Exercise 1. RNA-seq alignment and quantification

Part 1. Prepare the working directory.

1. Connect to your assigned computer. If you do not know how, follow the instruction at http://cbsu.tc.cornell.edu/lab/doc/Remote_access.pdf (Read the section under "Connection by ssh". There are separate instructions for Windows and Mac users)
2. Use the following commands to create a working directory, and copy Arabidopsis genome FASTA file (TAIR10.fa) and rice annotation file (TAIR10.gff3) for this homework to the working directory (replace "xxxxxx" with your user ID)

```
mkdir /workdir/xxxxx

mkdir /workdir/xxxxx/exercise1

cd /workdir/xxxxx/exercise1

cp /shared_data/RNAseq/exercise1/* ./

ls
```

Part 2. Examine qualities of the RNA-seq data files

1. Run fastqc on the fastq file

```
fastqc MUa.fastq.gz
```

2. The fastqc software would create a new file called "MUa_fastqc.html". You can download this file to your laptop. To do this, you need the software called FileZilla.
   Instruction to use FileZilla within Cornell campus:
   Host name: cbsuzzzzz.biohpc.cornell.edu (cbsuzzzzz is the name of your assigned computer)
   UserName and Password: your user ID and password
   Port: 22
   After click "Quickconnect", the left panel show files in your laptop, the right panel show files in the remote BioHPC computer. Next to "Remote site" on top of the right panel, enter "/workdir/xxxxx/" and press "return". You will see the "MUa_fastqc.html" file and drag it into the left panel.

   Instruction to use FileZilla outside Cornell campus:
   User VPN.

3. On your laptop by double clicking the file "MUa_fastqc.html" to open the page.

Part 3. Index the genome file for alignment with STAR

We are going to use STAR to align RNA-seq reads to the genome. First, we will need to index the reference genome.

Add STAR to the current path, so that you can run "STAR" without full path.

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

Index the reference genome.

```
mkdir genomedb

STAR --runMode genomeGenerate --runThreadN 2 \

--genomeDir genomedb \

--genomeFastaFiles genome.fa --sjdbGTFfile genome.gtf --sjdbOverhang 49 \

>& mystarlog &
```

- This process might take 20 minutes. We recommend to run this in "screen".
- Set --sjdbOverhang to "readLength -1", which is 49 in this example.
- If possible, use gtf instead of gff3, or convert gff3 to gtf file first.
- In this example, we set --runThreadN 2. In real work, you should increase the number to all cores of the computer.
- Sometime, you might want to terminate a program that is running in the background (e.g. you realized that you set the parameter wrong, need to restart the job). First use "top" or "ps -u my_user_ID" to find out the process ID (PID) of your program, it should be an integer number. Then use the command "kill PID" to terminate the program. Make sure you use "top" command again to confirm that the program is actually killed.
- If you need to terminate a program that is NOT running in the background, press "Ctrl-C" to stop it. If this does not work, you can open ssh connection in a new terminal window, and use "kill PID" to kill a program that is running.
- When you use a shell script to run a batch of commands, you will need to use "kill PID" to kill both the PID of the shell script, and the any of the commands that are still running.

Part 4. Run STAR to align reads and count number of reads per gene.

1. We have files from 4 RNA-seq libraries. Running STAR could take up to an hour. We will prepare a shell script to process all 4 fastq files. Create a text file with a Text editor (We recommend Text Wrangler for mac users, Notepad+ or Edit+ for Windows users). Using the content in the black box below. Name the file rnaseq.sh. Upload the file to your home directory.
   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.

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

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

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

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

Now, run the shell script:

```
cd /workdir/xxxxx/exercise1

sh ~/rnaseq.sh >& runlog &
```

- In Linux, "~" is a shortcut for your home directory. Use "~/rnaseq.sh" if your script is located in the home directory. If you keep the script in the working directory, then use "./rnaseq.sh".
- This process would take up to an hour. You need to use "nohup" to run the shell script.

Use "top" command to check whether the job is finished or not. If the job is finished, you should not see any jobs running under your user name. You should see "cuffdiffout" directory should have been corrected.

After running STAR, you will find many new files are created. You need to keep the following files for each sample:

- *Aligned.sortedByCoord.out.bam: Alignment file.
- *Log.final.out: Alignment statistics. This file tells you what % of reads can be mapped. You will need this information for QC and for publication.
- *ReadsPerGene.out.tab: read count per gene.

We will need the ReadsPerGene.out.tab file for next step to identify differentially expressed genes. It is a tab delimited text file. The columns are:

column 1: gene ID;

column 2: counts for unstranded RNA-seq;

column 3: counts for reads from forward strand of RNA;

column 4: counts for reads from reverse strand of RNA (we will use number of the column);

As this data set is strand specific, we will use the values in column 4 as read count per gene.

Part 5. Integrate read count from all 4 samples into one table.

STAR would produce one gene count file per sample. Here we will use Linux tools "paste" and "cut" to combine them into one single table. If you do not feel comfortable with the Linux

```
paste WTaReadsPerGene.out.tab WTbReadsPerGene.out.tab MUaReadsPerGene.out.tab
MUbReadsPerGene.out.tab | \

cut -f1,4,8,12,16 | \

tail -n +5 > tmpfile

cat tmpfile | sed "s/^gene://" >gene_count.txt
```

paste: merge files by columns;
cut: extract columns in the merged file;
tail -n +5: discard the first 4 lines of statistics summary and start from line 5;
`sed "s/^gene://"`: remove "gene:" from each line;
`>gene_count.txt`: Write the result into a file gene_count.txt

Part 6. Generate MDS plot of the libraries.

MDS plot can be used to evaluate the variance between biological replicates, and identify sample outliers and mislabeled samples. We are going to use EdgeR package to generate the plot.

1. Type "R" and press return. Now, you are in R console. Use the following steps to generate a PDF file with MDS plot of the samples.

```
library("edgeR")

x <- read.delim("gene_count.txt", header=F, row.names=1)

colnames(x)<-c("WTa","WTb","MUa","MUb")

group <- factor(c(1,1,2,2))

y <- DGEList(counts=x,group=group)

y <- calcNormFactors(y)

pdf("myplot.pdf")

plotMDS(y, col=c(rep("red",2), rep("blue", 2)))

dev.off()

quit()
```

Part 7. Use IGV to visualize the read alignment

We will use bam files produced by STAR to visualize the alignment results.

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 WTaAligned.sortedByCoord.out.bam

samtools index MUaAligned.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", "genome.fa", "genome.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 (http://www.broadinstitute.org/igv/), and download the Windows package or Mac app. You will need to register with your email address for the first. Double click the IGV file to start IGV. (After you double click the file, it might take a minute for IGV to start.) If it complains that you do not have JAVA on your computer, go to https://www.java.com/en/ to get JAVA.

4. Most commonly used genomes are already in IGV. For this genome, we will need to create our own genome database. Click "Genomes"->"Create .genome" file. Fill out the following fields:

   Unique identifier: testgenome
   Descript name: testgenome
   Fasta: use the "Browse" button to find the genome.fa file
   Gene file: use the "Browse" button to find the genome.gtf file

   Then save the genome database on your computer.

5. From menu "File" -> "Load file", open the
   "`WTaAligned.sortedByCoord.out.bam`" and
   "`MUaAligned.sortedByCoord.out.bam`".
   Inspect the following regions by enter the text in the box next to "Go" and click "Go".

   5:26,964,129-26,973,009

Part 8. Parallelization of jobs

If you have many RNA-seq data samples, you can re-do part 4 with this command:

```
cd /workdir/my_user_ID/exercise1

perl_fork_univ.pl ~/rnaseq.sh 2 >& runlog &
```

In this example, the PERL script perl_fork_univ.pl (developed by BioHPC team) would take jobs from the file ~/rnaseq.sh, run two jobs at a time, until all jobs are finished.