Single cell RNAseq Data Pre-Processing

Cellranger Singel Cell Gene Expression Workflow

Dr. Rudramani Pokhrel

Computational Research Scientiest, The University of Arizona, Immunobiology Department
rpokhrel@email.arizona.edu

Last modified: 30 Apr 2022

Cell Ranger is a set of analysis pipelines that process Chromium single-cell data to align reads, generate feature-barcode matrices, perform clustering and other secondary analysis, and more. In this tutorial we will start with installing software, use ***cellrange mkfastq*** module to demultiplex the illumina basecall files and finally, use ***cellranger count*** moduel to obtain gene count matrices.

Software Requirement

Download and install cellranger from:

https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest?

Install illumina software bcl2fastq2 from:

https://support.illumina.com/downloads/bcl2fastq-conversion-software-v2-20.html

Alternatively you can use conda to download both packages:

```
conda install -c hcc cellranger
conda install -c dranew bcl2fastq
```

Since the pipeline requires minimum of 32GB RAM, I am using provided module in University of Arizona hpc If you are using your own hpc environment; please chek the available module;

```
module spider bcl2fasq2
module spider mkfastq
```

Demultiplexing the Illumina basecall files (BCL)

The cellranger mkfastq is a wrapper for Illumina's bcl2fastq tool for demultiplexing illumina basecall files. BCL files are raw sequencing data from a sequencer. Demultiplexing is assigning the cluster of raw files to samples provided with index information.

I have skin cancer Melanoma single cell data for cell line A375 and A375_treated with some drugs. Check it with **ls** command
ls

```
Data
RTAComplete.txt
RTAConfiguration.xml
RTARead1Complete.txt
RTARead2Complete.txt
RTARead3Complete.txt
RunCompletionStatus.xml
RunInfo.xml
RunParameters.xml
```

Inside the ***Data/Intensities/BaseCalls/*** folder we have bcl files.

```
ls Data/Intensities/BaseCalls/
L001  L002  L003  L004
```

Which contains data from all four flow cells.

RunInfo.xml contains information about the sequence and index length:

```
more RunInfo.xml

<Reads>
      <Read Number="1" NumCycles="26" IsIndexedRead="N" />
      <Read Number="2" NumCycles="8" IsIndexedRead="Y" />
      <Read Number="3" NumCycles="98" IsIndexedRead="N" />
    </Reads>
```

Running makefastq

The single cell data are sequenced over the platform Single index plates used with Chromium Single Cell 3' v3.1 (Single Index). You can get the information about the index in : https://support.10xgenomics.com/single-cell-gene-expression/index/doc/specifications-sample-index-sets-for-single-cell-3
We have sequenced the data using i7 index set **SI-GA-A11** for control samples and **SI-GA-A12** for treated samples. From this information I generated a ***SampleSheet.csv*** files:

```
Lane,Sample,Index
*,A375,SI-GA-A11
*,A375_treated,SI-GA-A12
```

where * represents all four lanes in flowcell.

Create a file *mkfastq.sh* to submit the job in hpc which uses slurm job scheduler.

```
#!/bin/bash
#SBATCH --account=rpokhrel
#SBATCH --job-name=mkfastq
#SBATCH --err=mkfastq.err
#SBATCH --out=mkfastq.out
#SBATCH --time=4:0:0
#SBATCH --partition=standard
#SBATCH --nodes=1
# number of tasks (processes) per node, need minimum of 32 GB RAM
#SBATCH --ntasks-per-node=10
#SBATCH --mail-type=END
#SBATCH --mail-user=rpokhrel@arizona.edu

module load gnu8/8.3.0
module load bcl2fastq2/2.20.0
module load cellranger/6.1.1
cellranger mkfastq --id=Demultiplexed \
                   --run=/xdisk/rpokhrel/Test/scRNA/Melanoma \
                   --csv=SampleSheet.csv
```

Where **–id** is the name of output folder and **–run** is path to the folder which contains **Data** folder and **SampleSheet.csv** file.
Submit your job: `sbatch mkfastq.sh`

Once you finish the run you can find corresponding fastq files in folder **Demultiplexed/outs/fastq_path/HNN53BGXF**.
It generated two folders for two seperate samples: **A375 and A375_treated**
Inside each folder you have demultiplexed fastq files.

Cellranger count:

The cellranger count pipeline aligns sequencing reads in FASTQ files to a reference transcriptome and returns UMI count data per cell. First download the reference file from https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest
In this case I am using human species.

```
wget https://cf.10xgenomics.com/supp/cell-exp/refdata-gex-GRCh38-2020-A.tar.gz
tar -xvf refdata-gex-GRCh38-2020-A.tar.gz
```

Prepare a cout.sh hpc submission file

```
#!/bin/bash
#SBATCH --account=rpokhrel
#SBATCH --job-name=count
#SBATCH --err=count.err
#SBATCH --out=count.out
#SBATCH --time=50:0:0
#SBATCH --partition=standard
#SBATCH --nodes=1
# number of tasks (processes) per node, need minimum of 32 GB RAM
#SBATCH --ntasks-per-node=10
#SBATCH --mail-type=END
#SBATCH --mail-user=rpokhrel@arizona.edu

module load gnu8/8.3.0
module load bcl2fastq2/2.20.0
module load cellranger/6.1.1

# run for first library
cellranger count \
    --transcriptome=/xdisk/rpokhrel/Test/scRNA/refdata-gex-GRCh38-2020-A \
    --id=A375_count \
    --fastqs=/xdisk/rpokhrel/Test/scRNA/Melanoma/Demultiplexed/outs/fastq_path/HNN53BGXF/A375 \
    --sample=A375 \
    --include-introns \
    --localcores=10

# run for second library
cellranger count \
    --transcriptome=/xdisk/rpokhrel/Test/scRNA/refdata-gex-GRCh38-2020-A \
    --id=A375_count \
    --fastqs=/xdisk/rpokhrel/Test/scRNA/Melanoma/Demultiplexed/outs/fastq_path/HNN53BGXF/A375_treated \
    --sample=A375_treated \
    --include-introns \
    --localcores=10
```

–transcriptome is path to your downloaded reference file. **–id** path of output folder, and **–sample** is the sample name.
In **–sample** put prefix of demultiplexed fastq files like in this example we have **A375_S1_L001_R1_001.fastq.gz**

For more options please visit the site: https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/using/count

Submit job to hpc: `sbatch count.sh`

Once run finished, you have two out files **A375_out and A375_treated_out**

Inside **A375_count/outs** have following files:

```
analysis
cloupe.cloupe
filtered_feature_bc_matrix
filtered_feature_bc_matrix.h5
metrics_summary.csv
molecule_info.h5
possorted_genome_bam.bam
possorted_genome_bam.bam.bai
raw_feature_bc_matrix
raw_feature_bc_matrix.h5
web_summary.html
```

The basic analysis is in **analysis** folder and summary statistics is in **web_summary.html** file.
The **filtered_feature_bc_matrix** has three files: `barcodes.tsv.gz` `features.tsv.gz` `matrix.mtx.gz` We will use this files to do further analysis using **seurat** https://satijalab.org/seurat or **scanpy** https://scanpy.readthedocs.io/en/stable/tutorials.html packages. Please follow my Seurat analysis pipeline at:
https://github.com/githubrudramani/Pipelines/tree/main/sc-RNAseq/scRNA_data_analysis_complete_workflow.pdf