## June 2017

# An Introduction to Deep-Sequencing Data Analysis

## Exercise #2: Using WEXAC for mapping and sequence QC

***Dena Leshkowitz and Ester Feldmesser***

### Introduction

In this workshop we will learn how to evaluate the sequence quality and how to map the reads to a reference genome. The data set in this workshop is a collection of RNA-Seq data from mRNAs extracted from acute lymphoblastic leukemia (ALL) precursor in B cell line.

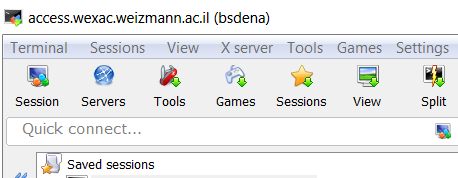
In addition we will start to analyze RNA-Seq data from Arabidopsis downloaded from the public domain <https://www.ncbi.nlm.nih.gov/pubmed/26084880>, in future exercises we will continue with additional steps of the analysis.

We will be using the WEXAC cluster and write Linux commands in a terminal.

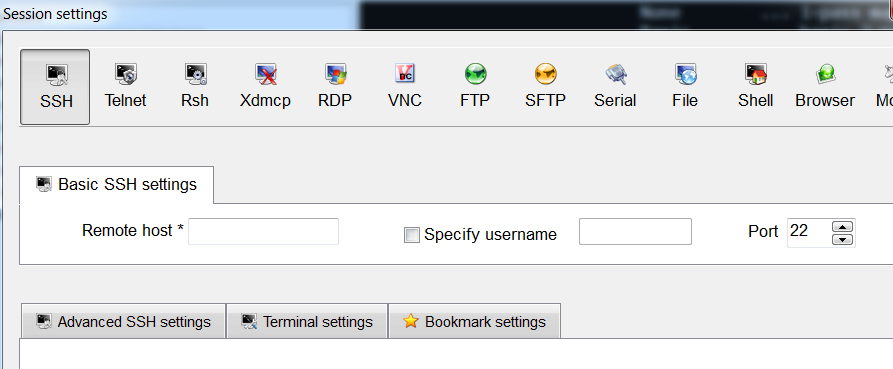
### Instructions

### Accessing the data

* 1. Find the icon MobaXterm on your desktop and click to open.



Click on session (red arrow), then select SSH



You are prompted to insert the host (server) name – *access.wexac.weizmann.ac.il* and the username *– please ask us for your* username and password. Then in the terminal you will be requested to enter your password.

### Running the FASTQC program

### The sequences are in your home directory in the myreads.fastq file.

### To view the file type:

*more myreads.fastq*

* 1. Look at the quality value of the first base from the first sequence. Convert the character into a numeric value using the Supplementary converter below. Our sequences comply with the Sanger convention.

Question 1: What is the probability of an error in the first base?

### To create a QC report for myreads.fastq using the fastqc program, type:

*module load fastqc*

*module load jre*

*mkdir output\_exercise2*

*cd output\_exercise2*

*mkdir output\_fastqc*

*fastqc -o output\_fastqc ~/myreads.fastq*

*ls output\_fastqc/*

*firefox output\_fastqc/myreads\_fastqc.html*

Question 2:

* + - * 1. How many sequences does the fastq file contain?
        2. Is the base quality the same for all the cycles?
        3. Do all the cycles have an equal base content?

The sequences are from a RNA-Seq experiment. During the course we will discuss the reason for the unequal base content at the beginning of the sequences.

1. **Running the Bowtie program and understanding its output**

We are going to run bowtie, let’s find the name of bowtie2 module by listing all available modules on wexac access server.

*module avail*

Load the required module for bowtie2

To see all the modules you loaded type:

*module list*

If you have loaded bowtie2 module you are ready to run bowtie.

To see the manual type:

*bowtie2 -h*

You will see the following:

Bowtie 2 version 2.2.9 by Ben Langmead (langmea@cs.jhu.edu, www.cs.jhu.edu/~langmea)

Usage:

bowtie2 [options]\* -x <bt2-idx> {-1 <m1> -2 <m2> | -U <r>} [-S <sam>]

The -x parameter is the path and name of the genome index to be used by bowtie2.

In our case we should write -x /shareDB/iGenomes/Homo\_sapiens/UCSC/hg38/Sequence/Bowtie2Index/genome

To run Bowtie2 type:

*bsub -q bio-guest -R 'rusage[mem=50000]' -o  ~/output\_exercise2/bowtie\_job.txt  -e ~/output\_exercise2/error\_bowtie.txt -J bow "bowtie2 -x /shareDB/iGenomes/Homo\_sapiens/UCSC/hg38/Sequence/Bowtie2Index/genome -U ~/myreads.fastq -S ~/output\_exercise2/myreads.sam****> log.txt 2>&1****"*

You should get the following message:

Memory reservation is (MB): 50000

Memory Limit is (MB): 50000

Job <888078> is submitted to queue <bio-guest>.

Type:

*bjobs*

You should either see your job at RUN or PEND status or if it is finished you will not see it in this list.

When you job is done type

*ls*

You should have two important file outputs: bowtie\_log.txt and myreads.sam

Type:

*more log.txt*

Question 3:

What percentage of reads was mapped to the genome? Why are there reads that did not map?

To view the alignment information, type:

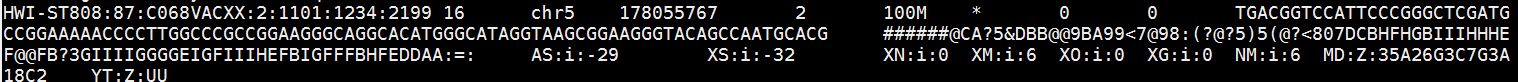
*less myreads.sam*

To understand the SAM format look in:

<https://samtools.github.io/hts-specs/SAMv1.pdf> (page 4) or below in Supplementary.

Click on the enter key until you reach a line that does not start with @, these lines are headers describing chromosome information.

In the lines after the headers, the first field is the read name and the second one is a flag that explains the read mapping status (red arrow).



To help you understand this flag go to the utility below. This utility explains SAM flags in plain English:

<https://broadinstitute.github.io/picard/explain-flags.html>

Looking at the **third read** information this flag is 0.

However this 0 flag is missing from the above picard utility web page. The flag 0 represents a read mapped to the forward strand that is unpaired.

Reads mapped uniquely will appear in one line only, and reads mapped to multiple locations in the genome will appear in multiple lines.

Question 4:

* + - * 1. Observe the first and third mapped reads, to which chromosome and location are they mapped?
        2. Do they have mismatches and if yes what is/are it/they?

Hint: Look for the CIGAR (blue arrow; see CIGAR explanation in Supplementary below)

1. **Running the STAR program**

We will now run the alignments that you will use for tomorrow’s exercise in order to analyze transcriptome (RNA-Seq) data. The aligner in this case is [STAR](https://github.com/alexdobin/STAR/blob/master/doc/STARmanual.pdf), this aligner knows how to map the reads that map to exon-exon junctions.

The genome we will use is Arabidopsis.

The sequence files are located in: ~/course\_2017/Arabidopsis\_RNAseq

You need to perform the following tasks:

* 1. Create the fastqc reports for all sequence files.

The sequence files are found at ~/course\_2017/Arabidopsis\_RNAseq/data

* 1. Run mapping with star according to the directions below.

To view the star command for a sequence file of one sample (SRR1660397.fastq) type:

*more ~/course\_2017/Arabidopsis\_RNAseq/commands/star\_command\_SRR1660397.txt*

Preparing for running star:

*module load star/2.5.2b*

*mkdir ~/RNA-Seq-Arabodopsis-map*

*cd ~/RNA-Seq-Arabodopsis-map*

Running the star command:

*bsub -q bio-guest -R "rusage[mem=50000] span[hosts=1]" -o ~/RNA-Seq-Arabodopsis-map/star\_job\_SRR1660397.txt -e ~/RNA-Seq-Arabodopsis-map/star\_error\_SRR1660397.txt -J st1 -n 10 < ~/course\_2017/Arabidopsis\_RNAseq/commands/star\_command\_SRR1660397.txt*

You should get a message such as the following:

Memory reservation is (MB): 50000

Memory Limit is (MB): 50000

Job <346524> is submitted to queue <bio-guest>.

To see the job running type:

*bjobs*

To list the files produced:

*ls -l*

Now you need to run a similar star command for all the other sequence files. The star command lines that are needed for the mapping are available, type the following to list them:

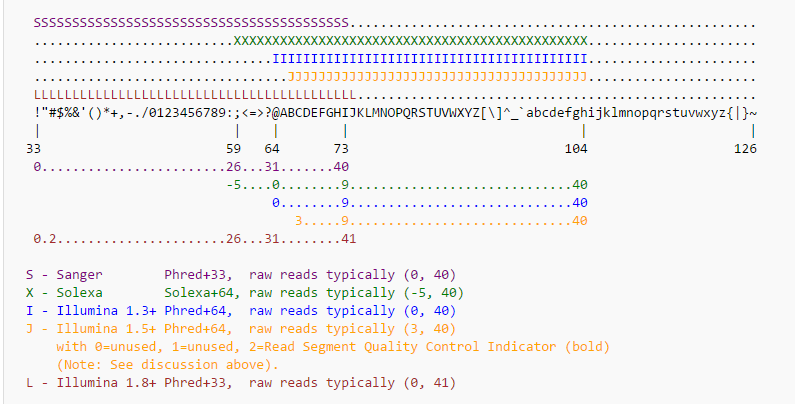
*ls -1 ~/course\_2017/Arabidopsis\_RNAseq/commands/*

You need to change the bsub command (above) in order to run star for all the sequence files.

Good job. You completed this exercise.

## Supplementary

### Converting ASCII Characters to quality values



### SAM file format in short

SAM file is a TAB-delimited version of the BAM which is a binary file. Apart from the header lines, which are started with the ‘@’ symbol, each alignment line consists of:

Column Fields Description

1. QNAME Query template/pair NAME
2. FLAG bitwise FLAG
3. RNAME Reference sequence NAME
4. POS 1-based leftmost POSition/coordinate of clipped sequence
5. MAPQ MAPping Quality (Phred-scaled)
6. CIGAR extended CIGAR string
7. MRNM Mate Reference sequence NaMe (‘=’ if same as RNAME)
8. MPOS 1-based Mate POSistion
9. LEN inferred Template LENgth (insert size)
10. SEQ query SEQuence on the same strand as the reference
11. QUAL query QUALity (ASCII-33 gives the Phred base quality)
12. OPT variable OPTional fields in the format TAG:VTYPE:VALUE

### CIGAR: CIGAR string.

The CIGAR operations are given in the following table:

M alignment match

I insertion to the reference

D deletion from the reference

N skipped region from the reference

S soft clipping (clipped sequences present in SEQ)

H hard clipping (clipped sequences NOT present in SEQ)

P padding (silent deletion from padded reference)

= sequence match

X sequence mismatch