Exercise 1. From Fastq data files to Read Count Matrix

Part 1. Prepare the working directory.

1. Find out the name of the computer that has been reserved for you (https://cbsu.tc.cornell.edu/ww/machines.aspx?i=123).
2. Connect to the computer using Putty (Windows) or Terminal (Mac).
3. From the command line, create a working directory and copy all data files required for this exercise to the working directory. (**Replacing "<my_user_ID>" in the commands with your actual BioHPC user ID**).

```
mkdir /workdir/<my_user_ID>/

cd /workdir/<my_user_ID>/

cp /shared_data/RNAseq/exercise1/* ./

ls -la
```

Part 2. Examine the quality of the fastq data files

1. Run fastqc on the fastq file

```
fastqc ERR458493.fastq.gz
```

2. The fastqc software would create a new file called "ERR458493_fastqc.html". You can use FileZilla to download the file to your laptop, and double click the file to check the results.

- The data used for this exercise are from this article: Schurch et al. (2016) *RNA* 22(6):839 PMID: PMID: 27022035

Part 3. Run read mapping software

We are going to use STAR to map the sequencing reads in the fastq files to the reference genome. STAR is a fast alignment software, but it requires a computer with large memory (30GB for the 3GB human genome).

1. Inspect the files in the working directory (/workdir/my_user_ID. If you are not in working directory already, type "cd /workdir/usre_user_ID" first)

```
ls -la
```

Description of the files in the directory.

R64.fa	Reference genome sequence file, in fasta format.
R64.gtf	Genome annotation file, in gtf format.
ERR458493.fastq.gz	RNA-seq data file, wt_sample1
ERR458494.fastq.gz	RNA-seq data file, wt_sample2
ERR458495.fastq.gz	RNA-seq data file, wt_sample3
ERR458500.fastq.gz	RNA-seq data file, mu_sample1
ERR458501.fastq.gz	RNA-seq data file, mu_sample2
ERR458502.fastq.gz	RNA-seq data file, mu_sample3

If you are interested in finding out what are in the files, or number of reads in the fastq files, use the following commands to examine the files.

```
gunzip -c ERR458493.fastq.gz | head

gunzip -c ERR458493.fastq.gz | wc -l

less R64.fa

less R64.gtf
```

- When inspecting files with "less" command, press "space" key to move on to the next page, or press "q" key to exit.
- "wc -l" is the command to count the number of lines in a file. The command "gunzip -c ERR458493.fastq.gz | wc -l" would tell you the number of lines in the file. As every sequence read takes up 4 lines in the fastq file, the line number divided by 4 gives you the number of sequencing reads in the file.

2. Map the reads to reference genome using STAR.

On the BioHPC computers, STAR is installed in the directory "/programs/STAR". The "export PATH=/programs/STAR:$PATH" command would put STAR in your current path. Now you can run the software by simply typing the command "STAR".

```
export PATH=/programs/STAR:$PATH
```

Then index the reference genome with STAR:

```
mkdir genome

STAR --runMode genomeGenerate --runThreadN 2 --genomeDir genome \

--genomeFastaFiles R64.fa --sjdbGTFfile R64.gtf  \

--sjdbOverhang 50
```

The parameters:

--runMode genomeGenerate: Set runMode to "genomeGenerate" to index the genome;

--runThreadN: Number of CPU cores;

--genomeDir: Output directory for indexed genome database;

--genomeFastaFiles: Reference genome file

--sjdbGTFfile: Genome annotation file and it should be GTF format.

--sjdbOverhang: Use the value (reads_length -1), the read length is 51 for this exercise.

In the next step, we will align sequencing reads to the indexed genome.

```
STAR --quantMode GeneCounts --genomeDir genome --runThreadN 2 \

--readFilesIn ERR458493.fastq.gz --readFilesCommand zcat \

--outFileNamePrefix wt1_ --outFilterMultimapNmax 1 --outFilterMismatchNmax 2 \

--outSAMtype BAM SortedByCoordinate
```

--quantMode GeneCounts: Output a file with read counts per gene;

--genomeDir: Reference genome index directory;

--runThreadN: Number of CPU cores;

--readFilesIn: Sequence data file;

--readFilesCommand zcat: Input file is a decompressed .gz file;

--outFileNamePrefix: Prefix of the output file names;

--outFilterMismatchNmax 2 : Only report alignment with up to 2 mismatches per read;

`--outSAMtype BAM SortedByCoordinate: Output sorted bam files.`

After running STAR software, many new files will be produced. The files you need to keep are:

1) wt1_Aligned.sortedByCoord.out.bam: BAM file with the alignment results;
2) wt1_Log.final.out: A report file that shows percentage of read can be mapped;
3) wt1_ReadsPerGene.out.tab: a tab delimited text file with read count per gene.

Inspect the files:

```
less wt1_Log.final.out

less wt1_ReadsPerGene.out.tab
```

In the file wt1_ReadsPerGene.out.tab, there are three numbers for each gene. The four columns are,

- *column 1: gene ID*
- *column 2: counts for unstranded RNA-seq*
- *column 3: counts for reads aligned with plus strand of RNA*
- *column 4: counts for reads aligned with minus strand of RNA*

Use column 2 if you use unstranded RNA-seq library prep kit. Use column 4 if you use stranded RNA-seq. Use column 3 if you do 3' RNA-seq. For this exercise, we will use column 2 (unstranded).

Part 4. Visualize the BAM file with IGV

1. Index the bam files
   We are going to use the IGV software to visualize the BAM files. For IGV to read the BAM files, the ".bam" files need to be indexed. We will use the samtools software:

```
samtools index wt1_Aligned.sortedByCoord.out.bam
```

After this step, you will see a ".bai" file created for each ".bam" file.

2. Using FILEZILLA to download the “*.bam” ,“*.bai”, “R64.fa” and “R64.gtf” files to your laptop computer.

3. IGV is a JAVA software that can be run on Windows, MAC or a Linux computer. To launch IGV on your laptop, go to IGV web site (https://software.broadinstitute.org/software/igv/), click “Download”, and download the Windows or Mac version for your laptop. Double click the IGV installation tool to install IGV. On Windows computer, the software is installed in the directory C:\Program Files\IGV_2.6.3. Double click “igv.bat” to start IGV. After double clicking, it might take a few seconds before you see the software starting.

4. Most commonly used genomes are already loaded in IGV. In this exercise, we will create our own genome database. Click “Genomes”->”Create .genome” file. Fill out the following fields:

   Unique identifier: R64
   Descript name: R64
   Fasta: use the “Browse” button to find the R64.fa file
   Gene file: use the “Browse” button to find the R64.gtf file

   Then save the genome database on your computer.

5. From menu “File” -> “Load file”, open the “`wt1_Aligned.sortedByCoord.out.bam`”.
   Inspect the following regions by enter the text in the box next to “Go” and click “Go”.
   II:265,593-282,726

Part 5. Run the commands in a shell script

In a typical RNA-seq experiment, you have many samples and it could take many hours to finish the alignments. There are two things you can do to make computing faster.

1. Create a batch command ("a shell script") to process all files;
2. Use the “Shared Memory” feature of STAR. (We do not use it in workshop, I will explain it at the end of this note.)

In order to do this, you can use a text editor to make a text file with the following lines. We recommend Mac users to use “BBEdit” (The free version is fine).
(https://www.barebones.com/products/bbedit/), Windows users can use “Notepad++”
(http://notepad-plus-plus.org/). You can give the script a name, normally with the extension

"sh", e.g."runSTAR.sh". If the file is made on a Windows computer, you need to make sure to save the file as a LINUX style text file. From NotePad++, used the "Edit -> EOL Conversion -> UNIX" option. If you are not sure about this, after uploading the script to Linux, run command "dos2unit runSTAR.sh" to convert to LINUX text file.

You can use FileZilla(win & mac) to upload the file to your home directory. To make things easier, both software include a function to directly save edited file to a remote LINUX machine. Here are the lines in your shell script.

You can also use the shell script that we have prepared for you. It is located in the data directory with the file name "runSTAR.sh";

```
STAR --quantMode GeneCounts --genomeDir genome --runThreadN 2 --readFilesIn
ERR458493.fastq.gz --readFilesCommand zcat --outFileNamePrefix wt1_ --
outFilterMultimapNmax 1 --outFilterMismatchNmax 2 --outSAMtype BAM SortedByCoordinate

STAR --quantMode GeneCounts --genomeDir genome --runThreadN 2 --readFilesIn
ERR458494.fastq.gz --readFilesCommand zcat --outFileNamePrefix wt2_ --
outFilterMultimapNmax 1 --outFilterMismatchNmax 2 --outSAMtype BAM SortedByCoordinate

STAR --quantMode GeneCounts --genomeDir genome --runThreadN 2 --readFilesIn
ERR458495.fastq.gz --readFilesCommand zcat --outFileNamePrefix wt3_ --
outFilterMultimapNmax 1 --outFilterMismatchNmax 2 --outSAMtype BAM SortedByCoordinate

STAR --quantMode GeneCounts --genomeDir genome --runThreadN 2 --readFilesIn
ERR458500.fastq.gz --readFilesCommand zcat --outFileNamePrefix mu1_ --
outFilterMultimapNmax 1 --outFilterMismatchNmax 2 --outSAMtype BAM SortedByCoordinate

STAR --quantMode GeneCounts --genomeDir genome --runThreadN 2 --readFilesIn
ERR458501.fastq.gz --readFilesCommand zcat --outFileNamePrefix mu2_ --
outFilterMultimapNmax 1 --outFilterMismatchNmax 2 --outSAMtype BAM SortedByCoordinate

STAR --quantMode GeneCounts --genomeDir genome --runThreadN 2 --readFilesIn
ERR458502.fastq.gz --readFilesCommand zcat --outFileNamePrefix mu3_ --
outFilterMultimapNmax 1 --outFilterMismatchNmax 2 --outSAMtype BAM SortedByCoordinate
```

- In these commands, I set --runThreadN to 2. You would want to increase the number in real work.
- You might want to run multiple jobs in parallel. Read the instructions at https://biohpc.cornell.edu/lab/doc/using_BioHPC_CPUs.pdf for using BioHPC computer efficiently, or get help form our office hours.

To run the shell script, start "screen", and in a screen session run these commands:

```
cd /workdir/<your_User_ID>

export PATH=/programs/STAR:$PATH

sh runSTAR.sh >& log &
```

After the run starts, detach from "screen" by pressing "Ctrl-a" "d". Use the command "top" to check the whether the job is still running.

Alternatively, especially when you are analyzing your own data, more likely you would use STAR to process multiple samples simultaneously. On BioHPC, we recommend to use a script called "perl_fork_univ.pl". As each STAR job would use several CPU cores and significant amount of memory, make sure they would not exceed the total CPU cores and amount of RAM on the computer. The following command would produce the same results as the previous one, but as it runs 2 jobs at a time, it would be twice as fast.

```
cd /workdir/<your_User_ID>

export PATH=/programs/STAR:$PATH

perl_fork_univ.pl  runSTAR.sh 2 >& log &
```

After running the shell script, you will get 6 files read count files, with one file per sample (*_ReadsPerGene.out.tab). Now you will need to combine the 6 files into one single file for statistical analysis. You can use Excel to do this, and then save the merged file as a tab-delimited text file. Or you can use the following commands:

```
paste wt1_ReadsPerGene.out.tab wt2_ReadsPerGene.out.tab wt3_ReadsPerGene.out.tab
mu1_ReadsPerGene.out.tab mu2_ReadsPerGene.out.tab mu3_ReadsPerGene.out.tab | \

cut -f1,2,6,10,14,18,22 | \

tail -n +5 > gene_count.txt
```

paste: Merge the 5 files side by side;
`cut -f1,2,6,10,14,18,22`: Extract columns 1,2,6,10,14,18,22 from the merged data (column 1 is the gene name, columns 2-22 are second column from each individual file);
tail -n +5: Discard the first 4 lines of statistics summary and start from line 5;
>gene_count.txt: Write the result into a file gene_count.txt

You can open the gene_count.txt file in Excel.

Part 6. Load the matrix into R and make PCA Plot with DESeq2

In the exercise data directory, there is a file named “samples.txt”. It is tab-delimited text file, you can inspect this file with “less samples.txt”. When you work with your own data, you can create this file with Excel and save as a tab-delimited text file.

In this workshop, we will use the BioHPC computer to do this step. You can also install R and DESeq2 module on your laptop to do this exercise.

The default R on BioHPC computers does not work for DESeq2 due to its parallel BLAS library. You will need to start R with “/programs/R-3.5.0s/bin/R”.

You will need to use X-windows to see the plot (Instructions of using X-windows on BioHPC: https://biohpc.cornell.edu/lab/userguide.aspx?a=access)

```
cd /workdir/<your_User_ID>

/programs/R-3.5.0s/bin/R

library(DESeq2)

library(ggplot2)

cts <- as.matrix(read.csv("gene_count.txt", sep="\t", row.names=1, header=FALSE))

coldata <- read.csv("samples.txt", sep="\t", row.names=1)

colnames(cts) <- rownames(coldata)

dds <- DESeqDataSetFromMatrix(countData = cts,

                    colData = coldata,

                    design = ~ Genotype)

vsd <- vst(dds, blind=FALSE)

pcaData <- plotPCA(vsd, intgroup=c("Genotype"), returnData=TRUE)

percentVar <- round(100 * attr(pcaData, "percentVar"))

ggplot(pcaData, aes(PC1, PC2, color=Genotype)) +

 geom_point(size=3) +

 xlim(-2.5, 2.5) +

 ylim(-1, 1) +

 xlab(paste0("PC1: ",percentVar[1],"% variance")) +

 ylab(paste0("PC2: ",percentVar[2],"% variance")) +

 geom_text(aes(label=name),vjust=2)

ggsave("myplot.png")
```

Use the "shared memory" feature of STAR

The first step of running STAR is to load the genome database into memory. There are two issues here:

1. Each job would take several minutes to load the same genome database into memory;
2. Each job would use significant amount of memory to keep its own copy of the genome database;

STAR provides a feature, which allows you to pre-load the genome database into the shared memory space, which can be used by all STAR alignment jobs.

Here are the steps:

1. Load genome into database and keep it there.

```
STAR --genomeLoad LoadAndExit --genomeDir genome
```

2. Make a shell script with STAR alignment commands as you did in step 5. Add these two parameters into each STAR command: "--genomeLoad LoadAndKeep --limitBAMsortRAM 4000000000" . The genomeLoad instruct STAR to use shared memory, and limitBAMsortRAM to instruct STAR to limit 4GB for bam sorting step. You can decrease or increase the sorting memory based on the computer you are using. Now you can run multiple jobs of STAR with "perl_fork_univ.pl" script, and each job will use the same shared memory.

3. After you are done, make sure you remove the genome database from the shared memory. Otherwise it will stay there.

```
STAR --genomeLoad Remove --genomeDir genome
```