Master RNA-Seq Script

Scott Teresi

Contents

Perform Indexing, Trimming, Alignment, and Mapping for RNA-Seq:

Indexing:

You will only need to perform this step once. The below code creates a genome index, it runs rather fast. I use **STAR** to do this step. Make sure to edit your options in this command to fit your needs (directory structure), especially the `sjdbOverhang` option as it depends on your read length.

```
#!/bin/bash -login
#SBATCH --time=01:00:00
#SBATCH --nodes=1-3
#SBATCH --cpus-per-task=2
#SBATCH --mem-per-cpu=16G
#SBATCH --job-name STAR_Genome_Index_Teresi
#SBATCH -o %j.out
#SBATCH -e %j.err


# ------------------------------
# Load Modules
module purge
module load GCC/7.3.0-2.30
module load OpenMPI/3.1.1
module load Perl
module load Java
module load Trimmomatic
module load STAR

#cd /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Scripts/


# BUILD A GENOME INDEX FOR STAR
STAR --runThreadN 2 \
--runMode genomeGenerate \
--genomeDir /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Output_Files/Index \
```

```
--genomeFastaFiles /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Input_Files/Vvinifera_145_hardmasked
--sjdbGTFfile /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Input_Files/Vvinifera_145_gene_exons.gff3
--sjdbGTFtagExonParentTranscript Parent \
--sjdbOverhang 149
```

Trimming:

Initialize a File List for FastQ Pairs:

The code below is a *shell* script named **file_generator.sh** but you may name it whatever you like. It creates a file list by going through a directory containing paired *fastq.gz* files and writing the pairs to a csv. It uses *ls* and *grep* to get the pairs and it outputs them to **File_List.csv**.

To-Do: Make a looping structure to avoid using the same code multiple times.

```
cd /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Input_Files/Seq/20190809_mRNASeq_PE150/
ls -d "$PWD/"*gz | grep R1 > R1.txt
ls -d "$PWD/"*gz | grep R2 > R2.txt
ls -d *gz | grep R1 > R3.txt # get name
paste -d "," R1.txt R2.txt R3.txt > File_List.csv


cd /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Input_Files/Seq/20190802_mRNASeq_PE150/
ls -d "$PWD/"*gz | grep R1 > R1.txt
ls -d "$PWD/"*gz | grep R2 > R2.txt
ls -d *gz | grep R1 > R3.txt # get name
paste -d "," R1.txt R2.txt R3.txt > File_List.csv


cd /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Input_Files/Seq/20190724_mRNASeq_PE150/
ls -d "$PWD/"*gz | grep R1 > R1.txt
ls -d "$PWD/"*gz | grep R2 > R2.txt
ls -d *gz | grep R1 > R3.txt # get name
paste -d "," R1.txt R2.txt R3.txt > File_List.csv
```

Read File List

I have reproduced the perl code below here for clarity but you will need to have this code in a separate perl file in the working directory. I have named this file **Trim_Bulk.pl**, but you may name it whatever you like, as it is the first step in the pipeline and nothing else refers to it.

This reads the previously created file list and runs the submission commands using the columns of the file list as arguments for the submission commands.

```
use strict;
use warnings;
use Cwd;
# Reads my master pair file and sends each unit in that csv to be submitted
my $file = $ARGV[0] or die "Need to get CSV file on the command line\n";
open(my $data, '<', $file) or die "Could not open '$file' $!\n";
while (my $line = <$data>) {
    chomp $line;
```

```
    my @fields = split "," , $line;
    my $file1 = $fields[0];
    my $file2 = $fields[1];
    $file1 =~ s/\//\\\//g;
    $file2 =~ s/\//\\\//g;

    `cp submit_trim_files_BASE.sh Trim_Submit/submit_trim_files_$fields[2].sh`;
    chdir("/mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Scripts/Trim/Trim_Submit/") or die "Cannot cl
    `perl -pi -e "s/FILE1/$file1/g" submit_trim_files_$fields[2].sh`;
    `perl -pi -e "s/FILE2/$file2/g" submit_trim_files_$fields[2].sh`;
    `perl -pi -e "s/NAME/$fields[2]/g" submit_trim_files_$fields[2].sh`;
    `sbatch submit_trim_files_$fields[2].sh`;
    chdir("../") or die "Cannot change";
}
```

Submission of Pairs Script:

Attention: This script is where you will do most of your modification. I have made this script necessary to edit as it is advantageous to have your code run in chunks, that is chunks of fastq pairs. Below you will have to change the initial **cd** command to match the file list that you supply when running the pipeline, which I will describe at the end of the code presentation.

I have reproduced the below code here for clarity but you will need to have this code in a separate shell file in the working directory. This file should be called **trim_files.sh** It makes the output directories and subdirectories for the pairs. It names the output directory based on the first name of the pair presented. This is done below in the step `$3`.

Attention: You may want to edit the `Pre-Processing Step of Alignment` portion below where it runs the Trim command. Obviously your directory structure may change, but here I am specifically setting the amount of threads, 15, and performing a headcrop of 10, in addition to some other minor settings.

```
#!/bin/bash
# Declare output directory:
#   This is the master output folder
#   star_expression_analysis.sh will make subdirectories in master output
cd /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Output_Files/809_Seq/ # change this line as needed t
# 724_Seq --- directory
# 802_Seq --- directory
# 809_Seq --- directory

mkdir $3
cd $3

mkdir Trimmed
cd Trimmed

# Pre-Processing Step of Alignment
java -jar /mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Trimmomatic/Trimmomatic-0.39/trimmomatic-0.39
```

Submit the Pairs From the File List:

I have reproduced the below code here for clarity but you will need to have this code in a separate shell file in the working directory. This file should be called **submit_trim_files_BASE.sh**.

```
#!/bin/bash -login
#SBATCH --time=02:30:00
#SBATCH --cpus-per-task=15
#SBATCH --mem-per-cpu=20G
#SBATCH --job-name Trimmomatic_Teresi
#SBATCH -o %j.out
#SBATCH -e %j.err


# -------------------------------
# Load Modules
module purge
module load GCC/7.3.0-2.30
module load OpenMPI/3.1.1
module load Perl
module load Java
module load Trimmomatic
module load STAR

# -------------------------------
# Commands:

# Submit:
#   Path for shell script to submit
#   This is what is submitted to SLURM
/mnt/research/edgerpat_lab/Scotty/Grape_RNA_Seq/Scripts/Trim/trim_files.sh FILE1 FILE2 NAME
```

Directory Structure:

It is important to note than within the folder with my 3 scripts, Trim_Bulk.pl, trim-files.sh, and submit_trim_files_BASE.sh I have a folder called **Trim_Submit** where all of the submission shell files are put, and where the error and log files are put as the script runs on the HPCC.

Use:

To run the pipeline, make sure you have a **File_List.csv** constructed and you have made sure your directory structure is well-connected.

Run: `perl Trim_Bulk.pl /full/path/to/File_List.csv/appropriate/to/your/specified_one/in/trim_files.sh`

We will now move on to alignment, now that our fastq files are appropriately trimmed.

Alignment: