**QIIME2: Pinnell Pipeline TAMU HPRC Edition**

*June 2023 (updates by EDoster for qiime2-2023.2)*

*General notes: This will process demultiplexed 16S reads using DADA2 denoising. For further information go to qiime2.org and use the ‘Tutorials’ and ‘Plugins’ links under the ‘Docs’ tab.*

**Pre-QIIME Steps:**

Depending on how you received your reads you will use slightly different files to start the process, but in general it is a similar process. In our case, we need to create a manifest file pointing QIIME2 to the forward read and the reverse read. See the ‘Metadata in QIIME 2’ page on qiime2.org (<https://docs.qiime2.org/2023.5/tutorials/metadata/>) for more details.

*Note: I structure my project directory with subdirectories as follows*:

* metadata: $ projectID/metadata/
* rawreads: $ projectID/reads/
* for running QIIME2 and its products: $ projectID/QIIME2/

You should make a directory on your scratch space called ‘16S\_ReAnalysis’:

cd /scratch/user/ljpinnell ## navigate to your scratch directory (use your username)

mkdir 16S\_ReAnalysis ## make the project directory

Within this directory set up three sub-directories as I described above and put the manifest file into metadata/.

**Gathering Fastq samples**

**Option 1: Use fastq samples from a sequencing project.**

In the VERO group, this usually means you’ll get a link to access sequencing samples from a one drive folder. Then, you’ll use Globus to download the samples into either Grace, or Terra on TAMU’s HPRC for analysis. Download these samples into the “16S\_ReAnalysis” directory.

**Option 2: Use SRA list to download reads**

One example is to use reads from NCBI’s Sequence Read Archive (SRA) we need to use SRA toolkit to download the reads we are going to analyze. Navigate into your project directory:

cd 16S\_ReAnalysis

Using Cyberduck, FileZilla, SFTP, or the web portal upload the ‘SRR\_Acc\_list.txt’ file I gave you to the reads/ directory.

This file, ‘SRR\_Acc\_list.txt’, contains a list of SRA accessions, one per row and can be made using NCBI’s website or manually using a text editor.

Now we will use the ‘sra-download.sh’ script to download the sequence data. I sent you this script as an example so you can upload to HPRC and run. The script looks like this:

#!/bin/bash

#SBATCH -t 02:00:00 -p short --mem=25G --nodes=1 --ntasks-per-node=1

module load SRA-Toolkit

## part 1: prefetch some info about the runs

prefetch -O SRA --option-file SRR\_Acc\_List.txt

## download the sequence data into a directory called reads

fasterq-dump SRA/SRR\* -O reads

## compress the downloaded reads with gzip

gzip reads/\*.fastq

If you don’t have this example script, you can use the linux command “nano” or by making a text file called “sra-download.sh” on your computer and uploading it to the HPRC.

To run this (and every sbatch script on the HPRC) you use the following command:

sbatch sra-download.sh

From now I won’t be explicitly stating this part but run every script this way. This script will spit a bunch of sub-directories named SRR\* into a directory called SRA/ and then use those to download all the fastq files into the reads/ directory. Once finished you can remove the SRA/ directory since we don’t need those temporary files anymore:

rm -r SRA/

The result of this process is that in the reads/ directory we now have two read files for each sample: [sampleID]\_1.fastq.gz and [sampleID]\_2.fastq.gz representing the forward and reverse reads, respectively. The next step is create a manifest file QIIME2 will use to find the reads.

**STEP 1: CREATING MANIFEST FILE**

To import the reads, QIIME2 needs the absolute path to your forward and reverse reads. To do this you can make a manifest file that we will use to import the reads during the first step and contains the Sample IDs, and absolute paths to both reads. Follow the steps in the ‘QIIME\_manifest\_creation’ document I sent you. When you’re done your manifest file should be located here:

**[ProjectID]/metadata/manifest.tsv**

Now we are ready to start QIIME2. **From now on you will make and run all your scripts from within the [ProjectID]/QIIME2/ directory.** At each step you should make your script by using nano. You can nano into the filename you want to create and write your script there. For example, if you were going to make the script for step one (artifact\_creation.sh) you would type:

nano artifact\_creation.sh

Then you type out everything I’ve given you for that step and save it. **Copy and paste at your own peril!** Apostrophes and other characters in MS Word don’t copy right into command-line and it’ll mess up your code. I’d recommend you type it out as I’ve written it the first time. Then in future scripts just copy and paste the parts that are the same from your previous scripts in command-line.

**STEP 2: IMPORTING INTO QIIME2**

*If your reads are not in the demultiplexed manifest style, you’ll need to use a different ‘–type’ flag. Check qiime2 website for details. Directions here (*[*https://docs.qiime2.org/2023.5/tutorials/importing/*](https://docs.qiime2.org/2023.5/tutorials/importing/)*)*

Here we will create an ‘artifact’, which is just the filetype QIIME2 uses throughout. Basically, we are just importing the reads into QIIME2. The ‘\’ indicates that the command continues onto the next line. This may not work properly on the HPRC (I think it does though) and I’ve used them here so it’s easier to follow the commands. If it doesn’t work just put all the commands on a single line. You need to point it to the manifest file where you stored the \*.fastq.gz files. Call this script artifact\_creation.sh:

Notice the use of the relative path to point to the “manifest.tsv” file. So, make sure you navigate to the QIIME2 directory and place that “artifact\_creation.sh” script there.

#!/bin/bash

#SBATCH -t 01:00:00 -p short –mem=10G –nodes=1 –ntasks-per-node=1

module load QIIME2/2023.2

qiime tools import –type ‘SampleData[PairedEndSequencesWithQuality]’ \

--input-path ../metadata/manifest.tsv -–input-format PairedEndFastqManifestPhred33V2 \

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

This will output a qza ‘artifact’ with the demultiplexed reads in it.

**STEP 3: SUMMARIZE/VIEW THE DEMULTIPLEXED READS**

This step just converts the .qza file into a .qzv file that we can view on the interactive qiiime2 viewer online. We can see how many reads per sample and take a look at read QC. So, run the demux\_sumarize.sh script:

#!/bin/bash

#SBATCH -t 01:00:00 -p short --mem=10G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

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

Transfer the demux.qzv file off the HPRC to your local machine (SFTP, Cyberduck, or FileZilla) so you can drag and drop it into [view.qiime2.org](http://view.qiime2.org/). Click on the ‘Interactive Quality Plot’ tab and you can see box and whisker plots for each bp position of the sequenced reads. Based on read qualities at each position determine if/where you want to trim and/or truncate them. This is a bit subjective, but typically the reads (especially the reverse) will drop off near the end so you’ll have to truncate them at some point to get rid of low-quality bp positions. Make note of where you want to trim and/or truncate.

**STOPPING POINT: e-mail when you get here and will look at the file together before proceeding!**

**STEP 4: DENOISING/QC WITH DADA2**

*Note: QIIME also provides the option of using Deblur for this step, but its more finicky and it doesn’t seem to make a difference in the biological interpretations down-stream so I stick with DADA2. The difference between the two lies in their error correction methods.*

Here, we are merging paired-reads, denoising and dereplicating sequences, removing chimeras, and producing amplicon sequence variants (ASVs; essentially more precise OTUs). This is where we use the values we picked from trimming (from start) and truncating (from the end) for both reads.

#!/bin/bash

#SBATCH -t 24:00:00 -p medium --mem=250G --nodes=1 --ntasks-per-node=28

module load QIIME2/2023.2

qiime dada2 denoise-paired --i-demultiplexed-seqs paired-end-demux.qza \

--p-trunc-len-f [x] -–p-trim-left-f [x] --p-trunc-len-r [x] –-p-trim-left-r [x]\

--p-n-threads 28 --o-table table.qza \ --o-representative-sequences rep-seqs.qza \

--output-dir DADA2

You’ll need to insert the proper trim and truncate lengths. The --p-n-threads’ flag is telling QIIME how many threads to use. You can pick a number here, but 0 means use all available. Threads are another way of saying CPUs, and I suggest using 28 for the HPRC and the number of samples here.

This will output the ASV table (table.qza), and rep-seqs.qza, which contains a representative sequence for each ASV. The created DADA2 directory contains error correction stats.

**STEP 5: CLASSIFICATION OF ASVs**

There are additional steps to classification if you haven’t already trained a classifier, but I’ve got both a Greengenes and SILVA132 release ready to go for use with our lab’s primer pair (341/785). For training a new classifier follow the steps listed here: <https://docs.qiime2.org/2019.7/tutorials/feature-classifier/>

Additionally, I can give you the steps I used to make the classifier used here. The steps involve downloading the database you want to use, trimming to the region you’re interested in with your primer pair, then training a Naïve Bayes classifier using our target region and reference taxonomy. This produces the ‘classifier.qza’ artifact we’re going to use here. The path given in the script below will not work on the HPRC, so I’ll give you a new one.

**IF YOU GET HERE BEFORE I SEND YOU A CLASSIFIER LET ME KNOW (IT WONT WORK WITHOUT ONE)!!!**

To assign taxonomy run the following assign\_taxonomy.sh script:

#!/bin/bash

#SBATCH -t 24:00:00 -p medium --mem=250G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

qiime feature-classifier classify-sklearn \

--i-classifier /scratch/group/vero\_research/QIIME\_Classifiers/qiime2-2023.2/SILVA138.1/341f-785r/silva-138.1-ssu-nr99-341f-785r-classifier.qza \

--i-reads rep-seqs.qza --o-classification taxonomy.qza

In this case, we have our taxonomy assigned to our ASVs based on SILVA 138.1 classifications and the “341f, 785r” 16S sequencing primers. We could also use GreenGenes or another DB. Make sure you pay attention to the selection of classifier so that it also matches the primers that were used for 16S sequencing.

**STEP 6: FILTERING OUT ASVs ASSIGNTED TO CHLOROPLAST OR MITCHONDRIA**

Some people don’t do this, but I always do. The chloroplast removal is a leftover from my marine days, but since we have eukaryotic hosts it is generally necessary to remove mitochondrial mapped reads. I recommend doing both, as we do here. I split my new feature table into a new directory called ‘no-chloro-no-mito’ that you have to make before running the script. So, make that new directory:

mkdir no-chloro-no-mito

then run filter\_chloro.sh:

#!/bin/bash

#SBATCH -t 01:00:00 -p short --mem=10G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

qiime taxa filter-table --i-table table.qza --i-taxonomy taxonomy.qza \

--p-exclude mitochondria,chloroplast --o-filtered-table no-chloro-no-mito/table.qza

This will put a new ‘table.qza’ into the no-chloro-no-mito/ directory that you will use for everything after the next step.

**STEP 7: FILTERING OUT REP-SEQS ASSIGNTED TO CHLOROPLAST/MITCHONDRIA**

Same idea as step 9, but instead of filtering the ASVs out we are filtering out the rep. sequences assigned to those ASVs. Run filter\_chloro\_seqs.sh:

#!/bin/bash

#SBATCH -t 01:00:00 -p short --mem=10G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

qiime taxa filter-seqs --i-sequences rep-seqs.qza --i-taxonomy taxonomy.qza \

--p-exclude mitochondria,chloroplast \

--o-filtered-sequences no-chloro-no-mito/rep-seqs.qza

Remember, that from now on, everything we do is inside of no-mito-no-chloro/. This means start making and running your scripts from that directory from step 8. Navigate there.

**STEP 8: VISUALIZE THE NEW FEATURE TABLE**

Double reminder! From now on we should be inside the no-mito-no-chloro directory!

Similar to visualizing the demultiplexed reads, this will let us see how many ASVs were assigned to each sample. If you decide to rarefy your data (which you have to if you continue with QIIME for further down-stream analysis) you will use this visualization to help you pick a depth.

#!/bin/bash

#SBATCH -t 01:00:00 -p short --mem=10G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

qiime feature-table summarize --i-table table.qza --o-visualization table.qzv

Pull off the server, upload to view.qiime2.org and look at the ‘Interactive Sample Detail’ to help choose your sampling depth. You won’t need the sampling depth for a while if you decide to rarefy, and normally you actually won’t ever need it because we don’t typically rarefy*.*

**STEP 9: BUILDING A TREE FOR PHYLOGENETIC-BASED ANALYSES**

This is a four-step process that I just do in one script because I don’t want to use 4 scripts if I can use one. The four steps are: 1) alignment of the rep seqs, 2) masking unconserved and highly gapped columns, 3) making an unrooted tree, and 4) mid-point rooting the tree. Run tree.sh:

#!/bin/bash

#SBATCH -t 24:00:00 -p medium --mem=250G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

qiime alignment mafft --i-sequences rep-seqs.qza \

--o-alignment aligned-rep-seqs.qza

qiime alignment mask --i-alignment aligned-rep-seqs.qza \

--o-masked-alignment masked-aligned-rep-seqs.qza

qiime phylogeny fasttree --i-alignment masked-aligned-rep-seqs.qza \

--o-tree unrooted-tree.qza

qiime phylogeny midpoint-root --i-tree unrooted-tree.qza \

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

This will spit out 4 new files, rooted-tree.qza being the final one. Reminder: You should have run this inside of no-mito-no-chloro/. Go ahead and copy the taxonomy.qza file into this directory now. At this point you should have the following files in no-mito-no-chloro/:

* table.qza
* rep-seqs.qza
* rooted-tree.qza
* taxonomy.qza

… and some other files from the tree script that we don’t really need.

*Note: At this point I stop using QIIME2 and import everything into R where I use phyloseq for the analysis. To do so follow the exporting to phyloseq steps below.*

*If interested in just seeing what you get following QIIME2 feel free to follow the steps below starting with ‘Core Diversity Metrics’. NOTE: Those steps are not edited from previous versions so may contain errors. For further reading check out the ‘Moving Pictures Tutorial’ on qiime2.org.*

**STEP 10: Exporting from QIIME2 (converting the .qza to normal files)**

Before running this script, create the exported/ directory where you are sending the files. Remember this should be inside no-mito-no-chloro/.

mkdir exported

Now we can run the export\_phyloseq.sh script:

#!/bin/bash

#SBATCH -t 01:00:00 -p short --mem=10G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

qiime tools export --input-path rep-seqs.qza --output-path exported/

qiime tools export --input-path taxonomy.qza --output-path exported/

qiime tools export --input-path rooted-tree.qza --output-path exported/

qiime tools export --input-path table.qza --output-path exported/

This will spit 4 new files into the exported/ directory:

* feature-table.biom (the ASV table)
* taxonomy.tsv (the taxonomy table)
* dna-sequences.fasta (the representative sequences of the ASVs)
* tree.nwk (the tree file of the rep seqs)

**STEP 11: Prepping the taxonomy table**

Navigate into the exported/ directory and run everything from now on in there. We need to alter the headers of the taxonomy.tsv file. The headers are tab separated so try not to mess with the spacing of the headers. If you do just use tab to re-separate each of the headers. Use nano to edit the headers (or whatever editor you prefer).

In the original file (taxonomy.tsv) you should have the following tab separated headers:

Feature ID Taxon Confidence

Change these headers to the following and save as new file called biom-taxonomy.tsv:

#OTUID taxonomy confidence

These new headers will allow you to amend the feature-table.biom with taxonomic information.

**STEP 12: Amending the BIOM file**

Phyloseq wants the BIOM table to include taxonomy, so here we are amending our feature-table.biomwith the formatted taxonomic information found in biom-taxonomy.tsv. Use the convert\_biom.sh script to do so:

#!/bin/bash

#SBATCH -t 01:00:00 -p short --mem=10G --nodes=1 --ntasks-per-node=1

module load QIIME2/2023.2

biom add-metadata -i feature-table.biom -o table-with-taxonomy.biom \

--observation-metadata-fp biom-taxonomy.tsv --sc-separated taxonomy

This will result in a newly created biom table file (table-with-taxonomy.biom) that we will use to import into phyloseq. Your data is ready to be imported into R!

Pull the table-with-taxonomy.biom, tree.nwk, and dna-sequences.fasta off the HPRC and onto your local machine. These are what you will use in phyloseq.

-------------------------------------------------------- END --------------------------------------------------------

See below for continuing to use QIIME for analyses, but keep in mind I haven’t edited those steps in a few years.

**OPTIONAL QIIME2 PROCESSING:**

**CORE DIVERSITY METRICS**

This is a QIIME2 developed pipeline that calculates a bunch of alpha and beta diversity metrics. Here, we will need the sampling depth from our second table without chloro and mito seqs. Run this script (core\_diversity.sh) from within the no-chloro-no-mito/ directory:

qiime diversity core-metrics-phylogenetic --i-phylogeny ../rooted-tree.qza \

--i-table table.qza --p-sampling-depth 500 \

--m-metadata-file ../../metadata/metadata.txt --output-dir core-metrics-results

This will output everything to a new directory called ‘core-metrics-results’. This folder cannot exist when you run the script, or it will fail. The script itself will produce it. This should produce results using Shannon, jaccards, faiths PD, pielou’s eveness, bray-curtis, unifrac, and weighted unifrac. There should also be visualization’s you can look at at view.qiime2.org for PCoA of beta diversity. In this example I set the sampling depth at 500, based on the table.qzv file made after filtering chloro/mito. Transfer all the “.qzv” files onto your computer and explore them using view.qiime2.org

**MAKING RELATIVE ABUNDANCE BAR PLOTS**

Make the relative abundance plot commonly seen in 16S studies for all 7 taxonomic ranks. The resulting .qzv file can be viewed. Again, run taxa\_barplots.sh from within no-mito-no-chloro:

qiime taxa barplot --i-table table.qza --i-taxonomy taxonomy.qza \

--m-metadata-file ../../metadata/metadata.txt \

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

You can transfer the resulting .qzv file to your computer and open it on view.qiime2.org. Take a look at all the different taxonomic levels and the metadata categories. This ends the basics of using QIIME for processing and analysis.