = real bin count / real coverage Therefore, the normalized bin count is calculated as follows: real bin count * genome size / genome-wide coverage
Reads per kilobase per million reads (RPKM)	This method is similar to the normalization used for RNA-seq data. The formula is as follows: number of reads per bin/(number of million mapped reads * bin length in kbp) The resulting numbers are usually very small.
Total read count normalization	When comparing two BAM files the simplest way to account for differences in sequencing depth is to divide the coverage by the total number of sequenced reads.
signal extraction (SES)	Based on a method proposed by {Diaz, 2012 #8}. Not recommended for broad marks or when bamFingerprint indicates that the ChIP and input sample have very similar read coverages.

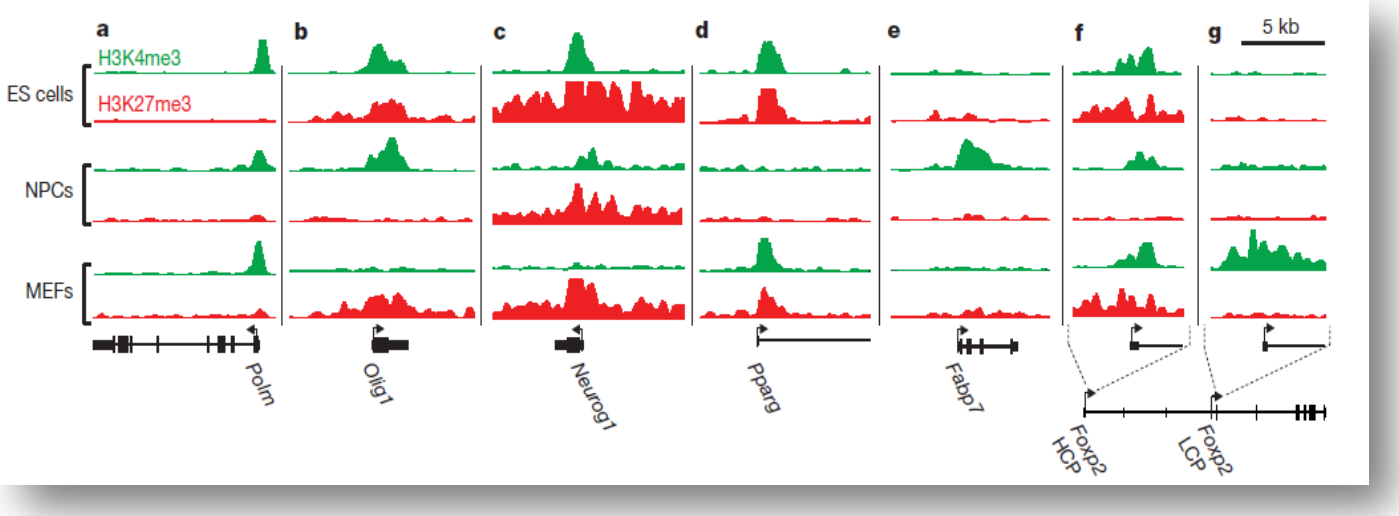
deepTools: bamCoverage with output format “bigwig”

Browsing the coverage profiles

We strongly recommend to spend considerable time with the visual inspection of **normalized** coverage profiles using a Genome Browser, e.g. IGV or the UCSC browser. As bigWig files are much smaller than BAM files, they can easily be uploaded.

The visual inspection should come before any other major down-stream analysis. It helps to “get a feeling for the data”, for example:

- identifying regions with extremely high or no coverage at all
- assess whether the distribution of the signal (broad vs. narrow enrichments) matches the expectation
- checking candidate regions where one expects (no) signal
- generate hypotheses regarding the pattern of the signal, e.g. enrichments at promoters or along gene bodies etc.



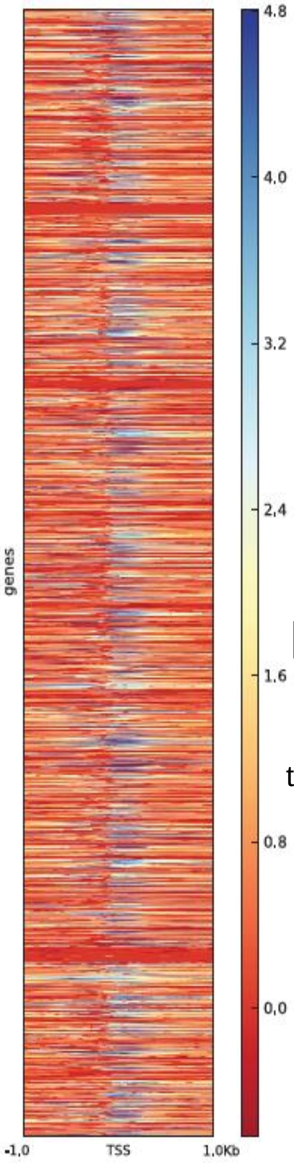
Famous example of Genome Browser screenshots illustrating the different combinations of the permissive H3K4me3 and the restrictive H3K27me3 marks in ES and differentiated cells. Figure taken from Mikkelsen et al. (2007) Nature.

deepTools: bamCoverage or bamCompare with output format “bigwig”, then use a external Genome Browser, e.g. from IGV or UCSC

Understanding signals within the genome I

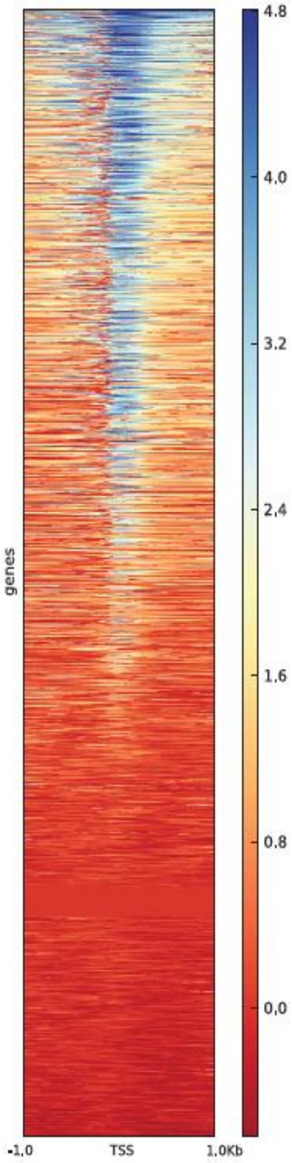
Example: Assessing the ChIP-seq signal of **RNA Polymerase II** (Pol II) at the transcription start site (TSS) of (Drosophila) genes

Heatmaps are very useful to get an overall feeling for the signal distribution.



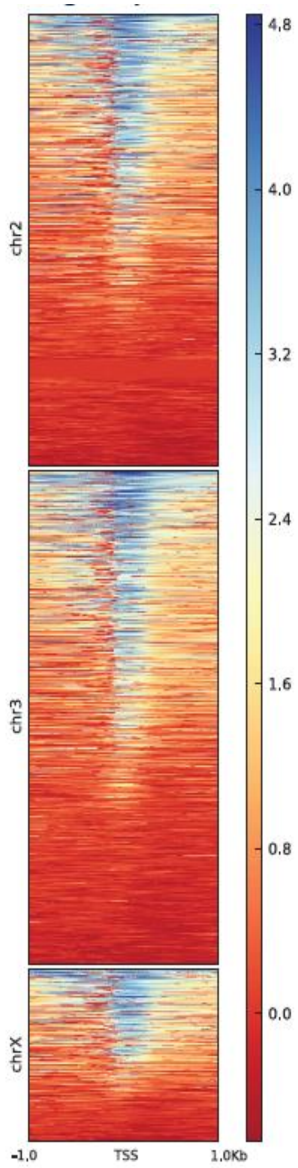
The unsorted heatmap displays the Pol II signals around the TSS for all genes. The strongest signals seem to surround the TSS.

sort TSS according to the Pol II signal



now it is clear, that ca. 1/3 of TSS have very high signals, ¼ have intermediate Pol II signals and 50 % have no signal

separate the genes into different groups based on their chromosome



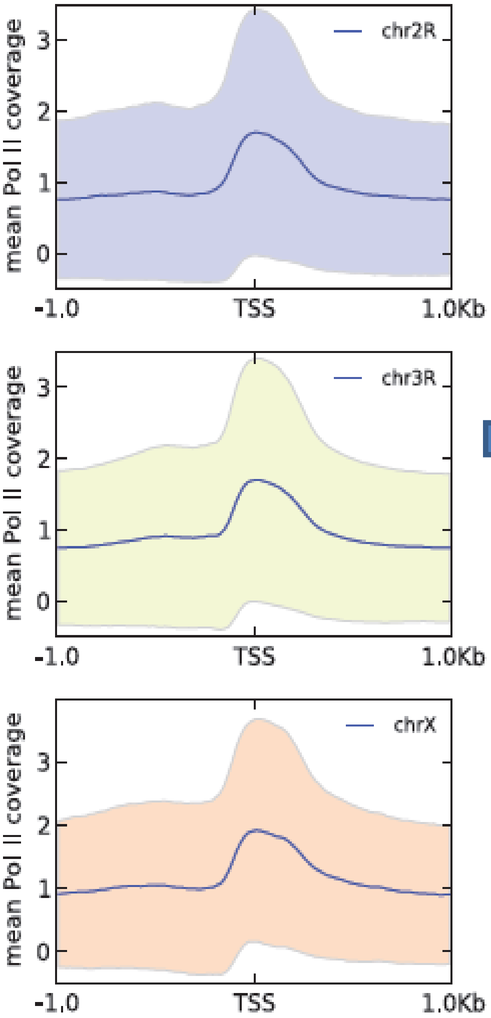
the numbers of TSS with strong and weak Pol II signal seem to be similar between the different chromosomes

deepTools: computeMatrix with "reference-point" and with or without sorting (advanced options), supplying either one file for all genes or three files for genes on each chromosome separately

deepTools: heatmapper with output file from computeMatrix

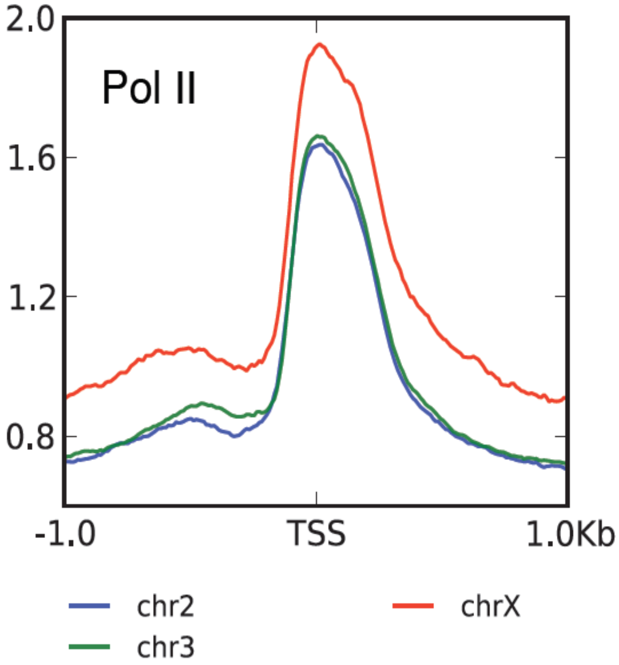
Understanding signals within the genome II

Summary plots (or average plots or “meta-gene” plots) summarize the heatmap findings.



plotted separately, the signals look very similar

note that the shaded region indicates the spread of the standard deviation



plotted within the same frame, there is an indication that Pol II might be more strongly enriched on the X chromosome than on autosomes (which is in line with current theories about the male X of fruit flies)

deepTools: computeMatrix with “reference-point”
deepTools: profiler with output from computeMatrix, choosing different plot types (advanced options)

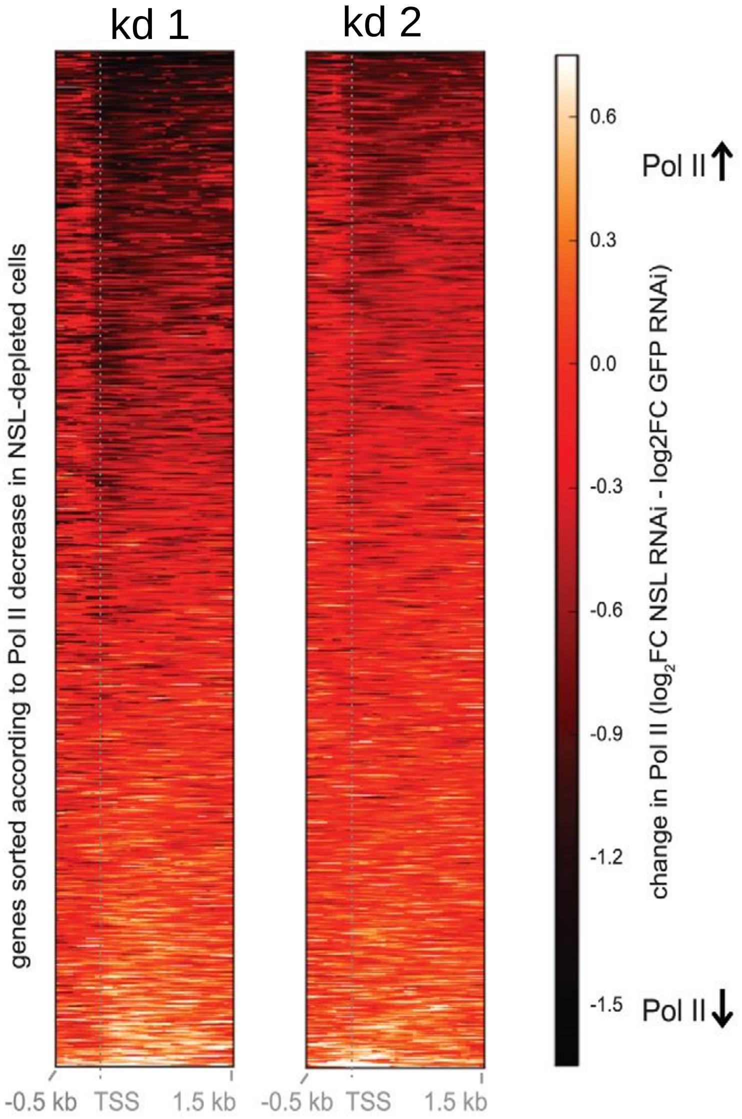
3. More examples of insightful heatmaps and average plots

Comparing signal **differences**

for example:

- ChIP vs. input
- wild type vs. knock-out
- day 1 vs. day 2

Example: **Difference in Pol II signal** in WT vs. 2 knock-down (kd) conditions (from PMID: 22723752)

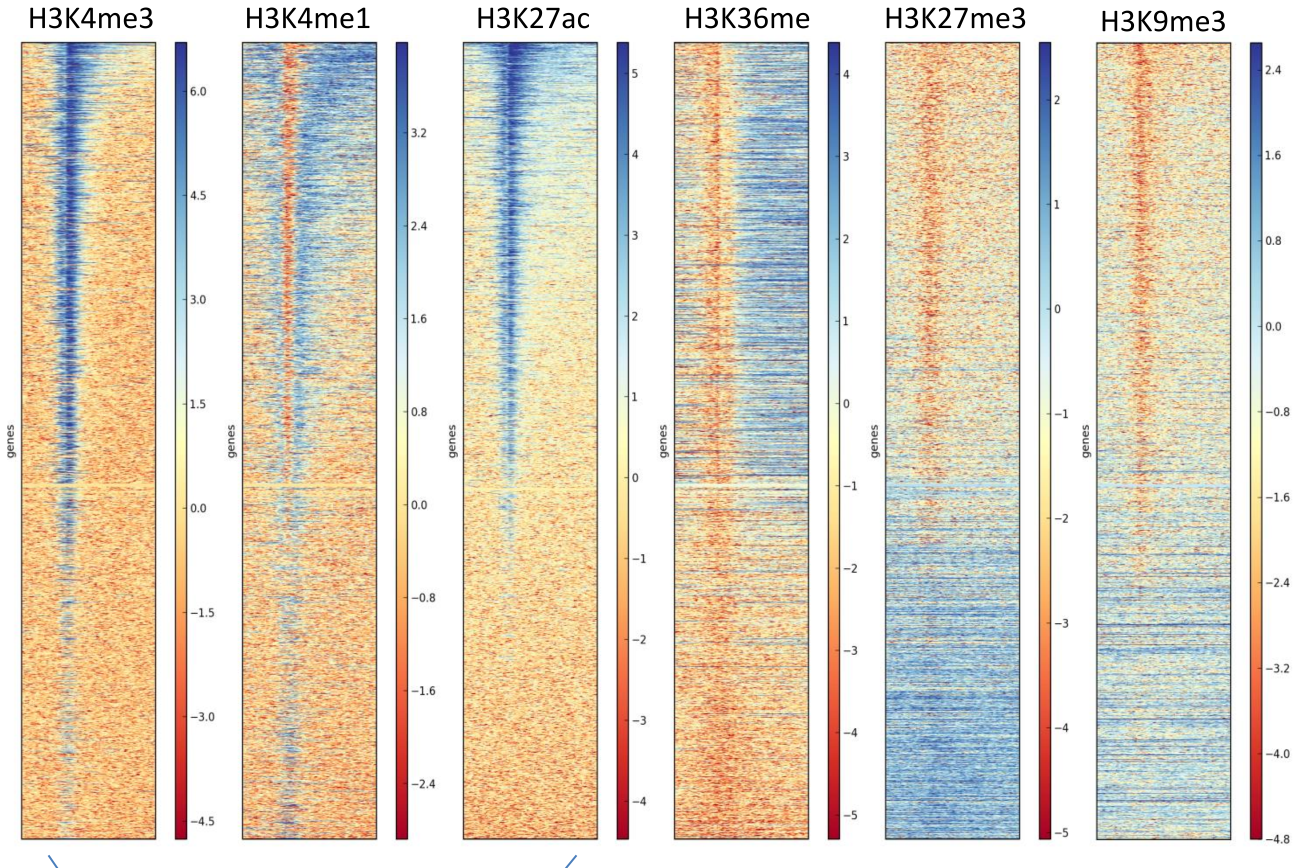


visible:
kd 1 leads to stronger changes than kd 2

deepTools: bamCompare with knock-out and wildtype sample, using the "difference" instead of default log2ratio option
deepTools: computeMatrix with reference-point, followed by **deepTools**: heatmapper

Distinct signals of histone marks around gene starts

- all genes are sorted according to H3K4me3 signal abundance
- clearly, H3K4me1, H3K27me3 and H3K9me3 are depleted (red) where H3K4me3 and H3K27ac are present (blue)



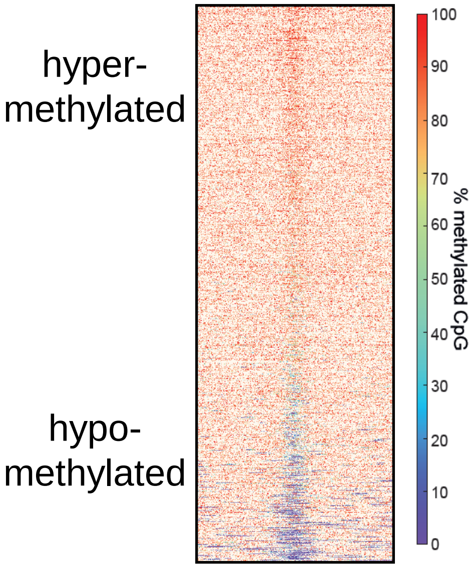
narrow enrichments

broad enrichments

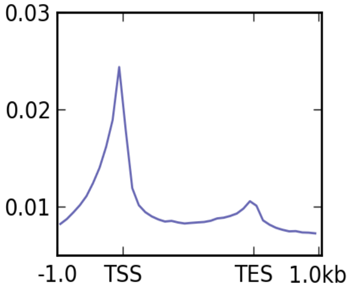
deepTools: bamCompare for each sample (ChIP vs. input)
deepTools: computeMatrix on normalized H3K4me3 bigWig file, with default sorting and saving the order of the regions to a BED file (advanded output options), this BED file is then used for **deepTools:** computeMatrix on all other normalized sample files choosing "no sorting"
finally, **deepTools**: heatmapper is run on all computeMatrix results without sorting

Any value contained in a bigWig file can be visualized

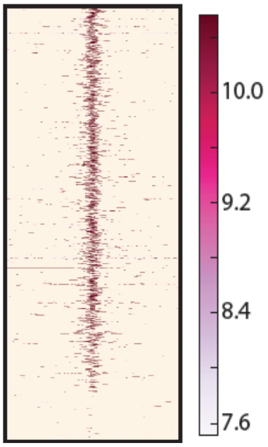
% GC methylation at arbitrary non-gene regions



% GC methylation along gene bodies

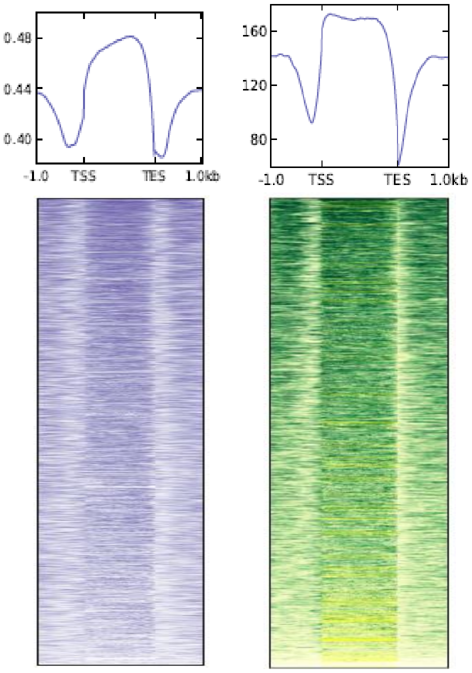


Motif presence



for the motif of interest, the chosen regions are highly enriched around their center points

GC content & raw reads



the raw read distribution of this sample largely reflects the GC content