mothur > make.contigs(file=GREATLAKES.files, processors=8)

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

mothur > screen.seqs(fasta=GREATLAKES.trim.contigs.fasta, group=GREATLAKES.contigs.groups, summary=GREATLAKES.trim.contigs.summary, maxambig=0, maxlength=254, minlength=251, maxhomop=8)

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

mothur > summary.seqs(fasta=GREATLAKES.trim.contigs.good.unique.fasta, name=GREATLAKES.trim.contigs.good.names)

mothur > count.seqs(name=GREATLAKES.trim.contigs.good.names)

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

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

mothur > remove.seqs(fasta=GREATLAKES.trim.contigs.good.unique.align, name=GREATLAKES.trim.contigs.good.names, group=GREATLAKES.contigs.good.groups, accnos=GREATLAKES.trim.contigs.good.unique.flip.accnos)

mothur > count.seqs(name=GREATLAKES.trim.contigs.good.pick.names, group=GREATLAKES.contigs.good.pick.groups)

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

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

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

mothur > filter.seqs(fasta=GREATLAKES.trim.contigs.good.unique.pick.good.align, vertical=T)

mothur > unique.seqs(fasta=GREATLAKES.trim.contigs.good.unique.pick.good.filter.fasta, count=GREATLAKES.trim.contigs.good.pick.good.count\_table)

mothur > pre.cluster(fasta=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.fasta, count=GREATLAKES.trim.contigs.good.unique.pick.good.filter.count\_table, diffs=2)

chimera.uchime(fasta=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.fasta, count=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.count\_table, dereplicate=t, processors=8)

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

summary.seqs(fasta=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.fasta, count=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.denovo.uchime.pick.count\_table)

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

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

PHYLOTYPE

# label: 5 = phylum; 4 = class; 3 = order; 2 = family; 1 = genus

mothur>

phylotype(taxonomy=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pds.wang.pick.taxonomy)

#shared file to use for

mothur > make.shared(list=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pds.wang.pick.tx.list, count=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.denovo.uchime.pick.pick.count\_table, label=3)

#Gives a taxonomy reference to the shared file created from phylotyping

mothur > classify.otu(list=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pds.wang.pick.tx.list, count=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.denovo.uchime.pick.pick.count\_table, taxonomy=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pds.wang.pick.taxonomy, label=3)

### remove singletons and doubletons

mothur > split.abund(fasta=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pick.fasta, list=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pick.an.unique\_list.list, count=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.denovo.uchime.pick.pick.count\_table, cutoff=2, label=0.03)

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

0.03

ALPHA DIVERSITY

### alpha diversity metrics

#need to retain singletons for indices

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

mothur > count.groups(shared=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pick.an.unique\_list.shared)

mothur > summary.single(shared=GREATLAKES.trim.contigs.good.unique.pick.good.filter.unique.precluster.pick.pick.an.unique\_list.shared, calc=chao-npshannon, subsample=T)

#Catchall: This has to be run on your own computer. Mothur needs to be installed, and the CatchAll program has to be in the same folder as the mothur program. All you need is a shared file with the OTU data (singletons and doubletons removed, but not normalized or rarefied).