SIBG INFORMATICS GROUP

PHYLUCE TUTORIAL

TUTORIAL MODIFIED FROM BRANT FAIRCLOTH'S HTTP://PHYLUCE.READTHEDOCS.ORG/EN/LATEST/TUTORIAL-ONE.HTML. IT HAS BEEN TAILORED TO THE SMITHSONIAN HPC CLUSTER, HYDRA, BY THE SIBG BIOINFORMATICS GROUP.

SLIDES ARE NOT COMPREHENSIVE, BUT ARE MEANT TO ACCOMPANY THE MARKDOWN TUTORIAL FILE, WHICH CONTAINS COMPLETE INSTRUCTIONS.

MAJOR PIPELINE STEPS

- 1. Get the data
- 2. Count the read data
- 3. Clean the read data
- 4. Assemble the data
- 5. Assembly QC
- 6. Finding UCE loci
- 7. Extracting UCE loci
- 8. Exploding the monolithic FASTA file
- 9. Aligning UCE loci
- 10. Alignment cleaning
- 11. Final data matrices
- 12. Preparing data for RAxML and ExaML

HINTS

- You will generate 17 job files during this tutorial. Name them so that you will remember what they are.
- You should be able to reproduce this pipeline for your own data pretty easily. The file names for your reads might be something you have to adjust.
- module: phyluce_tg
- Log in to either
  - USER@hydra-login01.si.edu or
  - USER@hydra-login02.si.edu

1. GET THE DATA

- Here, we will process raw Illumina UCE data for 4 taxa that were enriched with the 5000 UCE Tetrapod probe set:

  1. *Mus musculus* (PE100)
  2. *Anolis carolinensis* (PE100)
  3. *Alligator mississippiensis* (PE150)
  4. *Gallus gallus* (PE250)

- Data are in /pool/genomics/tutorial_data
- Copy them to /pool/genomics/USER/uce-tutorial/raw-fastq

2. COUNT THE READ DATA

- hint: use the QSub Generator:
  - https://hydra-3.si.edu/tools/QSubGen
- change to cwd: Checked (keep checked for all job files)
- join stderr & stdout Checked (Keep checked for all job files)
- command:

```
for i in *R1*.fastq.gz;
do echo $i;
gunzip -c $i | wc -l | awk '{print $1/4}';
done
```

- upload your job file using scp from your local machine into the raw-fastq directory
- hint: submit the job on Hydra using qsub

3. CLEAN THE READ DATA

- There are many tools for this. We have modified phyluce's illumiprocessor to use Trim Galore! instead of Trimmomatic. Trim Galore! has the needed functionality but behaves better on Hydra (i.e. does not use Java).
- JOB FILE #2: illumiprocessor
- command: illumiprocessor
- arguments: (Note: the arguments should start on the same line as the command. The '\' in the arguments allows them to span multiple lines.)

```
--input raw-fastq \
--output clean-fastq \
--config illumiprocessor.conf \
--paired \
--cores $NSLOTS
```

4. ASSEMBLE THE DATA

- We will use Trinity to assemble the data into contigs. There will be a separate Trinity run for each sample in your dataset. This is the most time consuming computer intensive portion of the pipeline. For today's tutorial, we will not have time to complete the assemblies.
    - Copy the directory with completed assemblies: pool/genomics/tutorial_data/trinity-assemblies to your uce-tutorial directory.
    - hint: use cp -r.
    - Find the contigs!
    - Skip to the next section (#5). If you have extra time now, feel free to make the Trinity job file (JOB FILE #3) but wait to submit it until later.

5. ASSEMBLY QC

- JOB FILE #4: get FASTA lengths
- command:

```
for i in trinity-assemblies/contigs/*.fasta;
 do phyluce_assembly_get_fasta_lengths --input $i --csv;
 done
```

- Check the log file for output similar to the below (header not included):

```
samples,contigs,total bp,mean length,95 CI length,min length,max length,median legnth,contigs >1kb
alligator_mississippiensis.contigs.fasta,10587,5820479,549.776046094,3.5939422934,224,11285,413.0,1182
anolis_carolinensis.contigs.fasta,2458,1067208,434.177379984,5.72662897806,224,4359,319.0,34
gallus_gallus.contigs.fasta,19905,8841661,444.192966591,2.06136172068,224,9883,306.0,1530
mus_musculus.contigs.fasta,2162,1126231,520.920906568,7.75103292163,224,6542,358.0,186
```

6. FINDING USE LOCI

- Now we want to run lastz to match contigs to the UCE probe set and to remove duplicates.The search matches are stored in a sqlite database.
- sqlite is a database system that phyluce uses to store information about the contigs such as presence/absence. Phyluce scripts access this data and advanced users can access this database directly.
- Before we locate UCE loci (in other words, match your contigs to the UCE probes), you need to get the probe set used for the enrichments.
- JOB FILE #5: Match contigs to probes

7. EXTRACTING UCE LOCI

- Now that we have located UCE loci, we need to determine which taxa we want in our analysis, create a list of those taxa, and also a list of which UCE loci we enriched in each taxon (the “data matrix configuration file”). We will then use this list to extract FASTA data for each taxon for each UCE locus.

- First, we need to decide which taxa we want in our “taxon set." Create a configuration file.

- hint: use nano to create a file called taxon-set.conf in uce-tutorial listing the taxa you want to include like so:

```
[all]
alligator_mississippiensis
anolis_carolinensis
gallus_gallus
mus_musculus
```

- JOB FILE #7: get FASTA data for taxa in our taxon set

- JOB FILE #6: Get match counts

8. EXPLODING THE MONOLITHIC FAST FILE

- We can "explode" the monolithic fasta file into a file of UCE loci that we have enriched by taxon in order to get individual statistics on UCE assemblies for a given taxon.
- JOB FILE #8: explode the monolithic FASTA file
- JOB FILE #9: get summary stats on the FASTAs

9. ALIGNING UCE LOCI

- When you align UCE loci, you can either leave them as-is, without trimming, edge trim, or internal trim. See the PHYLUCE docs for more details about why you might choose one over the other. By default, edge-trimming is turned on. We will use MAFFT.
- JOB FILE #10: align with edge-trimming
- JOB FILE #11: get alignment summary data
- JOB FILE #12: align with no-trim and output FASTA
- JOB FILE #13: internal trim with Gblocks
- JOB FILE #14: get alignment summary data

10. ALIGNMENT CLEANING

- Each alignment now contains the locus name along with the taxon name. This is not what we want downstream, so we need to clean our alignments. For the remainder of this tutorial, we will work with the Gblocks trimmed alignments, so we will clean those alignments:
- JOB FILE #15: clean alignments

11. FINAL DATA MATRICES

- To create a 75% data matrix (i.e. 25% or less missing), run the following. Notice that the integer following –taxa is the total number of organisms in the study.

- JOB FILE #16: create a 75% data matrix

12. PREPARING DATA FOR RAXML AND EXAML

- Here we will formatting our 75p data matrix into a phylip file for RAxML or ExaML.

- JOB FILE #17: generate a phylip file