Week_9

Assay for transposase accessible chromatin coupled with sequencing (ATAC-seq): motif discovery and integration with RNA-seq

ATAC-seq motif discovery

Tasks

Task 1

Read the motimatchr vignette

Task 2

Load the rtracklayer package.

```
library(rtracklayer)
GRanges_data = import("/student_data/BBT.BI.203_2022/data/ex9/atac/unified.bed", format = "BED")
```

Task 3

Extract all human motifs from JASPAR2014

```
library(motifmatchr)
library(TFBSTools)
library(JASPAR2014)
suppressMessages(library(JASPAR2014))
opts <- list()
opts[["species"]] <- 9606
PFMatrixList <- getMatrixSet(JASPAR2014, opts)
```

Task 4

Use the matchMotifs function to find the motifs inside the peaks from the previous section

```
match_motif = matchMotifs(PFMatrixList,
                          GRanges_data,
                          genome = "hg38",
                          out = "scores",
                          p.cutoff = 5e-05, w = 7)
```

Task 5

Use the motifCounts function to extract the matrix of motif counts from the object in task4

```
counts_data = motifCounts(match_motif)
```

Task 6

Find the motif with highest occurrence across all peaks

```
a = colSums(counts_data)
which.max(colSums(counts_data))
```

Task 7

Extract the position Weight Matrix (PWM) representing the motif from taks 6 from the Jaspar databased object MA0528.1, ZNF263

```
opts_ZNF263 = list()
opts_ZNF263[["species"]] <- 9606
opts_ZNF263[["name"]] <- "ZNF263"
pfm = getMatrixByID(JASPAR2014, ID = "MA0528.1")
pwm_data = toPWM(pfm, pseudocounts = 0.8)
```

Task 8

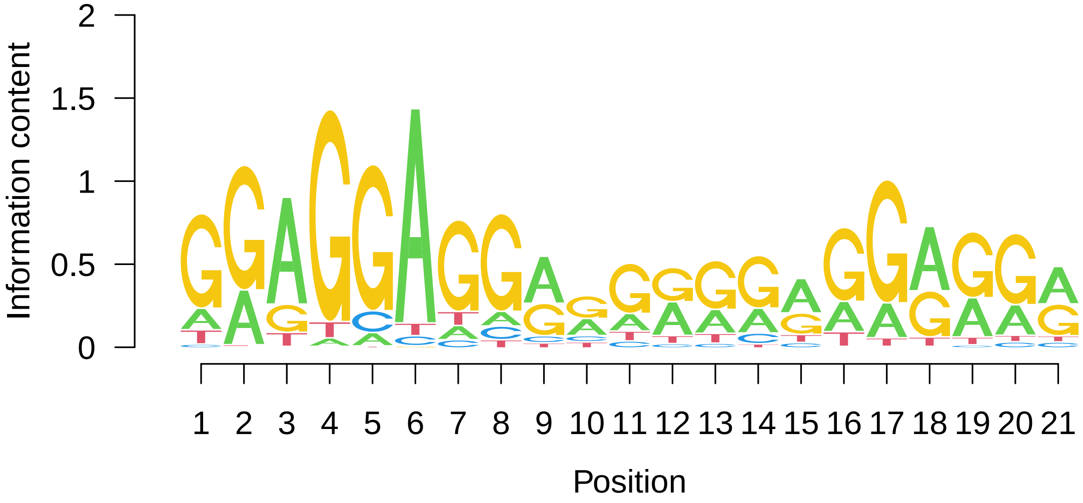
Convert the PWM from task 7 to Information Content Matrix (ICM)

```
icm <- toICM(pfm, pseudocounts=0.8, schneider=TRUE)
```

Task 9

Plot the sequence logo of the motif from Tasks 6,7,8 using its ICM

```
seqLogo(icm)
```



## Questions

Question 1

Give the definitions of the following terms: a. Position frequency matrix summarize occurrences of each nucleotide at each position in a set of observed transcription factor-DNA interactions.

b. Position weight matrix is known as a position-specific weight matrix which presents a scoring matrix composed of the log likelihood of each nucleotide in a motif.

c. Information content matrix presents how different a given position weight matrix is from a uniform distribution.

Question 2

The transcription factor is MA0528.1, ZNF263

Question 3

Integration of ATAC-seq with RNA seq

Tasks

Task 1

```
annotatePeaks.pl /student_data/BBT.BI.203_2022/data/ex9/atac/unified.bed hg38 -annStats annotated_peaks
```

Task 2

Load the results from Task 1 into a table in R with read.delim() function.

```
annotated_peaks = read.delim("anotated_peaks.tsv")
colnames(annotated_peaks)[1] = "Peak_id"
```

Task 3

Subset the results from task 2 by Annotation.

```
nm <- annotated_peaks$Annotation
start_with_promoter <- nm %in% grep("^promoter", nm, value = TRUE)
annotation_subset <- subset(annotated_peaks, start_with_promoter, select=Peak_id:Gene.Type)
```

Task 4

Loading the mapping table into R

```
mapping_table = read.table(file = "/student_data/BBT.BI.203_2022/data/ex9/tables/mapping.txt",
                           header = TRUE,
                           sep ="")
mapping_table = as.data.frame(mapping_table)
```

Task 5

```
merge_data <- merge(x = annotation_subset,
                    y = mapping_table,
                    by.x = "Nearest.PromoterID",
                    by.y = "transcript_id")
```

Task 6

Load RNA-seq log-fold changes in to R

```
TSA_vs_DMSO = readRDS("/student_data/BBT.BI.203_2022/data/ex9/rna/TSA_vs_DMSO.rds")
```

Task 7

Subset the log-fold cahnge table from task 6 to contain only the log2FoldChange column.

```
log2FoldChange_rna = TSA_vs_DMSO[[2]]
log2FoldChange_rna_data_frame = as.data.frame(log2FoldChange_rna)
row.names(log2FoldChange_rna_data_frame) = row.names(TSA_vs_DMSO)
```

Task 8

Use the merge function to add the RNA-seq log-fold change values to the annotation table from the task 5

```
merge_data_RNA <- merge(x = merge_data,
                    y = log2FoldChange_rna_data_frame,
                    by.x = "gene_id",
                    by.y = 0)
```

Task 9

Load the unified peaks quantification table into R

```
peaks_quantification = read.csv("/student_data/BBT.BI.203_2022/data/ex9/atac/unified_peaks.multicov",
                                sep = "\t",
                                header = FALSE)
colnames(peaks_quantification) = c("chr", "start", "end", "name", "score", "strand", "SRR3622814", "SRR3
```

Task 10

Extract the last six columns from the Task 9 data.frame and convert it to a matrix

```
data_frame_peaks = as.data.frame(cbind(peaks_quantification$SRR3622814,
                                       peaks_quantification$SRR3622815,
                                       peaks_quantification$SRR3622816,
                                       peaks_quantification$SRR3622817,
                                       peaks_quantification$SRR3622818,
                                       peaks_quantification$SRR3622819))
matrix_peaks = as.matrix(data_frame_peaks)
colnames(matrix_peaks) = c("SRR3622814", "SRR3622815", "SRR3622816", "SRR3622817", "SRR3622818", "SRR362
rownames(matrix_peaks) = peaks_quantification$name
```

Task 11

Perform CPM normalization on the matrix from the Task 10 CPM (Counts Per Million) are obtained by dividing counts by the library counts sum and multiplying the results by a million.

```
matrix_peaks_cpm = cbind(matrix_peaks[,1]/sum(matrix_peaks[,1]),
                         matrix_peaks[,2]/sum(matrix_peaks[,2]),
                         matrix_peaks[,3]/sum(matrix_peaks[,3]),
                         matrix_peaks[,4]/sum(matrix_peaks[,4]),
                         matrix_peaks[,5]/sum(matrix_peaks[,5]),
                         matrix_peaks[,6]/sum(matrix_peaks[,6]))
colnames(matrix_peaks_cpm) = colnames(matrix_peaks)
```

Task 12

Compute the log2fold changes

```
data_frame_log2foldchange = as.data.frame(log2(rowMedians(matrix_peaks_cpm[,1:3])/rowMedians(matrix_peak
row.names(data_frame_log2foldchange) = annotated_peaks$Peak_id
colnames(data_frame_log2foldchange) = "log2FoldChange_atac"
```

Task 13

Merge the ATAC-seq log-fold changes to data.frame from Task 8

```
merge_data_RNA_ATAC <- merge(x = merge_data_RNA,
                             y = data_frame_log2foldchange,
                             by.x = "Peak_id",
                             by.y = 0)
```

Task 14

Subset the values for ATAC-seq log2FoldChanges in the range [-2,2]

```
value_foldchange_ATAC <- merge_data_RNA_ATAC$log2FoldChange_atac
range_value <- value_foldchange_ATAC <2 & value_foldchange_ATAC > -2
subset_atac_rna <- subset(merge_data_RNA_ATAC,
                          range_value,
                          select=log2FoldChange_rna:log2FoldChange_atac)
```

Task 15

Make a scatterplot of the log-fold changes

```
plot(subset_atac_rna$log2FoldChange_atac,
     subset_atac_rna$log2FoldChange_rna,
     main ="Scatter plot",
     xlab = "ATAC-seq fold change",
     ylab = "RNA fold change",
     ylim = c(-4,10))
```

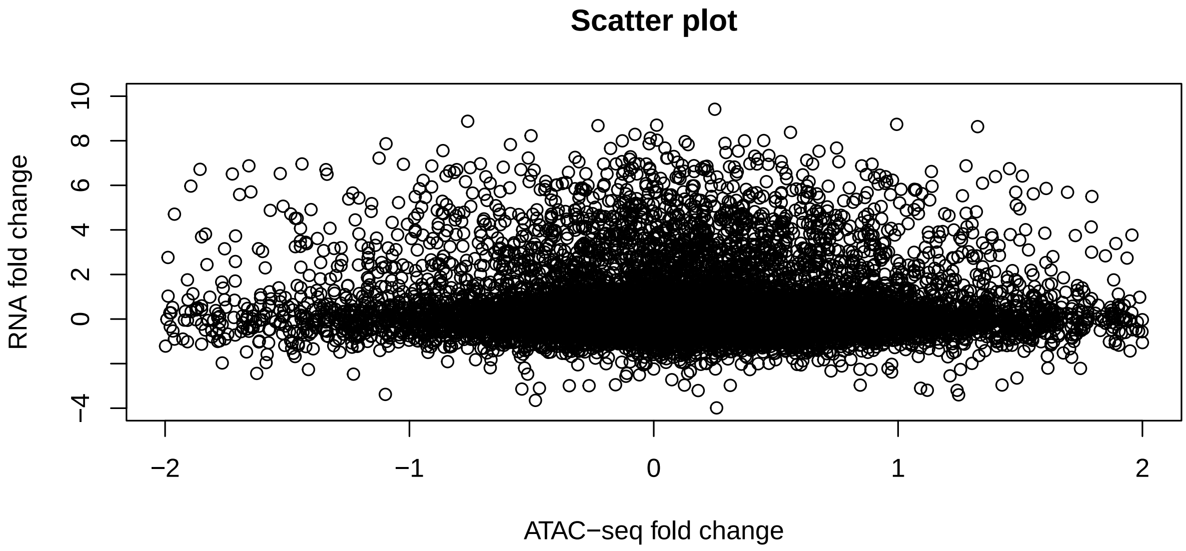


Figure 1: A scatter plot and a smoothed scatterplot

```
## Warning in plot.window(...): "buylrd" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "buylrd" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "buylrd" is not a
## graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "buylrd" is not a
## graphical parameter

## Warning in box(...): "buylrd" is not a graphical parameter
## Warning in title(...): "buylrd" is not a graphical parameter
```

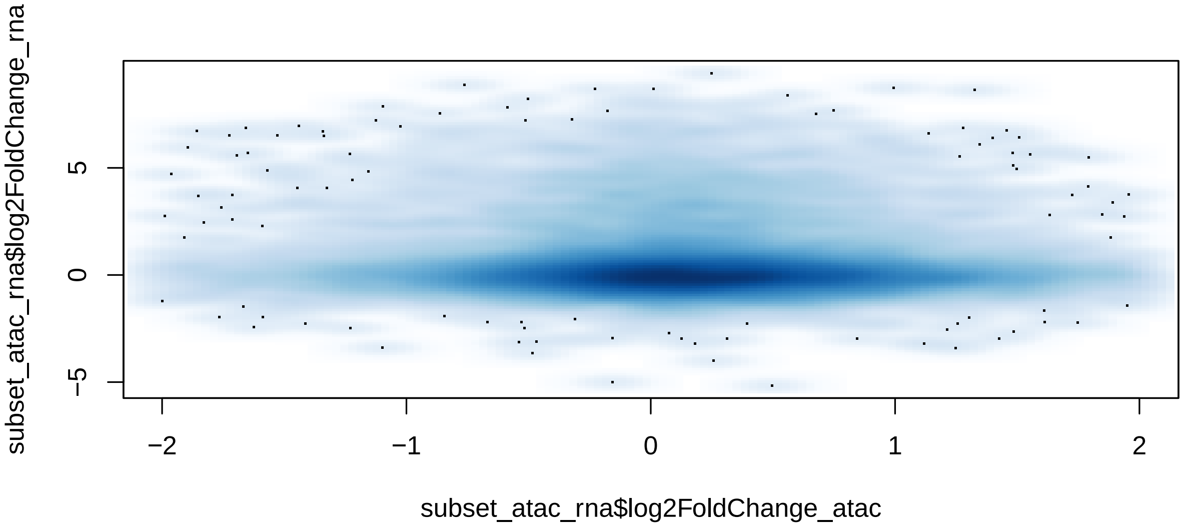


Figure 2: A scatter plot and a smoothed scatterplot

Task 16

Draw a smoothed scatterplot

```
smoothScatter(subset_atac_rna$log2FoldChange_atac,
              subset_atac_rna$log2FoldChange_rna,
              buylrd = c( "#313695", "#4575B4", "#74ADD1","#ABD9E9", "#E0F3F8", "#FFFFBF", "#FEE090", "
```

Questions

Question 1

Describe the columns of the final data.frame of Task 13:

The data frame include 12104 rows with 22 columns. The columns are following: 1. Peaks ID, unique Peaks ID; Show the annotation of peak ID 2. gene ID; show the gene ID 3. Nears promoter ID: present the nearest promoter ID to the peak identification 4. Chromosome names. 5. Start of annotation peaks. 6. End of annotation peaks. 7. Strand. 8. Peak score. 9. Focus ration region size. 10. Annotation. 11. Details of annotation. 12. Distance to TSS. 13. Enterz ID. 14. Nearest unigene. 15. Nearest refseq 16. Nearest ensembl. 17. Gene name. 18. Gene alias. 19. Descriptions of genes. 20. Gene types. 21. Log2Foldchange of RNA expresison. 22. Log2Foldchange of ATAC seq analysis.

Question 2

Merge function could be used to merge two data frames by common columns or row names, or do ther version of database join operations. However, the sequence of data as row names is not changes, it likes merges together.

Question 3

CPM normalization has proven essential to ensure accurate inferences and replication of findings. The findings of CPM is found based on by dividing reads counts by gene lengths (expressed in million-nucleotides). The changes of gene lengths by amplifications and deletions could make the effect of the CPM. How to deal with the problem?

Question 4

Yes, based on a scatter plot, I see that I observed a much stronger correlation between ATAC-seq and RNA-seq log2FoldChange. I think that while correlating annotated chromatin peaks identified by ATAC-seq and differential expressed genes detected by RNA-seq can home in the similar or close functionally-relevatn chromatin regions.