Epigenetics Project

María Fernanda García, María Fernanda Requena and Maria Jose Rodriguez Barrera

2023-04-04

Introduction

Epigenetics is a change in the genetic expression due to the specific cellular marks that can occur as a result of DNA modifications and its associated proteins. In the paper *"Lineage-Specific Genome Architecture Links Enhancers and Non-coding Disease Variants to Target Gene Promoters"* authors use Chip-Seq to study epigenetic modifications in human blood cell types and its promoter and fragment enrichment. Also Hi-C techniques were developed to obtain major promoter interaction information to match collaborating regulatory regions and identify genes regulated by noncoding disease-associated variants.

In our report we present you an analysis of one of the epigenetic marks found in the paper related to H3K4me3. What we will try to analyze is the difference of the H3K4me3 enrichment in promoters and fragments and the transcription factors involved.

The H3K4me3 mark was chosen because, according to literature, is involved in transcriptional start sites and could regulate transcription initiation. Also, we focused on the erythroblast cells because the paper analysis demonstrates with this type of cells the specificity of the enhancer-promoter interactions. To make this report possible, we analyzed the bigwig signal files of this type of marks from the Blueprint database and developed statistical and data management techniques.

Analysis

Data Download

Resulting interactions After following the path of the data used in the paper we can download it. Working directory: /mnt/Timina/bioinfoII/mgarcia/Epigenetics/blueprint_data.

https://www.cell.com/cell/fulltext/S0092-8674(16)31322-8

```
wget https://www.cell.com/cms/10.1016/j.cell.2016.09.037/attachment/5bc79f6f-1b69-4192-8cb8-4247cc2e0f3
```

Signal files (bigWig) from bluePrint

Data processing Summary Blueprint ChIPSeq analysis pipeline

Mapping

The mapping was carried out using bwa 0.7.7 to human genome GRCh38 reference

Filtering

The output bam file was then filtered to remove unmapped reads and reads with Mapping Quality less than 5

Wiggle plots

The fragment size was modeled using the PhantomPeakQualTools R script

Modeling Fragment Size

Signal plots are produced using align2RawSignal (output: bigwigs). The detailed description of the ChipSeq analysis performed for the blueprint project is in the next link: http://ftp.ebi.ac.uk/pub/databases/blueprint/protocols/Analy

In the following commands we donwloaded the BluePrint data:

```
wget http://ftp.ebi.ac.uk/pub/databases/blueprint/data/homo_sapiens/GRCh38/cord_blood/S002R5/erythroblas

wget http://ftp.ebi.ac.uk/pub/databases/blueprint/data/homo_sapiens/GRCh38/cord_blood/S002S3/erythroblas

wget http://ftp.ebi.ac.uk/pub/databases/blueprint/data/homo_sapiens/GRCh38/cord_blood/S002R5/erythroblas

wget http://ftp.ebi.ac.uk/pub/databases/blueprint/data/homo_sapiens/GRCh38/cord_blood/S002S3/erythroblas
```

Filtering promoters-fragments

We created a bed file to filter the erythroblast cells in the *ActivePromoterEnhancerLinks.tsv*, a file obtained from the total resulting interactions, specifically the *mmc4.zip*.

```
grep "Ery" ../mmc4/ActivePromoterEnhancerLinks.tsv > eritrocitos.bed
```

Afterwards, we decided to divide the data into two, promoters and enhancer data. To do so we chose the first 3 columns of the eritrocitos.bed file. These first 3 columns correspond to the promoters and the next 3 to the Hi-C fragments:

```
awk '{ print $1, $2, $3 }' eritrocitos.bed | sed -e 's/ /\t/g' > promotores.bed
awk '{ print $5, $6, $7 }' eritrocitos.bed | sed -e 's/ /\t/g' > fragmentos.bed
```

Liftover

Then, the positions in the bed files were according to the reference version GRCh37 *(hg19)*, so we had to change them to match with the ones in the version GRCh38 *(Hg38)* using the module of UCSC-executables/ucsc in the cluster, but the online option is also available Lift Genome Annotations ucsc.edu.

Commands

```
module load UCSC-executables/ucsc

liftOver promotores.bed ../hg19ToHg38.over.chain.gz promotores38.bed notpassed-promotores38.bed

liftOver enhancers.bed ../hg19ToHg38.over.chain.gz enhancers38.bed notpassed-fragmentos38.bed
```

Normalizing Data

In the next commands we use the *bigwigCompare* tool from *deeptools 2.5.3* module. This command allows us to normalize our data. We work in the directory: /mnt/Timina/bioinfoII/mgarcia/Epigenetics/norm .

Commands

```
bigwigCompare -b1 ../blueprint_data/S002S3H1.ERX300734.H3K4me3.bwa.GRCh38.20150528.bw -b2  ../blueprint_

bigwigCompare -b1  ../blueprint_data/S002R5H1.ERX300721.H3K4me3.bwa.GRCh38.20150528.bw -b2  ../blueprint
```

Brief explanation

The command bigwigCompare compares two bigWig files :

- *b1*: specifies the first bigWig file to compare (treatment file)
- *b2*: specifies the second bigWig file (control file)
- *-o* indicates the output normalized file

We merge both .bw normalized files with *bigWigMerge* command. The first two files correspond to the normalized data files and the third file is the output file (.bedGraph).

```
bigWigMerge norm_2S3H1.bw norm_2R5H11.bw norm_merge_H3K4me3.bedGraph
```

Sorting data

In the following command we organize our bedGraph data to speed up the following processes.

```
sort -k1,1 -k2,2n norm_merge_H3K4me3.bedGraph > nm_sorted_H3K4me3.bedGraph
```

- k1,1 specifies sorting based on the first column
- k2,2n specifies secondary sorting based on the numerical value in the second column. The n after the 2 specifies that the second column should be sorted numerically.

Afterwards, a conversion from bedGraph to Bigwig is done with the following command:

```
bedGraphToBigWig nm_sorted_H3K4me3.bedGraph ../blueprint_data/GRCh38_EBV.chrom.sizes.tsv nm_sorted.H3K4
```

In this command the input file is *nm_sorted_H3K4me3.bedGraph* and the output file is *nm_sorted.H3K4me3.bw*

Computematrix

Create a matrix from the sorted file for each bed file containing the promoters and the fragments. Directory: /mnt/Timina/bioinfoII/mgarcia/Epigenetics/norm

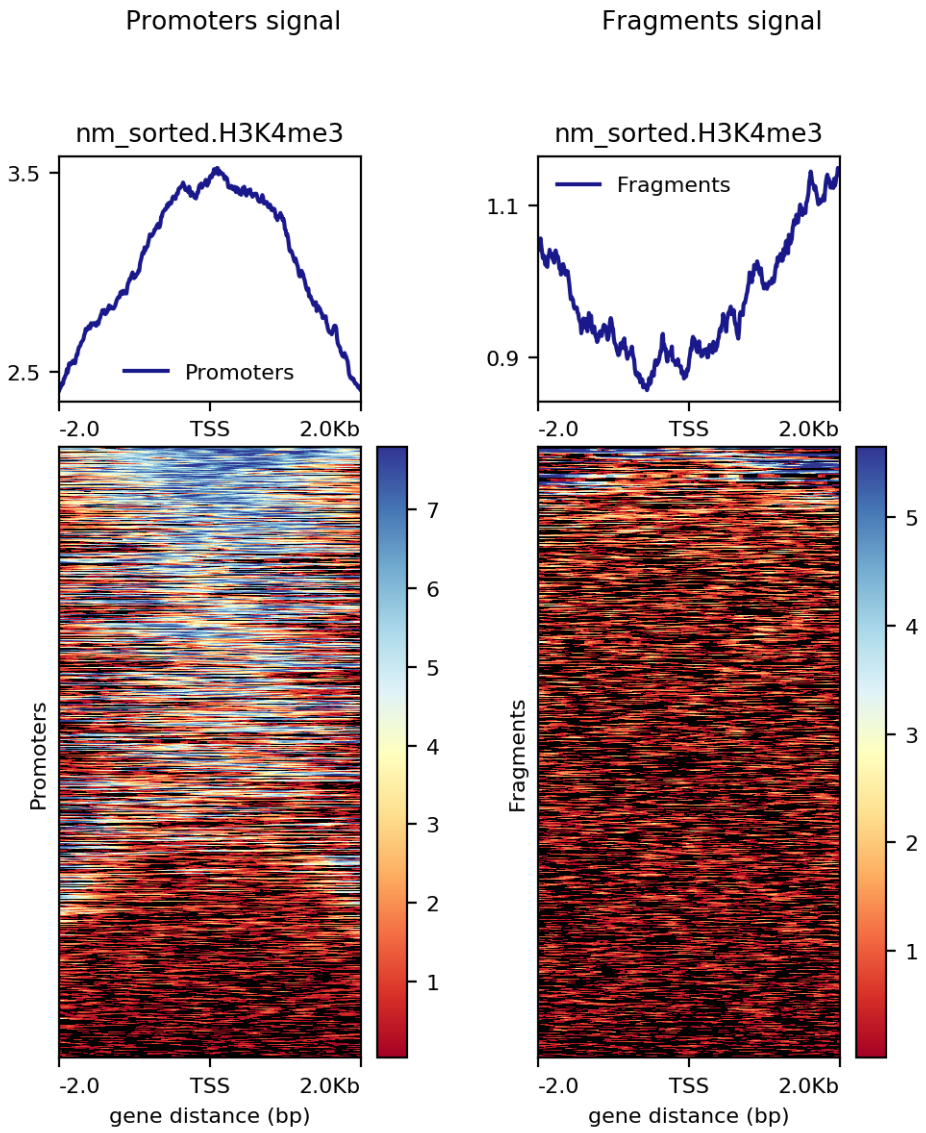
```
‘computeMatrix reference-point -S nm_sorted.H3K4me3.bw -R ../filtered_data/promotores38.bed  --referenc

computeMatrix reference-point -S nm_sorted.H3K4me3.bw -R ../filtered_data/fragmentos38.bed  --referencel
```

We visualized data matrices with the plotHeatmap command

```
plotHeatmap -m Promoters_H3K4me3_nmatrix.tab.gz -out heatmap_prom_H3K4me3.png --heatmapHeight 10 --regi

plotHeatmap -m Fragments_H3K4me3_nmatrix.tab.gz -out heatmap_frag_H3K4me3.png --heatmapHeight 10 --regi
```



Interpretation

After obtaining the heatmaps from the command *plotHeatmap* we can observe what seems to be a difference between Promoters and Fragments **H3K4me3 enrichment**. These differences are represented by the different colors, ranges and patterns seen in comparison between them.

Firstly, we can clearly visualize how the Promoters signal has a higher coverage depth, represented by the color blue it represents in certain regions. On which blue color indicates a higher coverage. On the other hand, Fragments signal heat map looks almost homogenous in a low range color.

In the same way, if we compare the ranges represented in the graph above from both signals, we can observe again that the coverage intensity, when H3K4me3 is present, from the Fragments signal is lower than the Promerts one. With a value of almost always lower than 1.1 in Fragments signal coverage and with Promoters coverage above 2.5.

Last but not least, in the Promoters Heatmap we are able to observe almost a perfect bell peak, as already mentioned with values above 2.5, with its highest point represented in the Transcription Starting Site (TSS). This distribution not only can give us an idea of the difference of H3K4me3 enrichment in general, but also tell us that the H3K4me3 is associated with transcription start sites, as literature says. In contrast, Fragment signals present peaks and valleys all over the graph and without distinction in the TSS.

Testing Data Normality

To test the enrichment of the histone in promoters or fragments we verify if each matrix is following a normal distribution with qqplots in R.

```
module load r/4.2.2
R
```

Load Data

We use the command read.delim to read two tabular separated files *(n_promotor_mat.tab.gz and n_fragmento_mat.tab.gz)* and ask R to skip the first line and #not to look for a header (the files do not include header).

```
promotores<-read.delim("n_promotor_mat.tab.gz", header = F, skip=1)
fragmentos<-read.delim("n_fragmento_mat.tab.gz", header = F, skip=1)
```

The nam values in the files were replaced for zeros

```
promotores[is.na(promotores)] = 0 #NAs to 0
fragmentos[is.na(fragmentos)] = 0
```

Remove columns

```
promotores<-promotores[,-c(1:6)]
fragmentos<-fragmentos[,-c(1:6)]
```

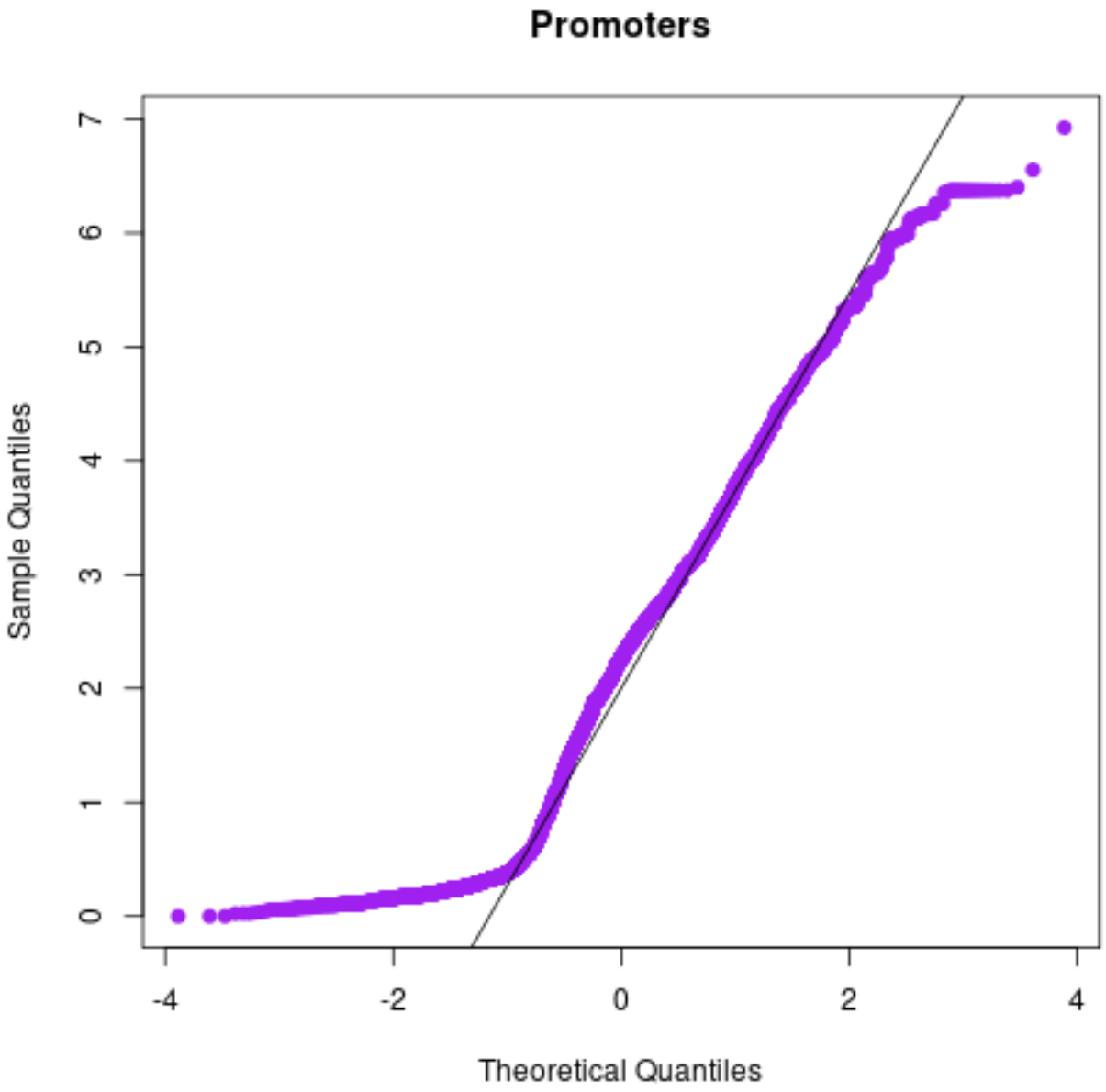
Mean for each row

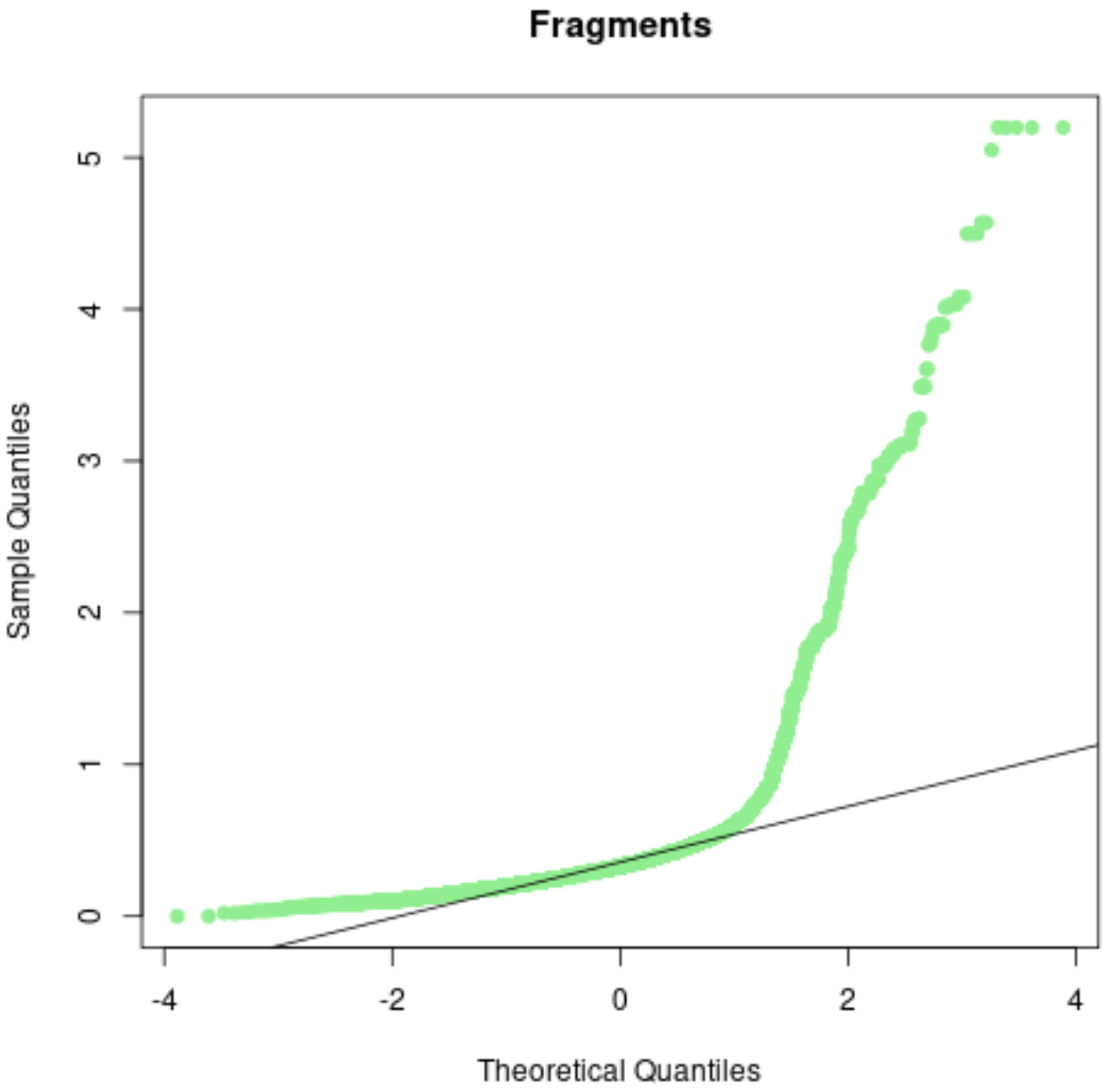
```
promotores_row_means <- data.frame(ROW_MEANS=rowMeans(promotores))
fragmentos_row_means <- data.frame(ROW_MEANS=rowMeans(fragmentos))
```

qqplots to assess the normality of the data

```
png(file ="qqplot_promotores.png")
qqnorm(promotores_row_means$ROW_MEANS, pch = 19, col = "purple",main="promotores")
qqline(promotores_row_means$ROW_MEANS)
dev.off()

png(file ="qqplot_fragmentos.png")
qqnorm(fragmentos_row_means$ROW_MEANS, pch = 19, col = "lightgreen",main="fragmentos")
qqline(fragmentos_row_means$ROW_MEANS)
dev.off()
```


Promoters
Sample Quantiles
Theoretical Quantiles
0
1
2
3
4
5
6
7
-4
-2
0
2
4



As we can see in the fragments plot, the Promoter's graph dots align better to the bar than in the Fragment's graph. This could vaguely tell us that the histone H3K4me3 is more enriched to promoters than to possible enhancers (fragments).

However we consider that there are some points that fall outside the line, which may affect the validity of assuming normality for the statistical test. Consequently, we applied the Shapiro-Wilk's method, which is a well known statistical technique to prove if the data follows a normal distribution.

Shapiro-Wilk's method

```
#arrays made with each row mean
prom.enr<-promotores_row_means[,1]
frag.enr<-fragmentos_row_means[,1]
```

```
#Shapiro-Wilk
SW.promotores <- sample(prom.enr, 5000)
shapiro.test(SW.promotores)
#
# Shapiro-Wilk normality test
#
# data:  SW.promotores
# W = 0.9572, p-value < 2.2e-16

SW.fragmentos <- sample(frag.enr, 5000)
shapiro.test(SW.fragmentos)
#
# Shapiro-Wilk normality test
#
# data:  SW.fragmentos
# W = 0.57199, p-value < 2.2e-16
```

According to the results, the p value is lower than 2.2e-16, so we can conclude that the data does not follow a normal distribution.

Histone enrichment in promoters and fragments

Enrichment hypothesis test

Is there greater enrichment of the histone H3K4me3 in promoters or in enhancers?

Considering the above conclusion, we can develop a statistical method to test the enrichment of the histone in promoters and fragments (enhancers). To make this possible, we implemented a Wilcox test, which is a non parametric test and is used as an alternative test for the t-test when the data given is not normal. This test gives the possibility of comparing two independent samples to compare if there is a significant difference between them.

Wilcoxon test

```
wilcox.test(prom.enr,frag.enr, alternative = "two.sided",mu=0, paired=F, conf.int = T, conf.level = 0.9
```

The results given are the following:

- data: prom.enr and frag.enr
- W = 81823562, p-value < 2.2e-16
- alternative hypothesis: true location shift is not equal to 0
- 99 percent confidence interval: 1.733800 1.826695
- sample estimates: difference in location: 1.779897

As we can see, the p value and the difference in location declare that there is a difference between the means of the two samples given (promoters and enhancers in our case). The confidence interval (99 percent confidence) also establishes the difference between the samples with a positive value that demonstrates that the promoter enrichment is higher than in enhancers.

TFs enriched

In order to find the transcription factors present we had to create bed files containing the sequences enriched in promoters and enhancers sorted previously. The module used to do this was bedops and its function –intersect that determines genomic regions common to all input sets.

```
module load bedops/2.4.20
bedops --intersect promotores38.bed fragmentos38.bed > prom_frag_38.bed
```

Download the resulting file to our device.

```
rsync -chavzP --stats mgarcia@dna.lavis.unam.mx:/mnt/Timina/bioinfoII/mgarcia/Epigenetics/filtered_data/
```

We entered to RSAT (Regulatory Sequence Analysis Tools (RSAT) and used the guide of which program to use, this asks for the type of data to analyze (Coordinates (BED)), your biological question / analysis to perform (We want to extract the sequences corresponding to these coordinates) and the Relevant RSAT programs (fetch sequences from UCSC). Then we uploaded the previous downloaded file and chose the genome reference hg38. The output files were a .bed file with genomic coordinates, a .fasta with the fetched sequences and a .txt

Result file(s)

Genomic coordinates (bed)

http://rsat.france-bioinformatique.fr/metazoa//tmp/www-data/2023/04/08/prom_frag_38_7ep3_20230408_195016.bed

Fetched sequences (fasta)

http://rsat.france-bioinformatique.fr/metazoa//tmp/www-data/2023/04/08/prom_frag_38_7ep3_20230408_195016.fasta

Log file (txt)

http://rsat.france-bioinformatique.fr/metazoa//tmp/www-data/2023/04/08/prom_frag_38_7ep3_20230408_195016_log.t

```
wget -O fetched_H3K4me3.fasta http://rsat.france-bioinformatique.fr/metazoa//tmp/www-data/2023/04/08/pr
```

Non redundant Jaspar collection

We used the characterized TFBs non redundant Jaspar collection, this will help to:

- Avoid redundance: JASPAR non redundant collection is high cured data base that does not include duplicated and redudant TF.
- Data Confidence: JASPAR Data Base is a well established and known data base used by scientist that is often updated and review.
- JAPAR collection has wide information of Transcription Factors.

To download it we visited the page: https://jaspar.genereg.net/ / Below the search bar we clicked in “Advanced Options” and specified that we wanted the core collection, of Homo sapiens, all classes, latest version, vertebrates, ChiP-seq and all families. 233 profiles agreed with those characteristics so we added them to the cart and downloaded them as a combined text file in mem format. The link to download the files is useful only during 5 days.

```
wget -O JASPAR_m_meme.txt http://jaspar.genereg.net/temp/20230409184822_JASPAR2022_combined_matrices_52
```

It is important to use permuted sequences as control because this can give us a higher confidence in our analysis and conclusions:

- Permuted sequences are used as reference to establish the significance level and determine if the observed patterns are real or random.
- It sort allows us to identify if patterns observed have a biological reason or result of introduced artefacts during the data analysis.

Then we downloaded the permuted matrices with the default type in meme format:

```
wget -O JASPAR_perm_meme.txt http://jaspar.genereg.net/temp/20230409190344_permuted_matrices_5229.txt
```

Pattern matching

The pattern matching could be realized with the rsat program 'matrix-scan' with the following job:

```
#!/bin/bash
#
# Use Current working directory
#$ -cwd
#
# Join stdout and stderr
#$ -j n
#
# Run job through bash shell
#$ -S /bin/bash
#
# You can edit the script since this line
#
# Your job name
#$ -N pat_matching
#
# Send an email after the job has finished
#$ -m e
#$ -M ****@gmail.com
#
#
# If modules are needed, source modules environment (Do not delete the next line):
. /etc/profile.d/modules.sh
#
# Add any modules you might require:
module load rsat/8sep2021
#
# Write your commands in the next line
matrix-scan -m JASPAR_m_meme.txt -matrix_format meme -i fetched_H3K4me3.fasta -bginput -markov 0 -o mat
```

Explanation

- *-m* specifies the input motif matrix file to be used for scanning
- *-matrix_format meme*: Specifies the format of the input motif matrix file, which is MEME format in this case.

- *-i fetched_H3K4me3.fasta*: Establishes the input DNA sequence file to be scanned for transcription factor binding sites
- *-bginput:* will compare the input sequences to the provided motif matrix to identify potential transcription factor binding sites.
- *-markov 0*: This specifies the order of the Markov background model to be used for calculating motif scores. A value of 0 indicates that no Markov background model should be used, which means a simple nucleotide frequency background model will be used instead.
- *-o matrix_scan_tf*: output file

We tried to run the command using the permuted sequences matrix:

```
matrix-scan -m JASPAR_m_meme.txt -matrix_format meme -i fetched_H3K4me3.fasta -bgfile JASPAR_perm_meme.
```

Following the format:

```
matrix-scan -m matrixfile [-i inputfile] [-o outputfile] [-v] [-bgfile backgroundfile|-bgorder #]
```

But an error was displayed in the terminal. Then we used -markov 0 to indicate the use of a simple nucleotide frequency background, this let us run the command. We realized that the use of this background may not be the suited option because it can lead to negative results in the "weight" column.

Fetch sequences larger than 20 nucleotides and weight scores greater than 22:

```
awk '{if(($6 - $5)> 20 && $8>22) print ($3)}' matrix_scan_tf| sort | uniq > IDmatrix.txt
```

Output

- MA1589.1

In the following command we obtained the TF name:

```
grep -f IDmatrix.txt JASPAR_m_meme.txt | cut -d "." -f 4 >> TF_name.txt
Contenido de TF_name.txt:
```

Output

- ZNF140
- 1

The obtained TF is the Zinc finger protein 140 which "enables DNA-binding transcription repressor activity, RNA polymerase II-specific and sequence-specific double-stranded DNA binding activity. Involved in negative regulation of transcription by RNA polymerase II. Predicted to be active in nucleus." (genecards.org).

With the weight score of 22 we obtained just one TF (ZNF140), the same goes for the weight values 21, 20, 19 and 18. We can decrease the weight value to find more TFs although what is obtained may not be as reliable.

Fetch sequences larger than 20 nucleotides and weight scores greater than 16:

```
awk '{if(($6 - $5)> 20 && $8>16) print ($3)}' matrix_scan_tf| sort | uniq > weight.txt
```

Output

- MA0050.2
- MA0138.2
- MA1589.1
- MA1929.1
- MA1930.1

In the following command we obtained all the TF names:

```
grep [Insert each ID] JASPAR_m_meme.txt | cut -d "." -f 4
```

Output

- IRF1
- REST
- ZNF140
- CTCF
- CTCF

TR Fuctions

- **IRF1**: Interferon Regulatory Factor 1, transcriptional regulator and tumor suppressor, activates genes involved in immune responses (innate and acquired).
- **REST**: RE1-Silencing Transcription Factor, member of the Kruppel-type zinc finger transcription factor family. Also represses neuronal genes in non-neuronal tissues.
- **ZNF140**: Zinc finger protein 140
- **CTCF** Extended motif with zinc finger 8 (5bp): 11-zinc finger protein or CCCTC-binding factor, involved in transcriptional regulation, insulator activity, recombination and regulation of chromatin architecture.
- **CTCF** Extended motif with zinc finger 8 (6bp)

Conslusion

This project focuses on the modification H3K4me3, in the first part we can observe that the modification is more enriched in promoters than in enhancers for erythrocytes. In the second part we had some problems with the command using the permuted sequence matrix so we decided to use markov 0 that indicates a simple nucleotide frequency. Although it was not the best decision because we obtained negative results, this let us run the command. When the matrix-scan command was run with weight scores greater than 22, just 1 result was obtained: ZNF140 (Zinc finger protein 140). The same command was run but the weight value was decreased in order to obtain more than 1 result. with weight scores greater than 16 we obtained *ZNF140, IRF1 (Interferon Regulatory Factor 1), REST (RE1-Silencing Transcription Factor), CTCF (Extended motif with zinc finger 8 - 5bp)* and *CTCF (Extended motif with zinc finger 8 - 6bp).* The proteins associated with the TF are Zinc fingers and proteins that play an important role in transcription regulation.

Modules

- deeptools/2.5.3
- r/4.2.2
- bedops/2.4.20
- rsat/8sep2021

Bibliography

- Javierre BM, Burren OS, Wilder SP, Kreuzhuber R, Hill SM, Sewitz S, Cairns J, Wingett SW, Várnai C, Thiecke MJ, Burden F, Farrow S, Cutler AJ, Rehnström K, Downes K, Grassi L, Kostadima M, Freire-Pritchett P, Wang F; BLUEPRINT Consortium; Stunnenberg HG, Todd JA, Zerbino DR, Stegle O, Ouwehand WH, Frontini M, Wallace C, Spivakov M, Fraser P. Lineage-Specific Genome Architecture Links Enhancers and Non-coding Disease Variants to Target Gene Promoters. Cell. 2016 Nov 17;167(5):1369-1384.e19. doi: 10.1016/j.cell.2016.09.037. PMID: 27863249; PMCID: PMC5123897.
- https://deeptools.readthedocs.io/en/develop/content/example_step_by_step.html#heatmaps-and-summary-plots
- https://www.cienciadedatos.net/documentos/17_mann%E2%80%93whitney_u_test
- (PDF) Using regulatory genomics data to interpret the function of disease variants and prioritise genes from expression studies
- rCOGS Quickstart
- make_pchic: This function computes loads in a set of promoter capture... in ollyburren/rCOGS: Computes gene prioritisation scores based on promoter capture Hi-C data
- liftOverPlink/README.md at master · sritchie73/liftOverPlink · GitHub
- matrix-scan manual
- ZNF140 Gene - GeneCards | ZN140 Protein | ZN140 Antibody
- JASPAR - CTCF - MA1929.1 - profile summary
- JASPAR - CTCF - MA1930.1 - profile summary
- 6.1.1. bedops — BEDOPS v2.4.41