1. **Purpose**

This SOP describes the steps in the RNA-seq pipeline, the programs used in each step with the defaults parameters, and examples of how to run the pipeline on the UNC Longleaf cluster.

1. **Scope**

This SOP is valid for RNA-seq data processing from human and mice.

1. **References/Related Procedures**

This SOP is related to all other SOPs associated with pipelines in the Furey Lab:

(LIST SOP NAMES)

1. **Responsibilities**

When running this pipeline, the version of the pipeline should be noted along with any deviations from the default parameters described here.

1. **Definitions and Diagrams**

A flow diagram of steps in the pipeline is attached. This pipeline consists of the following steps:

5.1) **Purpose:** Assess sequence quality

**Program:** Fastqc, longleaf module: fastqc

**Parameters:** output directory, input fastq file

**Command:** fastqc --outdir <output dir> --noextract -t 1 <fastq file>

**Notes:** fastq file can be gzip’d, run on both paired read files

**Dependencies:** none

5.2) **Purpose:** Trim reads to a given length

**Program:** fastx\_trimmer, longleaf module fastx\_toolkit

**Parameters:** last base to keep [50], input fastq file, output fastq file

**Command:** fastx\_trimmer -l <last base> -i <fastq file> -o <out file>

**Notes:** fastq file can be gzip’d, run on both paired read files

**Dependencies:** none

5.3) **Purpose:** Trim low quality bases from ends of reads

**Program:** fastq\_quality\_filter, longleaf module fastx\_toolkit

**Parameters:** quality score [33], minimum quality score to keep [20], minimum percent bases above minimum quality score [90], input fastq file, output fastq file

**Command:** fastx\_quality\_filter -Q 33 -q [min score] -p [min percent] -i <fastq file> -o <out file>

**Notes:** fastq file can be gzip’d, run on both paired read files

**Dependencies:** Step 2

5.4) **Purpose:** Filter adapter contamination

**Program:** tagdust, longleaf module tagdust

**Parameters:** input fastq file, output fastq file, adapter fasta file

**Command:** tagdust -ref /proj/fureylab/bin/pipelines/RNA/truseq-library-seqs.fasta -a <fastq file> -o <out file> (-q ? -f 0.001 ?)

**Notes:** run on both paired read files

**Dependencies:** Step 3

5.5) **Purpose:** After potentially eliminating sequences from one of the read pair files, remove the corresponding read from the other read pair file

**Program:** SyncFastqsForPipeline.pl

**Parameters:** input paired fastq files, output paired fastq files

**Command:** /proj/fureylab/bin/pipelines/RNA/SyncFastqsForPipeline.pl -read1 [input read 1 fastq] -read2 [input read 2 fastq] -out1 [output read 1 fastq] -out2 [output read 2 fastq]

**Notes:** run only with paired read files, must be run on bigmem node [50G]

**Dependencies:** Step 4 (both paired read files)

5.6) **Purpose:** Align and quantify sequences

**Program:** rsem, star, longleaf module rsem, star

**Parameters:** temp directory, number of threads [4], input fastq file(s), output genome bam file, output transcript bam

**Command:** rsem --star --num-threads [number of threads] --temporary-folder [temp directory] --sort-bam-memory-per-thread 10M --keep-intermediate-files --output-genome-bam --paired-end [read 1 fastq] [read 2 fastq]

**Notes:** if single end reads, remove --paired-end and read 2 fastq options, must be run on bigmem node [100G]

**Dependencies:** Step 4 (single end read file) or Step 5 (paired end read files)

5.7) **Purpose:** Sort bam file(s)

**Program:** samtools sort, longleaf module samtools

**Parameters:** input genome bam file, output sorted genome bam file

**Command:** samtools sort [input genome bam file] -o [output sorted genome file]

**Notes:**

**Dependencies:** Step 6

5.8) **Purpose:** Index bam file

**Program:** samtools index, longleaf module samtools

**Parameters:** input sorted genome bam file, output sorted genome bam index file

**Command:** samtools index [input sorted genome bam file] -o [output sorted genome index file]

**Notes:**

**Dependencies:** Step 7

5.9) **Purpose:** Determine per base coverage for browser display

**Program:** bamCoverage index, longleaf module deeptools

**Parameters:** input sorted genome bam file, input sorted genome bam index file, output sorted genome bam index file

**Command:** bamCoverage --bam [input sorted genome bam file] -o [output bigwig file] --binSize 10 --normalizeUsing RPKM --ignoreForNormalization chrX chrY

**Notes:** set slurm time to 2 days

**Dependencies:** Step 8

5.10) **Purpose:** Determine count statistics for each step

**Program:** line\_count.py

**Parameters:** output directory for above steps, temp directory for above steps, base naming label, initial read fastq file(s), output linecount file

**Command:** /proj/fureylab/bin/pipelines/RNA/line\_count.py count --out-dir [output directory --tmp-directory [temp directory] --base-name [base name] --input-file [initial read 1 fastq] --input-file2 [initial read 2 fastq] linecount.out

**Notes:** For single end, remove --input-file2 parameter

**Dependencies:** Step 6

1. **Procedure**

6.1) Log on to longleaf.unc.edu

6.2) Create config file. See attached for an example. Configuration file should be grouped in sections by assembly, i.e. hg38, mm10. The following parameters should be set in the config file:

--lookup [lookup file]

--rsem-database [path to rsem database]

6.3) Run pipeline using the following command:

/proj/fureylab/pipelines/RNA-seq/rna.py --rsem --num-of-bp [50] --quality [20] --percent [90] --read1 [first paired end fastq file] --read2 [second paired end fastq file] --config [configuration file] --section [section header in configuration file, i.e. hg38] --slurm --max-num-threads [4] --out-dir [permanent output directory] --tmp-dir [temporary file directory on /netscr] --run

1. **Attachments**

Any attachments that are needed to make the SOP complete should be included here as a list with names.

--Attach an example of the configuration file, schematic of the pipeline

1. **Revision History**

RNA-seq Pipeline Version 1.0 3/19/2020 Supersedes: None