## Next Generation Sequence Analysis Homework Week 8

Week 8 serves as the first of three weeks on differential gene expression (DGE) with RNA-seq. In this exercise, you will align RNA-seq reads to a reference genome assembly.

**About the data**

The RNA-seq data are from date palm fruit. In this experiment, researchers wished to test for differential gene expression between varieties of date palm with high fruit sucrose content (n=4) versus those with trace amounts of sucrose (n=4). The goal was to determine if a group of linked invertase enzymes identified by Genome Wide Association Study (GWAS) showed DGE between varieties with the two sugar phenotypes.

The RNA-seq data in this experiment were generated on a NextSeq sequencer, processed to exclude reads that failed Illumina's quality control filter, and then adapters removed with Trimmomatic.

**Completing your assignment**

The preferred way to submit your assignments is to submit an .html file produced by RMarkdown/Markdown and occasionally other files as requested. Your code should be embedded in readable code chunks.

Where possible please upload the answers to questions as a single markdown .html (preferred) report. Other formats are discouraged but .txt, .pdf, and .docx are accepted. Occasionally you may also submit screenshots or additional image files.

Please always include your name in the filename AND at the top of the document.

Upload this weeks report to the Assignments link in NYU Classes.

**Task 1: Align RNA-seq reads to the reference genome**

For this task you will align and coordinate sort RNA-seq reads to a reference genome assembly with the STAR aligner. You will then create a BAM index file for each sample.

The desired output for this task is a coordinate-sorted BAM alignment and BAM index file for each of the eight date palm fruit RNA-seq samples.

STAR documentation is here here:

https://github.com/alexdobin/STAR/blob/master/doc/STARmanual.pdf

For a quick intro, begin by reading 'Section 1.2 Basic Worflow' to get the main idea.

The eight samples you will align and their fastq file names are in the following tab-delimited file:

`/scratch/work/courses/BI7653/hw8.2022/fastqs.txt`

This is a tab-delimited file with sample, forward fastq file and reverse fastq file name. Note that columns two and three have the file names not the full paths to the files.

The fastqs are located in:

`/scratch/work/courses/BI7653/hw8.2022/fastqs`

Your instructor has created the index files for STAR using the genomeGenerate command in the STAR package. The index files are located here:

`/scratch/work/courses/BI7653/hw8.2022/STAR.genome`

The reference fasta used to create the index files is:

`/scratch/work/courses/BI7653/hw8.2022/Pdac_Barhee_chr_unan_cp_180126.fa`

Before proceeding, log in to Greene and create a directory ngs.week8 in your `/scratch` directory. Then, use the instructions below to create an array job script to align RNA-seq reads.

To prepare your array job script, you must review the STAR documentation section 3 "Running mapping jobs", section 4 for how to specify output directory and file name prefix for output files, and section 4.3 for how to output a coordinate-sorted BAM in the STAR documentation.

STAR, like many softwares in NGS, has a few quirks that require reading documentation carefully. To save you some time, consider the following when constructing your array job script.

1. Use the slurm directive `#SBATCH --array 1-8` to run array job on 8 samples and their fastqs
2. Consider parsing the table with fastq file names using the approach used in Week 2 Task 2 script provided to you by your instructor. See syntax in `/scratch/work/courses/BI7653/hw2.2022/wk2_task2.sh`. The idea is to parse the 3-column table and define variables with sample name, forward fastq, and reverse fastq in `/scratch/work/courses/BI7653/hw8.2022/fastqs.txt` in your array job script.
3. Consider creating a separate directory for each sample with mkdir in your array job script. If you create a directory in the script, you may then `cd` into the directory before executing the STAR aligner on the current sample in your array job. This is not necessary, but can help prevent unwanted file name collisions. Even when running each job array index in a different directory, it is always preferable to give output files a unique name.
4. To give each output a different name, consider defining the sample name in a variable called `sample` (see example in wk2_task2.sh) in your array job script and then include option `--outFileNamePrefix $sample` in your STAR command line. Note that STAR is a bit unconventional in that unfortunately does not provide full control over output file names. However, this should prepend the sample name to the default output alignment file name. See section '5 Output files' of documentation for details.
5. Include a `–runThreadN` argument to your STAR command. Choose the number of threads (between 1 and 4) and consider using `$SLURM_CPUS_PER_TASK` as the argument to make sure the number of threads matches the number available to requested by your job script.
6. Include in your STAR alignment command the option to output a coordinate-sorted BAM (see section 4.3 of documentation). Otherwise, you would have to run a separate tool to coordinate sort each BAM after the alignment is completed.
7. STAR does not work natively with g-zipped (compressed) fastq files. To get STAR to work with compressed files, add `--readFilesCommand zcat` to your STAR command line, where `zcat` is a binary/executable in BASH that uncompresses a g-zipped file and outputs text to STDOUT (similar to `gunzip -c`). Effectively, STAR will use the BASH `zcat` command to uncompress the files.
8. Request 16 GB of RAM (coordinate-sorting is memory intensive) and include the –limitBAMsortRAM 20000000000 argument in your command.
9. You do not normally need to worry about temporary directories when running NGS software, but your instructor found the documentation for STAR to be ambiguous on the `--outTmpDir` default (see section 16.12 at documentation listed above). Therefore, lets force the temporary directory to be unique for each STAR alignment command by adding the argument `--outTmpDir ${SLURM_TMPDIR}/${SLURM_ARRAY_JOB_ID}_${SLURM_ARRAY_TASK_ID}` to your STAR command. This specifies a local temporary directory on the compute node

where the slurm job executed (stored in `SLURM_TMPDIR` environmental variable) and will instruct STAR to create a unique directory for the array job and task identifier inside that directory.

10. Given that each output BAM is coordinate sorted, we can create a BAM index for each file in the same script that we align the RNA-seq reads. Given that the output BAMs will have file name ${sample}Aligned.sortedByCoord.out.bam, add an appropriate samtools command to create an index file in your array job script after the STAR alignment command.

11. Make sure to load the most recent most recent STAR and samtools modules in your job array script.

When you are ready execute your script with sbatch.

Q1.1. Report the contents of your array job script and the job id on Greene [ 3 points ].

Q1.2 Review the file “Log.final.out” for sample PDAC253 and report the following [ 1 point ]:

1. The number of uniquely mapped reads
2. The percentage of uniquely mapped reads
3. The total number of input reads

Q1.3 It is common to only work with uniquely mapped reads and not include multiply-mapped reads in downstream analysis. For Q1.3a and Q1.3b you need to review the STAR documentation [ 1 point ]

Q1.3a What is the mapping quality used by STAR by default to indicate uniquely mapped reads?

Q1.3b If you want to make sure STAR output only uniquely mapped reads, how might you do this (see documentation)?

Q1.4 The number and percentage of reads mapped to too many loci is very high for PDAC253 sample Provide a hypothesis for this observation and how you might go about evaluating it. [ 1 point ]

Q1.5. Report the first 20 lines of the header for one output BAM (using samtools view). Then answer is your BAM coordinate-sorted? Please include your samtools view command in your answer for full credit. [ 1 point ]?

Q1.6. Recall from the SAM format specification that mapping quality of a read alignment is a PHRED-scaled probability that the read is aligned in error. However, this quantity is difficult to estimate and different alignment tools use different rules/heuristics to assign mapping quality scores to aligned reads. In many cases, these qualities are only loosely correlated with an accurate PHRED score.

Q1.6a What mapping quality scores are present in the alignment for PDAC253 (note: you may need to convert BAM to SAM)? Include either your command line or how you answered this question in your answer [1 point]

Q1.6b. What does each of the observed mapping quality mean? (hint see STAR documentation) [ 1 point ]

Q1.7. Imagine that you are working on a pair of recently duplicated genes and want to independently test for differential gene expression for the duplicated genes with the RNA-seq data in this assignment. Do you think this is possible? What factor(s) should be considered in order to do so? [ 1 point ]

**You are finished, see Completing your assignment section for how instructions to submit your assignment.**