How to get to HPCC and load mothur

1. open terminal. Type >ssh –X [username@hpcc.msu.edu](mailto:username@hpcc.msu.edu)
2. enter password
3. select node. Type >ssh devintel-07 – Now you are already in your home space
4. Find the listed files (>ls) or change directory (>cd) to the desired files or folder
5. If you intend to work interactively, see number 6. If you want to work through job schedule, see number 7.
6. WorkLoad tools – mothur. Type >module load mother/1.31.2b – Tada~~ you are already in mother :D Get started! To quit, type >quit()

\*if you are working with mother installed on your computer, use command below

cd mothur

./mothur

1. Type > qsub {script name}. Wait for a while, then type >qstat –u {netID}. After a while (again), >showq –u {netID}, or qstat –f {netID} All those 3 commands will pull info about the status and others.

MiSeq workflow

#make contigs from sequencing data

MiSeq workflow – cheatsheet for Shairah

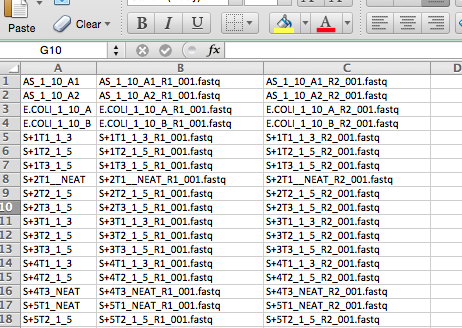
I. download all sequence data (make BACK-UP)

II. need to create contigs file (example SA.txt) consist of fwd and reverse fastq file names

# copy n paste the sample and fastq file names in excel

#formula for copy all rows in Excel (=A1&F$1)

# save as tab delimited (txt)



Fwd seq

Samples / treatment

Reverse seq

III. create contigs (need to be done in HPCC account)

#use command below

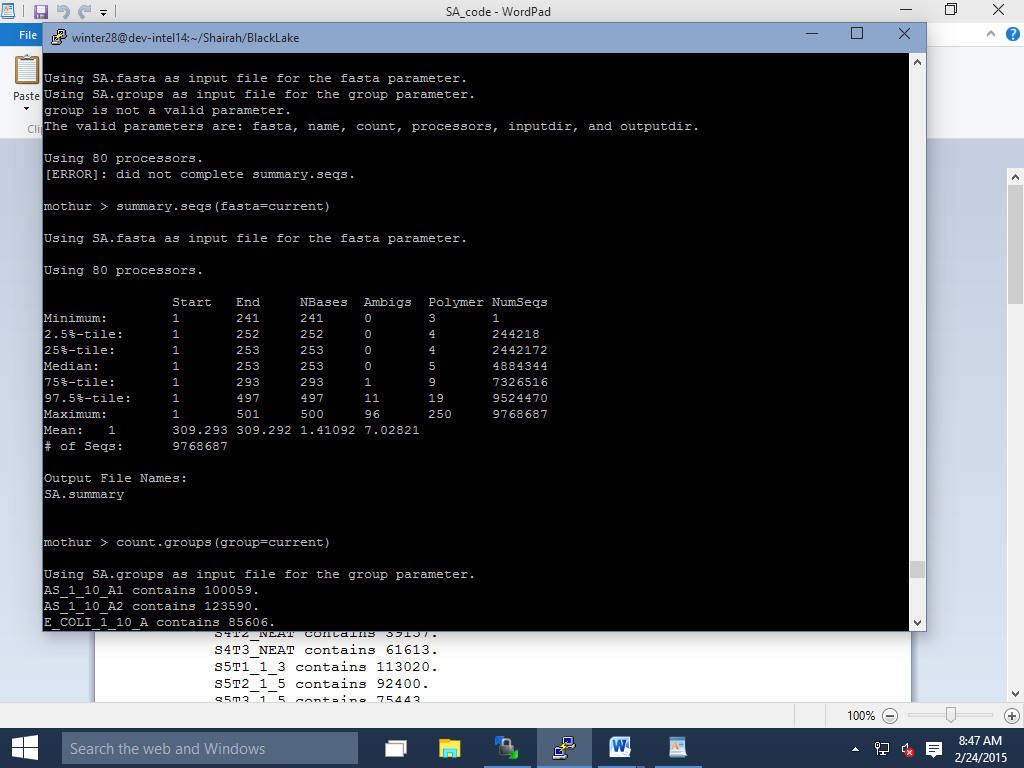
#the output fasta file (SA.fasta will be generated)

**make.contigs(file=SA.txt, bdiffs=0, pdiffs=0, processors=80)** \*processor depend on your comp/pc

IV. Get to know your data a.k.a Quality Control

#next command will gives info abt the starts, stops, and length of your sequence data

**summary.seqs(fasta=SA.fasta)**



#next command will gives info abt how many sequence generated in samples (first screening) \*advise to be also used when everything being cleaned and processed later on

**count.groups(group=SA.groups)**

AS\_1\_10\_A1 contains 100059.

AS\_1\_10\_A2 contains 123590.

E\_COLI\_1\_10\_A contains 85606.

E\_COLI\_1\_10\_B contains 100853.

S1T1\_1\_3 contains 132779.

S1T2\_NEAT contains 91650.

S1T3\_1\_5 contains 94308.

S2T1\_1\_3 contains 100447.

S2T2\_1\_5 contains 77972.

S2T3\_1\_5 contains 103105.

S3T1\_1\_3 contains 114794.

S3T2\_1\_5 contains 47104.

S3T3\_1\_5 contains 76477.

S4T1\_NEAT contains 81994.

S4T2\_NEAT contains 39157.

S4T3\_NEAT contains 61613.

S5T1\_1\_3 contains 113020.

S5T2\_1\_5 contains 92400.

S5T3\_1\_5 contains 75443.

S6T1\_1\_3 contains 54089.

This can be discarded – very low # of sequences

S6T2\_1\_5 contains 101923.

S6T3\_1\_5 contains 90208.

SBS1A contains 73869.

SBS1B contains 67990.

SBS2A contains 46759.

SBS2B contains 119097.

SBS3A contains 165.

#next command will help to remove seq that don't fit the parameters ( seq with wrong alignment region, truncated based on length, start or end position, contain homoplymer, etc)

**screen.seqs(fasta=SA.fasta, group=SA.groups, summary=SA.summary, maxambig=0, maxhomop=8, maxlength=300)**

#output files names:

SA.good.summary

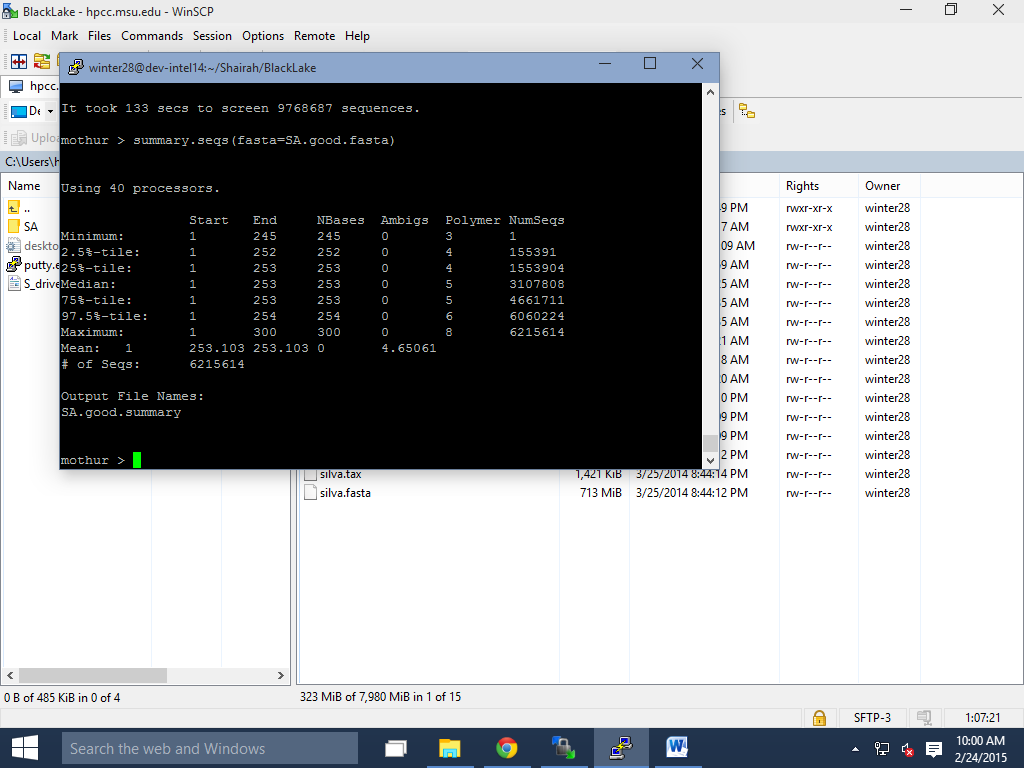
# then can run check again

# use **summary.seqs (fasta=SA.good.fasta)**

SA.good.fasta

SA.bad.accnos

SA.good.groups



V. Sequence analyses

- find unique sequence

-do sequence alignment

-Chimera checking

#next command will create names files that bin sequence that are fully identical

**unique.seqs(fasta=SA.good.fasta)**

#output file names:

# then can run check again

# use **summary.seqs (fasta=SA.good.unique.fasta, name=SA.good.names)**

#output file names SA.good.uniques.summary

SA.good.names

SA.good.unique.fasta

#next command will gives info abt name of the group for each column, name of unique sequences in each row and how many of uniques seq appear in each group

**count.seqs(**name=stability.trim.contigs.good.names, group=stability.contigs.good.groups)

#next command will do the sequence alignment

**align.seqs(fasta=SA.good.unique.fasta, reference=silva.fasta, processors=40)**

**\*what’s the difference between align.seqs and pcr.seqs command?**

1.  Can  do  a  global  Clustal  alignment,  which  calculates  all  pairwise  distances  and  builds  distances  from  alignment  tree. Tends  to  increase  distances.

2.  Pairwise  alignments  calculted  without  guide  tree.  Tends  to  have  smaller  distances.

3.  Profile‐based  alignment.  Align  to  a  reference  sequence\*\*

4.  RDP  aligner  based  on  Infernal  (Hidden  Markov  models)  uses  a  reference  for  the  secondary  structure.

File names can be replaced by using <current>

#output files names:

# then can run check again

# use **summary.seqs (fasta=current, name=current)**

SA.good.unique.align

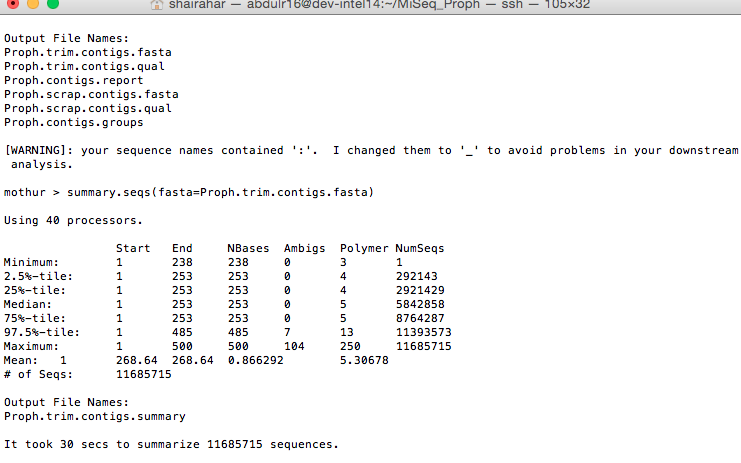
SA.good.unique.align.report

SA.good.unique.flip.accnos

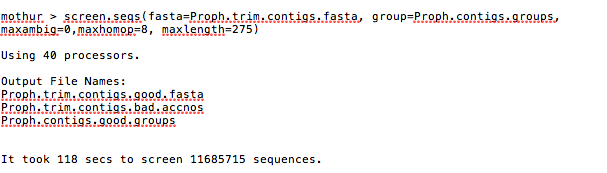
MiSeq Proph

Make contigs

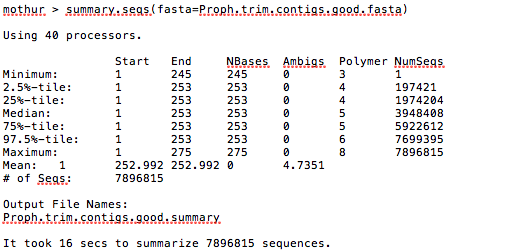
>Summary.seqs



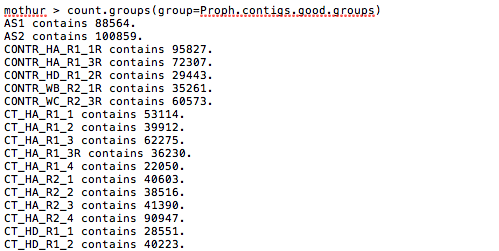
>Screen.seqs

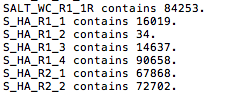


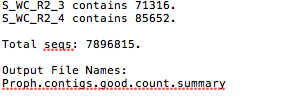
>summary.seqs



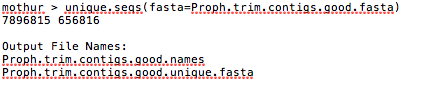
>count.groups

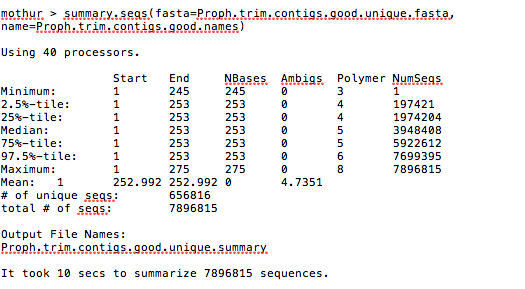






>unique.seqs

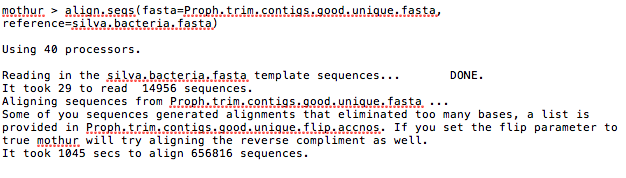




>count.seqs!!

generate a file called stability.trim.contigs.good.count\_table.

>align.seqs



Output File Names:

Proph.trim.contigs.good.unique.align

Proph.trim.contigs.good.unique.align.report

Proph.trim.contigs.good.unique.flip.accnos

\*\*\*\*To align the sequences, have to make sure that Silva database or greengene are there (within the same folder where all your files are). Can always get the latest version of Silva database –the latest as on June 5 is Silva v123

### match, mismatch, gapopen, and gapextend

In the pairwise alignment portion of the aligning procedure, the default reward for a match is +1 and the penalties for a mismatch, opening and extending a gap are -1, -2, and -1. Our experience has shown that these produce the best alignments for 16S rRNA gene sequences. You are encouraged to play around with these to suit your own purposes as shown below:

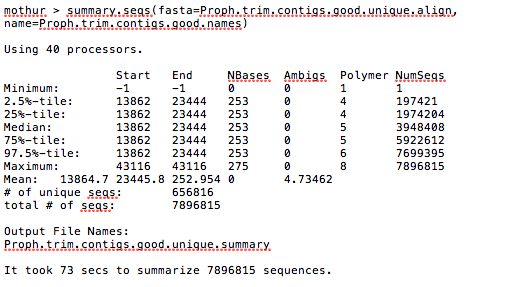
mothur > align.seqs(candidate=abrecovery.fasta, template=core\_set\_aligned.imputed.fasta, align=gotoh, match=1, mismatch=-3)

or

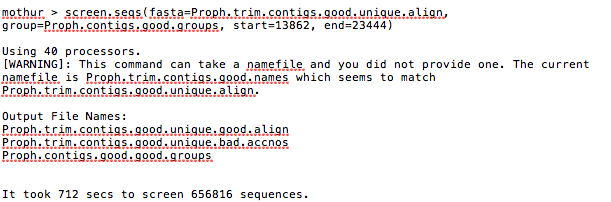
mothur > align.seqs(candidate=abrecovery.fasta, template=core\_set\_aligned.imputed.fasta, align=gotoh, gapopen=-5)

etc.

Keep in mind that if you are using the align=blast option, blast will limit the combinations of match, mismatch, gapopen, and gapextend that you can use. Hopefully, we've scared you off of using blast at all so that this won't be an issue.



##do screen.seqs.. see most reads (more than 90%) start with 13862 and end 23444. We need to take all sequences with those criteria, and remove unaligned sequence



or can try screen.seqs command with maxlength = 253

check this logfile mothur.1445536902.logfile in MiSeq\_Proph

### optimize && criteria

The optimize and criteria parameters allow you set the start, end, maxabig, maxhomop, minlength, maxlength, minoverlap, ostart, oend, mismatches, maxn, minscore, maxinsert and minsim parameters relative to your set of sequences. The following command would remove all sequences that started after the position that 90% of the sequences do.

mothur > screen.seqs(fasta=sogin.unique.align, optimize=start, criteria=90)

Using the optimize parameter trumps the value given for the start parameter if you have provided it. You can optimize to several parameters by separating by dashes. Example: optimize=start-end-minlength, would remove any sequence that starts after the position that 90% of the sequences do, or ends before the position that 90% of the sequences do, or whose length is shorter than 90% of the sequences.

##THINGS TO REMEMBER##

screen take group

summary take names

\*look at the same alignment space 🡪 start 13862, end 23444

screen – for any garbage

unique

align 🡪 use silva

screen 🡪 remove unaligned sequence

filter

download database into the folder

commands

File used fasta/group

contigs

screen

group

File used names

unique

align

better to use names & groups

count.group

CATFISH

Extract all gz file > gunzip \*.gz

remove it, just do a rm -rf \*.gz

1. make.contigs
2. summary to check the contigs
3. screen to remove suspicious sequences (too many homoplymer, too big in bp size)
4. unique.seqs
5. count.seqs (or count.groups?)

-Counts the number of sequences represented by the representative sequence in a name file

-counts sequences from a specific group or set of groups from the following file types: group, count or shared file

1. pcr.seqs to remove pcr junk

to know the start and end region

1. align.seqs –do screen.seqs as required

screen.seqs is actually looking for sequences that start at \*or before\* 6428 and end at \*or after\* 23444. So your results make sense. When you do filter.seqs with trump=T those extra columns will go away

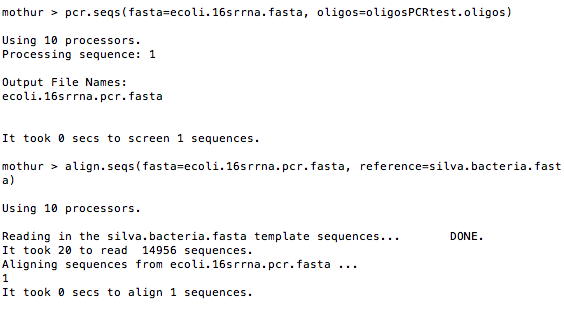
1. filter seqs > then re-run unique.seqs
2. pre.cluster –needed to merge seqs (slow and long process)
3. remove.seqs
4. classify.seqs
5. remove.lineage
6. dist.seqs
7. cluster / cluster.split (PLEASE, be patient. This step is crazy. Have to determine the size of dist.file, in order to allocate enough memory, # of processors, time to complete the analyses)
8. make.shared
9. classify.otu
10. remove.groups

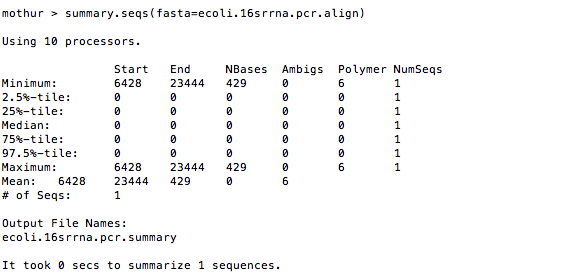
* I am curious if there is code available to eliminate a single sample from my analysis. I reached the part of the SOP where it is recommended to subsample to the lowest number of sequences. However, I found a sample with only 44 sequences. As this number is really low and after going through my lab notes this sample likely did not have enough DNA extracted, I would like to ignore this sample and subsample to the sample with the next lowest number of sequences (31,482)

http://www.mothur.org/forum/viewtopic.php?t=3513

1. <http://www.mothur.org/forum/viewtopic.php?t=1486> (namefile and groupfile mismatch)

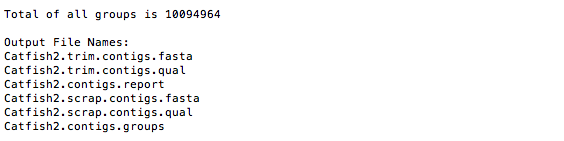
1)  
mothur "#pcr.seqs(fasta=ecoli.16srrna.fasta, oligos=pcrTest.oligos)"  
  
A)  
I obtained my ecoli 16S rRNA sequence from NCBI and saved it as a fasta file.  
<http://www.ncbi.nlm.nih.gov/nuccore/174375?report=fasta>  
  
B)  
pcrTest.oligos is a two line file containing the primers. See <http://www.mothur.org/wiki/Oligos_File> for more details.  
forward ACTCCTACgggAggCAgCAg  
reverse GGACTACHVGGGTWTCTAAT  
  
2)  
mothur "#align.seqs(fasta=ecoli.16srrna.pcr.fasta, reference=silva.bacteria.fasta)"   
  
3)  
mothur "#summary.seqs(fasta=ecoli.16srrna.pcr.align)"  
  
Results  
My primers amplify the V3 and V4 regions and their 5' positions correspond to positions 6428 and 23444 in the 50000 bp SILVA alignment. See below.

Macintosh HD:Users:shairahar:Desktop:Screen Shot 2016-06-05 at 4.46.36 PM.png

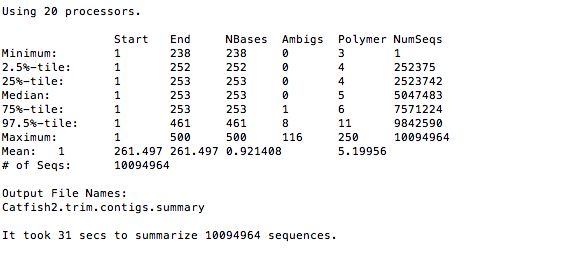


start again

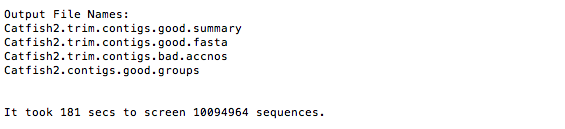
mothur > make.contigs



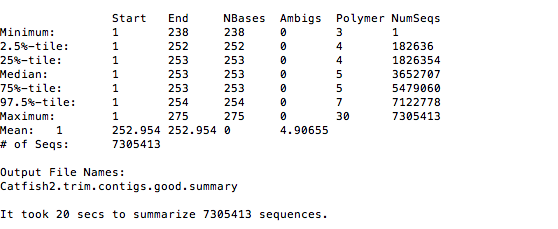
mothur > summary.seqs(fasta=Catfish2.trim.contigs.fasta)



mothur > screen.seqs(fasta=Catfish2.trim.contigs.fasta, group=Catfish2.contigs.groups, summary=Catfish2.trim.contigs.summary, maxambig=0, maxlength=275)



mothur > summary.seqs(fasta=Catfish2.trim.contigs.good.fasta)



# by this time, we already have unique seqs , and name file

**(fasta=\*good.unique.fasta; name=\*good.names)**

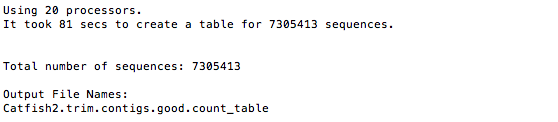
mothur > unique.seqs(fasta=Catfish2.trim.contigs.good.fasta)

Macintosh HD:Users:shairahar:Desktop:Screen Shot 2016-06-07 at 3.11.34 PM.png

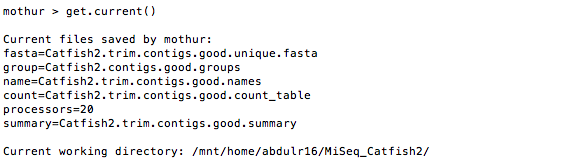
mother > count.seqs(name=Catfish2.trim.contigs.good.names, group=Catfish2.contigs.good.groups)

# by this time, we already have table file

**(count=\*count\_table)**

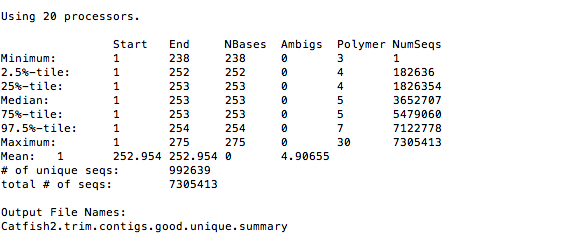


#check for what file we have so far using get.current()



summary.seqs(fasta=Catfish2.trim.contigs.good.unique.fasta, count=Catfish2.trim.contigs.good.count\_table)

mothur > count.groups(group=Catfish.contigs.good.groups)



@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

How to determine start and stop in silva ref alignment? In pcr.seqs

V3-v4

Start 6428 and 23444

V3-v5

the start position for silva reference alignment v119 is 6388 and the stop position is 22096

##Pat says above that 27f (aka 8f) and 1492r primers go from 1044 to 43116. One can infer from the SOP that the V4 region is 11894 to 25319.

The only thing Pat suggest people use at this point is the v2 chemistry with the 500 cycle kit (2x250 nt) to sequence the V4 region. Anything else is just asking for problems.

mothur > pcr.seqs(fasta=silva.nr\_v123.align, start=11894, end=25319, keepdots=F)

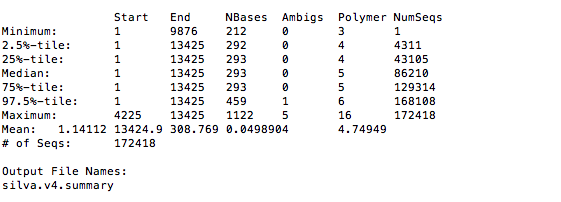
# pcr.seqs trim inputted seq (in this case the silva v123 ref file) based on user-defined option (in this case reset start and stop)

# rename the file \*pcr.align to \*fasta to avoid confusion, then do summary

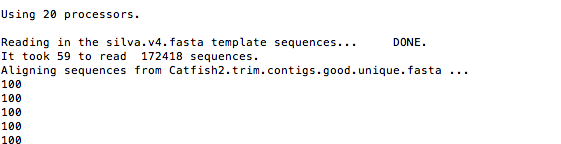
Macintosh HD:Users:shairahar:Desktop:Screen Shot 2016-06-07 at 4.54.50 PM.png

mothur > system(mv silva.nr\_v123.pcr.align silva.v4.fasta)

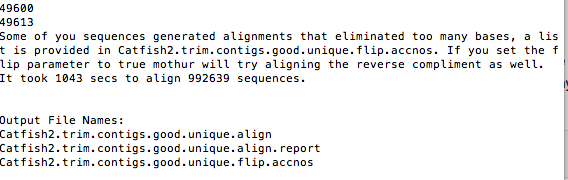
mothur > summary.seqs(fasta=silva.v4.fasta)



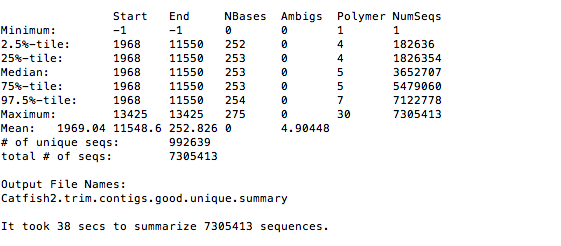
mothur > align.seqs(fasta=Catfish2.trim.contigs.good.unique.fasta, reference=silva.v4.fasta)



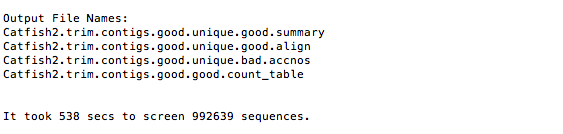
….

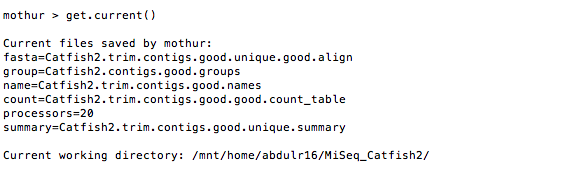


mothur > summary.seqs(fasta=Catfish2.trim.contigs.good.unique.align, count=Catfish2.trim.contigs.good.count\_table)

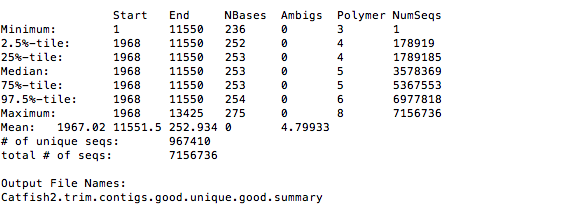


mothur > screen.seqs(fasta=Catfish2.trim.contigs.good.unique.align, count=Catfish2.trim.contigs.good.count\_table, summary=Catfish2.trim.contigs.good.unique.summary, start=1968, end=11550, maxhomop=8)

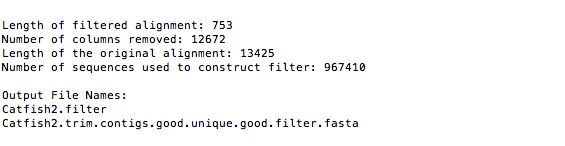




mothur > summary.seqs(fasta=Catfish2.trim.contigs.good.unique.good.align, count=Catfish2.trim.contigs.good.good.count\_table)

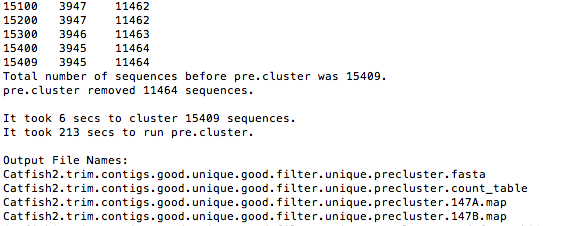


mothur > filter.seqs

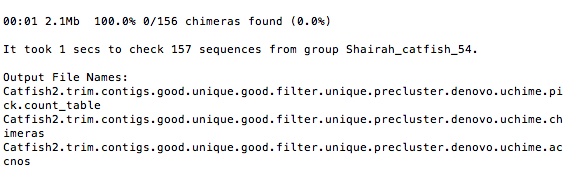


# total number of sequences before precluster was 15409. pre.cluster removed 11464 sequences.  
This output indicates, by column, the number of sequences processed, the number of sequences that will be found in the final dataset, and the number of sequences that have been clustered away.

mothur > pre.cluster

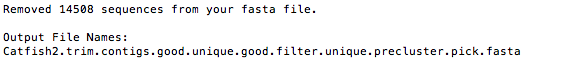


mothur > chimera.uchime(fasta=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.fasta, count=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.count\_table, dereplicate=t)

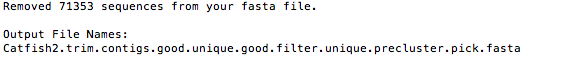


remove.seqs(fasta=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.fasta, accnos=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.accnos)

1st run



2nd run

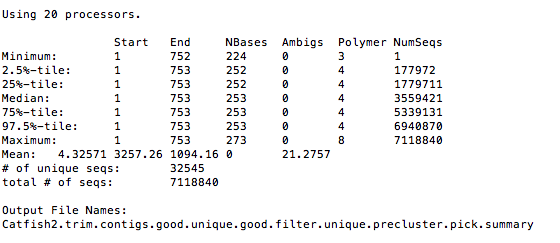


summary.seqs(fasta=current,count=current)

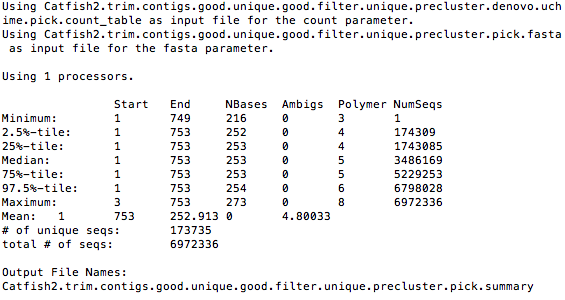
Using Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.count\_table as input file for the count parameter.

Using Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta as input file for the fasta parameter.

1st run



2nd run



mothur > classify.seqs(fasta=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta, count=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.count\_table, reference=trainset9\_032012.pds.fasta, taxonomy=trainset9\_032012.pds.tax, cutoff=80)

1st run

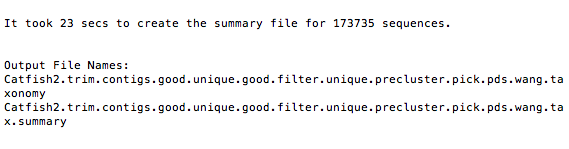
\*\*no results produced using v1.36.1 (ask [mothur\_bugs@gmail.com](mailto:mothur_bugs@gmail.com))

\*\*files provided by westscott:

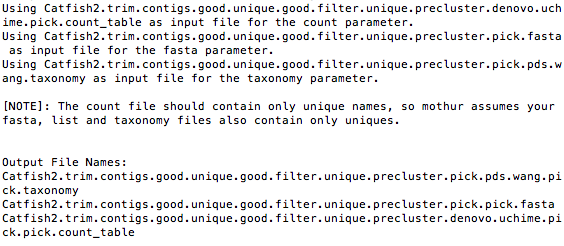
results/Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.wang.tax.summary

inflating: results/Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.wang.taxonomy

2nd run



remove.lineage(fasta=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta, count=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.count\_table, taxonomy=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.wang.taxonomy, taxon=Chloroplast-Mitochondria-unknown-Archaea-Eukaryota)



#get into analyses

following the 2nd run

dist.seqs

cluster

cluster(column=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.dist, count=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.pick.count\_table)

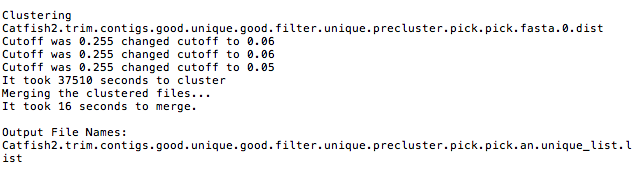
<http://www.mothur.org/forum/viewtopic.php?t=4076> (for cluster split)

@@this is working!@@

>cluster.split(fasta=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.fasta, count=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.pick.count\_table, taxonomy=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.wang.pick.taxonomy, taxlevel=4, cluster=f, processors=8, cutoff=0.15)

>cluster.split(file=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.file, processors=4)

try qsub cluster.sh 33218023.mgr-04.i

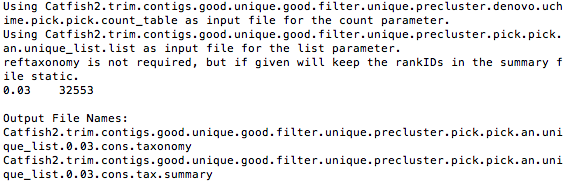


>make.shared(list=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.an.unique\_list.list, count=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.pick.count\_table, label=0.03)

0.03

Macintosh HD:Users:shairahar:Desktop:Screen Shot 2016-06-22 at 11.52.53 AM.png

>classify.otu(list=current, count=current, taxonomy=Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.wang.pick.taxonomy, label=0.03)



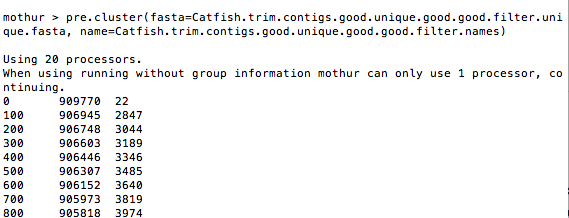
mothur > system(mv Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.an.unique\_list.shared Catfish2.an.shared)

mothur > system(mv Catfish2.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.an.unique\_list.0.03.cons.taxonomy Catfish2.an.cons.taxonomy)

mothur > count.groups(shared=Catfish2.an.shared)



pre.cluster(fasta=Catfish.trim.contigs.good.unique.good.good.filter.unique.fasta, name=Catfish.trim.contigs.good.unique.good.good.filter.names)



Total number of sequences before precluster was 91++++, pre.cluster removed 39741 sequences.

This output indicates, by column, the number of sequences processed, the number of sequences that will be found in the final dataset, and the number of sequences that have been clustered away. This should accelerate as the function runs. In this example, this step merged 3974 sequences with other sequences, leaving you with a set of 90,5818 sequences to work with.

Error on mismatch name file and group file

http://www.mothur.org/forum/viewtopic.php?t=1486