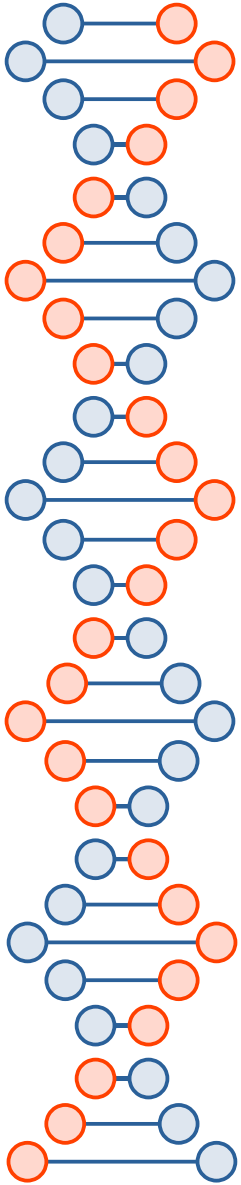


Tutorial

Running the Processing Pipeline on High-Throughput Raw Reads Data

Presented by: Eric Arezza

Prerequisite

- Familiarity with Linux command-line interface
- Familiarity with Python and virtualenv
- Access to high-performance computing (HPC) resources
  - Using Digital Research Alliance of Canada's Advanced Research Computing services here (a.k.a. Compute Canada)

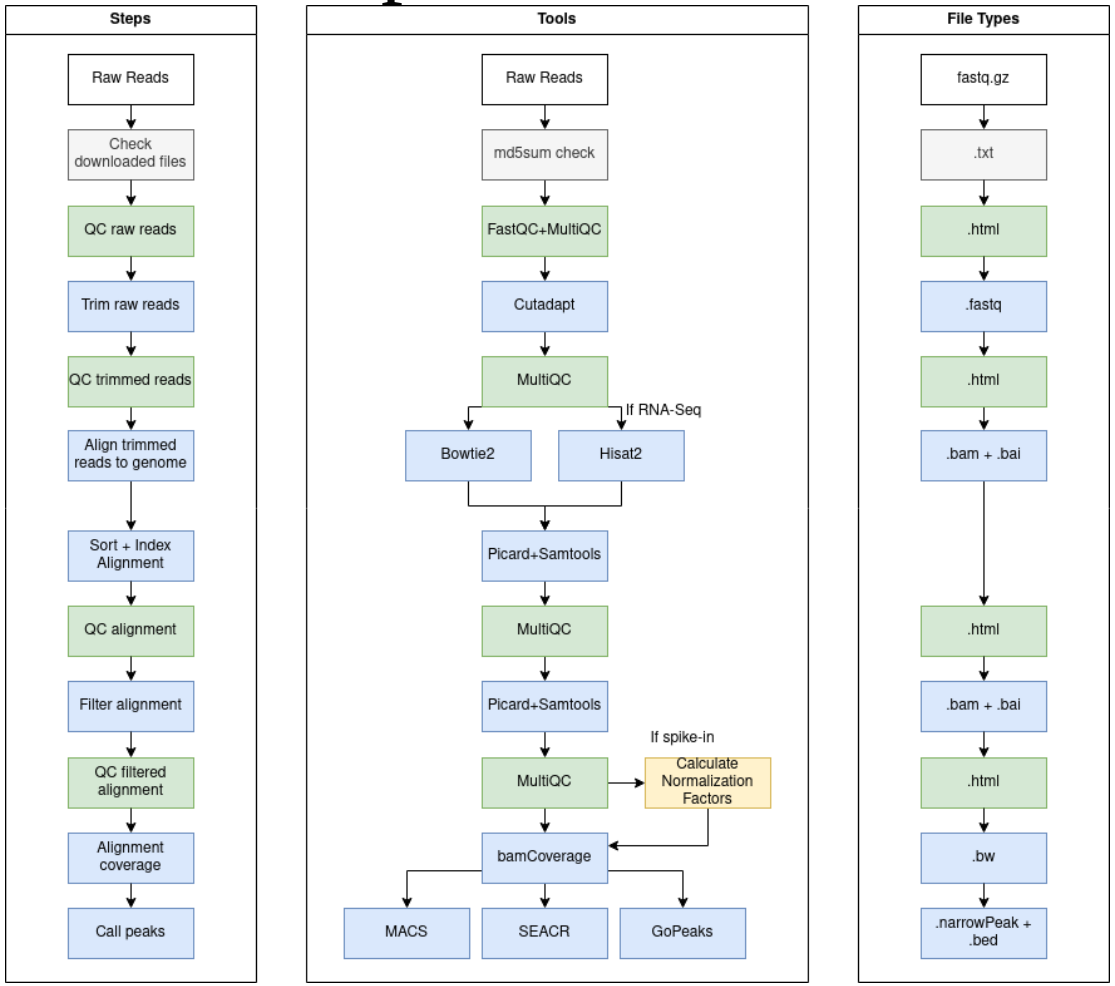
    https://alliancecan.ca/en/services/advanced-research-computing/federation/national-host-sites

    https://docs.alliancecan.ca/wiki/Technical_documentation

Preface

- Many programs available in bioinformatics for many uses and even for similar uses
  - Lots of support for Python-based and R-based tools
  - Many are also built in other languages
- Pipeline here is run using Python, executing functions from many programs

Pipeline Preface

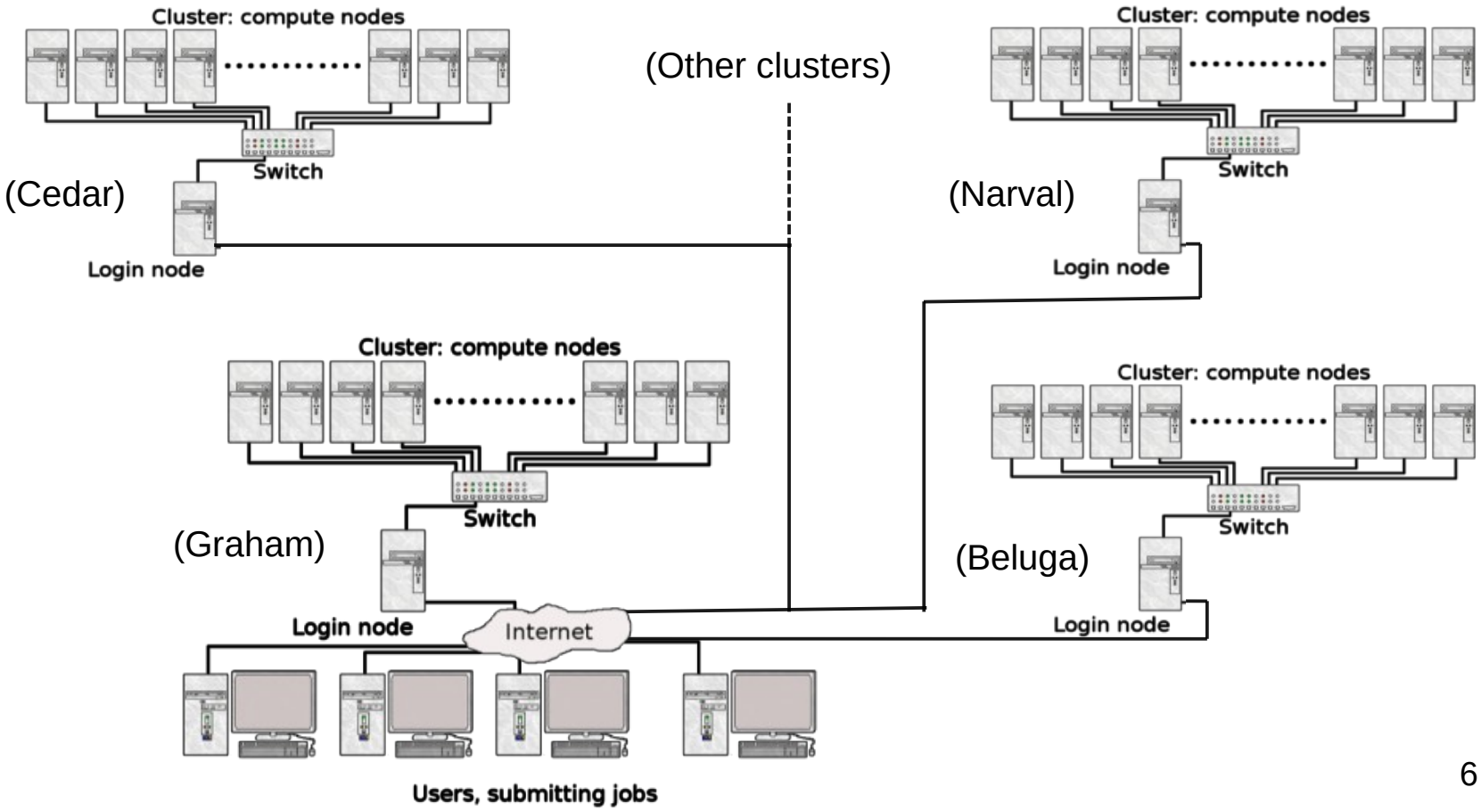


Pipeline Preface

- Maps reads from next-generation (high-throughput) sequencers to a reference genome to obtain enrichment levels of regions
- Capable of unspliced and spliced alignment

- Runs on HPC platform
  - Benefit from multi-core parallelism to speed-up processing

HPC Preface

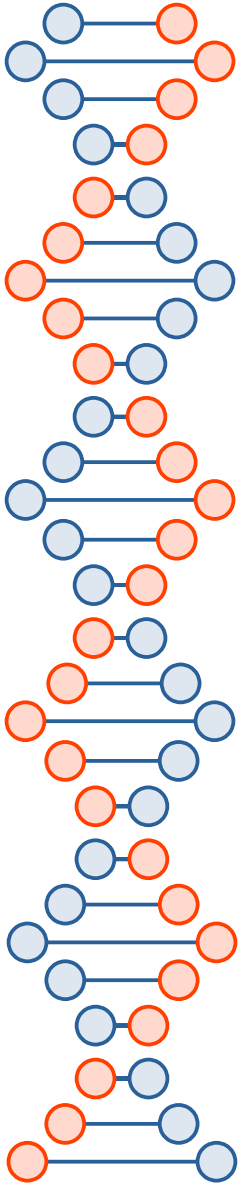

Cluster: compute nodes
Switch
(Cedar)
Login node
(Other clusters)
Cluster: compute nodes
Switch
(Narval)
Login node
Cluster: compute nodes
Switch
(Graham)
Login node
Internet
Cluster: compute nodes
Switch
(Beluga)
Login node
Users, submitting jobs

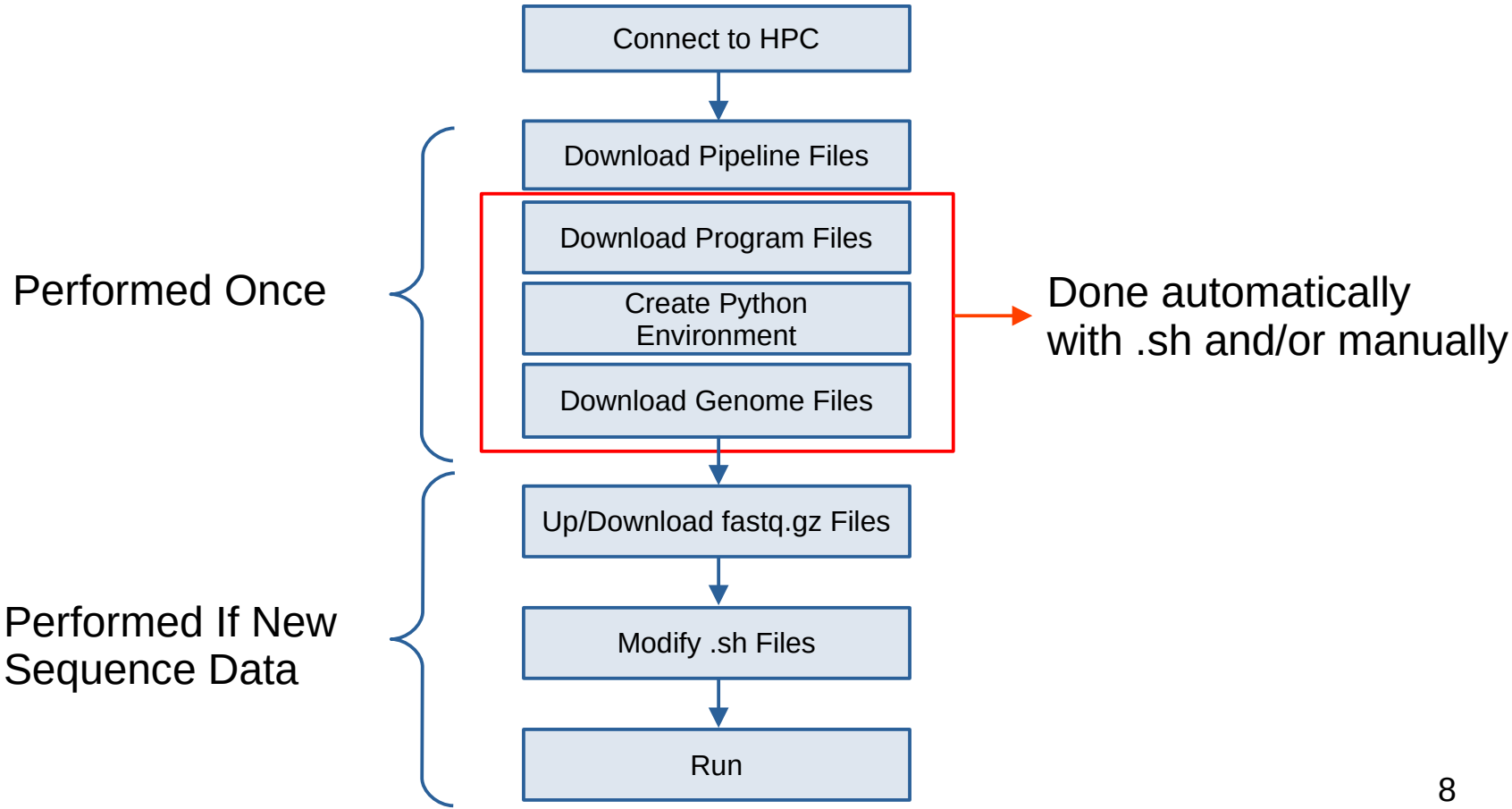
Learning Goals

<u>Running the pipeline:</u>

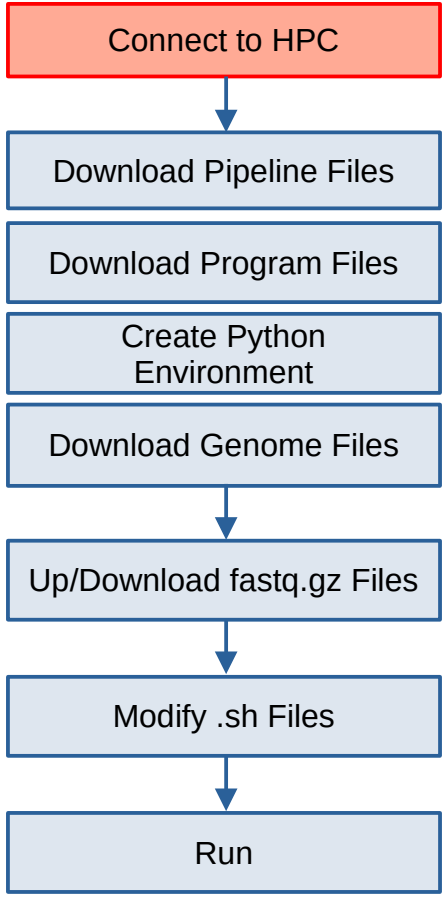
1) Setup on HPC:
   - Downloading required files and programs
   - Creating a Python virtualenv and installing packages
   - Downloading genome files
   - Down/Uploading .fastq.gz files
2) Run options:
   - Batch job scripts
   - Pipeline script
   - Debugging and output files

Overview




Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run
Performed Once
Performed If New Sequence Data
Done automatically with .sh and/or manually

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

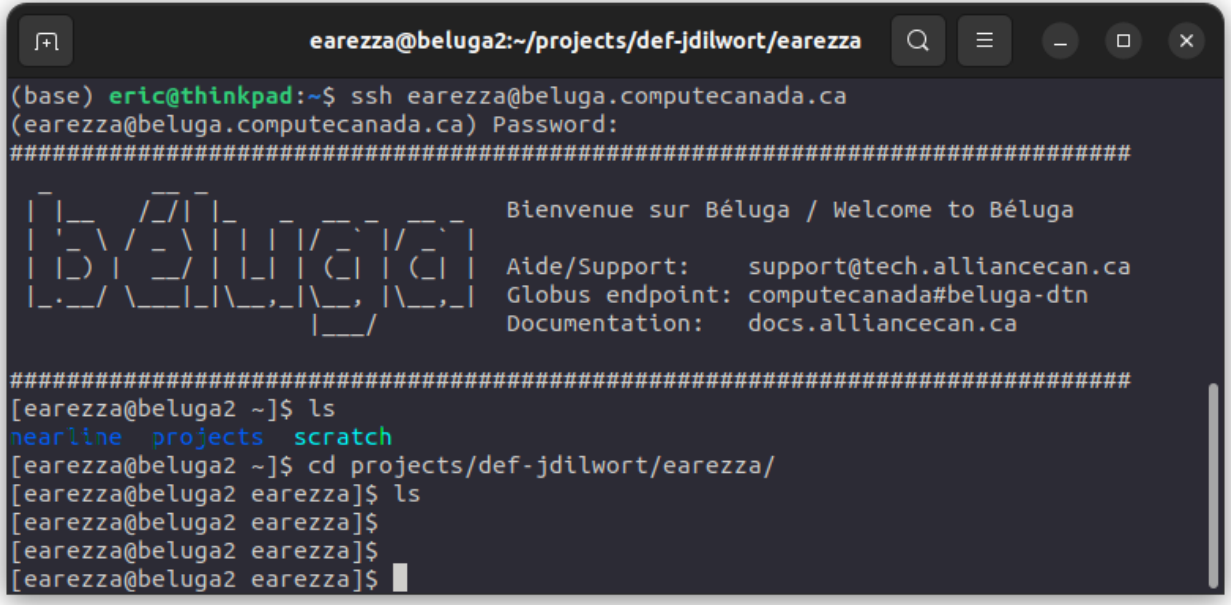
Setup – Connect to HPC

1) Login to HPC

ssh earezza@beluga.computecanada.ca

2) Navigate to your project directory

cd projects/def-jdilwort/earezza/



Setup – HPC Storage Allocations

Storage spaces on Compute Canada:

Cedar | Graham | Béluga and Narval | Niagara

Filesystem Characteristics

Filesystem	Default Quota	Lustre-based	Backed up	Purged	Available by Default	Mounted on Compute Nodes
Home Space	50 GB and 500K files per user[1]	No	Yes	No	Yes	Yes
Scratch Space	20 TB and 1M files per user	Yes	No	Files older than 60 days are purged.[2]	Yes	Yes
Project Space	1 TB and 500K files per group[3]	Yes	Yes	No	Yes	Yes
Nearline Space	10 TB and 5000 files per group	Yes	Yes	No	Yes	No

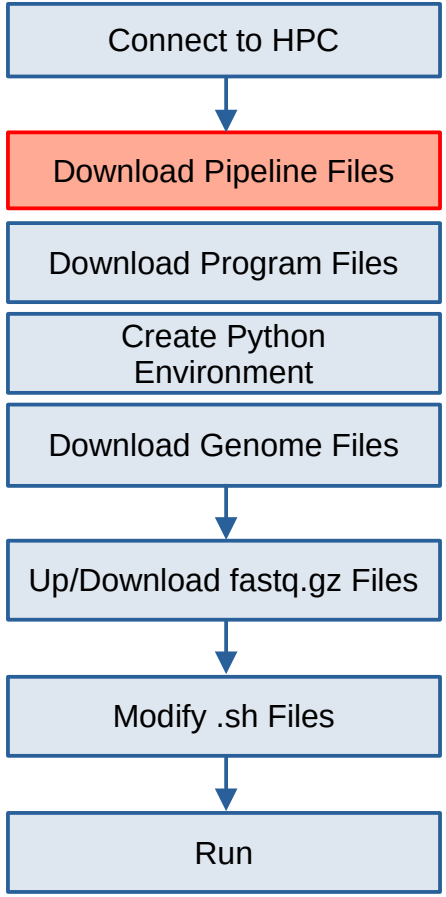
1. ↑ This quota is fixed and cannot be changed.
2. ↑ See Scratch purging policy for more information.
3. ↑ Project space can be increased to 10 TB per group by a RAS request. The group's sponsoring PI should write to technical support to make the request.

https://docs.alliancecan.ca/wiki/Storage_and_file_management

Takeaway:

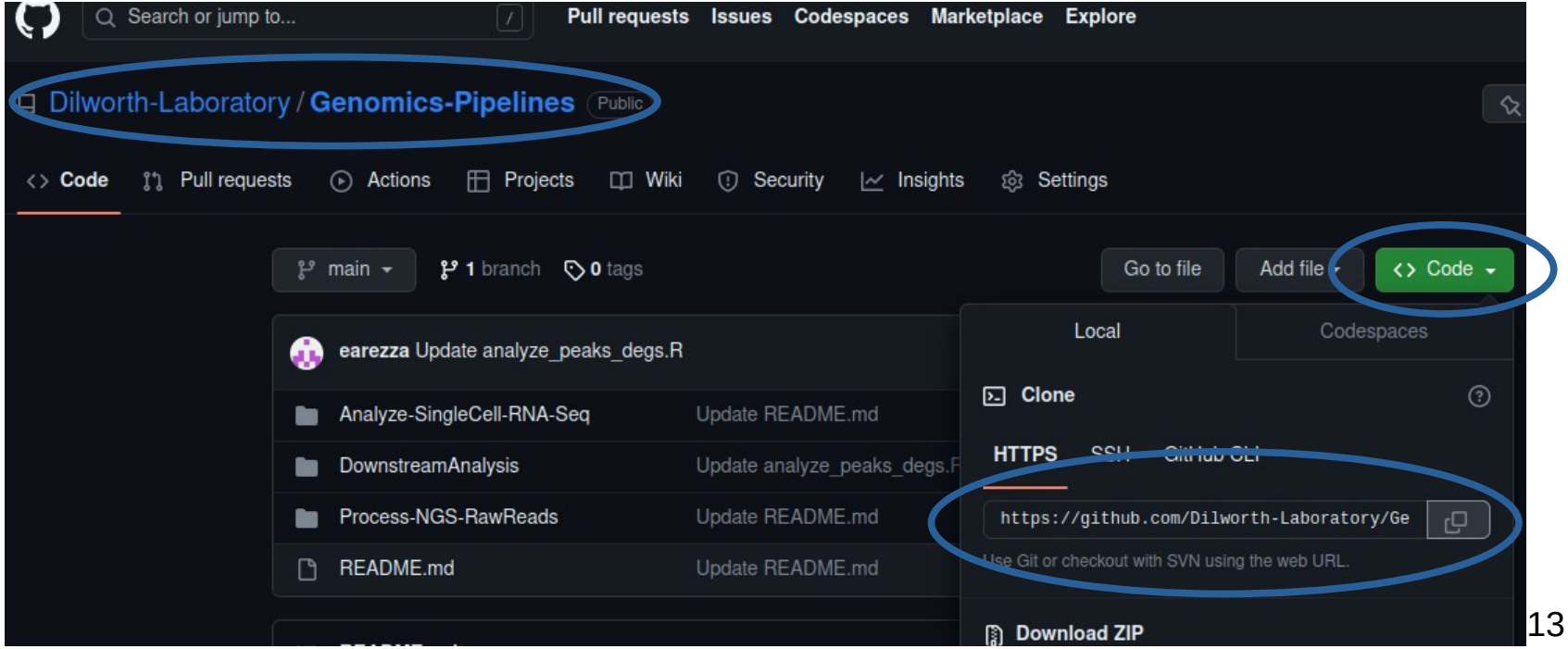
- Keep pipeline + script files in project/
- Keep data + output files in scratch/ (until backing-up/archiving)

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Setup – Downloading GitHub Files

In a web browser, copy the .git URL link from https://github.com/Dilworth-Laboratory/Genomics-Pipelines



Setup – Downloading GitHub Files

Clone the repo to your directory from the copied link:

- *git clone https://github.com/Dilworth-Laboratory/Genomics-Pipelines.git*

Main files from GitHub:

Dilworth-Laboratory/Genomics-Pipelines/Process-NGS-RawReads/

- ngs_processing_pipeline.py (main script)
- .sh files (batch script templates for running on HPC)
- cc_requirements.txt (Python environment info)

Setup – Automatically

Run setup script:

- *bash Genomics-Pipelines/Process-NGS-RawReads/ngs_setup.sh*

On Compute Canada (Beluga) and only mm10 genome files:

Time ~ 1.5 hours

Storage ~ 50GB

- 46GB is genome files (bowtie2 & hisat2)
- 2.5GB is python environment
- 33MB is all other files

If last step of downloading genomes is too long (or at any point), you can cancel with *Ctrl+C* and download manually using same lines in ngs_setup.sh script.

Setup – Manually

1) Copy .py script out to projects/def-jdilwort/$USER/

- *cp Genomics-Pipelines/Process-NGS-RawReads/ngs_processing_pipeline.py .*

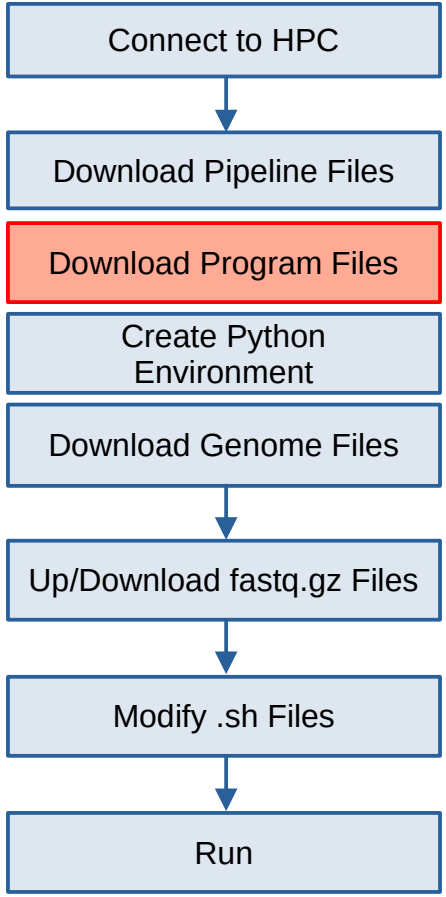
2) Copy .sh scripts too…

- *cp Genomics-Pipelines/Process-NGS-RawReads/ngs_pipeline_*.sh .*

earezza@beluga2:~/projects/def-jdilwort/earezza

```
[earezza@beluga2 earezza]$ git clone https://github.com/Dilworth-Laboratory/Genomics-Pipelines.git
Cloning into 'Genomics-Pipelines'...
remote: Enumerating objects: 501, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 501 (delta 13), reused 10 (delta 10), pack-reused 487
Receiving objects: 100% (501/501), 371.32 KiB | 1.34 MiB/s, done.
Resolving deltas: 100% (289/289), done.
[earezza@beluga2 earezza]$ ls
Genomics-Pipelines
[earezza@beluga2 earezza]$ ls Genomics-Pipelines/Process-NGS-RawReads/
cc_requirements.txt      ngs_pipeline.png        ngs_processing_pipeline.py
ngs_pipeline_CUTnTag.sh  ngs_pipeline_RNASeq.sh  README.md
[earezza@beluga2 earezza]$
[earezza@beluga2 earezza]$
[earezza@beluga2 earezza]$ cp Genomics-Pipelines/Process-NGS-RawReads/ngs_processing_pipeline.py .
[earezza@beluga2 earezza]$
```

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Setup – Manually Download Programs

1) Picard Tools (managing bam files)

- wget https://github.com/broadinstitute/picard/releases/download/3.0.0/picard.jar

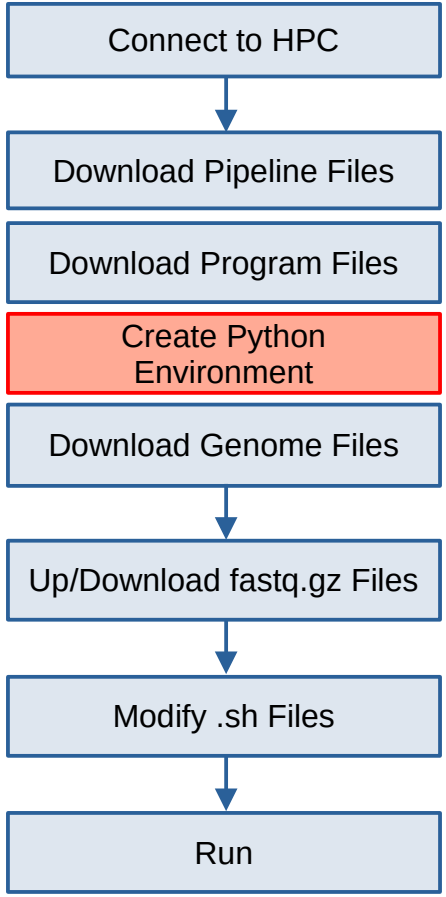
2) SEACR (peak caller)

- git clone https://github.com/FredHutch/SEACR.git

3) GoPeaks (peak caller)

- wget -O gopeaks https://github.com/maxsonBraunLab/gopeaks/releases/download/v1.0.0/gopeaks-linux-amd64
- chmod +x gopeaks

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Setup – Manually Create Python Environment

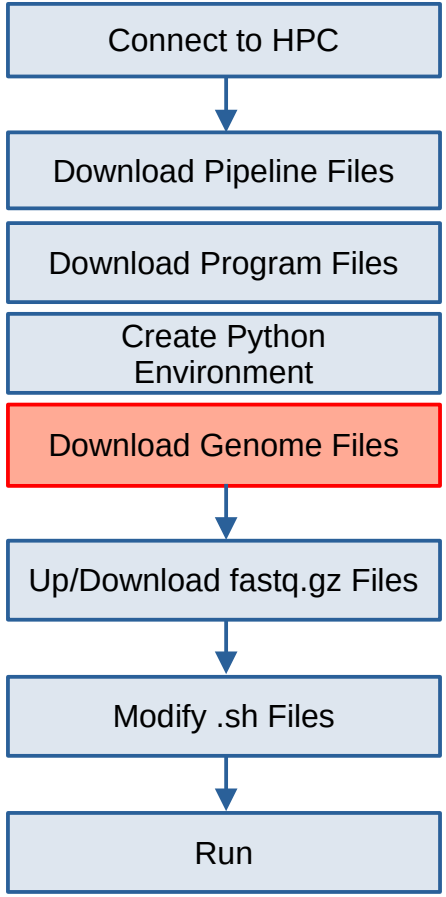
Create a virtualenv from cc_requirements.txt

- *module load python/3.9*
- *virtualenv --no-download ngsENV*
- *source ngsENV/bin/activate*
- *pip install --no-index --upgrade pip*
- *pip install -r Genomics-Pipelines/Process-NGS-RawReads/cc_requirements.txt*

Reference for Compute Canada: https://docs.alliancecan.ca/wiki/Python

Note: Many packages + software already on Compute Canada and can be loaded without installing beforehand

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Setup – Manually Downloading Genome Files

Reference genome files:

- Bowtie2 index files for mm10, hg38, etc…
  (unspliced alignment for CUT&Tag, ChIP-Seq, etc…)
  https://support.illumina.com/sequencing/sequencing_software/igenome.html

- Hisat2 index files for mm10, hg38, etc…
  (spliced alignment for RNA-Seq)
  https://daehwankimlab.github.io/hisat2/download/

Setup – Manually Downloading Genome Files

1) Copy URL from desired assembly and download with wget command:
   - wget <URL>

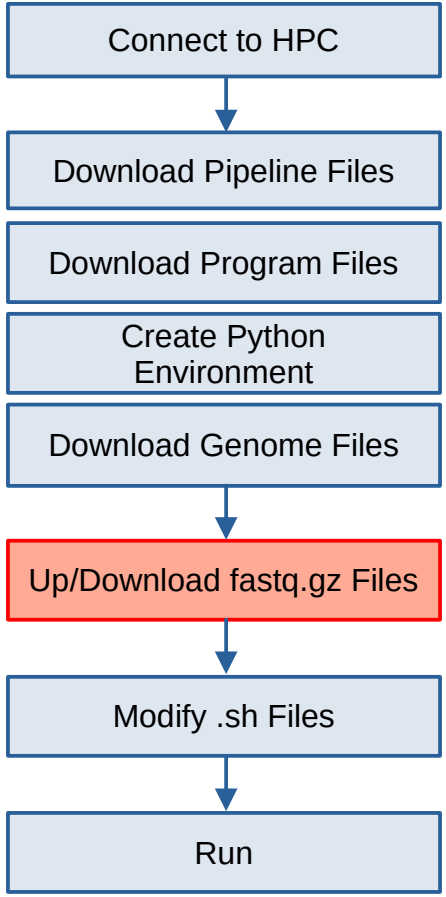
```
earezza@beluga4:~/projects/def-jdilwort/earezza/Reference_Files
[earezza@beluga4 Reference_Files]$ wget http://igenomes.illumina.com.s3-website-us-east-1.amazonaws.com/Mus_musculus/UCSC/mm10/Mus_musculus_UCSC_mm10.tar.gz
--2023-05-12 11:52:18--  http://igenomes.illumina.com.s3-website-us-east-1.amazonaws.com/Mus_musculus/UCSC/mm10/Mus_musculus_UCSC_mm10.tar.gz
Resolving igenomes.illumina.com.s3-website-us-east-1.amazonaws.com... 52.217.206.61, 52.217.116.229, 52.217.71.187, ...
Connecting to igenomes.illumina.com.s3-website-us-east-1.amazonaws.com|52.217.206.61|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 17681885302 (16G) [application/x-tar]
Saving to: ‘Mus_musculus_UCSC_mm10.tar.gz’

Mus_musculus_UCSC_mm10.tar.gz          100%[===================================================================>]  16.47G  21.8MB/s    in 18m 25s

2023-05-12 12:10:43 (15.3 MB/s) - ‘Mus_musculus_UCSC_mm10.tar.gz’ saved [17681885302/17681885302]
```

2) Extract tarball
   - tar -xzvf <filename.tar.gz>
3) Bowtie2 index files start with “genome” found in:
   - Mus_musculus/UCSC/mm10/Sequence/Bowtie2Index/

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Setup – Preparing Reads

Reads filenames must follow the conventions:

- Paired-end:
  - name-of-read_R1.fastq.gz
  - name-of-read_R2.fastq.gz
- Single-end:
  - name-of-read_1.fastq.gz
  - name-of-read_2.fastq.gz
  - name-of-read_3.fastq.gz

Underscores ONLY to indicate replicate & forward/reverse read numbers

Setup – Preparing Reads

Reads should be organized into their respective folders by <u>conditions & replicates</u>:

- WT-1/:
  - WT-rep1_R1.fastq.gz
  - WT-rep1_R2.fastq.gz
- WT-2/:
  - WT-rep2_R1.fastq.gz
  - WT-rep2_R2.fastq.gz
- KO-1/:
  - KO-rep1_R1.fastq.gz
  - KO-rep1_R2.fastq.gz
- KO-2/:
  - KO-rep2_R1.fastq.gz
  - KO-rep2_R2.fastq.gz

Setup – Preparing Reads

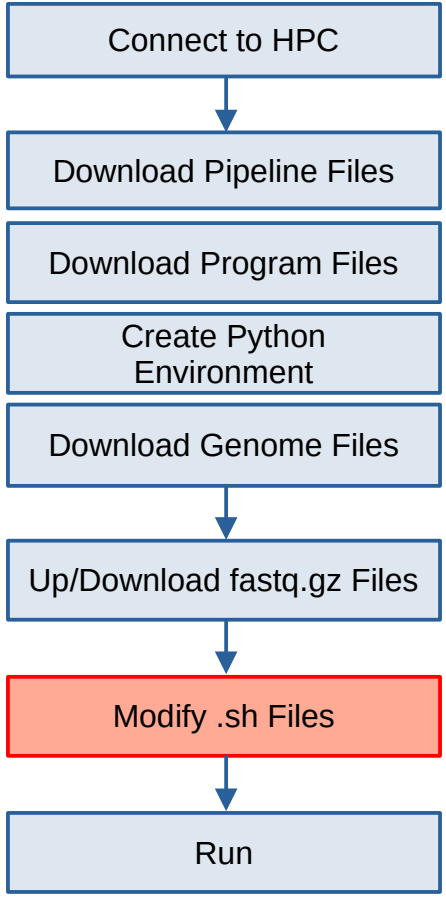
Folders with reads should be stored in ~/scratch/

If uploading from local computer to HPC:

- *scp -r WT-1/ earezza@beluga.computecanada.ca:~/scratch/*
- *scp -r WT-2/ earezza@beluga.computecanada.ca:~/scratch/*
- *scp -r KO-*/ earezza@beluga.computecanada.ca:~/scratch/*

Otherwise, download reads as necessary from sequencing facility and format to filename conventions.

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Setup – Modify Batch Scripts

```
#!/bin/bash
#SBATCH --time=24:00:00
#SBATCH --account=def-jdilwort
#SBATCH --cpus-per-task=32
#SBATCH --mem-per-cpu=1G
#SBATCH --array=0-2
#SBATCH --job-name=CUTnTag_Process
#SBATCH --output=%x-%j.out

module load python/3.9 scipy-stack/2021a
module load samtools>=1.11 r>=4.0.5 bowtie2>=2.4.1 fastqc>=0.11.9
module load hisat2
module load java
source ngsENV/bin/activate
rm *=*

files=(SAMPLE_1 SAMPLE_2 SAMPLE_3)

python ngs_processing_pipeline.py ...
```

Defines requested HPC parameters and batch job options

24 hrs requested
Jeff's group account
32 CPUs
1G per CPU

3 samples of reads (0,1,2)

Generic batch job name and output filename

Load pre-installed programs (on Compute Canada)

Activate Python environment

List of reads folders to process, space-separated

Run pipeline with supplied options

Setup – Modify Batch Scripts

Example:

python ngs_processing_pipeline.py --reads READS_DIR/ --species Mus -length 100 -technique cnt

Executes pipeline script

Define options

- Common Options to Change:

  --species

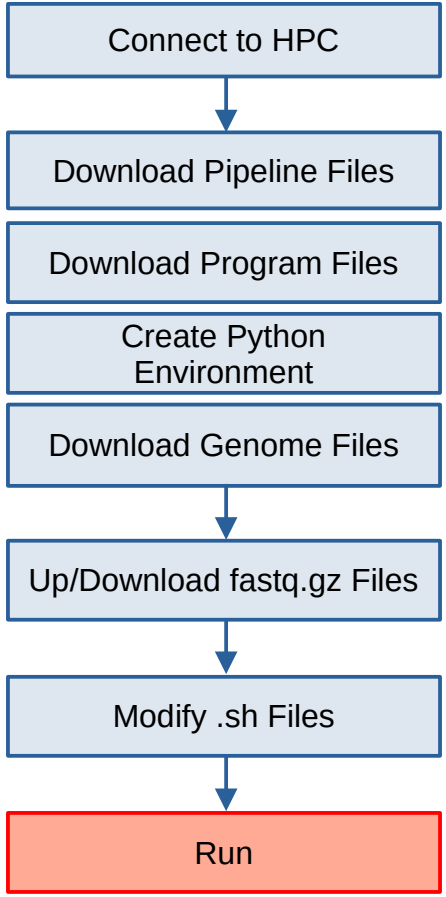
  --technique

  --reads_type

  --genome_index

- python ngs_processing_pipeline.py --help

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Running – Monitoring & Debugging

Run your modified batch script

- *sbatch ngs_pipeline_RNASeq.sh*

Monitor submitted jobs with:

- *sq*

```
[earezza@beluga3 earezza]$ sq
          JOBID     USER      ACCOUNT           NAME  ST  TIME_LEFT NODES CPUS TRES_PER_N MIN_MEM NODELIST (REASON)
     37626950_0  earezza def-jdilwort RNASeq_Process   R   23:25:59     1   32        N/A      1G bc21113 (None)
     37626950_1  earezza def-jdilwort RNASeq_Process   R   23:42:08     1   32        N/A      1G bc11934 (None)
     37626950_2  earezza def-jdilwort RNASeq_Process   R   23:42:08     1   32        N/A      1G bc11935 (None)
     37626950_3  earezza def-jdilwort RNASeq_Process  PD 1-00:00:00     1   32        N/A      1G  (Nodes required for
 job are DOWN, DRAINED or reserved for jobs in higher priority partitions)
 37626961_[0-3]  earezza def-jdilwort CUTnTag_Proces  PD 1-00:00:00     1   32        N/A      1G  (Priority)
[earezza@beluga3 earezza]$
```

Cancel submitted jobs with:

- scancel <jobID number>

Running – Monitoring & Debugging

Alternatively, once “ST” is “R” or job ended, you can view output log files

- Main log file is ~/scratch/Sample/logs/Sample.log

```
[earezza@beluga4 earezza]$ tail ~/scratch/CUTNTAG/WT_1/logs/WT_1.log
INFO : __main__ : Step 10/13 - Compileresults_filtering
INFO : __main__ : PASSED - duration: 0.0775 minutes
INFO : __main__ : Step 11/13 - GetBigwigs_BamCoverage
INFO : __main__ : PASSED - duration: 15.764 minutes
INFO : __main__ : Step 12/13 - Peak_Calling
INFO : __main__ : PASSED - duration: 5.1771 minutes
INFO : __main__ : Step 13/13 - Clean_up
INFO : __main__ : PASSED - duration: 0.0002 minutes
INFO : __main__ : Pipeline COMPLETE!
INFO : __main__ : Duration: 75.1989 minutes
[earezza@beluga4 earezza]$
```

- More detailed logs for each pipeline step are also in ~/scratch/Sample/logs/

Output Files

Once complete, you can navigate the resulting folders for specific files (.bw, .bed, etc…) and download (scp) results

```
[earezza@beluga4 earezza]$
[earezza@beluga4 earezza]$ ls ~/scratch/CUTNTAG/WT_1/
All_output        logs        WT-H3-3-1_R1.fastq.gz      WT-H3-3-1_R2.fastq.gz
Analysis_Results  md5sum.txt  WT-H3-3-1_R1.fastq.gz.md5  WT-H3-3-1_R2.fastq.gz.md5
[earezza@beluga4 earezza]$
```

logs/...log (monitor the progress of the script and troubleshoot problems)

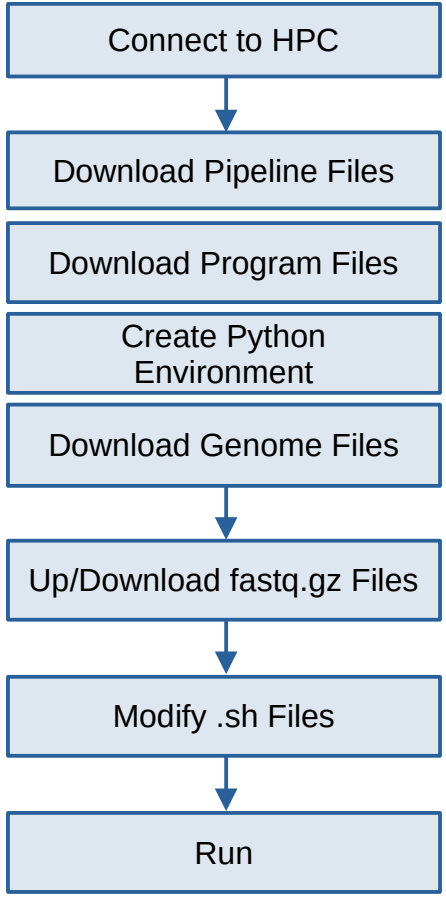
Analysis_Results/QC_Rawreads/...html (quality check raw reads and modify input options/re-run if required)

Analysis_Results/Peaks/...stringent.bed...peaks.narrowPeak...gopeaks_peaks.bed (peaks files identifying enriched regions, useful in downstream analysis)

Analysis_Results/Normalized_and_Unnormalized_BigWigs/Normalized/...bw (normalized bigwigs for viewing coverage in genome browsers)

All_output/Processed_reads/...bam..bai (alignment+index files (should always be together), required for many analysis tools)

Overview


Connect to HPC
Download Pipeline Files
Download Program Files
Create Python Environment
Download Genome Files
Up/Download fastq.gz Files
Modify .sh Files
Run

Questions?

Contact:
earezza@ohri.ca