<https://github.com/beiko-lab/mimb_16S/blob/master/qiimeCommands.sh>

#1 Demultiplexing Raw Sequences

1. Usearch – using usearch tool

<https://www.drive5.com/usearch/manual/pipe_demux.html>

Illumina paired with dual index (i1 + i2 + r1 + r2) - Here, you need to make a barcodes FASTA file (bar.fa) which matches the output from fastq\_join, which is I1, NNN spacer, reverse-complemented I2. If you have raw Illumina dual index reads (I1 + I2 + R1 + R2), you can hack a solution by using fastq\_join to concatenate the index reads into a single sequence, then make barcodes FASTA file which matches the format (I1, NN.. spacer, reverse-complemented I2).

usearch -fastq\_join I1.fq -reverse I2.fq -fastqout indexes.fq

usearch -fastx\_demux R1.fq -reverse R2.fq -index indexes.fq \

-fastqout fwd\_demux.fq -output2 rev\_demux.fq

The 64-bit version is not free.

The binary file is in usr/bin but it isn’t called by the terminal

Usearch -> vsearch

<https://github.com/torognes/vsearch>

<https://github.com/torognes/vsearch/wiki/VSEARCH-pipeline>

Installed in conda:

conda activate bioinfo

First join indexes to form barcodes:

vsearch --fastq\_join B2\_S150\_L001\_I1\_001.fastq.gz --reverse B2\_S150\_L001\_I2\_001.fastq.gz --fastqout ../barcodesmerge/B2\_S150\_L001\_I1I2\_001.fastq

vsearch --fastq\_join B2\_S150\_L001\_I1\_001.fastq.gz --reverse B2\_S150\_L001\_I2\_001.fastq.gz --fastqout ../barcodesmerge/B2\_S150\_L001\_I1I2\_001.fastq

1. Bayexer – in Perl

<https://github.com/HaisiYi/Bayexer/tree/7e49bd58fd1c38b1228bf3fa4d8b107e67231529>

perl Bayexer -i 1C\_S149\_L001\_R1\_001.fastq.gz 1C\_S149\_L001\_R1\_001.fastq.gz -j 1C\_S149\_L001\_I1\_001.fastq.gz 1C\_S149\_L001\_I2\_001.fastq.gz -o ./demux/

#### asking for a sample file????

1. qiime 1

<http://qiime.org/tutorials/processing_illumina_data.html>

<https://nbviewer.jupyter.org/github/biocore/qiime/blob/1.9.1/examples/ipynb/illumina_overview_tutorial.ipynb>

<https://forum.qiime2.org/t/importing-and-demultiplex-process-for-4-fastq-files-r1-r2-index1-and-index2/2586>

$ python merge\_bcs\_reads.py Undetermined\_S0\_L001\_I2\_001.fastq Undetermined\_S0\_L001\_R1\_001.fastq Full\_Read1\_wBarcodes.fastq

$ python merge\_bcs\_reads.py Undetermined\_S0\_L001\_I1\_001.fastq Undetermined\_S0\_L001\_R2\_001.fastq Full\_Read2\_wBarcodes.fastq

$ extract\_barcodes.py --input\_type barcode\_paired\_end -f Full\_Read1\_wBarcodes.fastq -r Full\_Read2\_wBarcodes.fastq -m Pilot\_QIIME\_MapFile\_2.txt --rev\_comp\_bc2 --switch\_bc\_order --bc1\_len 8 --bc2\_len 8 -o parsed\_barcodes\_fullreads/

$ join\_paired\_ends.py -f reads1.fastq -r reads2.fastq -b barcodes.fastq -o /Users/Sara\_Jeanne/Desktop/QIIME/fastq-join\_joined\_FullReads

$ split\_libraries\_fastq.py -i fastqjoin.join.fastq -b fastqjoin.join\_barcodes.fastq -m Pilot\_QIIME\_MapFile\_2.txt -o Split\_Lib\_FullReads\_attachedBC\_q20/ --store\_qual\_scores --barcode\_type 16 -q 19

1. Mothur –

<https://mothur.org/wiki/sffinfo/>

<https://mothur.org/wiki/miseq_sop/>

<https://forum.mothur.org/t/demultiplex-dual-index-fastq-files/2534/5>

mothur > make.contigs(ffastq=yourForwardFastqFile, rfastq=yourReverseFastqFile, findex=yourForwardIndexFile, rindex=yourReverseIndexFile, oligos=yourOligosFile, pdiffs=2, bdiffs=1)

Here’s a link to **the oligos file** wiki page, [http://www.mothur.org/wiki/Oligos\_File 6](http://www.mothur.org/wiki/Oligos_File).

mothur > make.contigs(ffastq= 1C\_S149\_L001\_R1\_001.fastq.gz, rfastq= 1C\_S149\_L001\_R2\_001.fastq.gz, findex= 1C\_S149\_L001\_I1\_001.fastq.gz, rindex= 1C\_S149\_L001\_I2\_001.fastq.gz)

qiime tools import \

--type 'SampleData[PairedEndSequencesWithQuality]' \

--input-path rawsequences \

--output-path paired-end-demux.qza

\

qiime tools import --type 'SampleData[PairedEndSequencesWithQuality]' --input-path rawsequences --output-path paired-end-demux.qza --input-format FastqGzFormat

qiime tools import --type 'SampleData[PairedEndSequencesWithQuality]' --input-path rawsequences --output-path paired-end-demux.qza --input-format PairedEndFastqManifestPhred33V2

# 2 Metadata

biom add-metadata -i table\_even2762.biom -o seaurchintable.biom --observation-metadata-fp SeaUrchinMetadataQiime2.tsv

# 3 Import Biom table

qiime tools import \

--input-path seaurchintable.biom \

--type 'FeatureTable[Frequency]' \

--input-format BIOMV210Format \

--output-path seaurchinfeaturetable.qza

# 4 Import Phylogenetic Trees

qiime tools import \

--input-path gg\_13\_8\_otus/trees/97\_otus.tree \

--output-path unrooted-tree.qza \

--type 'Phylogeny[Unrooted]'

# 5 Import Per-feature unaligned Sequence Data

qiime tools import \

--input-path gg\_13\_8\_otus/rep\_set/97\_otus.fasta \

--output-path sequences.qza \

--type 'FeatureData[Sequence]'

# 6 Feature table & feature data

qiime feature-table summarize \

--i-table seaurchinfeaturetable.qza \

--o-visualization seaurchinfeaturetable.qzv \

--m-sample-metadata-file SeaUrchinMetadataQiime2.tsv

qiime feature-table tabulate-seqs \

--i-data sequences.qza \

--o-visualization sequences.qzv

# 7 Root Unrooted Tree

qiime phylogeny midpoint-root \

--i-tree unrooted-tree.qza \

--o-rooted-tree rooted-tree.qza

--type 'Phylogeny[Rooted]'

# in the step 7 fixing error

"All non-root nodes in ``tree`` must have a branch length.

import skbio"

in python:

1.

t = skbio.TreeNode.read('/home/qiime2/Desktop/QIIME\_DATA/core-2762/gg\_13\_8\_otus/trees/97\_otus.tree')

2.

for n in t.traverse():

if n.length is None:

n.length = 0.0

3.

import qiime2

4.

ar = qiime2.Artifact.import\_data('Phylogeny[Rooted]', t)

5.

ar.save('97\_otus.qza')

# 8 Diversity (alpha & beta)

qiime diversity core-metrics-phylogenetic \

--i-phylogeny 97\_otus.qza \

--i-table seaurchinfeaturetable.qza \

--p-sampling-depth 2762 \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--output-dir core-metrics-results

# 9 Exploring Microbial Composition with metadata

qiime diversity alpha-group-significance \

--i-alpha-diversity core-metrics-results/faith\_pd\_vector.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--o-visualization core-metrics-results/faith-pd-group-significance.qzv

qiime diversity alpha-group-significance \

--i-alpha-diversity core-metrics-results/evenness\_vector.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--o-visualization core-metrics-results/evenness-group-significance.qzv

# 10 PERMANOVA

by Reef Habitat

qiime diversity beta-group-significance \

--i-distance-matrix core-metrics-results/unweighted\_unifrac\_distance\_matrix.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--m-metadata-column Reef\_Habitat \

--o-visualization core-metrics-results/unweighted-unifrac-reefhabitat-significance.qzv \

--p-pairwise

by Current:

qiime diversity beta-group-significance \

--i-distance-matrix core-metrics-results/unweighted\_unifrac\_distance\_matrix.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--m-metadata-column Current \

--o-visualization core-metrics-results/unweighted-unifrac-current-significance.qzv \

--p-pairwise

by Sargassum

qiime diversity beta-group-significance \

--i-distance-matrix core-metrics-results/unweighted\_unifrac\_distance\_matrix.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--m-metadata-column Sargassum \

--o-visualization core-metrics-results/unweighted-unifrac-sargassum-significance.qzv \

--p-pairwise

by Relative\_Gut\_Content

qiime diversity beta-group-significance \

--i-distance-matrix core-metrics-results/unweighted\_unifrac\_distance\_matrix.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--m-metadata-column Relative\_Gut\_Content \

--o-visualization core-metrics-results/unweighted-unifrac-Relative\_Gut\_Content-significance.qzv \

--p-pairwise

# 11 Extension of step 8

numerical

qiime emperor plot \

--i-pcoa core-metrics-results/unweighted\_unifrac\_pcoa\_results.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--p-custom-axes Size \

--o-visualization core-metrics-results/unweighted-unifrac-emperor-size.qzv

qiime emperor plot \

--i-pcoa core-metrics-results/bray\_curtis\_pcoa\_results.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--p-custom-axes Size \

--o-visualization core-metrics-results/bray-curtis-emperor-size.qzv

# 12 Alpha rarefaction plotting

qiime diversity alpha-rarefaction \

--i-table seaurchinfeaturetable.qza \

--i-phylogeny 97\_otus.qza \

--p-max-depth 2762 \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--o-visualization alpha-rarefaction.qzv

# 13 Taxonomic Analysis

Training Feature Classifiers: recomended to train classifiers with own sequences.

https://docs.qiime2.org/2020.2/data-resources/

Used: Greengenes 13\_8 99% OTUs from 515F/806R region of sequences (MD5: 6df67fb01e2f3305e76c61a1c16136b4)

qiime feature-classifier classify-sklearn \

--i-classifier gg-13-8-99-515-806-nb-classifier.qza \

--i-reads sequences.qza \

--o-classification taxonomy.qza

qiime metadata tabulate \

--m-input-file taxonomy.qza \

--o-visualization taxonomy.qzv

qiime taxa barplot \

--i-table seaurchinfeaturetable.qza \

--i-taxonomy taxonomy.qza \

--m-metadata-file SeaUrchinMetadataQiime2.tsv \

--o-visualization taxa-bar-plots.qzv

Merging & grouping

<https://forum.qiime2.org/t/best-way-to-merge-or-group-runs-samples/2855/2>

<https://docs.qiime2.org/2017.12/plugins/available/feature-table/group/>

<https://docs.qiime2.org/2017.12/plugins/available/feature-table/merge/>

group by LOCATION

qiime feature-table group --i-table seaurchinfeaturetable.qza --p-axis sample --m-metadata-file SeaUrchinMetadataQiime2.tsv --m-metadata-column Location --p-mode sum

OUTPUT: grouped\_table.qza

This is the same as seaurchinfeaturetable.qza

Summarize:

qiime feature-table summarize \

--i-table grouped\_table.qza \

--o-visualization grouped\_table.qzv \

--m-sample-metadata-file SeaUrchinMetadataQiime2.tsv #####the location ids are not present in the metadata

2. Taxonomy

qiime taxa barplot \

--i-table grouped\_table.qza \

--i-taxonomy taxonomy.qza \

--m-metadata-file grouped\_meta.tsv \

--o-visualization grouped-taxa-bar-plots.qzv

group BY Reef\_Habitat

qiime feature-table group --i-table seaurchinfeaturetable.qza --p-axis sample --m-metadata-file SeaUrchinMetadataQiime2.tsv --m-metadata-column Reef\_Habitat --p-mode sum --o-grouped-table reef\_habitat\_grouped.qza

Summarize:

qiime feature-table summarize \

--i-table reef\_habitat\_grouped.qza \

--o-visualization grouped\_table.qzv \

--m-sample-metadata-file grouped\_meta\_reef.tsv

2. Taxonomy

qiime taxa barplot \

--i-table reef\_habitat\_grouped.qza \

--i-taxonomy taxonomy.qza \

--m-metadata-file grouped\_meta\_reef.tsv \

--o-visualization grouped-taxa-bar-plots.qzv

group BY Size – Didn’t work

qiime feature-table group --i-table seaurchinfeaturetable.qza --p-axis sample --m-metadata-file SeaUrchinMetadataQiime2.tsv tsv --m-metadata-column Size --p-mode sum --o-grouped-table size\_grouped.qza

merge by size --i-table seaurchinfeaturetable.qza --p-overlap-method --m-metadata-file SeaUrchinMetadataQiime2.tsv tsv --m-metadata-column Size --o-grouped-table size\_grouped.qza

qiime feature-table merge

Summarize:

qiime feature-table summarize \

--i-table size\_grouped.qza \

--o-visualization size\_grouped\_table.qzv \

--m-sample-metadata-file grouped\_meta\_size.tsv

2. Taxonomy

qiime taxa barplot \

--i-table size\_grouped.qza \

--i-taxonomy taxonomy.qza \

--m-metadata-file grouped\_meta\_size.tsv \

--o-visualization size-grouped-taxa-bar-plots.qzv