Differential Gene Expression Analysis

Following is an example of project/publication with paired-end read data on NCBI

https://pubmed.ncbi.nlm.nih.gov/27739497/

Data has been downloaded from NCBI's SRA and deposited on logrus. Please make sure you copy over all the relevant files into your workspace ahead of time before you start analysis. Also, if you decide to download all fastq input files from SRA, use fastq-dump (a software) to do the same. Fastq-dump may be time consuming depending on size of file and the number of samples, hence run on screen.

Download data (using fastq-dump) or copy data into your workspace and run sequence data quality check using FASTQC

```
#In order to download data from SRA you will have to use fastq-dump, a tool in bio
#environment. You have to provide SRR #'s associated with each fastq upload on NCBI SRA
#since you have paired-end reads, you have to include --split-files option to split
#as R1 and R2
#run on screen (recommended)

source activate bio

======EXAMPLE OF FAST_DUMP COMMAND==================

fastq-dump.2.10.0 --split-files \
--outdir /home/asundara/paired-end_example/ SRR3194957 SRR3194956 SRR3194955 SRR3194954

#Rename input files to something intuitive instead of having SRR prefixes.
=====================================================

#if you want to copy fastq files from instructor workspace,

cp /home/asundara/paired-end_example/*.fastq /path/to/your/directory/

#Run fastqc on files to perform quality check

source activate bio

#cd into the folder that has all the fastq files
#ALWAYS DO --HELP FOR HELP WITH SOFTWARE USAGE

/home/usrname/practice_workflow/raw_data/

mkdir fastqc_out

fastqc -o fastqc_out/ -f fastq *.fastq
```

To download reference genomes and annotations (In this case, it is a S. aureus genome and annotation file as this was from the publication link pasted above). I found a link to the reference genome from a 2016 ASM genome announcement submission: https://www.ncbi.nlm.nih.gov/pmc/articles/

Genome Assembly: Create a folder to download reference fasta in your workspace.

```
#create a directory called S-aureus_reference in your practice_workflow
#folder to download your reference files

mkdir S-aureus_reference

cd /home/usrname/practice_workflow/S-aureus_reference

wget https://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/001/267/715/\
GCA_001267715.2_ASM126771v2/GCA_001267715.2_ASM126771v2_genomic.fna.gz
```

Genome Annotation: Download the annotation gff file into the same folder as your reference fasta

```
#download gff file associated with your fasta into the same folder.  Once you
#have both files, check the identifiers to make sure they match.

cd /home/usrname/practice_workflow/S-aureus_reference

wget https://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/001/267/715/\
GCA_001267715.2_ASM126771v2/GCA_001267715.2_ASM126771v2_genomic.gff.gz

#at this point ensure you have the fasta and gff file from the same database
```

Generating Alignments using STAR Alignment Program

- Align/Map sequence reads generated for each individual sample to the downloaded genomes.
- Use STAR program to generate the alignments
- Generating alignments is a two step process
  - Step 1: Create/Build a genome index/database.
  - Step 2: Run alignments

```
cd /home/usrname/practice_workflow/S-aureus_reference/

mkdir star_index

cd star_index


/home/asundara/sw/star/STAR-2.7.6a/bin/Linux_x86_64/STAR \
```

```
--runThreadN 4 \
--runMode genomeGenerate \
--genomeDir /home/usrname/practice_workflow/S-aureus_reference/star_index \
--genomeFastaFiles /home/usrname/practice_workflow/S-aureus_reference/\
GCA_001267715.2_ASM126771v2_genomic.fna \
--genomeSAindexNbases 9
```

* For paired-end reads
  - To run one sample

```
cd /home/usrname/practice_workflow/
mkdir STAR_alignments/
cd STAR_alignments/

/home/asundara/sw/star/STAR-2.7.6a/bin/Linux_x86_64/STAR \
--runThreadN 4 \
--genomeDir /home/usrname/practice_workflow/S-aureus_reference/star_index \
--readFilesIn /home/usrname/practice_workflow/raw_data/ATCC_1_R1.fastq \
/home/usrname/practice_workflow/raw_data/ATCC_1_R2.fastq \
--outFileNamePrefix /home/usrname/practice_workflow/STAR_alignments/ATCC_1
```

  - To run multiple samples at once using a for loop on the command line

```
cd /home/usrname/practice_workflow/
mkdir STAR_alignments/
cd STAR_alignments/

for file in ATCC_1 ATCC_2 1679a_1 1679a_2; \
do /home/asundara/sw/star/STAR-2.7.6a/bin/Linux_x86_64/STAR \
--runThreadN 4 \
--genomeDir /home/usrname/practice_workflow/S-aureus_reference/star_index \
--readFilesIn /home/usrname/practice_workflow/raw_data/${file}_R1.fastq \
/home/usrname/practice_workflow/raw_data/${file}_R2.fastq
--outFileNamePrefix /home/usrname/practice_workflow/STAR_alignments/${file}; \
done
```

Generating Alignments using HISAT2 Alignment Program

- Generating alignments is a two step process
  - Step 1: Create/Build a genome index/database.
  - Step 2: Run alignments

```
cd /home/usrname/practice_workflow/S-aureus_reference

source activate bio

#create a folder hisat2_index in your reference folder

mkdir hisat2_index

hisat2-build GCA_001267715.2_ASM126771v2_genomic.fna hisat2_index/\
GCA_001267715.2_ASM126771v2_genomic
```

* For paired-end reads
  - To run one sample at a time

```
#make a directory called hisat2_alignments in /home/usrname/practice_workflow/
#run alignments from where your input files exist

cd /home/usrname/practice_workflow/raw_data

hisat2 \
-x /home/usrname/practice_workflow/S-aureus_reference/hisat2_index/\
GCA_001267715.2_ASM126771v2_genomic \
-p 2 \
-1 /home/usrname/practice_workflow/raw_data/ATCC_1_R1.fastq \
-2 /home/usrname/practice_workflow/raw_data/ATCC_1_R2.fastq \
-S /home/usrname/practice_workflow/hisat2_alignments/ATCC_1.sam
```

  - To run multiple samples at once using a for loop on the command line

```
#run alignments on screen

cd /home/usrname/practice_workflow/raw_data

for sample in ATCC_1 ATCC_2 1679a_1 1679a_2; \
do hisat2 \
-x /home/usrname/practice_workflow/S-aureus_reference/hisat2_index/\
GCA_001267715.2_ASM126771v2_genomic \
-p 2 \
-1 /home/usrname/practice_workflow/raw_data/${file}_R1.fastq \
-2 /home/usrname/practice_workflow/raw_data${file}_R2.fastq \
-S /home/usrname/practice_workflow/hisat2_alignments/${file}.sam; \
done
```

Generating Alignment Metrics - The following commands should work for alignment files (.sam) files generated using both STAR & HISAT2 alignment programs. You should be able to run this using the workflow from class.

Total Reads from SAM file

All alignments (MAPPED/UNMAPPED/MULTIMAPPED)

Total UNIQUE & MULTIMAPPED Reads

Total number uniquely aligning reads

Generating Read Counts using HT-Seq and FeatureCounts. HT-Seq is an alternative read counter in addition to featureCounts used in class. Before you start running a new software, please read the manual or do a –help on the server. HT-seq is installed in another environment so do not source activate bio. Source the environment where this sw has been installed

```
#Create a folder to deposit counts from HTSeq using HiSat2 aligner.
#Run HT-seq command from the folder that contains raw reads.
#Look at the output files.  cat, head, tail, etc
#identify lines that might interfere with downstream analysis and remove them
#ALWAYS RUN SW ON SCREEN
#source the environment for ht-seq
#to learn more about the flags, do htseq-count --help

source /sw7/compbio/htseq/HTSEQ_ENV/bin/activate

cd /home/usrname/practice_workflow/

mkdir HTseq-HISAT2

cd hisat2_alignments


for file in *.sam; \
do \
htseq-count ${file} \
--type=gene \
--idattr=Name \
--order name \
--stranded=no \
/home/asundara/paired-end_example/reference_S-aureus/\
GCA_001267715.2_ASM126771v2_genomic.gff \
> /home/asundara/paired-end_example/htseq_hisat/${file}.HTSEQ.txt; done

#At every step make sure you do some housekeeping to make sure you clean
#up file names as you go.
#You can use a simple rename command to do this.
#Once you have the files, cat, head, tail, less to view contents.
```

```
#Make it a practice to view contents everytime you generate new files.
#Last 10 lines of the output from HT-Seq looks like this,
#ie, when you cat the file and | tail

============
cat ATCC_2.sam.HTSEQ.txt | tail
rpsN    1
rrf     0
ssrA    286
ssrS    4
ureE    3
__no_feature
17037042 __ambiguous
5176 __too_low_aQual
220743 __not_aligned
4258516 __alignment_not_unique  6589
============
#You have to delete the last 5 lines so you can use it as input for DESEQ2.
#Last 5 lines start with double underscore.
#You can do something like this

cat ATCC_2.sam.HTSEQ.txt | grep -v '^--' > ATCC_2.sam.HTSEQ.DESEQ2.txt

#To run a for loop,

for file in *.HTSEQ.txt; do \
cat ${file} \
|grep -v '^__' \
> ${file}.DESEQ2.txt; done
```

```
#Generating read counts using FeatureCounts for HISAT2 Alignments

#you have to sort your sam files by name before you run featureCounts on paired-end data
#Many aligners would have have sorted paired end reads while generating sam files
#You can use a software called samtools if this is nor done for you
#create a file called sorted.sam from your sam files and run
#featureCounts on this sorted sam file

#ONLY RUN THE FOLLOWING STEP IF YOUR SAM FILES ARE NOT SORTED

================================================

#sort sam file as follows using a for loop

source activate bio

for file in *.sam; do samtools sort -n -o ${file}.sorted.sam ${file}; done

#run feature counts on sorted sam files
#to learn more about the flags, do featureCounts --help

================================================

source activate bio
```

```
cd /home/usrname/practice_workflow/
mkdir featurecounts_HISAT2
cd hisat2_alignments/


#Using a for loop on multiple .sam files at once

for file in *.sam; \
do \
echo ${file}; \
featureCounts \
-p \
-a /home/usrname/practice_workflow/S-aureus_reference/GCA_001267715.2_ASM126771v2_genomic.gff \
-o /home/usrname/practice_workflow/featurecounts_HISAT2/${file}.featurecounts.txt \
-T 1 \
-t gene \
-g Name \
${file}; \
done

#Once files are generated, parse the files to only get the appropriate
#columns for downstream step.
#Please refer to your DGE workflow. Example command to clean up file is as given below

for file in *.counts.txt; do cat ${file} | sed '1,2d' | awk '{print $1 "\t" $7}' \
> ${file}.DESEQ2.txt; done
```

You can run any aligner/read counter combination. For instance, you can run STAR for generating alignments and then use either HT-Seq or FeatureCounts for generating read counts. Similarly, you can use HISAT2 for mapping/generating alignments and use either HT-Seq or FeatureCounts for generating read counts. In the example provided above, I have used HISAT2 for generating alignments and used sam files from this software as input for both HT-Seq read counter as well as FeatureCount read counter.

Differential Gene Expression Analysis - You should be able to run this using the workflow from class.