**Methods Paper Workflow**

By Sarah Daly

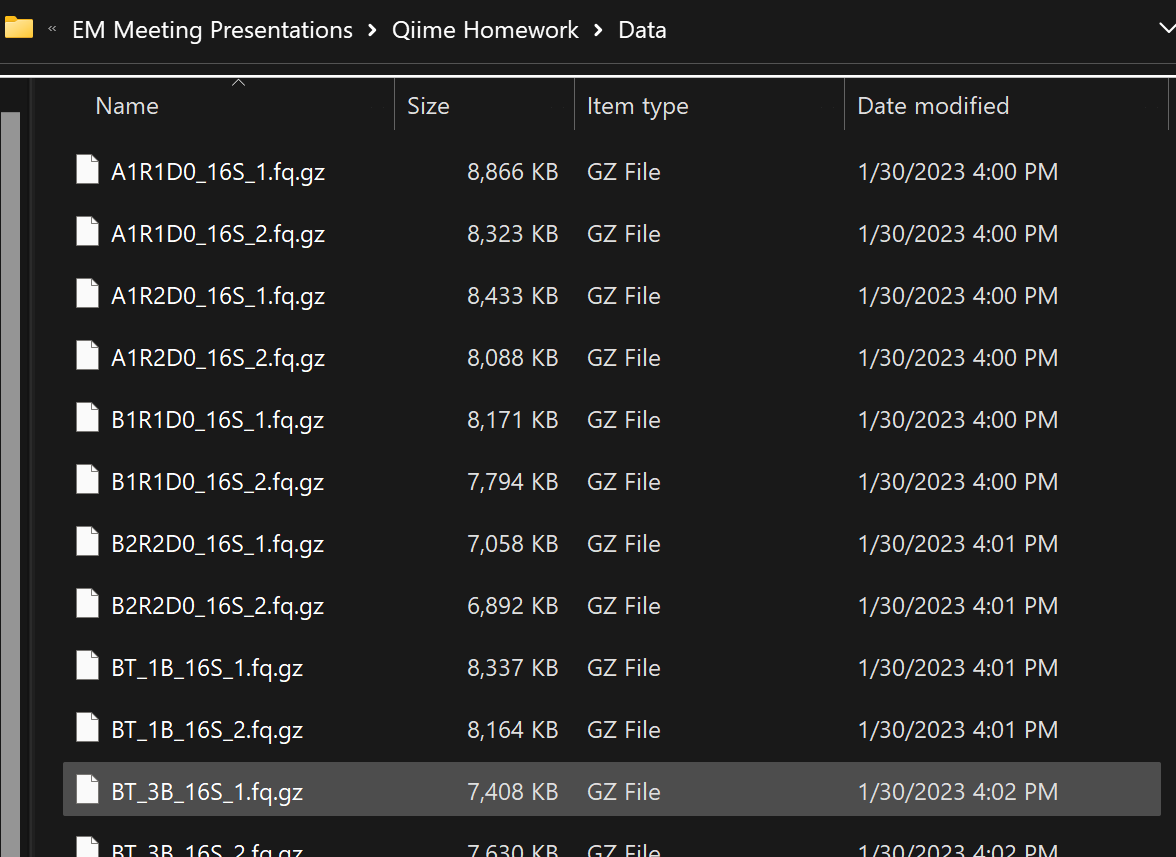
1. Processing ITS Sequences in QIIME
2. Export QIIME files to Rstudio for metagenomic analysis and visualization
3. Perform multivariate analysis in Rstudio

# QIIME Workflow

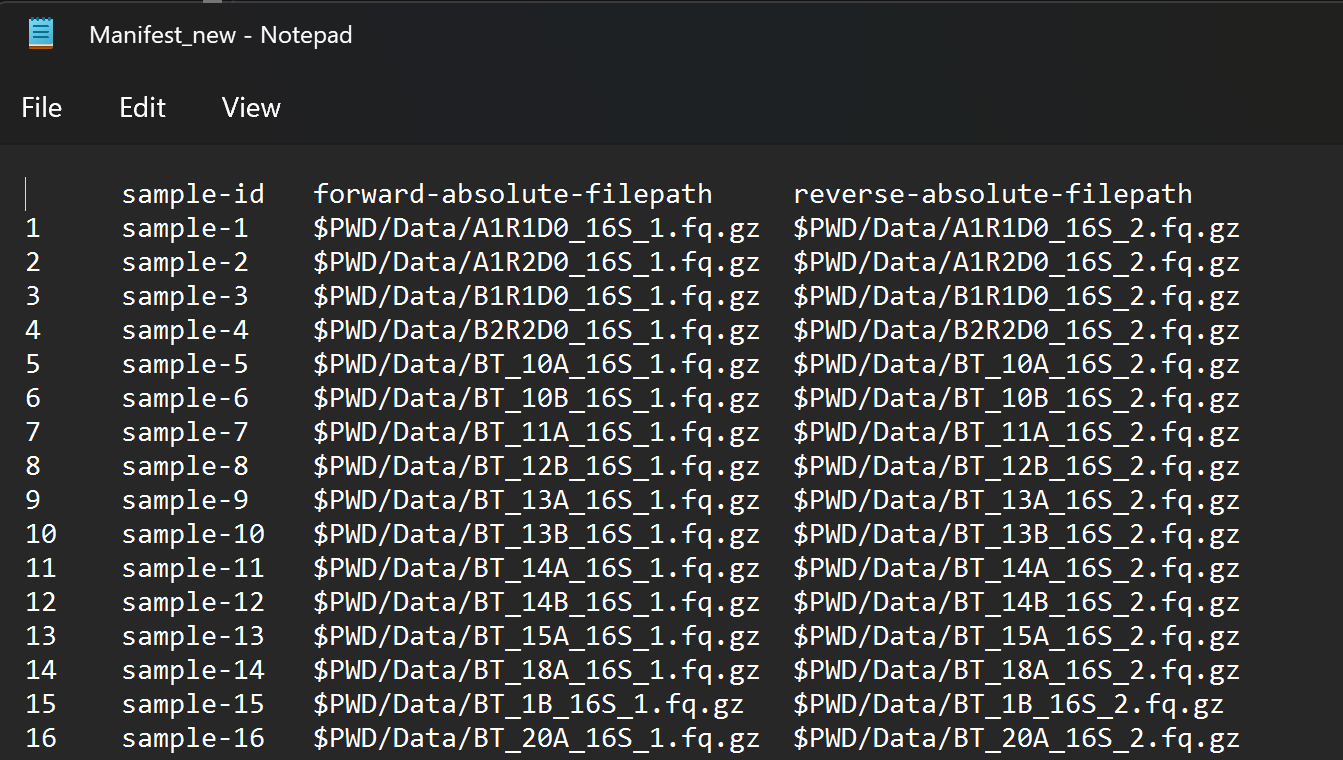
# Starting QIIME

1. type “qiime” into command line to open QIIME 2. (I did this tutorial in qiime2022.2; machine was gen1 64 cores, 512 GB RAM)
2. Type “source tab-qiime” so you can tab complete file names
3. Type in “qiime --help" to see the list of plug-ins (similar to “packages” in R) installed (dada2 is the most important we are using)
4. Type “qiime dada2 denoise-paired--help" for whatever command to get a list of parameters for a command

# Importing Sequence Files to QIIME

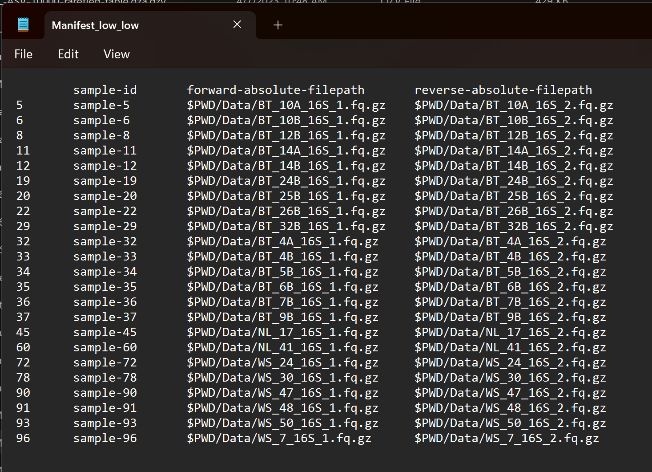
1. Download or copy the 16S sequence folder from Box into your personal directory.
2. Filter for the demultiplexed gz files (Search in windows for files that end in : \*\_1.fq.gz & \*\_2.fq.gz) and then drag them in your folder. Transfer these files into a new folder marked “Data” (there should be no subfolders)
3. 
4. Transfer “Data” to your home directory in QIIME. Check that all files have the same extension (.fq.gz) or QIIME will not import it.
5. In Qiime, change your working directory (cd ‘directory’) to “Data” and check the compression of each file (should be all Oks). 
   1. **for** f **in** \*.fq.gz; **do** gunzip -t -v $f; **done**
6. Go back to your working directory (cd ..)
7. The file names do not match the basic QIIME defaults. Therefore, we need to create a Manifest File with the name and path of all sequence files. Go to R Studio on your home computer and open Manifest File Generator.Rmd.
8. Read the directions and create a Manifest file. Export it as a .csv and open. Save it as a .txt. Note the file must be in this format.

***Troubleshooting:*** Find and replace $PWD with the path to the file if you have issues with QIIME opening it.

****

* 1. **More information on Manifest File Generation:**
  2. [**https://docs.qiime2.org/2022.2/tutorials/importing/**](https://docs.qiime2.org/2022.2/tutorials/importing/)
  3. [**https://bioinfo.ird.fr/index.php/tutorials-fr/qiime2-in-command-line/**](https://bioinfo.ird.fr/index.php/tutorials-fr/qiime2-in-command-line/)
  4. [**https://forum.qiime2.org/t/automatic-manifest-maker-in-python3/8595**](https://forum.qiime2.org/t/automatic-manifest-maker-in-python3/8595)
  5. [**https://forum.qiime2.org/t/automatic-manifest-maker-in-r/2921**](https://forum.qiime2.org/t/automatic-manifest-maker-in-r/2921)

1. In QIIME, convert the sequence files to a .qza artifiact & view basic sequence and sample info. It will generate an .html that you can view in a web browser. You should perform this command in your working directory.

**

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

qiime demux summarize --i-data paired-end-demux-ITS.qza --o-visualization paired-end-demux-ITS.qza

qiime tools view paired-end-demux-ITS.qza.qzv

qiime tools validate paired-end-demux-ITS.qza

*Result paired-end-demux.qza appears to be valid at level=max.*

\*Note in BioHPC (in PuTTy) you may not be able to “view” .qza and .qzv files ; you can view them online at this website: <https://view.qiime2.org/> (In VNC it seems like you can view these files)

\*\*Forward and Reverse read “Total” must match. If not, check each sample and remove samples that don’t match\*\*

# Importing Metadata in QIIME

1. Got to the .csv version of your manifest file (Manifest\_new.csv
2. Keep sample-id column (first column must be this); you can delete other columns as you add more
3. If you need barcodes, they can be found in SampleSeq\_info in the original sequence files
4. Keep cells blank for missing data
5. Data must be categorical or numeric
6. Use simple headings; use (-) between words if you need a heading with more than 2 words
7. Remember that the original sampling metadata files are in Box (20221115\_july 2022 sampling metasheet\_v03\_sd.csv) ; copy and paste in categories you want.
8. Extract Facility, Sampling Season, and original sample ID from this file info in the metadata file and input into basic metadata sheet.
9. Save as a .txt file (Metadata16S\_corrected.txt) and transfer to your QIIME working directory
10. Visualize the metadata in your working directory.

qiime metadata tabulate --m-input-file Metadata.txt --o-visualization tabulated-sample-metadata.qzv

qiime tools view tabulated-sample-metadata.qzv

**More information on creating metadata files**

* 1. <https://docs.qiime2.org/2022.8/tutorials/metadata/>
  2. <https://docs.google.com/spreadsheets/d/1lVOXPvK51wvV5lm40AQSAW9XdQgMxd3QsjN-wY5vNho/edit#gid=0>

# Truncating and Denoising Reads

1. We need to determine at what quality ends can be trimmed and reads can still be merged without losing info. This can be somewhat subjective.
2. *Denoise* to remove and/or correct noisy reads, correct errors in marginal sequences (DADA2), remove chimeric sequences, remove singletons, join denoised paired-end
3. According to the report we have, the minimum base length of forward reads is 226 and reverse reads is 223. Generally, the sequence quality looks good.
4. There are multiple criteria for trimming
   1. **Manual trimming**: Trim if median quality score < 20 [this may be better for ITS and paired end data] (will be uneven lengths)
5. **ITS:** The IT2S and ITS1 region is highly variable (200-600p) in length. Unlike in 16S sequence processing, fixed truncation length is no longer appropriate, since it will remove real ITS variants with lengths shorter than the truncation length. Primer removal is more complicated since some, but not all reads, may extend into the opposite primer when the amplified ITS region is shorter than the read length. [*https://benjjneb.github.io/dada2/ITS\_workflow.html*](https://benjjneb.github.io/dada2/ITS_workflow.html)

\*\*I suggest using a machine with 96 cores and letting the process run with all the cores (add the --p-n-threads 0 flag to the command). This seems to save considerable time—see below.

**Manual Trimming by Quality Score (took about half an hour to run)**

qiime dada2 denoise-paired --i-demultiplexed-seqs paired-end-demux-ITS.qza --p-trunc-len-f 0 --p-trunc-len-r 0 --p-trunc-q 20 --p-n-threads 8 --o-table table-ITS-q20.qza --o-representative-sequences ASV-rep-seqs-ITS-q20.qza --o-denoising-stats denoising-stats-ITS-q20.qza --verbose

<https://docs.qiime2.org/2018.8/tutorials/otu-clustering/>

**OTU Method**

***De novo* clustering of a feature table can be performed as follows. In this example, clustering is performed at 99% identity to create 99% OTUs.**

qiime vsearch cluster-features-de-novo **\**

--i-table table-ITS-q20.qza **\**

--i-sequences ASV-rep-seqs-ITS-q20.qza **\**

--p-perc-identity 0.99 **\**

--o-clustered-table otu-table-dn-99-q20.qza **\**

--o-clustered-sequences otu-rep-seqs-dn-99-ITS-q20.qza

**View output from quality score trimming:**

**qiime feature-table tabulate-seqs --i-data otu-rep-seqs-dn-99-ITS-q20.qza --o-visualization ITS-rep-seqs-otu.qzv**

**Summarize feature count**

qiime feature-table summarize --i-table table-ITS-q20.qza --o-visualization table\_summary.qzv

qiime tools view table\_summary.qzv

# Classifying Sequence Taxonomy

Machine-learning-based classification methods are available through classify-sklearn. These classifiers must be *trained*, e.g., to learn which features best distinguish each taxonomic group, adding an additional step to the classification process. [Classifier training](https://docs.qiime2.org/2022.8/tutorials/feature-classifier/) is **reference database- and marker-gene-specific** and only needs to happen once per marker-gene/reference database combination; that classifier may then be re-used as many times as you like without needing to re-train!

We will train the [Naive Bayes](http://scikit-learn.org/stable/modules/naive_bayes.html#multinomial-naive-bayes) classifier using [Greengenes](http://qiime.org/home_static/dataFiles.html) reference sequences and classify the representative sequences. We select classify-sklearn with a Naive Bayes classifier because it can slightly outperform other methods of 16S rRNA gene and fungal ITS sequence (<https://microbiomejournal.biomedcentral.com/articles/10.1186/s40168-018-0470-z>)

List of taxonomy databases:

<http://qiime.org/home_static/dataFiles.html>

**#Classifying sequences from the ASV table**

**#With 99% identity between taxa**

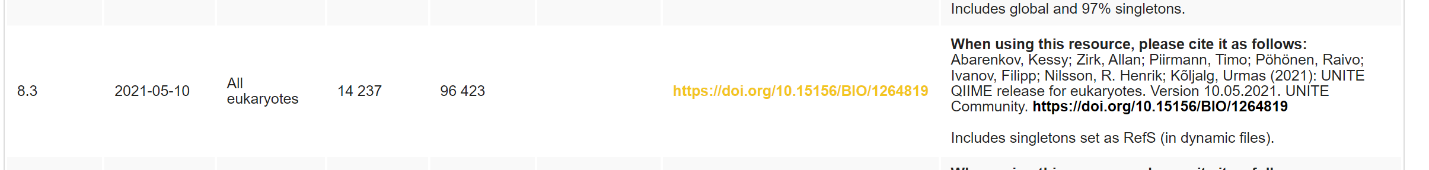
[**https://github.com/colinbrislawn/unite-train/releases/tag/9.0-qiime2-2023.2-demo**](https://github.com/colinbrislawn/unite-train/releases/tag/9.0-qiime2-2023.2-demo)

#**Download unite file version you want and transfer to QIIME working directory [I did this step already and you can upload it from Box**)

# <https://doi.plutof.ut.ee/doi/10.15156/BIO/2483915>

<https://unite.ut.ee/repository.php>

Select “dynamic” and “all” in order to capture everything

**

File will need to be unzipped

**#Unzip**

tar -xzf sh\_qiime\_release\_all\_29.11.2022.tgz

**#Change directory to that containing unite database files**

cd 29.11.22developer

**# Import the UNITE reference sequences into QIIME2.**

qiime tools import \

--type FeatureData[Sequence] \

--input-path sh\_refs\_qiime\_ver9\_dynamic\_29.11.2022\_dev\_uppercase.fasta \

--output-path unite-ver9-seqs\_dynamic\_29.11.2022.qza

**#Import Taxonomy File**

qiime tools import \

--type FeatureData[Taxonomy] \

--input-path sh\_taxonomy\_qiime\_ver9\_dynamic\_29.11.2022\_dev.txt\

--output-path unite-ver9-taxonomy\_dynamic\_29.11.2022.qza \

--input-format HeaderlessTSVTaxonomyFormat

**#Trainng the classifier-may take up to 30 minutes**

qiime feature-classifier fit-classifier-naive-bayes \

--i-reference-reads unite-ver9-seqs\_dynamic\_29.11.2022.qza \

--i-reference-taxonomy unite-ver9-taxonomy\_dynamic\_29.11.2022.qza \

--o-classifier unite-ver9-dynamic-fungi-classifier-29.11.2022.qza \

--verbose

cd..

**#Classifying sequences**

qiime feature-classifier classify-sklearn \

--i-classifier 29.11.22developer/unite-ver9-dynamic-fungi-classifier-29.11.2022.qza \

--i-reads otu-rep-seqs-dn-99-ITS-q20.qza \

--o-classification unite-dyn-11-22-ITS-taxonomy-otu-q20.qza \

--p-n-jobs -1 \

--verbose

*\*may take at least 30 minutes to run*

**#Make barplot**

qiime taxa barplot --i-table otu-table-dn-99-q20.qza --i-taxonomy unite-dyn-11-22-ITS-taxonomy-otu-q20.qza --m-metadata-file Metadata.txt --o-visualization ITS-otu-barplotbbb-unite-99-ver9-q20.qzv

qiime tools view ITS-otu-barplotbbb-unite-99-ver9-SFWS-q20.qzv

# #create a phylogeny tree

**OTU method**

qiime phylogeny align-to-tree-mafft-fasttree \

--p-n-threads 12 \

--i-sequences otu-rep-seqs-dn-99-ITS-q20.qza \

--output-dir phylogeny-align-to-tree-mafft-fasttree-ITS-otu --verbose

**Download & Open this folder on your computer.**

“rooted\_tree.qza” is the important file to important in R.

Cornell BioHPC: Deactivate the qiime environment after you are done:

conda deactivate

Make sure to copy files back to your home directory(/home/netid)!

Cancel your reservation in BioHPC if you are done!

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# R PHYLOSEQ ANALYSIS

Import pink files into Rstudio for each combination (Medium ASV, Medium OTU, ect)

-otu or asv table

-phylogenic tree

-taxonomic table

-metadata file

Process in “phyloseq package”.