The following describe the scripts, along with their inputs and outputs, that are involved in converting raw short reads into a taxonomy abundance table using qiime2.

All of the slurm scripts contain my email in the header and are set to run on Heather’s project bank, p32400. The bank may need to be changed for future use, and my email should be replaced with the users so you can completion or error notifications.

The file paths in all of the scripts are also set to my personal quest directory. This will need to be changed for future use.

1. **manifest.txt**

The first step in the pipeline is to import the raw data into qiime and convert the reads into a qiime2 object. Qiime requires a roadmap file that tells it which sample correspond to which pair of forward and reverse reads. The first column is named sample-ID and contains the list of user-defined sample names. The next two columns are named forward-read-absolute and reverse-read-absolute, and respectively contain the complete file paths to the forward and reverse paired reads that correspond to the sample name in the first column. This file must be a text file, but the columns are separated by a tab key. The best way to make it is to enter all of the text in excel and save it as a .txt. This must be in the saved in the directory where the following scripts are stored and run.

1. **import\_sep.sh**

SLURM script containing commands to read manifest.txt, call qiime2, and convert the reads to qiiime2 objects. To use, change the slurm header to fit your email and your quest bank, and change the file paths in the commands to the directory you’re working in. This script outputs files reads.qza and reads.qzv. These files contain the same information but have different uses. .qza files store the read information for further processing by qiime2. qzv files contain vizualization information. If you go to the qiime2 website and click view in the top right corner, you can input the .qzv file and it will display information like the number of reads and average length.

1. [**trim.sh**](http://trim.sh)

Slurm script that trims pcr primers from the reads. It calls the qiime function cutadapt and takes reads.qza. The primer sequences for forward and reverse reads must be added to the script in the –p-front-f and –p-front-r arguments.

This script produces two outputs, reads\_trimmed.qza and readquality-trimmed.qzv. reads\_trimmed.qza contains the trimmed reads and will be used for the next analysis step. readquality\_trimmed.qzv contains information for qiime to produce a graph that shows average read quality per base pair. Illumina sequencing gives a quality score for each base pair for each read that corresponds to accuracy confidence. This score generally declines as reads get longer, meaning base pairs and the ends of reads are less reliable than those towards the beginning. The graph shows, the average quality score for the ith base pair for all the reads.

This graph is essential for the next step. Upload it to the qiime2 website view page and navigate to the quality plot.

1. [**asv.sh**](http://asv.sh)

Slurm script that identifies amplicon sequence variants (ASV)s from the reads and calculates the abundance of each ASV for each sample. As inputs it takes reads\_trimmed.qza.

The script also needs to be edited to trim the reads according to the quality plot from the previous step. At a certain length threshold, the read quality dips to a point after which the low quality prevent qiime from being able to complete the ASV analysis. The reads need to be trimmed at a certain length to cut off these lower quality base pairs and produce useable data. The exact length threshold will vary based on your data and must be determined based on the quality plot. This process is relatively subjective. A general rule of thumb is to cut the reads at the length where the average quality score dips below 25 or 30. This needs to be done for both the forward and reverse reads, and qiime2 generates a plot for both. For this analysis, I chose to cut the reads at 235 base pairs for the forward and reverse reads.

There are four outputs to this script.

1. rep\_seqs\_dada2.qza
   1. This contains a list of every ASV that qiime2 identified and the corresponding ID code.
2. table\_dada2.qza
   1. Table containing abundance information for each ASV code in each sample.
3. Stats\_dada2.qza
   1. contains information about the asvs, including average length, quality, number, etc.
4. stats\_dada2.qzv
   1. vizualization file for ASV data. I did not use this fie.
5. rep\_seqs\_dada2.qz
   1. vizualitaion file for the representative sequences. This needs to be checked to insure a reasonable number of ASVs are identified. If the quality trimming is done wrong, qiime2 will find far fewer ASVs than what is expected. This file shows how many ASVs are outputted.

**5.** [**Taxonomy.sh**](http://taxonomy.sh)

Slrum script that calls qiime2 and uses a bayes classfier trained on the Midas database to classify all of the ASVs. It takes as inputs the trained classifier and rep\_seqs\_dada2.qza.

The classifier must be trained prior to use. Thankfully, McKenna has done this for us and it can be accessed with the filepath in the script. I do not know how training the classifier works and I never had to do it. McKenna goes into detail on her github.

Outputs:

taxonomy.qza / .qzv. This contains the list of ASV id codes and their taxonomic classification.

1. [**tree.sh**](http://tree.sh)

Slurm script that performs alignment to generate a phylogenetic tree from our data. I dont know much about how this works or why it is important, so again I would refer to the github. However, the tree file is needed for the R analysis that creates the biom file.

This takes as an imput the rep\_seqs\_dada2.qza.

Outputs include an alignment file, an unrooted tree, and a rooted tree. Only the rooted tree is needed for the next step.

1. **Combining the in R to produce the biom file.**

The tree, table, and taxonomy files are combined into a single data frame that shows the distribution ASV abundances along with taxonomic classifications for each sample. This is done by calling an R package called phyloseq, which can process the qiime2 objects. Mckenna’s code to perform these tasks is here: <https://github.com/mckfarm/16S_demo_2023/blob/main/scripts/analysis_example.Rmd>

1. **Exporting the to fasta**

To create a fasta file containing the ASV id codes and their corresponding sequences, use the following commands:

**module activate qiime2**

**qiime tools export \**

**--input-path rep\_seqs\_dada2.qza \**

**--output-path rep\_seqs\_dada2\_export**