# Gene Expression Analysis

## Introduction:

For the current lab, we performed feature counting for gene expression. The process by which a gene's information is used to create a functioning gene product, such as a protein or RNA molecule, is known as gene expression. It entails transforming the genetic data included in DNA into a gene product that can perform a particular biological function in a cell or organism.

We do feature counting for gene expressions to calculate the degree of gene expression in a sample, and we do feature counting for gene expressions. Counting the number of reads, or sequencing fragments, that correspond to particular genomic features, like genes or exons, is known as feature counting. We can estimate the abundance of a gene or exon in the sample and infer its degree of expression by counting the number of reads that map to that gene or exon.

The samples were available from dropbox and were uploaded to the cluster after downloading. Each of them had a forward read and a reverse read.

## Method:

### 2.1 Indexing and mapping:

```
module load bowtie2/2.4.1
bowtie2 -x CMCP6.GCA -U CMCP6_1ASWsample.1.fq CMCP6_1ASWsample.2.fq -S
CMCP6_1ASWsample.sam 11242150 reads; of these:
11242150 (100.00%) were unpaired; of these: 16283 (0.14%) aligned 0 times
11001879 (97.86%) aligned exactly 1 time 223988 (1.99%) aligned >1 times
bowtie2 -x CMCP6.GCA -U CMCP6_2ASWsample.1.fq CMCP6_2ASWsample.2.fq -S
CMCP6_2ASWsample.sam
9826782 reads; of these:
9826782 (100.00%) were unpaired; of these: 23564 (0.24%) aligned 0 times
9603439 (97.73%) aligned exactly 1 time 199779 (2.03%) aligned >1 times
bowtie2 -x CMCP6.GCA -U CMCP6_1HSsample.1.fq CMCP6_1HSsample.2.fq -S
CMCP6_1HSsample.sam
8504090 reads; of these:
8504090 (100.00%) were unpaired; of these: 16778 (0.20%) aligned 0 times
8262135 (97.15%) aligned exactly 1 time 225177 (2.65%) aligned >1 times
bowtie2 -x CMCP6.GCA -U CMCP6_2HSsample.1.fq CMCP6_2HSsample.2.fq -S
CMCP6_2HSsample.sam
10641448 reads; of these:
```

```
10641448 (100.00%) were unpaired; of these: 24645 (0.23%) aligned 0 times
10378281 (97.53%) aligned exactly 1 time 238522 (2.24%) aligned >1 times
```

**2.2 Conversion of SAM to BAM:**

```
module load samtools/1.10
samtools view -uS CMCP6_1ASWsample.sam | samtools sort - -o
CMCP6_1ASWsample-srt.bam
samtools view -uS CMCP6_2ASWsample.sam | samtools sort - -o
CMCP6_2ASWsample-srt.bam
samtools view -uS CMCP6_1HSsample.sam | samtools sort - -o CMCP6_1HSsample-srt.bam
samtools view -uS CMCP6_2HSsample.sam | samtools sort - -o CMCP6_2HSsample-srt.bam
```

**2.3 Feature counting:**

```
module load anaconda3
Conda create -n <feature_counts>
```

```
featureCounts -a Vibrio_vulnificus_cmcp6.GCA_000039765.1.23.gtf -o counts.txt
CMCP6_1ASWsample-srt.bam CMCP6_1HSsample-srt.bam CMCP6_2ASWsample-srt.bam
CMCP6_2HSsample-srt.bam
```

```
Load annotation file Vibrio_vulnificus_cmcp6.GCA_000039765.1.23.gtf ... ||

        ||  Features : 4572
        ||  Meta-features : 4572
        ||  Chromosomes/contigs : 2

        ||  Process BAM file CMCP6_1ASWsample-srt.bam...
        ||  Single-end reads are included.
        ||  Assign alignments to features...
        ||  Total alignments : 11242150
        ||  Successfully assigned alignments : 8346569 (74.2%)
        ||  Running time : 0.18 minutes

        ||  Process BAM file CMCP6_1HSsample-srt.bam...
        ||  Single-end reads are included.
        ||  Assign alignments to features...
        ||  Total alignments : 8504090
        ||  Successfully assigned alignments : 6225843 (73.2%)
        ||  Running time : 0.19 minutes
        ||  Process BAM file CMCP6_2ASWsample-srt.bam...
        ||  Single-end reads are included.
        ||  Assign alignments to features...
        ||  Total alignments : 9826782
        ||  Successfully assigned alignments : 6862529 (69.8%)
        ||  Running time : 0.16 minutes



        ||  Process BAM file CMCP6_2HSsample-srt.bam...
        ||  Single-end reads are included.
        ||  Assign alignments to features...
        ||  Total alignments : 10641448
        ||  Successfully assigned alignments : 7155286 (67.2%)
        ||  Running time : 0.17 minutes

        ||  Summary of counting results can be found in file "counts.txt.summary"
```

### 2.4 Moving to R for further analysis.

#### 2.4.1 Reading counts data and loading packages in R:

```
countdata<- read.table("CMCP6.counts", header = TRUE, row.names = 1)
library(“ggplot2”)
library(“pheatmap”)
library(“DESeq2”)
library(“calibrate”)
```

#### 2.4.2 Extracting required columns, renaming columns, and converting data frame to the matrix:

```
new_countdata <- countdata[c("output_sort_CMCP6_1_ASW.bam",
"output_sort_CMCP6_2_ASW.bam" ,  "output_sort_CMCP6_1_HS.bam" ,
"output_sort_CMCP6_2_HS.bam" )]
# Renaming column names
colnames(new_countdata) <- c('A','A_1','B','B_1')
# Converting data frame to a matrix
new_countdata<- as.matrix(new_countdata)
```

#### 2.4.3 Generating conditions, data frame, and DEseq data set

```
# generating experimental conditions
condition<-factor(c(rep("A",2), rep("B",2)))
# generating dataframe
colData<- data.frame(row.names = colnames(new_countdata), condition)
# Using new_countdata matrix and colData to generate DESeq data set
dds<- DESeqDataSetFromMatrix(countData = new_countdata, colData = colData, design =
~condition)
```

### 2.4.4 ordering, selecting, converting to DF, and logging:

```
# Order and selecting top 25 means of unnormalized counts
select <-order(rowMeans(counts(dds,normalized=FALSE)),decreasing=TRUE)[1:25]
# converting DESeq data set to a dataframe
df <- as.data.frame(colData(dds))
# log transforming count data
rld<- rlogTransformation(dds, fitType="mean")
```

### 2.5 Generating heatmap, and PCA:

```
# generating heatmap of log transformed count data
pheatmap(assay(rld)[select,], cluster_rows=FALSE, show_rownames=TRUE,
cluster_cols=FALSE, annotation_col=df)
# generating a PCA plot
plotPCA(rld, intgroup = "condition")
# testing for statistically significant genes
dds <- DESeq(dds)
# getting results and ordering them by p-values
res <- results(dds)
res <- res[order(res$padj), ]
res <- res[order(res$padj), ]
# merging res data frame with count data frame
resdata <- merge(as.data.frame(res), as.data.frame(counts(dds, normalized=TRUE)),
by='row.names', sort=FALSE)
names(resdata)[1] <- “Gene” png("DE_pvals.png", 1000, 1000, pointsize=20)
# plotting histogram of p values
hist(res$pvalue, breaks=50, col="grey")
```

### 2.6 writing data to file:

```
write.csv(resdata, file =”difExp_results.csv”)
```

```
# display the distribution of p-values
png("DE_pvals.png", 1000, 1000, pointsize=20)
hist(res$pvalue, breaks=50, col="grey")
```

## 3. Results and discussion:

### 3.1 Feature count:

```
GeneidChr Start End CMCP6_1HSsample-srt.bam

Strand LengthCMCP6_1ASWsample-srt.bam CMCP6_2ASWsample-srt.bam
CMCP6_2HSsample-srt.bam

VV1_0001 I VV1_0002 I VV1_0003 I VV1_0004 I VV1_0005 I VV1_0006 I VV1_0007 I
VV1_0008 I

1 1002 - 1131 2117 - 2180 3844 - 4490 5548 - 5621 5965 - 6115 7197 - 7740 10289 - 10356
10898 -

1002  6636  7257
987   316   376
1665  4757  7625
1059  728   537
345   429   145
1083  3015  666
2550  3896  777
543   2720  558

5556  9981
353   5332
3528  39910
863   617
380   248
3227  1340
4530  1057
2400  668
```

### 3.2 Histogram:

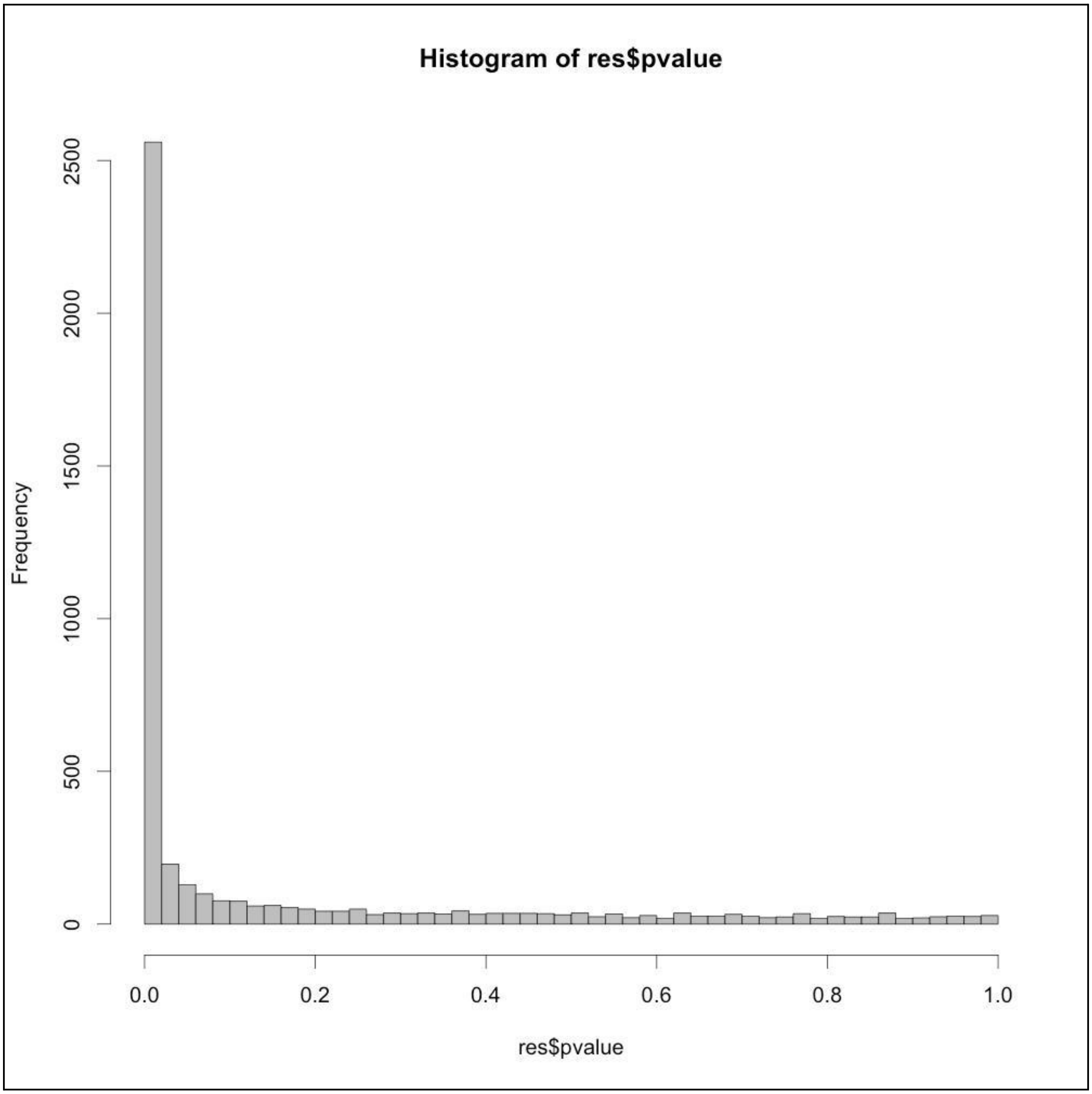

Fig: Histogram

The distribution of the p values was biased to the right, as shown by the plot above. The results (3178 non-statistically significant DE genes vs. 1294 DE genes) are shown in this graphic. The distribution is shifted to the right by the 3178 genes with higher p values.

### 3.3 PCA:

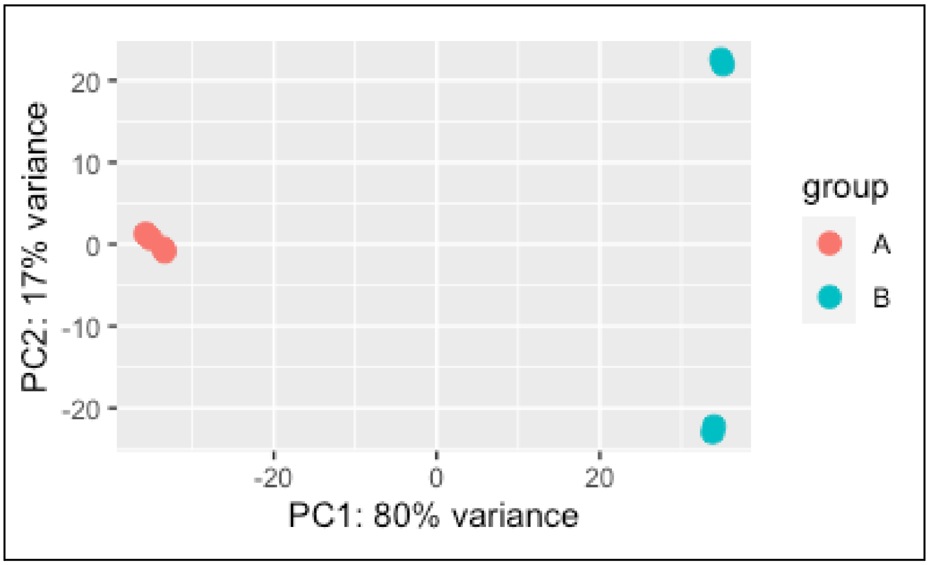

Fig: PCA for PC1 and PC2.

The association between the two replicates is displayed by the PCA plot. While there does not appear to be a clustering of the replicates in group B, there is a low variance clustering of the replicates in group A. However, because the non-statistically significant DE genes were not filtered out, this PCA plot does not give us useful information.

### 3.4 Heatmap:

#### 3.4.1 Heatmap of log transformed DE gene raw counts:

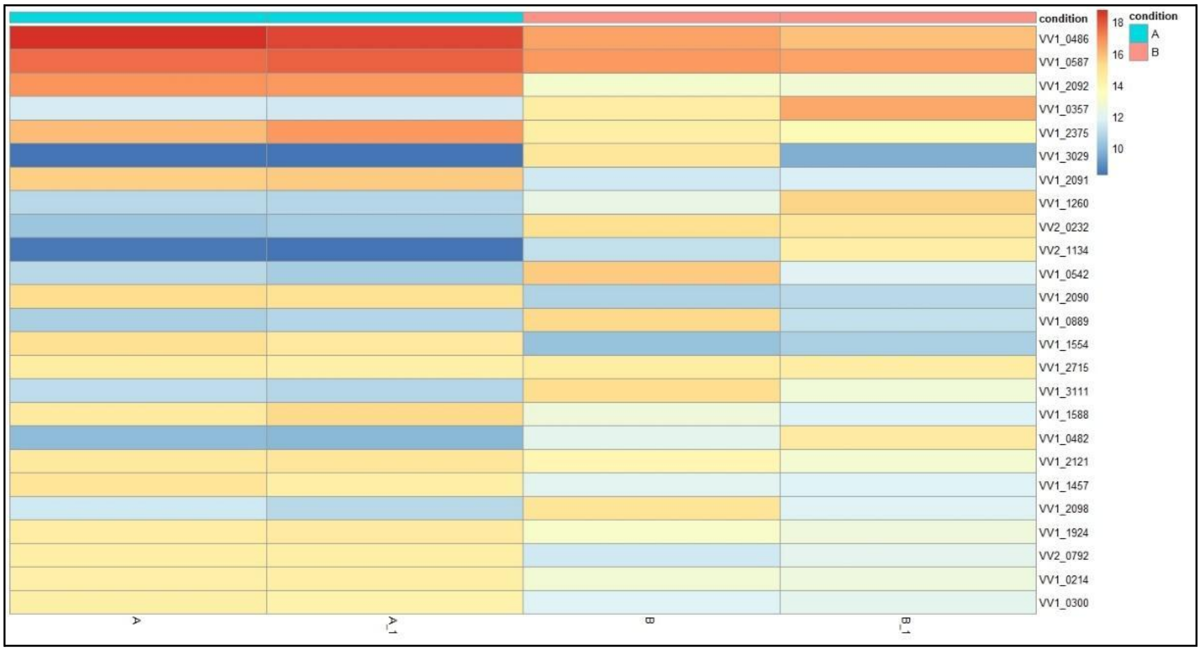

Fig: Heatmap of log transformed DE gene raw counts

This heatmap displays the differentially expressed genes in the repeats under the two circumstances after being log transformed, but it has not been statistically significant. Each gene's level or quantity of expression in the replicates under the two circumstances is indicated by a color key. As the magnitude of the DE genes are not corrected for statistical significance, this figure doesn't offer much information.

### 3.4.2 Heatmap of statistically significant DE genes in the replicates:

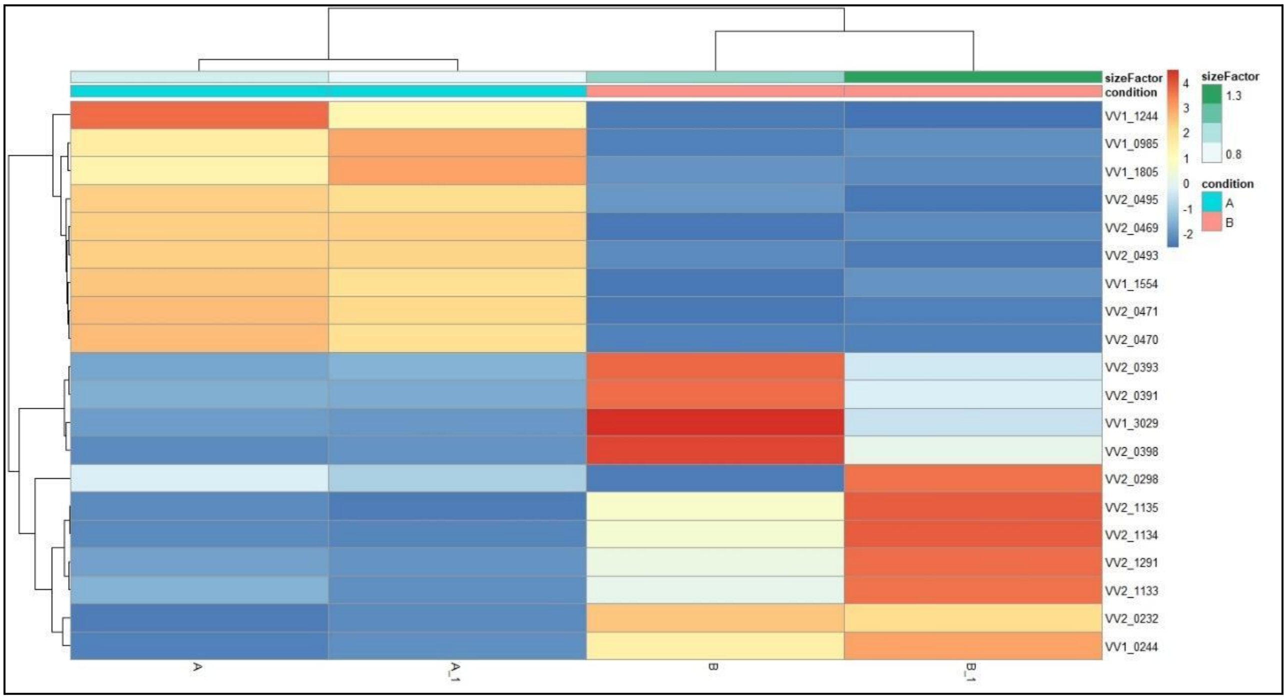

Fig: Heatmap of statistically significant DE genes in the replicates

This heat map of statistically significant differentially expressed genes demonstrates that the replicates under the two experimental conditions have distinct gene expression (upregulation and downregulation). Genes in replicates under the ASW condition are primarily upregulated in the top left quadrant, whereas genes under the HS condition are primarily downregulated in the top right quadrant. In contrast, genes that are elevated in the duplicates under the HS condition but downregulated under the ASW condition are seen in the bottom quadrants. It should be noted that some genes are somewhat downregulated (light blue hue) in replicates (B1) in the lower right quadrant under HS circumstances.