MG_HW8: Optimized assembly, read recruitment, and gene finding Version 3

Bonnie Hurwitz

Abstract

Citation: Bonnie Hurwitz MG_HW8: Optimized assembly, read recruitment, and gene finding. **protocols.io** dx.doi.org/10.17504/protocols.io.gbhbsj6

Published: 04 Nov 2016

Protocol

Step 1.

Login to the HPC and move over to the ICE cluster.

cmd COMMAND

```
ssh hpc
ice
```

NOTES

Bonnie Hurwitz 04 Nov 2016

If you have meno enabled, select option 3.

Step 2.

Move into the class directory

cmd COMMAND

```
cd /rsgrps/bh_class/username
```

NOTES

Bonnie Hurwitz 04 Nov 2016

change "username" to YOUR user name.

Step 3.

Create directories for each of the new analyses we will run. Note these steps will be similar to what you originally ran, but with optimized steps.

cmd COMMAND

```
mkdir assembly read_recruit gene_finding
```

Step 4.

We are ready to run the assembly. Create all of the necessary directories for storing the results.

cmd COMMAND

```
cd  /rsgrps/bh_class/username/assembly
mkdir std-err std-out
```

Step 5.

Create a script to run the assembly called run-assembly.sh.

cmd COMMAND

```
#!/bin/bash

#PBS -W group_list=bh_class
#PBS -q windfall
#PBS -l select=1:ncpus=12:mem=23gb
#PBS -l pvmem=22gb
#PBS -l walltime=24:00:00
#PBS -l cput=24:00:00
#PBS -M netid@email.arizona.edu
#PBS -m bea

FASTQ_DIR='/rsgrps/bh_class/username/unmapped'
ASSEM_DIR='/rsgrps/bh_class/username/assembly'
MIN_CONTIG_LEN=500
OUT_DIR='/rsgrps/bh_class/username/assembly/megahit-out'

cd $ASSEM_DIR

R1s=`ls $FASTQ_DIR/*.paired.1.fastq | python -
c 'import sys; print ",".join([x.strip() for x in sys.stdin.re
adlines()])'`
R2s=`ls $FASTQ_DIR/*.paired.2.fastq | python -
c 'import sys; print ",".join([x.strip() for x in sys.stdin.re
adlines()])'`
SINGLEs=`ls $FASTQ_DIR/*.singletons.fastq | python -
c 'import sys; print ",".join([x.strip() for x in sys.st
din.readlines()])'`

megahit -1 $R1s -2 $R2s -r $SINGLEs --preset meta-sensitive --min-contig-
len $MIN_CONTIG_LEN -o $OUT_DIR -t 12
```

NOTES

Bonnie Hurwitz 04 Nov 2016

don't forget to update with your netid and username.

ANNOTATIONS

Bonnie Hurwitz 04 Nov 2016

My exit status was 2 with this (below) in my std-err for this step. What should I change?

```
File "<string>", line 2
adlines()])
^
SyntaxError: invalid syntax
```

```
File "<string>", line 2
adlines()])
^
SyntaxError: invalid syntax
File "<string>", line 2
din.readlines()])
^
SyntaxError: invalid syntax
23.0Gb memory in total.
Using: 21.227Gb.
megahit: Number of paired-end files not match!
```

Step 6.

Run the assembly on the cluster.

cmd COMMAND

```
chmod 755 run-assembly.sh
qsub -e std-err/ -o std-out/ run-assembly.sh
```

Step 7.

Look at the quality of the assembly. How many contigs did you get? What is the average contig length, and N50? Is this an improvement over the first assembly you ran?

Step 8.

Now we will re-run the read recruitment back to the assembly as we did before. This time we will account for the read pairing in the read files.

cmd COMMAND

```
cd ../read_recruit
mkdir bam
mkdir bt2_index
```

Step 9.

Move into your assembly directory containing your contigs. We will need to simplify the fasta headers in the contig file

cmd COMMAND

```
cd ../assembly/megahit-out
module load fastx/0.0.14
fastx_renamer -n COUNT -i final.contigs.fa -o fixed-contigs.fa
```

Step 10.

Move into the contig indexing directory. And create the contig index.

cmd COMMAND

```
cd /rsgrps/bh_class/username/read_recruit/bt2_index
module load bowtie2/2.2.5
bowtie2-build -f /rsgrps/bh_class/username/assembly/megahit-out/fixed-
contigs.fa contig_index
```

Step 11.

Now move into the bam directory you created earlier and create std-err and std-out directories for running bowtie on the cluster.

cmd COMMAND

```
cd /rsgrps/bh_class/username/read_recruit/bam
mkdir std-err
mkdir std-out
```

Step 12.

Copy the following into a new script called bt2_align.sh:

cmd COMMAND

```
#!/bin/bash

#PBS -W group_list=bh_class
#PBS -q windfall
#PBS -l jobtype=cluster_only
#PBS -l select=1:ncpus=4:mem=15gb
#PBS -l pvmem=14gb
#PBS -l place=pack:shared
#PBS -l walltime=24:00:00
#PBS -l cput=24:00:00
#PBS -M netid@email.arizona.edu
#PBS -m bea

echo “my job_id is: ${PBS_JOBID}”

#####change here #######
FASTQ_DIR="/rsgrps/bh_class/username/unmapped"
BT2_INDEX="/rsgrps/bh_class/username/read_recruit/bt2_index/contig_index"
OUT_DIR="/rsgrps/bh_class/username/read_recruit/bam"
CONTIGS="/rsgrps/bh_class/username/assembly/megahit-out/fixed-contigs.fa"
#######################

cd $FASTQ_DIR
export FASTQ_LIST="$FASTQ_DIR/fastq-list"
ls *fastq | cut -d '.' -f 1 | sort | uniq > $FASTQ_LIST
echo "Samples to be processed:" $(cat $FASTQ_LIST)

module load bowtie2/2.2.5
module load samtools/1.3.1

while read FASTQ; do

    R1=$FASTQ".paired.1.fastq"
    R2=$FASTQ".paired.2.fastq"
    S=$FASTQ".singletons.fastq"
    OUT=$file

    bowtie2 -x $BT2_INDEX -1 $R1 -2 $R2 -U $S -q --maxins 800 --fr --very-sensitive-local -
p 4 -S $OUT_DIR/$FASTQ.sam

    cd $OUT_DIR
    echo "Converting $FASTQ.sam using reference $CONTIGS"
    samtools view -@ 16 -bT $CONTIGS $FASTQ.sam > $FASTQ.temp
    echo "Sorting $FASTQ"
    samtools sort -@ 16 $FASTQ.temp > $FASTQ.bam
    echo "Removing $FASTQ.temp"
    rm $FASTQ.temp
    cd $FASTQ_DIR

done < $FASTQ_LIST
```

NOTES

Bonnie Hurwitz 04 Nov 2016

Should this script be saved in the bam directory or the bt2_index directory?

Step 13.

Submit the job on the cluster to align reads via bowtie

cmd COMMAND

```
chmod 755 bt2_align.sh
qsub -e std-err/ -o std-out/ bt2_align.sh
```

NOTES

Bonnie Hurwitz 04 Nov 2016

Are the std-err and std-out directories in read_recruit/bt2_index directory or just read_recruit? (I also have a std-err and std-out in the read_recruit/bam dirctory as per step 11. Should I keep that?). I think I need a little more clarification as to which directories should have std-err and std-out directories and which shouldn't?

Step 14.

While the read recruitment is running, we are going to run the gene finding on the new assembly.

cmd COMMAND

```
cd ../gene_finding
mkdir std-err std-out
```

Step 15.

Create a script called run-prodigal.sh to get the gene calls.

cmd COMMAND

```
#!/bin/bash

#PBS -W group_list=bh_class
#PBS -q windfall
#PBS -l jobtype=cluster_only
#PBS -l select=1:ncpus=4:mem=15gb
#PBS -l pvmem=14gb
#PBS -l place=pack:shared
#PBS -l walltime=24:00:00
#PBS -l cput=24:00:00
#PBS -M netid@email.arizona.edu
#PBS -m bea

echo “my job_id is: ${PBS_JOBID}”
```

```
#####change here #######
CONTIG_DIR="/rsgrps/bh_class/username/assembly/megahit-out"
OUT_DIR="/rsgrps/bh_class/username/gene_finding"
#######################

GENE_CALLS="$OUT_DIR/gene_calls"
PROTEINS="$OUT_DIR/proteins.faa"
NUCLEO="$OUT_DIR/nucleotides.faa"
CONTIGS="$CONTIG_DIR/fixed-contigs.fa"

cd $CONTIG_DIR

module load prodigal/2.6.2

prodigal -i $CONTIGS -o $GENE_CALLS -a $PROTEINS -d $NUCLEO
```

Step 16.

Run the gene finding script on the cluster.

cmd COMMAND

```
chmod 755 run-prodigal.sh
qsub -e std-err/ -o std-out/ run-prodigal.sh
```

Step 17.

In the next protocol (MG_HW9), we will be converting these output files into files we can load into Anvio for data visualization. We will also be reporting on our findings in the google report.