Documentation for Aligning Fastq Files and Making CGmap files using bsbolt

Part 1: Alignment of FastqFiles

Step 1: Setting up sample ids for array job

* On computer:
  + Get sample identifiers of the trimmed fastq files listed in a txt file. The name that corresponds to each matched pair
    - Example for paired reads:
      * SAMPLEID\_R1\_trimmed.fastq.gz
      * SAMPLEID\_R2\_trimmed.fastq.gz
* On Hoffman2:
  + Convert the file to be unix compatible
  + On command line:

dos2unix sample\_ids.txt

Step 2:

* On Hoffman2:
  + Using rclone copy fastq.gz files to a folder in your account on Hoffman
  + Follow rclone instructions on Pellegrini Resources page

Step 3:

* On Hoffman2:
  + Create shell scripts using bsbolt Align script, one for the actual work script, and one for setting up an array job submission. See example scripts below:
    - Scripts have been adapted from scripts by Shawn Cokus (QCBio BS-Seq DNA Methylation Workshop) and scripts provided by Colin Farrell (Pellegrini Lab Post Doc)
    - Replace path folders and sample id files with your own corresponding folders/files

bsboltAlign\_WorkScript.bash

#!/bin/bash

# script adapted from scripts by Shawn Cokus (QCBio BS-Seq workshop) and scripts by Colin Farrell

#$ -pe shared 4

#$ -l h\_rt=6:00:00

#$ -l h\_data=8G

#$ -o /u/scratch/m/mpleasur/align\_outputs/

#$ -e /u/scratch/m/mpleasur/align\_outputs/

#$ -M mpleasure@ucla.edu

#$ -m a

#Get inputs from submission script

files="${1}";        shift

fastq\_files="${1}";  shift

output\_files="${1}"; shift

index\_file="${1}";  shift

#Move to the output folder

cd "${output\_files}"

#Load needed modules

. /u/local/Modules/default/init/modules.sh

module load python/3.7.3

#Get the sample name

sample=$(cat $files | head -${SGE\_TASK\_ID} | tail -1)

#Runs align script

python3 -m bsbolt Align -t 8 -DB ${index\_file} -F1 ${fastq\_files}${sample}\_R1\_trimmed.fastq.gz -F2 ${fastq\_files}${sample}\_R2\_trimmed.fastq.gz -O ${output\_files}${sample} -T 110 >${output\_files}${sample}.log

bsboltAlign\_JobSubmissionScript.bash

#!/bin/bash

# Script adapted from scripts by Shawn Cokus (QCBio BS-Seq workshop) and scripts by Colin Farrell

#$ -cwd

#$ -o joblog.$JOB\_ID.$TASK\_ID

#$ -j y

#$ -l h\_rt=8:00:00

#$ -l h\_data=8G

#This is your sample identifiers file

files="/u/scratch/m/mpleasur/sample\_id.txt"

#This goes through the sample identifiers file and gets the #number of lines for how many submissions to send

num\_files=$(cat "$files" | wc -l)

#Path to files, where the outputs should go, and the index file

fastq\_files="/u/scratch/m/mpleasur/HBFastqFiles/Human\_BuccalSwab\_Turkish/"

output\_files="/u/scratch/m/mpleasur/HBFastqFiles/alignments/"

index\_file="/u/scratch/m/mpleasur/practice1/BSBolt"

#How many jobs can run simultaneously

JOBS=30

#Job array submission – the variables following the script call #is what passes variables inputted in this script to the work #script.

qsub -M mpleasure@g.ucla.edu -m a -t 1-$num\_files:1 -tc "${JOBS}" bsboltAlign\_WorkScript.bash "${files}" "${fastq\_files}" "${output\_files}" "${index\_file}" "$@"

Part 2: bsbolt CallMethylation function to get CGmap files

Step 1:

* After you have aligned all your fastq files you can now use the CallMethylation function in bsbolt
* On Hoffman2:
  + Make sure all your aligned bam files are in one folder location and that the sample ids still match the sample ids list from before
  + Create shell scripts, one for work and one for job submission. Below are example scripts:
    - Scripts have been adapted from scripts by Shawn Cokus (QCBio BS-Seq DNA Methylation Workshop) and scripts provided by Colin Farrell (Pellegrini Lab Post Doc)
    - Replace path folders and sample id files with your own corresponding folders/files

bsboltMethylCall\_work.bash

#!/bin/bash

# Script adapted from scripts by Shawn Cokus (QCBio BS-Seq workshop) and scripts by Colin Farrell

#$ -pe shared 4

#$ -l h\_rt=4:00:00

#$ -l h\_data=8G

#$ -o /u/scratch/m/mpleasur/methyl\_outputs/

#$ -e /u/scratch/m/mpleasur/methyl\_outputs/

#$ -m a

#Get variable values from submission script

files="${1}"; shift

aligned\_bam\_files="${1}"; shift

output\_files="${1}"; shift

index="${1}"; shift

#Move into output file folder

cd "${output\_files}"

#Load modules needed

. /u/local/Modules/default/init/modules.sh

module load python/3.7.3

module load samtools

#Get sample name

sample=$(cat $files | head -${SGE\_TASK\_ID} | tail -1)

#Samtools deduplicate and sort process

samtools fixmate -p -m ${aligned\_bam\_files}${sample}.bam ${aligned\_bam\_files}${sample}.fixmates.bam

samtools sort -@ 8 -o ${aligned\_bam\_files}${sample}.sorted.bam ${aligned\_bam\_files}${sample}.fixmates.bam

rm ${aligned\_bam\_files}${sample}.fixmates.bam

samtools markdup ${aligned\_bam\_files}${sample}.sorted.bam ${aligned\_bam\_files}${sample}.dup.bam

rm ${aligned\_bam\_files}${sample}.sorted.bam

samtools index ${aligned\_bam\_files}${sample}.dup.bam

#Call methylation bsbolt function

python3 -m bsbolt CallMethylation -t 8 -BQ 5 -MQ 0 -IO -min 1 -CG -O ${aligned\_bam\_files}${sample}.bam -I ${aligned\_bam\_files}${sample}.dup.bam -DB ${index} -IO > ${output\_files}${sample}.meth.log

bsboltMethylCall\_submit.bash

#!/bin/bash

# script adapted from scripts by Shawn Cokus (QCBio BS-Seq workshop) and scripts by Colin Farrell

#$ -cwd

#$ -o joblog.$JOB\_ID.$TASK\_ID

#$ -j y

#$ -l h\_rt=12:00:00

#$ -l h\_data=8G

#Sample ids to run through folder of files

files="/u/scratch/m/mpleasur/sample\_id\_21.txt"

#Get number of files

num\_files=$(cat "$files" | wc -l)

#The aligned files from Part 1

aligned\_bam\_files="/u/scratch/m/mpleasur/HBFastqFiles/alignments/bam\_alignments/"

#Where you want the output files to go

output\_files="/u/scratch/m/mpleasur/HBFastqFiles/methylation\_CGmaps/"

#Index file

index="/u/scratch/m/mpleasur/BSBolt/"

#How many jobs to run simultaneously

JOBS=30

#Job array submission script

qsub -M mpleasur@g.ucla.edu -m a -t 1-$num\_files:1 -tc "${JOBS}" bsboltMethylCall\_work.bash "${files}" "${aligned\_bam\_files}" "${output\_files}" "${index}" "$@"