

Formation of the Super-Structures on the Inactive X

E. Ilic

M. Vaiyapuri

San Jose State University

Introduction

Looking at mammals, the X chromosome inactivation balances the dosage of X-linked genes between male and female organisms. What this means is that during development, female embryos have one X chromosome permanently “disabled”. The genes on the inactive X chromosome are not expressed in the organism, meaning that females have one functional X chromosome, eliminating the imbalance of males having one X chromosome while females have two. This project is an investigation into how the Xi (inactive X) is shaped and how the genes on this chromosome are repressed. The goal is to see if the presence of more H3K27me3 and less H3K4me3 may explain the repression of the genes on the Xi chromosome. To identify the differences between the Xi and X chromosome, histones were targeted with these modifications and CTCF using ChIP-seq. This way, it could be seen if there were differences in the levels of H3K27me3 and H3K4me3 between active and inactive X chromosomes, and whether this could have influenced gene expression.

The given sequences in this experiment are taken from mice. These sequences correspond to a portion of DNA that is linked to histones with either H3K27me3, H3K4me3 or CTCF. There are four subsets of data, each with two replicates:

1. CTCF – wt_CTCF_rep1 and wt_CTCF_rep2
2. H3K4me3 – wt_H3K4me3_rep1 and wt_H3K4me3_rep2
3. H3K27me3 – wt_H3K27me3_rep1 and wt_H3K27me3_rep2
4. Input – wt_input_rep1 and wt_input_rep2

In this experiment, input represents the control samples, done with the same treatment except for the immunoprecipitation step.

Methods & Materials

The original guide for this project used Galaxy to analyze data. In order to make the sharing of data more friendly for a group setting, Google Cloud was used.

0 . Environment Setup

The first task to facilitate the analysis of data was to set up the Google Cloud environment. A bucket was created to store all the data resulting from the analysis. An n1-standard-2 virtual machine was set up to process data, and miniconda was installed and used to create a new project environment. Using bioconda, the appropriate tools were installed. These tools include:

1. Bedtools - a toolset used for genome arithmetic.
2. Bowtie2 - a tool for aligning reads to long reference sequencing.
3. deepTools - a toolset used for analysis of deep sequencing data.
4. FastQC - a tool used for quality control.
5. Trim Galore! - a tool for quality and adapter trimming
6. MACS - a tool offering model based analysis for ChIP-Seq data.
 - 6.1. For this project MACS2 was used.
7. Samtools - a toolset for manipulating data in SAM, BAM, or CRAM format.

The environment was exported as a .yaml file to set up and work with the data. To create the environment, the command:

```
$ conda env create -f projenv.yaml
```

is run in the command line. To activate the environment, use

```
$ conda activate environment_name
```

I. Quality control and treatment of the sequences

To begin ChIP seq data analysis, quality control of the raw sequencing data must be performed.

The raw sequencing data is downloaded from the internet into the current working directory using the following commands:

```
$ wget https://zenodo.org/record/1324070/files/wt_H3K4me3_read1.fastq.gz
```

```
$ wget https://zenodo.org/record/1324070/files/wt_H3K4me3_read2.fastq.gz
```

The files are renamed to wt_H3K4me3_read1 and wt_H3K4me3_read2 using the following:

```
$ mv wt_H3K4me3_read1.fastq.gz wt_H3K4me3_read1
```

```
$ mv wt_H3K4me3_read2.fastq.gz wt_H3K4me3_read2
```

FastQC is run to analyze the quality of the reads in the FASTQ files using the command

```
$ fastqc wt_H3K4me3_read1 wt_H3K4me3_read2
```

Low quality bases are trimmed and report files are generated using Trim Galore!, which is run using

```
$ trim_galore --paired -q 15 --stringency 3 wt_H3K4me3_read1 wt_H3K4me3_read2
```

Where 15 is the Phred score requirement for trimming low-quality ends from reads (in addition to adapter removal) and 3 is the number of base pairs of overlapping sequence that will be trimmed from the 3' end of a read. After running Trim Galore!, the trimmed reads are now ready for mapping.

II. Mapping of the reads

In the case of this hands-on, the histone of interest is H3K4me3. H3K4me3 interacts with the chromatin, making the DNA more accessible for transcription—the genes nearby are transcribed

more. The interest in H3K4me3 lies in wanting to know if there is a difference in the quantity of DNA impacted by H3K4me3, and to see what genes are impacted.

To learn more about H3K4me3, the sequenced DNA needs to be mapped to a reference genome, to see where the fragments are located. In lieu of BLAST, Bowtie2 is used to map reads to a genome, as the exact alignment is not significant to the project.

Using Bowtie2, the two trimmed reads from part I are used as input. They will be used as a paired library. The reference genome for this project is: Mouse (*Mus musculus*): mm10, which can be found in the Bowtie2 manual. This genome can be uploaded to the Google Cloud bucket and accessed by the program using the command to copy the files:

```
$ gsutil cp "gs://bucketname/filename" .
```

It is important to get mm10 from the genome index in the Bowtie2 manual. The mm10 indexes used for Bowtie2 are already created, and running bowtie2-build is unnecessary.

With the two reads and the indexed genome, the command bowtie2 can be run using the command:

```
$ bowtie2 -x index -1 trimmed_reads_1.fastq -2 trimmed_reads_2.fastq -S bowtie2_output.sam
```

Where index is the index's name, trimmed_reads_1 is the read1 output of Trim Galore! and trimmed_reads_2 is the read2 output of Trim Galore!, bowtie2_output.sam is the resulting sam file from the alignment. Using IGV, we can visualize the aligned reads.

III. Assess quality of ChIP-seq experiment

The similarity between the replicates of the ChIP data and the control (input) is measured by computing the correlation of read counts on various regions for all samples. The sample BAM files are downloaded using the `wget` command detailed in Step I with the links provided on the Galaxy Training! website. `multiBamSummary` is run to calculate the read coverage (number of unique reads mapped at a given nucleotide) over many regions from each of the BAM files using the following command:

```
$ multiBamSummary bins --bamfiles wt_input_rep1 wt_input_rep2 wt_CTCF_rep1  
wt_CTCF_rep2 wt_H3K4me3_rep1 wt_H3K4me3_rep2 wt_H3K27me3_rep1  
wt_H3K27me3_rep2 --binSize 1000 --distanceBetweenBins 500 --region chrX -o results.npz
```

Where `binSize` (1000) is the number of bases of the genome-sampling window and `results.npz` is the filename of the coverage matrix to be generated. `plotCorrelation` is run to generate a heatmap to visualize the sample correlation matrix that was generated by `multiBamSummary` using the Pearson correlation method:

```
$ plotCorrelation --corData results.npz --corMethod pearson --whatToPlot heatmap --plotTitle  
"Pearson correlation between samples" --plotFile correlation.pdf
```


The resulting heatmap shows if samples cluster together or not (they cluster together if they are highly correlated). IP strength computation determines the clarity by which the ChIP-seq sample signals can be distinguished from the background distribution of reads in the control sample (input). `plotFingerprint` generates a fingerprint plot that visualizes the IP strength and is run using:

```
$ plotFingerprint -b wt_input_rep1 wt_H3K4me3_rep1 --region chrX --numberOfSamples  
10000 --plotFile fingerprint.pdf
```

The resulting fingerprint plot shows how much of the genome contains how much of the reads (more reads = higher strength).

IV. Extract coverage files

The two files being analyzed are wt_H3K4me3_rep1 as the ChIP data, and wt_input_rep1 as the input data. First, to estimate the sequencing depth, IdxStats was run on wt_H3K4me3_rep1.bam and wt_input_rep1.bam. The tool in Galaxy has the ability to run IdxStats on multiple datasets, this is because Galaxy will run separate jobs for each dataset entered, as seen here.

 This is a batch mode input field. Separate jobs will be triggered for each dataset selection.

First, index the files using samtools index. To run IdxStats for the two files in command line, the user has to run IdxStats two times, once for each file, like such:

```
$ samtools idxstats wt_H3K4me3_rep1.bam
```

```
$ samtools idxstats wt_input_rep1.bam
```

After running this, it can be seen that the samples have a different depth, which can affect the results. Using bamCoverage from deepTools can solve this. This tool is another that needs to be run in separate jobs. Following the parameters set by the project guidelines, the bamCoverage command with the two data files would look like this:

```
$ bamCoverage -b wt_H3K4me3_rep1.bam -o bamCoverage_H3K4me3.bed -of bedgraph -bs 25  
-r chrX --normalizeUsing RPGC --effectiveGenomeSize 2308125349
```

-b is the input file, -o is the file name, -of is the file type (bigwig or bedgraph), -bs is bin size (25), -r is region (X chromosome), --normalizeUsing is the normalization method (x1 in this case), and -effectiveGenomeSize is the mappable portion of the genome (mm10 is 2308125349). This is repeated with bigWig. Looking at the file in IGV, the coverage for wt_input_rep1 is stable, while wt_H3K4me3_rep1 has peaks.

To generate normalized coverage files, the tool bamCompare from deepTools is used.

```
$ bamCompare -b1 wt_H3K4me3_rep1.bam -b2 wt_input_rep1.bam -o wt_H3K4me3_rep1 -of bigwig -r chrX
```

-b1 is the treated file, -b2 is the control, -o is the output name, and -of is the output (repeated for bedgraph) and -r is region (X chromosome). The bin size is 50 and the files are compared with log2, but this does not need to be specified when running the tool with command line, as these are the default settings for bamCompare.

V. Call enriched regions

Since enriched regions (peaks) were observed in the ChIP data, MACS2 callpeak is run to generate files containing the coordinates of the peaks using the command

```
$ macs2 callpeak --treatment wt_H3K4me3_rep1 --control wt_input_rep1 --format BAMPE --gsize mm
```

Where wt_H3K4me3_rep1 is the treatment file, wt_input_rep1 is the control file, BAMPE (BAM Paired-End) is the format of the files, and mm is the shortcut for the mouse genome size (1.87e9). The resulting peak files can then be used to visualize gene expression.

VI. Plotting signal between samples

After normalizing the data and identifying the peaks, the final step is visualizing the ChIP enrichment values. The two samples of interest are the H3K4me3 and CTCF samples. The required files for the H3K4me3 sample had already been generated, so the steps needed to be repeated with the CTCF sample. The bamCompare step is repeated with wt_CTCF_rep1.bam to generate the coverage file for this. Then, MACS2 callpeak is also called on this sample to generate the peaks file.

The two peaks files are renamed to their respective sample names, wt_CTCF_rep1 and wt_H3K4me3_rep1, for simplicity. The files are concatenated, and bedtools sort and merge are called to sort and merge the concatenated output:

```
$ bedtools sort -i concatenatedFile to $ bedtools merge -i sortedFile
```

With this merged file, the signal on given regions can be computed using the previous bigwig coverage files. First, computeMatrix is called:

```
$ computeMatrix reference-point --referencePoint center -b 3000 -a 3000 -S wt_CTCF_rep1  
wt_H3K4me3_rep1 -R mergedFile -o matrixResult
```

reference-point indicates the “starting point” for computation (the center here), -b and -a are the distances up and downstream of the starting site, -S are the bigwig files from bamCompare, and -R is the merged file.

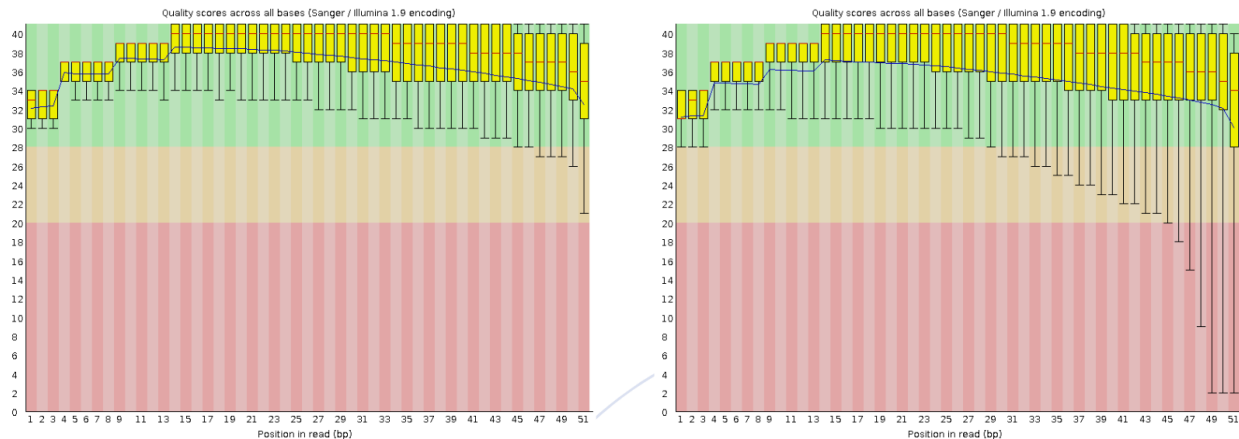
Finally, this matrix is used to plot the heatmap with plotHeatmap:

```
$ plotHeatmap -m matrixResult -o heatMap --refPointLabel center --kmeans 2
```

--refPointLabel is the label shown for the reference-point, and --kmeans is the number of clusters to compute (2 clusters).

Results

The per base sequence quality (included in the report generated by FastQC) decreases towards the ends of the reads for the H3K4me3 sample (especially in R2), indicating that we should trim the low quality bases for optimal results.



R2 has a greater percentage of base pairs trimmed (according to the trimming report generated by Trim Galore!), which makes sense given that the per sequence base quality diagram showed R2 experiencing a greater drop in quality towards the end.

```

=== Summary ===
Total reads processed:          50,000
Reads with adapters:           992 (2.0%)
Reads written (passing filters): 50,000 (100.0%)

Total basepairs processed:      2,550,000 bp
Quality-trimmed:               32,198 bp (1.3%)
Total written (filtered):       2,514,258 bp (98.6%)

```

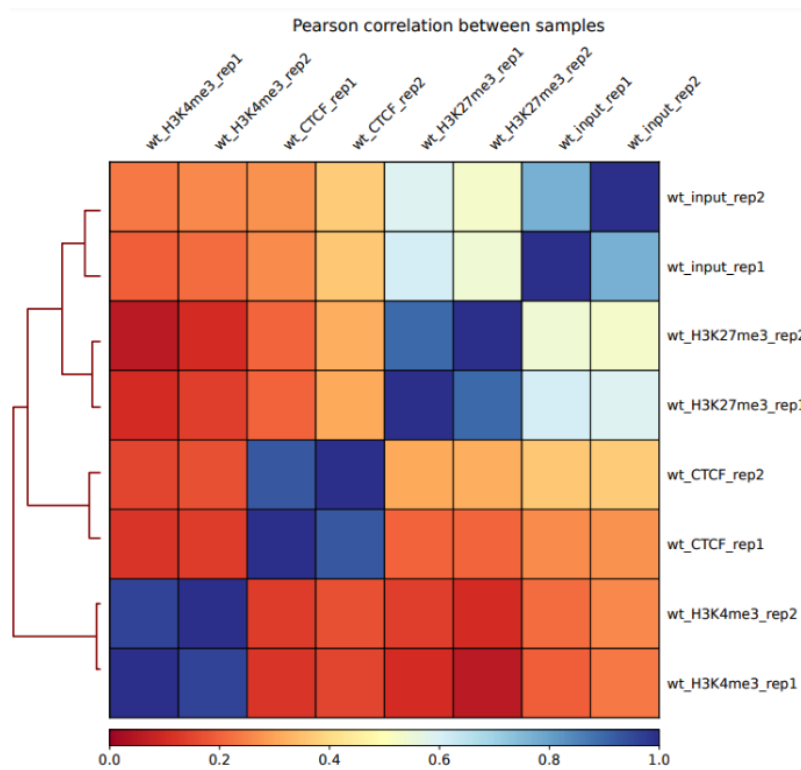
```

=== Summary ===
Total reads processed:          50,000
Reads with adapters:           1,010 (2.0%)
Reads written (passing filters): 50,000 (100.0%)

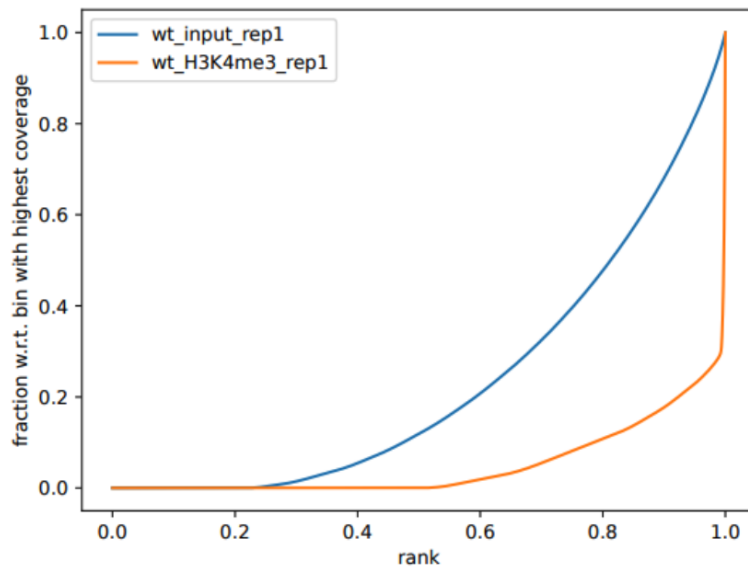
Total basepairs processed:      2,550,000 bp
Quality-trimmed:               116,414 bp (4.6%)
Total written (filtered):       2,430,246 bp (95.3%)

```

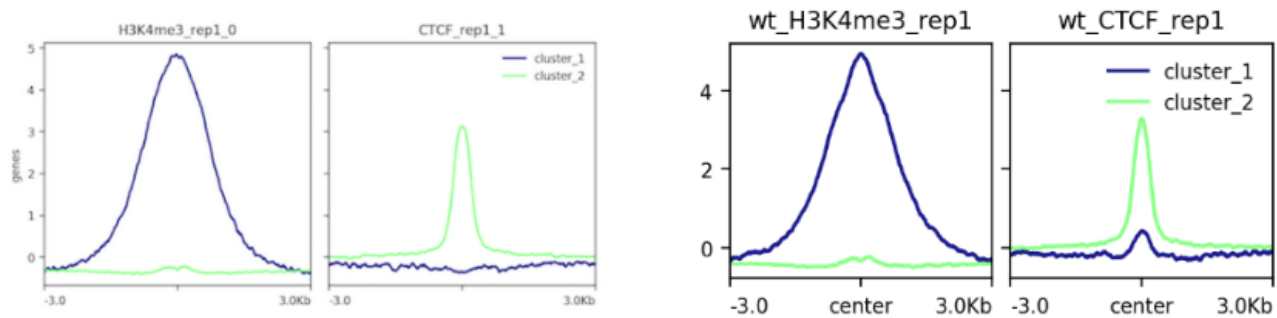
The correlation heatmap (output of plotCorrelation which took in the result matrix of multiBamSummary as input) shows high correlation between input replicates and ChIP-seq replicates.



According to the fingerprint plot generated by plotFingerprint, a small percentage of the genome has a large fraction of reads (especially in the H3K4me3 curve), while some of the chromosome X was not sequenced. This shows that the IP quality for the experiment may not be the most accurate.



The final peaks (right) and heatmap resemble that of the original project's (left):



Larger peaks are seen for the H3K4me3 sample, and shorter peaks are seen for the CTCF sample. Between the two, not many peaks are shared. The CTCF sample has a significantly higher number of peaks compared to the H3K4me3 sample.

Discussion

The results of the project indicate that the presence of the H3K4me3 epigenetic modification of the DNA packaging protein Histone H3 promotes gene expression, while the presence of the H3K27me3 modification of histone H3 represses gene expression. CTCF (CCCTC-binding factor) has both the ability to promote gene expression and the ability to repress gene expression. The tools that were used in this project can be used in future projects to analyze and improve the quality of bases in sequencing data, check alignment quality, analyze sample correlation, estimate IP strength, normalize coverage files, call peaks, compute signals between samples, and plot signals. Further research into the factors that cause CTCF to promote gene expression versus repress gene expression can be conducted in the future. It would also be interesting to research methods of collecting ChIP-seq data and use ChIP-seq data to observe other epigenetic modifications of H3 (as it is the most “extensively modified” of the five “main histones”) or look into epigenetic modifications of other histones.