RNAseq pipeline

The RNAseq pipeline is packaged in the Docker image with all the required tools pre-installed. You can download the image within Docker and then run it. The pipeline can run on any local operating system. The instructions in this document assume the pipeline running on Linux system.

Install Docker

The Docker is available for Mac, Window and Linux

- For MAC or Windows users:
  https://www.docker.com/products/docker-toolbox
- For Linux users: (Ubuntu and Centos)
  - Ubuntu: **https://docs.docker.com/engine/installation/linux/ubuntulinux/**
  - Linux: **https://docs.docker.com/engine/installation/linux/centos/**

Visiting **https://docs.docker.com/** for more details about Docker installation.

Load Docker image into a directory

The image of RNAseq pipeline is named as **ualrngs/rna-seq-pipeline**. Open a terminal and download the image into your computer by typing the following command

```
#load the image into your work directory
docker pull ualrngs/rna-seq-pipeline
```

You can check the image by typing

```
docker images
```

There are three entries in the pipeline.

- Sequencing quality assessment (**quality**)
- Transcripts assembly and differential expression analysis (**assemble**)
- Variants calling (**variant**)

We recommend sequencing quality assessment for all data to ensure the quality of analysis.

Data Structure in Docker Image

The Docker includes three main directories

- `/input`: Input data
- `/indexDir`: Index and annotation files
- `/variantOut`: Result of variant callers
- `/working`: Output and intermediate data

Inside the `/working` directory, there are five subdirecories. The description of each subdirectory is given the Table 1

Table 1: The subdirectory inside /working directory

Subdirectory	**Description**
fastQCOut	The results of FASTQC assessment
trimmed	The trimmed reads, which are output of trim and required by mapping. Two additional folders indicate the conditions/phenotypes such as normal and tumor are required.
tophatMAP	Aligned data, which are output of TopHat2 and are used by alignment quality control and Cufflinks for assembly. Two additional folders indicate the conditions/phenotypes such as normal and tumor are required.
alignmentQC	The alignment quality assessment output.
assembledTranscripts	The assembled transcripts based on RNA-seq data.
diffExpr	The reports of the differential expression analysis between the two conditions/phenotypes.

Input Data and Sequencing Quality Assessment

The pipeline takes paired-end RAN-seq data in FASTQ format as the input. The data file of forward and reverse strand should be named as *_1.fastq* and *_2.fastq* with same prefix, indicating the two strands of the same region.

To organize your data, you can create directories for input and output data. Furthermore, if you have different type of data, for instance, normal versus cancer. You can create subdirectory within directory of input data. For example, you want to start a project for analysis tumor and normal RNAseq data.

```
mkdir /home/yourAccount/Project1/inputRNA #make a new directory called inputRNA for input RNAseq data.
cd ./inputRNA #get into the directory.
mkdir normal tumor #make two sub-directories, normal and tumor for different type of sample
```

Then you can copy your data into input directory.

```
mkdir /home/yourAccount/Project1/RNA_results #make a directory for the results
```

Then perform quality assessment of raw reads

```
docker run --rm -v /home/yourAccount/Project1/inputRNA:/input -v /home/yourAccount/RNA_results:/working
    ualrngs/rna-seq-pipeline quality
```

Here

- docker run: the command to start a new Docker container to process the data
- –rm: means this Docker container will be removed after it finished
- -v /home/yourAccount/Project1/inputRNA:/input: Maps the directory `/home/yourAccount/Project1/inputRNA/` of the local machine to the directory `/input/` in the Docker container
- -v /home/yourAccount/RNA_results:/working: Obtain the results at `/working` in the Docker image to the local directory `/home/yourAccount/RNA_results`
- ualrngs/rna-seq-pipeline: the name of Docker image containing RNAseq pipeline
- **quality**: the name of the job for assessing reads quality, and one of three entries of the pipeline

The results of quality assessment of the RNA-seq data by FastQC can be founded in `/home/yourAccount/Project1/RNA_results`

Note: The directory `/working` in the Docker container is the folder in which intermediate and final results are saved and stored.

Differential Expression Analysis

Prepare reference and index files

As the size of index files is often very large, our Docker image does not contain index files. You need to build the reference and index files in your computer. The files that you need to download are

- The sequence of the whole reference genome
- The annotation of the genes in the reference genome
- The index files used buy Tophat2 for mapping

Create a directory to install the reference and index files

```
#This directory for index files, reference sequence files and annotation files in user's local computer
mkdir /home/yourAccount/Project1/refGenome/refFile
```

Various versions of reference and annotation of human and mouse genome are available at NCBI website. You can download zip file and then create index file in your local directory.

The following commands show a example to download human annotation and index files from NCBI website and then copy the index files, annotation files into the corresponding directory in the local computer.

```
# Step 1; #Download a compressed file from NCBI website
wget ftp://igenome:G3nom3s4u@ussd-ftp.illumina.com/Homo_sapiens/NCBI/build37.1/
    Homo_sapiens_NCBI_build37.1.tar.gz

#Step 2. Unzip the file. Then you should obtain a folder named Homo_sapiens contain various reference
    of human genome in version 37.1. The file in the folder includes index files for Tophat2
tar -xzvf Homo_sapiens_NCBI_build37.1.tar.gz

#Step 3: Copy the index files and annotation files
cp -r ./Homo_sapiens/NCBI/build37.1/Sequence/Bowtie2Index ./refFile/tophat2
cp ./Homo_sapiens/NCBI/build37.1/Sequence/WholeGenomeFasta/genome.fa ./refFile
```

```
cp ./Homo_sapiens/NCBI/build37.1/Annotation/Genes/genes.gtf ./refFile/annotatedGenes.gtf #copy and
    rename.
# After this command, you should have a folder named tophat2 and two files genome.fa and annotatedGenes
    .gtf in /home/yourAccount/Project1/refGenome/refFile
```

In the above example,

- Bowtie2Index is a folder containing the index files for TopHat2 to do the mapping; here we rename it to tophat2.
- genome.fa contains sequence of the whole genome
- annotatedGenes.gtf contains the annotation of the human genes with their coordination information, gene id, gene symbol, gene biotype, strand, source et.al

Start the full-run of the pipeline

Once you have index files and annotation files, you are ready to run the RNAseq pipeline.

```
docker run --rm -v /home/yourAccount/Project1/inputRNA:/input -v /home/yourAccount/Project1/refGenome/
    refFile:/indexDir -v /home/yourAccount/Project1/RNA-results:/working ualrngs/rna-seq-pipeline
    assemble full -k -r

#You copy your data files, index file and reference file to Docker image, run the assemble, your result
     is available at local directory /home/yourAccount/Project1/RNA-results/
```

Here,

- assemble: the name of job
- full: refers to run the entire pipeline
- -k: if specified, intermediate output is kept; Otherwise, only the results of differential expression analysis is reported
- -r: if specified, the annotation of genes from NCBI or Ensembl is used; Otherwise, the assembled transcripts from the RNA-seq data is used as annotation
- -f: library-type. Default is unstranded; Otherwise, use -f first or -f second to specify the library types respectively.

Library type	RNA-seq protocol	Description
fr-unstranded (default)	Illumina TruSeq	Reads from the leftmost end of the fragment (in transcript coordinates) map to the transcript strand, and the rightmost end maps to the opposite strand
fr-firststrand	dUTP, NSR, NNSR	Same as above except we enforce the rule that the rightmost end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during first strand synthesis is sequenced
fr-secondstrand	Directional Illumina (Ligation), Standard SOLiD	Same as above except TopHat/Cufflinks enforce the rule that the leftmost end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during second strand synthesis is sequenced

- -p:the number of thread. 8 is the default value
- -h: help, show the description of all the arguments

Run individual steps in the pipeline

Our RNA-seq pipeline also supports individual process for specific purpose. For example, you may want to do the quality control with your own parameters only or you may want to assemble the transcripts based on the RNA-seq data to discover some novel genes.

Here, **MODE** of assemble includes

- full (full pipeline as explained in the previous section)
- trim (trim the low quality reads)
- map (map raw reads to the reference genome)
- alignqc (alignment quality assessment)
- assembly (assemble transcriptome)
- de (differential expression analysis)

Now we will explain the usage of each mode.
Suppose you need to analyze a RNAseq dataset for normal and tumor tissues. Assume, you have data files and required index files stored in the following folders in your computer

- Input sequencing data for normal samples: `/home/yourAccount/Project1/inputRNA/normal/`
- Input sequencing data for tumor samples: `/home/yourAccount/Project1/inputRNA/tumor/`
- Index and annotation files: `/home/yourAccount/Project1/refGenome/refFile/`

trim

Function: trim the reads based on the quality of base-pairs.

- Input
  - Data type: fastq files with .fastq/.fq suffix.
  - Location of input data files
    - Local computer: You should have data files located at your computer, assume path of the files `/home/yourAccount/Project1/inputRNA/normal` for normal samples, and `/home/yourAccount/Project1/inputRNA/` for tumor samples
    - Docker: To run the docker, the data file needs to be copied to Docker in the directory `/input` using -v showing the following command
- Output:
  - Docker: the trimmed output fastqc is available `/working/normal/` for normal samples, and `/working/tumor/` for tumor samples at Docker.
  - Local computer: The result can be copied to your computer using -v as shown in the following command.
- Parameters:

  -h: print the description of the arguments

  -a: use specified adapter file, *adapter.txt* in input directory. (Default: Trimmomatic adapters). The format of adapter file is following

  ```
  >adapter1
  accagtacataccgtacgtaaatttgggccc
  ```

  -l: cut bases off the start of a read, if below a threshold quality, optional. (Default: 28)

  -t: cut bases off the end of a read if below this threshold, optional. (Default: 28)

  -m: drop the read if it is below a specified length, optional. (Default: 36)

  -w: perform a sliding window trimming, cutting once the average quality within the window falls below a threshold, optional. (Default is 4:24)

- Command

```
#The command for running trim mode
docker run --rm -v /home/yourAccount/Project1/inputRNA:/input -v /home/yourAccount/Project1/RNA-
    results:/working ualrngs/rna-seq-pipeline assemble trim -l 30 -t 30 -m 76
```

map

Function: apply TopHat2 to map the reads to the reference genome.

- Input
  - Data type: fastq files with .fastq/.fq suffix. The data should have a good quality evaluated by FASTQC. Often the output of trimmer is used for map to ensure the quality
  - Location of input data files
    - Local computer: You should have data files containing good quality reads located at your computer, assume path of the files `/home/yourAccount/Project1/cleanReads/normal/` for normal samples, and `/home/yourAccount/Project1/cleanReads/tumor/` for tumor samples

    * Docker: To run the docker, the data file needs to be copied to Docker in the directory `/working/trimmed/normal/`for normal samples, and `/working/trimmed/tumor/` using -v showing the following command

- Output:
  - Data type: .BAM files
  - Docker: the mapping results is available `/working/tophatMAP/normal/` for normal samples, and `/working/tophatMAP/tumor/` for tumor samples at Docker.
  - Local computer: The result can be copied to your computer using -v as shown in the following command.
- Parameters:

  -h: print the description of the arguments

  -f: library-type, default: fr-unstranded

  -p: the number of threads to be used, default is 8
- Index files: For mapping, user should already have index and annotation files at local computer. In this case, assume the index files locates
  - Local computer: `/home/yourAccount/Project1/refGenome/refFile/`
  - Docker: using -v to copy the index and annotation files to the Docker `/indexDir`
- Command

```
#The command for running map mode
docker run --rm -v /home/yourAccount/Project1/cleanReads:/working/trimmed -v /home/yourAccount/
    refGenome/refFile:/indexDir -v /home/yourAccount/Project1/RNA-results:/working ualrngs/rna-
    seq-pipeline assemble map
```

alignqc

Function: assess the quality of the alignment. Here *Picard*(InsertSizeMetrics and RnaSeqMetrics), *SAMtools* (flagstat) and *Qualimap*(bamqc) are applied for the alignment quality assessment

- Input
  - Data type: BAM files generated by TopHat2
  - Location of input data files
    * Local computer: Assume you have .BAM data files located at your computer, `/home/yourAccount/Project1/mappedReads/normal/` for normal samples, and `/home/yourAccount/Project1/mappedReads/tumor/` for tumor samples
    * Docker: To run the docker, the input .BAM data file needs to be copied to Docker in the directory `/working/tophatMAP/normal/`for normal samples, and `/working/tophatMAP/tumor/` using -v showing the following command
- Output:
  - Docker: the alignment quality assessment results are available `/working/alignmentQC/normal/` for normal samples, and `/working/alignmentQC/tumor/` for tumor samples at Docker.
  - Local computer: The result can be copied to your computer using -v as shown in the following command.

- Parameters:

  -h: print the description of the arguments

  -f: library-type, default: fr-unstranded

- Command

```
#The command for running alignqc mode
docker run --rm -v /home/yourAccount/Project1/mappedReads:/working/tophatMAP -v /home/yourAccount
    /Project1/RNA-results:/working ualrngs/rna-seq-pipeline assemble alignqc
```

assembly

Function: use Cufflinks and Cuffmerge to assemble transcripts

- Input
  - Data type: BAM files generated by TopHat2
  - Location of input data files
    - Local computer: Assume you have .BAM data files located at your computer, `/home/yourAccount/Project1/mappedReads/normal/` for normal samples, and `/home/yourAccount/Project1/mappedReads/tumor/` for tumor samples
    - Docker: To run the docker, the input .BAM data file needs to be copied to Docker in the directory `/working/tophatMAP/normal/`for normal samples, and `/working/tophatMAP/tumor/` using -v showing the following command
- Index files: For differential expression analysis, user should already have index and annotation files at local computer. In this case, assume the index files locates
  - Local computer: `/home/yourAccount/Project1/refGenome/refFile/`
  - Docker: using -v to copy the index and annotation files to the Docker `/indexDir`
- Output:
  - Docker: The output of assembly is the annotation of assembled genes in GTF format. All the samples including normal and tumor are used for generating the annotation. As a result, a single file named as assembledGenes.gtf is reported in the output directory `/working/assembledTranscripts/`
  - Local computer: The result can be copied to your computer using -v as shown in the following command.
- Parameters:

  -h: print the description of the arguments

  -f: library-type, default: fr-unstranded

  -p: the number of threads to be used, default is 8

- Command

```
#The command for running assembly mode
docker run --rm -v /home/yourAccount/Project1/mappedReads:/working/tophatMAP -v /home/yourAccount
    /refGenome/refFile:/indexDir -v /home/yourAccount/Project1/RNA-results:/working ualrngs/rna-
    seq-pipeline assemble assembly
```

de

Function: use Cuffdiff to detect the differential expression

- Input
  - Data type: BAM files generated by TopHat2, and gene list as reference. The reference genes could be annotated genes or assembled transcripts
    - annotated genes: download from website such as Ensembl, NCBI
    - assembled transcripts: generated by assembly mode
  - Location of input data files
    - Local computer: Assume you have .BAM data files located at your computer, `/home/yourAccount/Project1/mappedReads/normal/` for normal samples, and `/home/yourAccount/Project1/mappedReads/tumor/` for tumor samples
    - Docker: To run the docker, the input .BAM data file needs to be copied to Docker in the directory `/working/tophatMAP/normal/`for normal samples, and `/working/tophatMAP/tumor/` using -v showing the following command
- Reference genes: In order to run the differential expression individually, you are required to supply the reference genes to guide the analysis. Hence, map the annotated gene list to /indexDir in Docker container and always add -r to ask the pipeline to use annotatedGenes.gtf
- Index files: For assembly, user should already have index and annotation files at local computer. In this case, assume the index files locates
  - Local computer: `/home/yourAccount/Project1/refGenome/refFile/`
  - Docker: using -v to copy the index and annotation files to the Docker `/indexDir`
- Output:
  - Docker: The output of differential expression analysis is available at `/working/diffExpr/`
  - Local computer: The result can be copied to your computer using -v as shown in the following command.
- Parameters:

  -h: print the description of the arguments

  -f: library-type, default: fr-unstranded

  -p: the number of threads to be used, default is 8

  -r: if specified, use the annotated genes to measure the differential expression
- Command

```
#The command for running de mode
docker run --rm -v /home/yourAccount/Project1/mappedReads:/working/tophatMAP -v /home/yourAccount
    /refGenome/refFile:/indexDir -v /home/yourAccount/Project1/RNA-results:/working ualrngs/rna-
    seq-pipeline assemble de -r
```

Variant Call using GATK

Prepare the index for STAR and GATK

STAR and GATK also require index files for execution. The sequence of the whole genome (genome.fa) is used to build the index for both of these two applications. Map the genome.fa and annotatedGenes.gtf to the Docker container, the pipeline will build the index files. The same with section 2, organize the folder /refFile

Variant call

To run variant call, you need to have index files for STAR and GATK, as well sequence file in your local computers. And these files will be copied to Docker to run variant caller.

- Input files: fastqc file located at your computer `/home/yourAccount/Project1/inputRNA`
- Required index files and reference sequence files
  - index files for STAR at `/home/yourAccount/Project1/refGenome/refFile/indexSTAR`
  - index files for GATK at `/home/yourAccount/Project1/refGenome/refFile/indexGATK`
  - sequence file *genome.fa*: `/home/yourAccount/Project1/refGenome/refFile/`
- Output: The output of variant caller is available at the directory `/variantOut` in Docker. The results can be copied to the local machine using -v as shown in the following command
- Parameters

  -b: run the pipeline from building the index without save it on Docker

  -t: the pipeline apply trim as default. You can skip trimming by adding -t

  -p: the number of threads to be used, default is 8
- Command

```
#The command for running variant call
docker run --rm -v /home/yourAccount/Project1/inputRNA:/input -v /home/yourAccount/refGenome/refFile:/
    indexDir -v /home/yourAccount/Project1/RNA-results:/variantOut ualrngs/rna-seq-pipeline variant
    call -p 16


#Run vaiant call without saving index files and skip trim step.
docker run --rm -v /home/yourAccount/Project1/inputRNA:/input -v /home/yourAccount/refGenome/refFile:/
    indexDir -v /home/yourAccount/Project1/RNA-results:/variantOut ualrngs/rna-seq-pipeline variant
    call -p 16 -b -t
```