Apr 21, 2025

Intro to Sequence Analysis in QIIME2

DOI
https://dx.doi.org/10.17504/protocols.io.kxygx757ol8j/v1

Elle Barnes[1]

[1]Rochester Institute of Technology

MEGA Lab

E Elle Barnes
Rochester Institute of Technology

DOI: https://dx.doi.org/10.17504/protocols.io.kxygx757ol8j/v1

Protocol Citation: Elle Barnes 2025. Intro to Sequence Analysis in QIIME2. **protocols.io** https://dx.doi.org/10.17504/protocols.io.kxygx757ol8j/v1

Protocol status: Working
We use this protocol and it's working

Created: April 18, 2025

Last Modified: April 21, 2025

Protocol Integer ID: 130980

Keywords: amplicon sequencing, QIIME2, DADA2, 16S rRNA, qiime2 from sequence import, sequencing analysis, other amplicon dataset, rrna dataset, sequencing dataset, sequence import, microbiome, microbiome of zoo animal, environmental microbiology students at rit, sequence analysis, sequencing, qiime2 this protocol, environmental microbiology student, qiime2, archaea, bacteria, taxonomy table export for downstream analysis, downstream analysis, sequence, throughput amplicon, dataset, microbiology

Disclaimer

This protocol is built from the advice and guidance found in the QIIME2 documentation as well as other sources.

Abstract

This protocol is designed for students who are new to high-throughput amplicon sequencing analysis. It mainly focuses on 16S rRNA datasets (i.e., bacteria & archaea), but can be modified for other amplicon datasets (e.g., ITS, 18S). This protocol includes all the necessary steps for analysis in QIIME2 from sequence import to feature/taxonomy table export for downstream analysis in other programs, such as R.

The files provided as examples were generated by my Environmental Microbiology students at RIT and represent a small sequencing dataset generated from the gut (fecal) microbiome of zoo animals.

Understanding File Formatting

1 All amplicon sequencing experiments begin, at some point or another, as raw sequence data. This is usually a set of **FASTQ files** (.fastq.gz) which are a form of zipped sequence file containing:

- DNA sequences
- Quality scores for each base

At minimum, you will have **three** files if you performed *paired-end* sequencing:

- forward.fastq.gz
- reverse.fastq.gz
- a metadata file with a column of per-sample barcodes for use in FASTQ demultiplexing (or two columns of dual-index barcodes)

Note

In order to separate your files into individual samples, you will need to perform **demultiplexing**. However, I will not cover that here because we provided our metadata file of barcodes to the Illumina sequencer so it was able to demultiplex our files for us.

Importing Sequences into QIIME2

2 We will use the **Fastq Manifest** format to import our fastq.gz files.

First, you will need to create a text file called a "manifest" that will map the sample identifiers to the absolute file paths that contain your sequence files. The manifest file is a tab-separated (**.tsv**) text file.

The manifest has three columns:

- sample-id
- forward-absolute-filepath
- reverse-absolute-filepath

```
sample-id     forward-absolute-filepath       reverse-absolute-filepath
sample-1      $PWD/some/filepath/sample0_R1.fastq.gz  $PWD/some/filepath/sample1_R2.fastq.gz
sample-2      $PWD/some/filepath/sample2_R1.fastq.gz  $PWD/some/filepath/sample2_R2.fastq.gz
sample-3      $PWD/some/filepath/sample3_R1.fastq.gz  $PWD/some/filepath/sample3_R2.fastq.gz
sample-4      $PWD/some/filepath/sample4_R1.fastq.gz  $PWD/some/filepath/sample4_R2.fastq.gz
```

Figure 1. Example format for manifest file. If you are using the RIT RC cluster, your file path should start with /shared/rc/... with everything following the "..." corresponding to the project where the data is stored on the cluster and the folders within down to the each fastq.gz file.

3 **Import your manifest.tsv** to the same folder where your sequence files are located. This folder should *only* contain your fastq.gz files and manifest.tsv.

4 Enter the following command replacing "datafolder" for your own input-path (i.e., the folder where your manifest and sequences can be found).

This input path will depend on the directory you are already in (e.g., if you are in: /folder1, but your sequences are in: /folder1/folder2/folder3/sequence.fastq.gz then your input-path should be: folder2/folder3).

Command

new command name

```
qiime tools import \
  --type 'SampleData[PairedEndSequencesWithQuality]' \
  --input-path datafolder \
  --output-path demux_pairedend_raw.qza \
  --input-format PairedEndFastqManifestPhred33V2
```

Note

All recent Illumina sequencing data should use the PHRED offset = 33. Older sequencing files (e.g., before 2014) may use PHRED64.

5 The output of the previous step will be a **.qza file**. It is essentially one large "data artifact" file containing all the sequence-by-sample information we want to analyze.

To view this file, we first need to convert it to a **.qzv file** using the following command:

Command

new command name

```
qiime demux summarize \
  --i-data demux_pairedend_raw.qza \
  --o-visualization demuxpair.qzv
```

6 Download this .qzv file and drop it into QIIME2 view: **https://view.qiime2.org/**
Here is an example of a qzv output from a previous dataset:

FILE demuxpair.qzv 313.8KB

7 Take a look at the **interactive quality plots**. These will tell us where the quality in our forward and reverse reads dip below our quality threshold (usually we use Q > 30), usually at the ends of the reads. You can zoom in on sections of the forward and reverse read to see them in more detail.

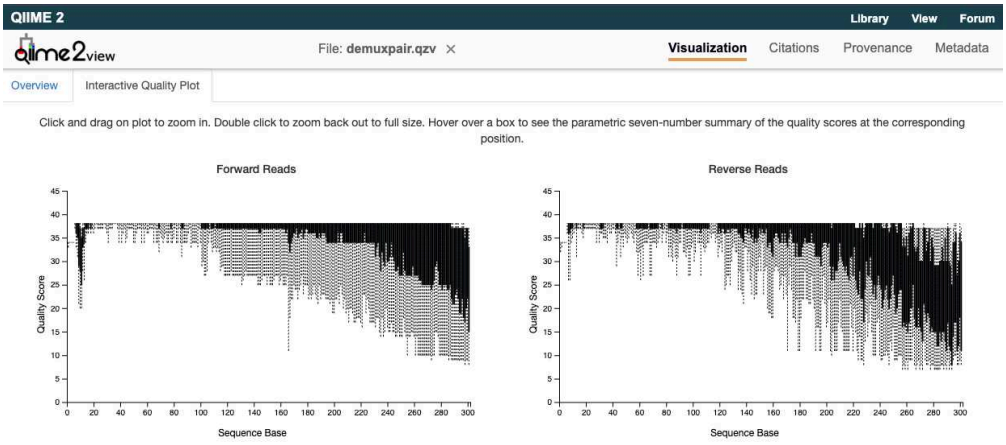


Figure 2. Example visualization of sequence quality after import as viewed in QIIME2 view.

Take note of which basepair position Q < 30 (median) for the forward and reverse. For instance, in the file attached in the previous step, that would be bp position 294 for the forward and either 228 (stringent) or 257 (less stringent) for the reverse. These positions will be used in downstream commands.

Primer/Adapter Removal using Cutadapt

8 We need to make sure to **remove our primer and adapter sequences** from our reads since they represent non-biological sequences. We can use cutadapt to do this which is available within QIIME2.

Run the following command replacing the *p-front-f* and *p-front-r* with the primer sequences relevant to your study:

Command

new command name

```
qiime cutadapt trim-paired \
  --i-demultiplexed-sequences demux_pairedend_raw.qza \
  --p-front-f GTGCCAGCMGCCGCGGTAA \
  --p-front-r GACTACHVGGGTWTCTAAT \
  --o-trimmed-sequences seqtrim.qza \
  --verbose
```

Note

If you're going to use **DADA2** to denoise your sequences, you can remove non-biological sequences at the same time as you call the denoising function. DADA2's denoise functions have the --p-trim parameter where you can specify to remove base pairs from the 5' end of your reads.

9 To view this file, we first need to convert it to a **.qzv file** and import it into QIIME2 view as in the step above. Re-evaluate the quality of your sequences.

Command

new command name

```
qiime demux summarize \
  --i-data seqtrim.qza \
  --o-visualization demuxtrim.qzv
```

Denoising via DADA2

10 Reads should then be denoised into **amplicon sequence variants (ASVs)** or clustered into operational taxonomic units (OTUs) to:

- reducing sequence errors
- remove chimeric sequences
- remove singletons
- and finally, join denoised paired-end reads & dereplicate sequences

In our lab, we generally choose to work with ASVs to start and these reads can always be clustered into OTUs later on. For more explanation regarding the difference between ASVs and OTUs and why denoising/clustering is important, check out this article by **Zymo Research**.

Remember, you should decide on where to *trim* (remove from 5' end) and *truncate* (remove from 3' end) based on the median quality scores at each base. We want to remove redundant sequences at the 5' end (if you did not first use cutadapt) and low-quality regions (Q-score < 30, often on the 3' end), while *maintaining enough overlap sequence to merge the F and R reads*.

This step will take a long amount of time (several hours). It is often better to submit it via a batch script if you have > 150 files. If you have less than that, I have found that you can successfully run it within the time limit of RIT's RC interactive node session.

Command

new command name

```
qiime dada2 denoise-paired \
  --i-demultiplexed-seqs demux_pairedend_raw.qza \
  --p-trim-left-f 10 \
  --p-trim-left-r 10 \
  --p-trunc-len-f 250 \
  --p-trunc-len-r 220 \
  --p-n-threads 0 \
  --o-representative-sequences repseqsdada16S.qza \
  --o-table tabledada16S.qza \
  --o-denoising-stats statsdada16S.qza \
  --verbose
```

11 To view the output, convert it to **.qzv** and import into QIIME2 view. Here is an example of a DADA2 output from a previous dataset:

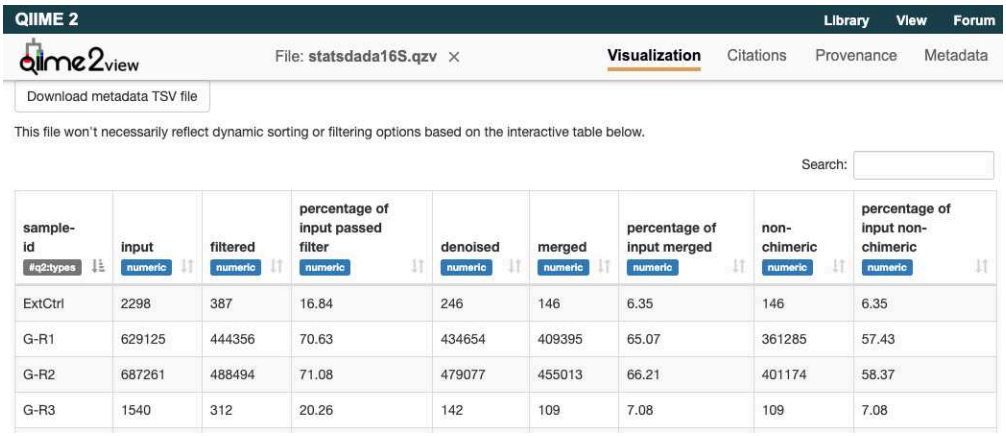
statsdada16S.qzv 1.2MB

Command

new command name

```
qiime metadata tabulate \
  --m-input-file statsdada16S.qza \
  --o-visualization statsdada16S.qzv
```

12 **Evaluate the output of DADA2** by considering how many sequences you had at start (input) vs. how many sequences passed each step of DADA2 (filtered, denoised, merged, non-chimeric).



sample-id #q2:types	input numeric	filtered numeric	percentage of input passed filter numeric	denoised numeric	merged numeric	percentage of input merged numeric	non-chimeric numeric	percentage of input non-chimeric numeric
ExtCtrl	2298	387	16.84	246	146	6.35	146	6.35
G-R1	629125	444356	70.63	434654	409395	65.07	361285	57.43
G-R2	687261	488494	71.08	479077	455013	66.21	401174	58.37
G-R3	1540	312	20.26	142	109	7.08	109	7.08

Figure 3. Example visualization of DADA2 output as viewed in QIIME2 view.

There is technically no "right" way to evaluate your DADA2 results as this will vary by dataset. Ultimately, we want to retain as many reads as possible post-DADA2 and this will depend on the quality of your sequences to start as well as the parameters you set for *trim* & *truncate* (which attempt to remove low quality reads). You may want to consider running DADA2 multiple times with slightly different trim/truncate parameters and comparing each output.

For more details on how to best evaluate your denoising results, check out this **video lecture**.

Assign Taxonomy

13 There are many **databases** containing 16S rRNA sequences and each has their strengths and weaknesses. In our lab, we use **GTDB** since it has the most up-to-date taxonomic names for bacteria and archaea (which have changed quite a lot over the last few years). Other common databases include SILVA and Greengenes.

For use with other types of amplicon datasets (e.g., ITS for fungi or 18S for algae), you will need a different database than the one linked below, but all other steps remain unchanged.

In QIIME2, these databases are known as **feature classifiers**. A list of available **pre-trained classifiers** is available **here**.

However, it is technically better to train whichever database you use on your own specific data instead of using a pre-trained classifier. This also is required if you plan to use your own database. For details of how to train your own classifier, see the **QIIME2 documentation**.

14 Let's proceed with the **GTDB diverse weighted** pre-trained classifier for now.

Download the classifier from **here** and import it into your cluster directory (the same location as you have been placing your inputs/outputs for the previous commands).

15 Run the following command to **assign taxonomy** to your sequences.

This step also usually takes a long amount of time (hours) and so it is often better to submit it via a batch script for datasets with > 150 samples.

Command

new command name

```
qiime feature-classifier classify-sklearn \
    --i-classifier gtdb_diverse_weighted_classifier_r220.qza \
    --i-reads repseqsdada16S.qza \
    --p-reads-per-batch 1000 \
    --o-classification taxonomy16S.qza \
    --verbose
```

Export files for analysis in R

16 I prefer to use R to analyze my samples (e.g., generate plots, perform statistical tests, etc.).

The first command exports your ASVs into a **feature table** (i.e., OTU or ASV table) and the second command converts your feature table to a .tsv file. Your feature table is a matrix of sample x ASV, where each cell is the number of reads.

Command

new command name

```
qiime tools export \
  --input-path tabledada16S.qza \
  --output-path exported-feature-table
```

Command

new command name

```
biom convert \
  -i exported-feature-table/feature-table.biom \
  -o featuretable16S.tsv \
  --to-tsv
```

17 This command exports your **taxonomy file** as a .tsv, which links your ASV ID to its full taxonomy (Domain to Species).

Command

new command name

```
qiime tools export \
  --input-path taxonomy16S.qza \
  --output-path exported
```

18 This final command exports a **FASTA file** with the sequences associated with individual ASV IDs. This file is very useful for potential future analyses where you want to align the sequences in your dataset to other published sequences without having to redo the entire pipeline.

Command

new command name

```
qiime tools export \
  --input-path repseqsdada16S.qza \
  --output-path exported
```

Protocol references

Bolyen, Evan, et al. "Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2." *Nature biotechnology* 37.8 (2019): 852-857. https://doi.org/10.1038/s41587-019-0209-9

Callahan, Benjamin J., et al. "DADA2: High-resolution sample inference from Illumina amplicon data." *Nature methods* 13.7 (2016): 581-583. https://doi.org/10.1038/nmeth.3869

Parks, Donovan H., et al. "A complete domain-to-species taxonomy for Bacteria and Archaea." *Nature biotechnology* 38.9 (2020): 1079-1086. https://doi.org/10.1038/s41587-020-0501-8