ALGORITHM THEORETICAL BASIS DOCUMENT (ATBD):

MICROBIAL COMMUNITY COMPOSITION

PREPARED BY	ORGANIZATION	DATE
Geoffrey House	SCI	08/03/2021

APPROVALS	ORGANIZATION	APPROVAL DATE
Kate Thibault	SCI	09/08/2021
Steve Stone	CI	09/08/2021

RELEASED BY	ORGANIZATION	RELEASE DATE
Tanisha Waters	CM	10/12/2021

See configuration management system for approval history.

The National Ecological Observatory Network is a project solely funded by the National Science Foundation and managed under cooperative agreement by Battelle. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Change Record

REVISION	DATE	ECO #	DESCRIPTION OF CHANGE
A	10/12/2021	ECO-06667	Initial release

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

1 DESCRIPTION

This document describes how raw marker gene (16S and ITS rRNA gene) sequencing data (Soil: DP1.10108.001, Benthic: DP1.20280.001, Surface water: DP1.20282.001) are analyzed to provide the microbial community composition data products (Soil: DP1.10081.001, Benthic: DP1.20086.001, Surface water: DP1.20141.001). The microbial community composition data products consist of taxonomy assignments and the number of sequences represented in each sample for each taxonomy assignment.

1.1 Purpose

This document details the algorithms used for creating NEON Level 1 data products for microbial community composition from Level 0 data, and ancillary data as defined in this document (such as calibration data) obtained via instrumental measurements made by DNA sequencers (for example, Illumina MiSeq). It includes a detailed discussion of measurement theory and implementation, appropriate theoretical background, data product provenance, quality assurance and control methods used, and approximations and/or assumptions made.

1.2 Scope

The theoretical background and entire algorithmic process used to derive Level 1 data from the Level 0 data for marker gene sequencing (16S or ITS) is described in this document. The DNA sequencer type employed is the Illumina MiSeq, which generates paired-end sequences. This document does not provide computational implementation details, except for cases where these stem directly from algorithmic choices explained here.

2 RELATED DOCUMENTS, ACRONYMS AND VARIABLE NOMENCLATURE

2.1 Applicable Documents

AD[01]	NEON.DOC.000001	NEON OBSERVATORY DESIGN
AD[02]	NEON.DOC.005003	NEON Scientific Data Products Catalog
AD[03]	NEON.DOC.002652	NEON Level 1, Level 2 and Level 3 Data Products Catalog
AD[04]	NEON.DOC.001152	NEON Aquatic Sampling Strategy
AD[05]	NEON.DOC.003044	Aquatic Microbial Sampling
AD[06]	NEON.DOC.014048	Soil Biogeochemical and Microbial Sampling
AD[07]		NEON User Guide to Microbial Community Composition

2.2 Reference Documents

RD[01]	NEON.DOC.000008	NEON Acronym List
RD[02]	NEON.DOC.000243	NEON Glossary of Terms

2.3 Acronyms

Acronym	Explanation
16S	Portion of the ribosomal RNA (rRNA) gene used to identify Archaea and Bacteria
AOS	Aquatic Observation System
ASV	Amplicon sequence variant; a unique sequence of either the 16S or ITS portions of the rRNA gene after computational error correction that represents biological sequence variability in the input data; it is used as input to the taxonomy classification process
ATBD	Algorithm Theoretical Basis Document
CI	NEON Cyberinfrastructure
CVAL	NEON Calibration, Validation, and Audit Laboratory
DP	Data Product

ITS	Internal transcribed spacer; portion of the ribosomal RNA (rRNA) gene used to identify Fungi
L0	Level 0
L1	Level 1
OTU	Operational taxonomic unit; a group of similar sequences that are generally created by clustering sequences that have a minimum percentage of sequence similarity, usually 97%
QA/QC	Quality assurance and quality control

3 DATA PRODUCT DESCRIPTION

3.1 Variables Reported

The community composition-related L1 DPs provided by the algorithms documented in this ATBD are displayed in the accompanying data publication workbook files: mmg_benthic_datapub, mmg_soil_datapub, mmg_surfaceWater_datapub

3.2 Input Dependencies

Table 3-1: List of marker gene-related L1 DPs that are transformed into L1 community composition DPs in this ATBD.

Description	Data Product Number
Marker gene sequencing data from benthic samples	DP1.20280.001
Marker gene sequencing data from surface water samples	DP1.20282.001
Marker gene sequencing data from soil samples	DP1.10108.001

3.3 Product Instances

For all aquatic sites and core terrestrial sites, marker gene sequences are generated multiple times per year. At terrestrial gradient sites, marker gene sequences are generated once every few years. See the NEON User Guide to Microbial Community Composition (DP1.10081.001; DP1.20141; DP1.20086.001) [AD07] for more detail about where and when the marker gene sequence information is collected.

3.4 Temporal Resolution and Extent

Benthic microbe samples are collected three times a year at stream sites and are not collected at lake or river sites [AD04, AD05]. Surface water microbe samples are collected 12 times a year at stream sites, and

six times a year at river and lake sites [AD04, AD05]. Marker gene sequences and community composition analysis are run on all samples from aquatic sites.

Soil microbe samples are collected three times a year at core sites, and three times every five years at gradient sites [AD06]. At core sites, each soil sample results in a microbial community composition analysis sample. See TOS Protocol and Procedure: SLS - Soil Biogeochemical and Microbial Sampling [AD06] for full details on the soil sampling temporal resolution and extent. Briefly, during the majority of bouts (n = 3 per field season, except in Alaska), only the top horizon is sampled, either the organic (O) horizon if present, else the top 30 cm of mineral (M) soil. Every five years during the peak greenness bout, both O and M horizons are sampled and measured when both are present. At gradient sites, microbial community composition is only measured in 1 out of 5 field seasons, although during all 3 bouts in that season. For transition bouts, only the top horizon is sampled, whereas for the peak greenness bout, both O and M horizons are sampled when both are present. Historic temporal sampling design changes have been documented in detail in the NEON User Guide to Soil physical and chemical properties, periodic (DP1.10086.001), see “sampling design changes” section.

3.5 Spatial Resolution and Extent

Surface water microbe samples are collected in one location in stream and river sites, and at up to three locations in lake sites (with more than three total samples possible if the lake is thermally stratified) [AD04, AD05]. Benthic microbe samples are not collected at fixed locations in streams, but rather target specific habitat types and substrate types for each site [AD04, AD05]. Soil microbe samples are collected from each of three sub-plots within 10 plots during each round of sampling (30 samples total per sampling round); of these, four plots (12 samples) are within the tower airshed and the other six plots (18 samples) are distributed across the site [AD06].

4 SCIENTIFIC CONTEXT

Microbial communities represent a key, but usually understudied facet of biodiversity in ecosystems. In addition to their diversity, microbial communities underpin terrestrial and aquatic food webs, in part because they strongly affect nutrient cycling through ecosystems. Therefore measurements of microbial community composition are important not only to address questions about changes in microbial community biodiversity through time, but also questions about nutrient availability, chemical transformations, and broader ecosystem functions. Microbial community composition is measured by NEON using DNA sequencing of marker gene ‘barcodes’ for both bacteria/archaea, and for fungi.

4.1 Theory of Measurement

There is a long history of identifying species of animals by sequencing a carefully chosen portion of a gene (often referred to as a sequence ‘barcode’) that contains a large amount of sequence variation between different species but relatively little variation within the same species. Although species definitions are less precise for microbes compared to animals, which makes microbial taxonomic assignment more difficult, the same sequence ‘barcoding’ principal is still useful in order to identify groups of microbes that

share similar sequence 'barcodes'. The microbes represented in each of these groups are then generally assumed to have similar whole genomes, and by extension, similar functional characteristics. The marker genes used here (a portion of the 16S ribosomal RNA [rRNA] gene for bacteria and archaea (16S rRNA), and the first internal transcribed spacer (ITS1) region of the rRNA gene for fungi, are commonly used in microbial ecology studies focused on these microbial groups. Although the specific portion of these marker genes that is sequenced varies somewhat between studies based on which PCR primers are used, NEON's sequencing protocol for these marker genes allows general comparisons to be made between NEON's measurements and many other independent studies.

4.2 Theory of Algorithm

There can be a large amount of sequence variation represented in marker gene datasets, like those analyzed here for the 16S rRNA and the ITS rRNA marker genes. When analyzing these data, there are two major goals:

1) Reducing sample processing and sequencing artefacts that can generate sequence variation that is not biologically based (e.g. through chimeric sequence generation during PCR amplification, sequencing errors, or other artefacts that produce sequence data that does not represent the input biological samples).

2) Summarizing the biological sequence variation in a way that is useful for downstream analysis (e.g. taxonomic classification, assessing ecological correlations, etc.) while collapsing identical sequences in order to reduce computational storage and processing requirements.

Traditionally, microbial marker gene data have been analyzed by using sequence clustering algorithms that group sequences into 'operational taxonomic units' (OTUs) for downstream analysis, usually by using a sequence similarity threshold. However, there are two major limitations to this OTU clustering method that are particularly problematic for the goals of the NEON program's long-term monitoring:

1) The generation of OTUs is not deterministic and is instead influenced by the range of sequence variation present in each analysis batch. This means that the OTU results for a given sample will be different depending on which other samples were also analyzed with it, and therefore OTUs cannot be compared between different analysis runs. Over the course of the planned 30-year lifetime of the NEON project, this is a major limitation that would prevent the ability to use marker genes to reproducibly analyze microbial community composition through time.

2) The clustering that occurs during the formation of OTUs can mask ecologically important variations in the microbes found in the samples. This can artificially under-represent the diversity of microbes that are identifiable using differences in marker gene sequences.

The marker gene sequence analysis pipeline detailed here addresses both of the limitations of OTU-based analysis by using amplicon sequence variants (ASVs) to represent the biological sequence variation instead. ASVs represent every unique sequence that is found in a dataset after attempting to identify and remove any artefacts of sample processing and sequencing from the dataset. In this sense, ASVs can be

thought of as a special case of OTUs, where the OTUs are generated using a 100% sequence similarity threshold. Because of how they are constructed, ASVs have neither of the limitations of typical OTUs:

1) ASV assignment is deterministic as long as the marker gene region (gene location and amplified fragment length) remains the same and the analysis methods and analysis parameters are identical. This means that ASVs can be directly compared (and identical ASVs will be found) among samples that may be collected years apart. Note: although ASV generation is deterministic, the method for chimeric sequence identification and removal, which is the step preceding ASV generation, is not fully deterministic. In order to gain the most power in identifying chimeras, the data from all samples in each processing run are pooled before chimera identification is done (see below for more detail); any chimeric sequences are then removed from all samples. The chimera removal method used (DADA2 within the QIIME2 version 2020.8.0 analysis pipeline) is widely used and does not provide an option for fully deterministic chimeric sequence removal. In practice, this has minor effects on both the ASVs identified and the number of sequences assigned to each ASV depending on which other samples are also present.

2) ASVs do not themselves mask any ecologically important variation that is reflected in the marker gene sequence because if two sequences only differ by one nucleotide, they will still be assigned to different ASVs.

To help enable as broad relevance of NEON data to the scientific research community as possible, the analysis pipeline described here uses the open source and increasingly commonly used QIIME2 analysis pipeline (with the embedded DADA2 program) to generate ASV results from NEON microbial marker gene sequencing data. The analysis parameters used, including the taxonomic reference database for taxonomy classification, are different between the 16S marker gene data and the ITS marker gene data, and are given in more detail in Section 5 below.

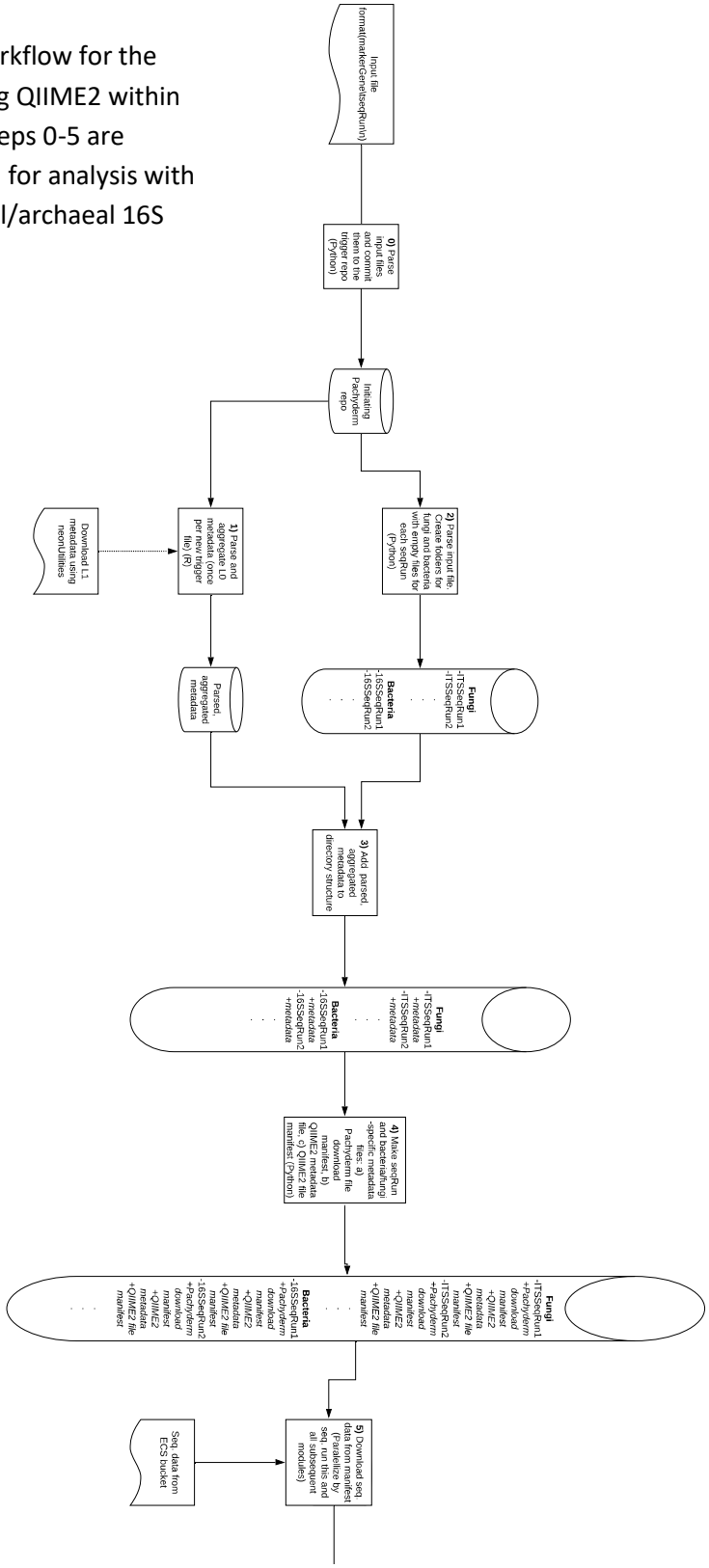
The QIIME2 pipeline (using QIIME2 version 2020.8.0) and its embedded sequence processing methods is used for all analysis from raw, de-multiplexed sequence data in paired fastq format, quality control, ASV generation, and taxonomic classification of ASVs. The ASV sequences, ASV names (MD5 hashes of the ASV sequence itself), the taxonomic classification of the ASVs, and the abundance of each ASV in each sample is reported in the NEON microbial community composition data products [AD07]. Other intermediate files or processing intermediates from QIIME2 are not reported in the data products.

5 ALGORITHM IMPLEMENTATION

The processing steps below are implemented within an automated pipeline using Pachyderm. Details of the Pachyderm pipeline implementation and source code can be found here: https://github.com/NEONScience/qiime2-pachyderm

The remainder of this document details the data analysis and processing steps done by QIIME2 within the Pachyderm pipeline.

Figure 1: Diagram (part 1 of 2) of processing workflow for the microbial community composition analysis using QIIME2 within the Pachyderm pipeline manager. Processing steps 0-5 are shown here. These steps are preparing the data for analysis with QIIME2, and are common between the bacterial/archaeal 16S rRNA and the fungal ITS rRNA analysis.



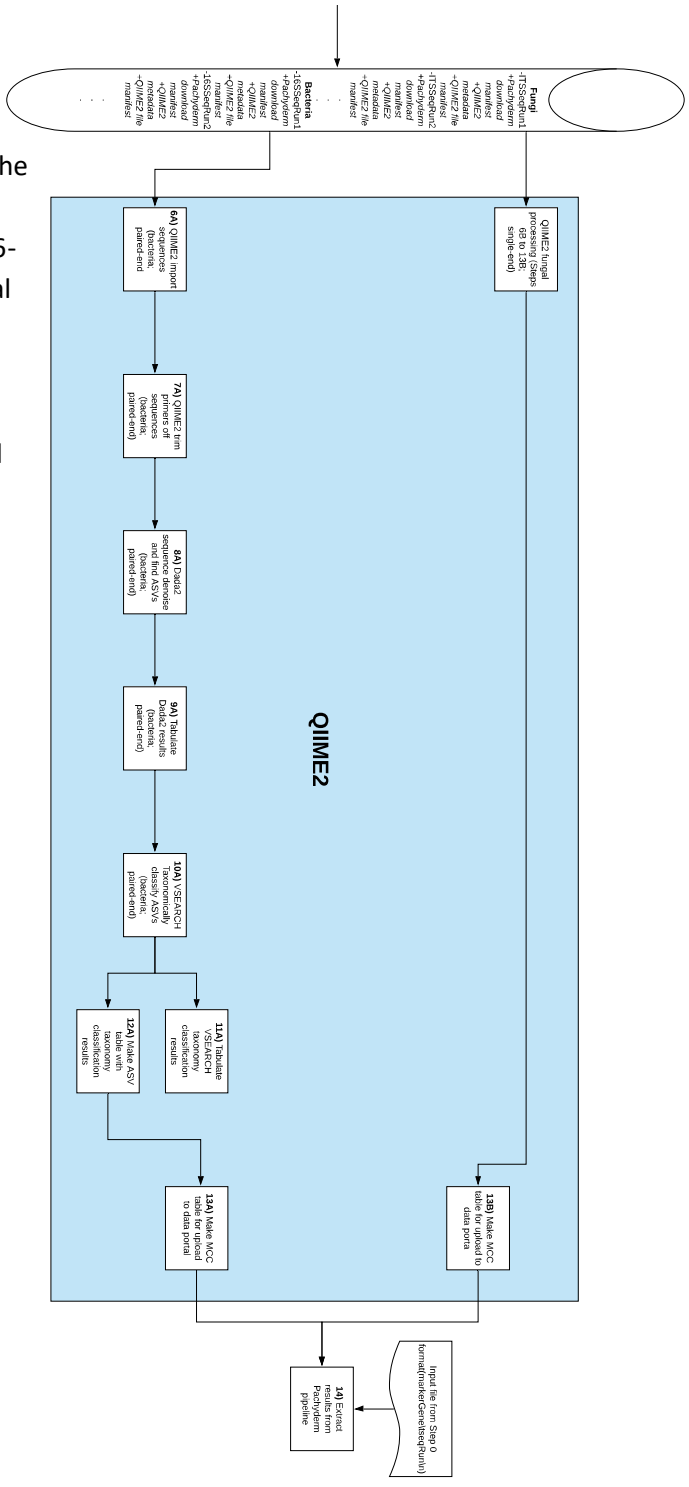


Figure 2: Diagram (part 2 of 2) of processing workflow for the microbial community composition analysis using QIIME2 within the Pachyderm pipeline manager. Processing steps 6-14 are shown here. Except for step 14 that extracts the final data product result from the Pachyderm pipeline, all steps take place within the QIIME2 analysis environment and are specific to either 16S or ITS analysis. Note: ITS analysis has the same number of processing steps as for 16S, but not all are shown.

DNA extraction data for each sample is first combined with the URL link to the files containing the raw sequence data to generate the information about sample metadata and raw sequence data files that QIIME2 requires. Note that raw sequence data is downloaded before QIIME2 is run.

The QIIME2 analysis pipeline uses the freely available Cutadapt, DADA2, and VSEARCH programs for ITS (single end, forward read only) and 16S (paired end, merged). Below are the details and the parameters used in each of these programs within QIIME2 to generate the microbial community composition data product.

Quality controlling raw sequences (Cutadapt)

ITS (Fungi)

Trimming the ITS marker gene data (Read 1 only) is done in two steps: 1) forward PCR primer removal, and then 2) reverse PCR primer removal (if present) and length filtering

1. Forward PCR primer removal
   Find and trim the 22 base long forward PCR primer from the start of the Read 1 sequences using a search that is anchored to the start of the read. The primer is 22 bases long. From spot-checks of raw sequences, the first 14bp of the primer sequence are much better conserved than the last 8bp. In addition, some sequences have a single base before the forward PCR primer. To set Cutadapt to remove up to 9bp from an expected 22 base PCR primer, the error-rate parameter is set to 0.41 (0.41 * 22 = 9.02; Cutadapt rounds down to allow 9 errors). Note that in this case, Cutadapt classifies any bases occurring before the PCR primer as an insertion in the PCR primer, so an anchored search with the exact PCR primer sequence is still possible. Any identified PCR primer sequence, within the range of error allowed, is trimmed from the sequence, and any sequences without an identifiable PCR primer are dropped from further analysis because they are likely sequencing artifacts.

   QIIME2 command used, with table of parameters:

   qiime cutadapt trim-single

Parameter name	Argument used	Default argument?
--p-cores	$NUM_THREADS <passed from Pachyderm configuration file>	No
--p-indels	True	Yes
--p-front	'^CTTGGTCATTTAGAGGAAGTAA'	No
--p-error-rate	0.41	No
--p-match-adapter-wildcards	True	Yes
--p-match-read-wildcards	False	Yes
--p-discard-untrimmed	True	No

2. Reverse PCR primer removal (if present) and length filtering
   Using the output from Step 1 above, now find and trim the reverse primer (if present) from the end of the sequences (note that the reverse primer sequence to find in the Read 1 sequences is the reverse complement of the PCR primer sequence used). Although in many cases the reverse primer should not be present in the R1 sequences due to the length of the PCR amplicon that these primers produce, for fungi with very short ITS1 sequences, the PCR amplicon can be sequenced through into the reverse primer sequence. The reverse primer is 20 bases long. Setting the error rate to be 0.4 allows for 8 mismatches with the primer sequence (20*0.4 = 8) to account for additional sequencing errors in the reverse reads. All sequences are kept regardless of whether the reverse primer was found, and only sequences with a minimum length of 185bp after all PCR primer removal are output.

   QIIME2 command used, with table of parameters:

   qiime cutadapt trim-single

Parameter name	Argument used	Default argument?
--p-cores	$NUM_THREADS <passed from Pachyderm configuration file>	No
--p-indels	True	Yes
--p-front	'GCATCGATGAAGAACGCAGC'	No
--p-error-rate	0.4	No
--p-minimum-length	185	No
--p-match-adapter-wildcards	True	Yes
--p-match-read-wildcards	False	Yes
--p-discard-untrimmed	False	YEs

16S (Archaea and Bacteria)

Trimming the 16S marker gene data (Read 1 and Read 2) is done in a single step for the paired reads. The PCR primers are represented with more accuracy in the 16S data, so the primer matching is more stringent (error rate of 0.1 allows 1 error over the 17bp Read 1 primer, and 2 allowed errors over the 22bp Read 2 primer).

QIIME2 command used, with table of parameters:

qiime cutadapt trim-paired --verbose

Parameter name	Argument used	Default argument?
--p-cores	$NUM_THREADS <passed from Pachyderm configuration file>	No
--p-indels	True	Yes
--p-error-rate	0.1	Yes

--p-front-f	'CCTACGGGNBGCASCAG'	No
--p-front-r	'GACTACNVGGGTATCTAATCC'	No
--p-match-adapter-wildcards	True	Yes
--p-match-read-wildcards	False	Yes
--p-discard-untrimmed	False	Yes

Using quality-controlled sequences to generate ASVs (DADA2)

After PCR primer removal, the remaining sequences are further trimmed and filtered, before being used to: 1) computationally identify and attempt to resolve sequencing errors, and 2) to generate ASVs, both using DADA2. As with the PCR primer removal step, this is done separately for the ITS sequence data and the 16S sequence data, although all sequences are truncated at the first instance (from left to right) of a sequencer-assigned quality score of less than or equal to 2. The remaining screening based on quality score uses maximum expected errors instead of average quality scores, because the expected error calculation correctly incorporates the fact that these quality scores are proportional to the base-10 logarithm of the calculated error rate.

ITS (Fungi)

To remove remaining low quality bases from the start of the sequences (bases that originally were located immediately after the primer sequence), remove 10bp from the start of each sequence, making each sequence passed to DADA2 at least 175bp. Because a minimum length was already enforced during the primer removal step (no similar minimum length functionality was available in QIIME2's implementation of DADA2 as of the pipeline construction), there is no other length-based screening done here (the parameter --p-trunc-len 0 means that no length truncation is done).

The DADA2 algorithm is trained on a subset of the data and is then used to model and attempt to correct for remaining sequencing errors in the reads. The overall quality of the ITS sequences is worse than the 16S sequences, but to provide as many moderate quality sequences to DADA2 as possible for error correction, we set the maximum number of expected errors allowed per sequence to eight. Across all NEON ITS sequencing runs considered (samples collected from 2015-2018), at least 70% of all sequences were at least 190 bp long and had fewer than eight expected errors (bright yellow in upper left-hand corner of Figure 1). All remaining sequences are pooled for chimera detection and removal to remove as many chimeras as possible.

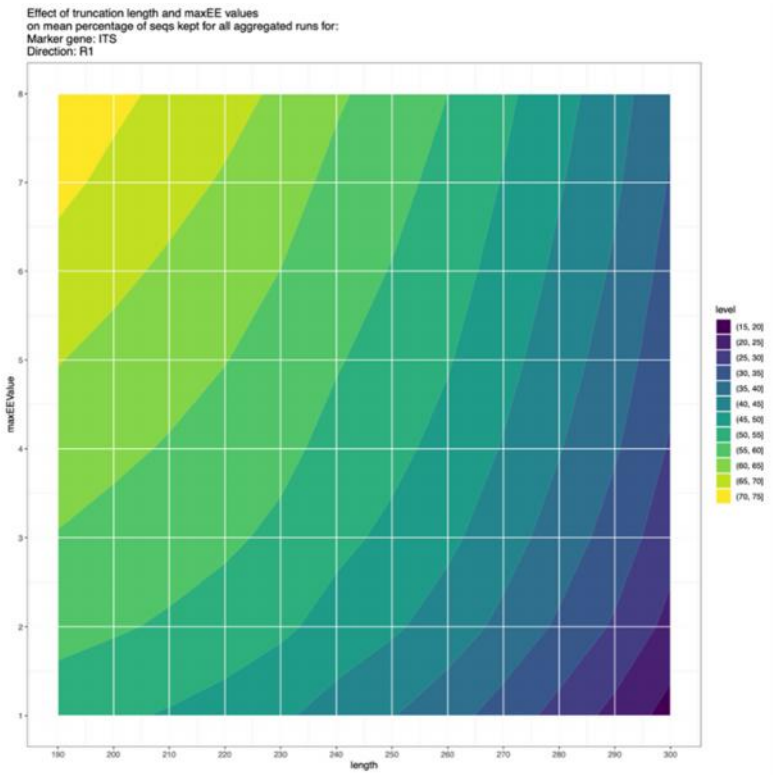


Figure 3: Percentage of sequences remaining (colors) for different combinations of sequence length allowed (x-axis) and maximum number of expected errors allowed (y-axis) for all ITS R1 sequences from samples collected from 2015-2018.

QIIME2 command used, with table of parameters:

qiime dada2 denoise-single

Parameter name	Argument used	Default argument?
--p-trunc-len	0	No
--p-trim-left	10	No
--p-max-ee	8	No
--p-trunc-q	2	Yes
--p-pooling-method	'independent'	Yes
--p-chimera-method	'pooled'	No
--p-min-fold-parent-over-abundance	1.0	Yes
--p-n-threads	$NUM_THREADS <passed from Pachyderm configuration file>	No
--p-n-reads-learn	1000000	Yes
--p-hashed-feature-ids	True	Yes

16S (Archaea and Bacteria)

In order to ensure consistent and reproducible read merging, the paired 16S reads were truncated to 265bp (forward read) and 200bp (reverse read); any reads shorter than this were dropped from further analysis. Then, to remove remaining low-quality bases from the start of the paired sequences (bases that originally were located immediately after each of the two primer sequences), 5bp were removed from the start of the forward reads, and 10bp were removed from the start of the reverse reads, leaving 260bp for the forward reads and 190bp for the reverse reads.

The DADA2 algorithm is trained on a subset of the data and is then used to model and attempt to correct for remaining sequencing errors in the reads. The overall quality of the 16S sequences is better than the ITS sequences. When setting the maximum number of expected errors to allow in the reads used as input for DADA2, it was important to ensure roughly equal numbers of forward and reverse reads were kept to allow efficient read merging; forward reads were permitted to have up to five expected errors, and reverse reads up to eight (Figure 2, note different color scales between the forward reads and the reverse reads).

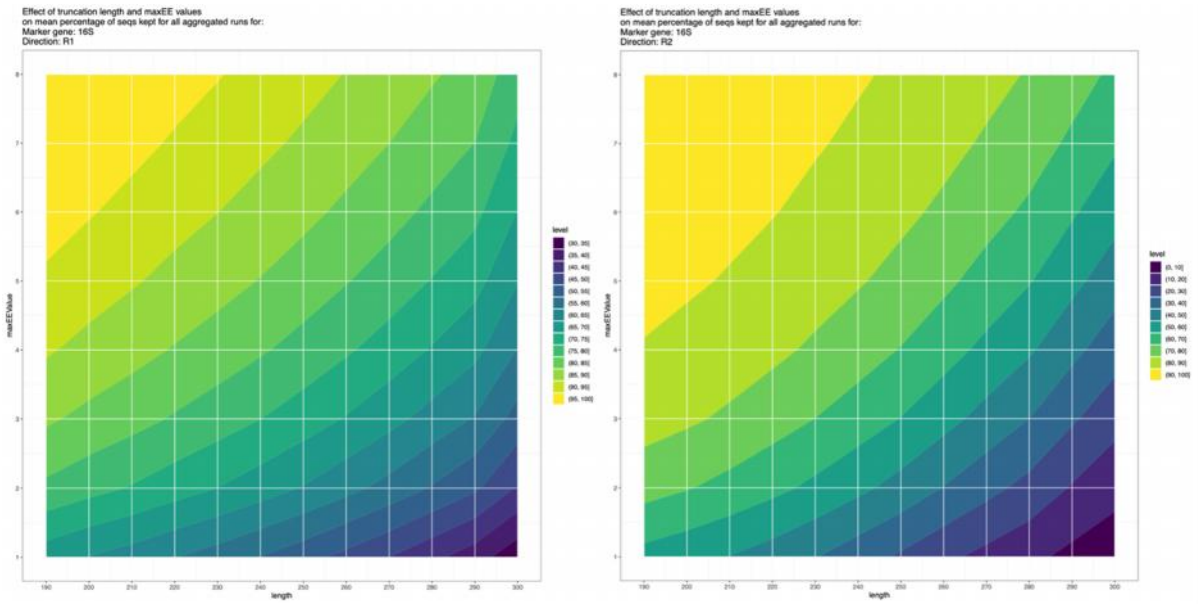


Figure 4: Percentage of sequences remaining (colors) for different combinations of sequence length allowed (x-axis) and maximum number of expected errors allowed (y-axis) for all 16S R1 sequences (left) and R2 sequences (right) from samples collected from 2015-2018 (Note: the color scale values differ between the two panels).

QIIME2 command used, with table of parameters:

qiime dada2 denoise-paired

Parameter name	Argument used	Default argument?
--p-trunc-len-f	265	No
--p-trunc-len-r	200	No
--p-trim-left-f	5	No
--p-trim-left-r	10	No
--p-max-ee-f	8	No
--p-max-ee-r	5	No
--p-trunc-q	2	Yes
--p-pooling-method	'independent'	Yes
--p-chimera-method	'pooled'	No
--p-min-fold-parent-over-abundance	1.0	Yes
--p-n-threads	$NUM_THREADS <passed from Pachyderm configuration file>	No
--p-n-reads-learn	1000000	Yes
--p-hashed-feature-ids	True	Yes

Assigning taxonomic classifications to ASVs (VSEARCH)

At this stage, DADA2 has generated ASVs, each with the following information: 1) the unique sequence represented by that ASV, 2) the MD5 hash of the ASV sequence, which is used as the name of the ASV, and 3) the number of times that ASV sequence is represented in the input data. The final analysis step is to assign taxonomic information to each ASV by comparing the ASVs to a reference database of sequences with known taxonomic assignments. The VSEARCH taxonomy classification module in QIIME2 was used to classify both 16S and ITS sequences, but different reference databases were used for each sequence type.

ITS (Fungi)

Steps to make UNITE database for use with QIIME2:

1) Downloaded version 8.2 of the UNITE database (released 2020-02-20) that only includes singletons that are set as RefS (manually curated). It does not include the global and 97% singletons. This version of the database is DOI: 10.15156/BIO/786385. Web link: https://plutof.ut.ee/#/doi/10.15156/BIO/786385

2) Following advice from QIIME2 (https://docs.qiime2.org/2020.8/tutorials/feature-classifier/#classification-of-fungal-its-sequences), the non-trimmed and non-ITS localized database files in the "developer" directory were selected, and the version of the database using "dynamic"

thresholding of taxa that were curated manually (instead of using blanket 97% or 99% sequence similarity thresholds).

3) The FASTA file containing the reference database sequences contains lowercase base characters (i.e. ‘a’, ‘c’, ’t’, ’g’) and some trailing blank spaces that prevent immediate importing into QIIME2. Both issues were fixed at the same time:

```
awk '/^>/ {print($0)}; /^[^>]/ {print(toupper($0))}'
~/sh_qiime_release_04.02.2020/developer/sh_refs_qiime_ver8_dynamic_04.02.2020_dev.fasta
| tr -d ' ' >
~/sh_qiime_release_04.02.2020/developer/sh_refs_qiime_ver8_dynamic_04.02.2020_dev_upp
ercase.fasta
```

4) The resulting FASTA file with uppercase bases then imported into QIIME2, as well as the corresponding taxonomic classification data:

```
qiime tools import \
  --type 'FeatureData[Sequence]' \
  --input-path
~/sh_qiime_release_04.02.2020/developer/sh_refs_qiime_ver8_dynamic_04.02.2020_dev_upp
ercase.fasta \
  --output-path UNITE_v8-2_DOI_786385-RefSSingleton_dev_dynamic_ITS_db.qza

qiime tools import \
  --type 'FeatureData[Taxonomy]' \
  --input-format HeaderlessTSVTaxonomyFormat \
  --input-path
~/sh_qiime_release_04.02.2020/developer/sh_taxonomy_qiime_ver8_dynamic_04.02.2020_de
v.txt \
  --output-path UNITE_v8-2_DOI_786385-RefSSingleton_dev_dynamic_ITS_tax.qza
```

5) The taxonomic classification of the ASVs was then done with the VSEARCH classification plugin in QIIME2. All parameters used are default values but are specified explicitly to better record data processing settings

QIIME2 command used, with table of parameters:

qiime feature-classifier classify-consensus-vsearch

Parameter name	Argument used	Default argument?
--p-maxaccepts	10	Yes
--p-perc-identity	0.8	Yes
--p-query-cov	0.8	Yes
--p-strand	'both'	Yes
--p-min-consensus	0.51	Yes

--p-unassignable-label	'Unassigned'	Yes
--p-search-exact	False	Yes
--p-top-hits-only	False	Yes
--p-maxhits	'all'	Yes
--p-maxrejects	'all'	Yes
--p-output-no-hits	True	Yes
--p-weak-id	0.0	Yes
--p-n-threads	$NUM_THREADS <passed from Pachyderm configuration file>	No

16S (Archaea and Bacteria)

Steps to make SILVA database for use with QIIME2:

1) Downloaded version 132 of the SILVA database (released 2018-04-10). Web link: https://www.arb-silva.de/fileadmin/silva_databases/qiime/Silva_132_release.zip

2) The FASTA file of reference sequences was then imported into QIIME2, as well as the corresponding taxonomic classification data

```
qiime tools import \
  --type 'FeatureData[Sequence]' \
  --input-path ~/SILVA_132_QIIME_release/rep_set/rep_set_16S_only/99/silva_132_99_16S.fna
\
  --output-path silva_132_99_16S_otus.qza


qiime tools import \
  --type 'FeatureData[Taxonomy]' \
  --input-format HeaderlessTSVTaxonomyFormat \
  --input-path
~/SILVA_132_QIIME_release/taxonomy/16S_only/99/majority_taxonomy_7_levels.txt \
  --output-path silva_132_99_16S_7LevelTaxon_ref-taxonomy.qza
```

3) The sequences were trimmed to the expected amplicon region based on the 16S PCR primers that were used to create a custom classifier, following the methods outlined in the QIIME2 tutorial here: https://docs.qiime2.org/2020.2/tutorials/feature-classifier/

The min-length and max-length thresholds used are very lenient around the expected value of roughly 450bp, so the range covers 350bp on each side of the expected amplicon length.

QIIME2 command used, with table of parameters:

qiime feature-classifier extract-reads

Parameter name	Argument used	Default argument?
--p-f-primer	CCTAYGGGRBGCASCAG	No
--p-r-primer	GGACTACHVGGGTWTCTAAT	No
--p-min-length	100	No
--p-max-length	800	No

4) The custom database was then used for the taxonomic classification of the ASVs with the VSEARCH classification plugin in QIIME2. All parameters used are default values but are specified explicitly to better record data processing settings

QIIME2 command used, with table of parameters:

qiime feature-classifier classify-consensus-vsearch

Parameter name	Argument used	Default argument?
--p-maxaccepts	10	Yes
--p-perc-identity	0.8	Yes
--p-query-cov	0.8	Yes
--p-strand	'both'	Yes
--p-min-consensus	0.51	Yes
--p-unassignable-label	'Unassigned'	Yes
--p-search-exact	False	Yes
--p-top-hits-only	False	Yes
--p-maxhits	'all'	Yes
--p-maxrejects	'all'	Yes
--p-output-no-hits	True	Yes
--p-weak-id	0.0	Yes
--p-n-threads	$NUM_THREADS <passed from Pachyderm configuration file>	No

Adding taxonomic assignments from VSEARCH to the DADA2 feature table

The feature table from DADA2 (counts of the number of times each ASV is encountered in each sample) then needs to be merged with the taxonomic identifications of each ASV from VSEARCH to give a feature table with taxonomic identifications of the ASVs present in each sample and the number of times each ASV was encountered. To do this, the taxonomic assignments are converted to tab separated value files, and the feature table is converted into a BIOM-formatted file. The taxonomic assignments are then added to the feature table BIOM file, and the resulting feature table with taxonomic assignments is finally converted from BIOM format into a tab-separated value (tsv) file for downstream merging with

other information to make the microbial community composition results table for ingest. The steps below for the conversion of ITS or 16S sequence data are the same – only the file names differ.

ITS (Fungi)

```
# Outputs taxonomy.tsv
qiime tools export --input-path "${inputDirStem}/${seqRun}/QIIME2_ITS_single-end_dada2_repSeqs_VSEARCH_taxonAssignOut.qza" --output-path "$pathForOutputs"

# The header line for taxonomy.tsv is 'Feature ID   Taxon   Consensus' and it needs to be changed to: '#OTUID   taxonomy   confidence'.
# Do this by running
sed -i "1s/.*/#OTUID\ttaxonomy\tconfidence/" "${pathForOutputs}/taxonomy.tsv"

# Outputs feature-table.biom
qiime tools export --input-path "${inputDirStem}/${seqRun}/QIIME2_ITS_single-end_dada2_table.qza" --output-path "$pathForOutputs"

# convert the feature-table.biom (without the taxonomy added) into a tsv file
biom convert -i "${pathForOutputs}/feature-table.biom" -o "${pathForOutputs}/feature-table.tsv" --to-tsv

# Add the taxonomy data ';' separated (what -sc-separated stands for) to the feature table
biom add-metadata -i "${pathForOutputs}/feature-table.biom" -o "${pathForOutputs}/feature-table-with-taxonomy.biom" --observation-metadata-fp "${pathForOutputs}/taxonomy.tsv" --sc-separated taxonomy
# Convert the feature table with the taxonomy added to be a .tsv file
biom convert -i "${pathForOutputs}/feature-table-with-taxonomy.biom" -o "${pathForOutputs}/feature-table-with-taxonomy.tsv" --to-tsv --header-key taxonomy
```

16S (Archaea and Bacteria)

```
# Outputs taxonomy.tsv
qiime tools export --input-path "${inputDirStem}/${seqRun}/QIIME2_16S_paired-end_dada2_repSeqs_VSEARCH_taxonAssignOut.qza" --output-path "$pathForOutputs"

# The header line for taxonomy.tsv is 'Feature ID   Taxon   Consensus' and it needs to be changed to: '#OTUID   taxonomy   confidence'.
# Do this by running
sed -i '1s/.*/#OTUID\ttaxonomy\tconfidence/' "${pathForOutputs}/taxonomy.tsv"

# Outputs feature-table.biom
qiime tools export --input-path "${inputDirStem}/${seqRun}/QIIME2_16S_paired-end_dada2_table.qza" --output-path "$pathForOutputs"
```

```
# convert the feature-table.biom (without the taxonomy added) into a tsv file
biom convert -i "${pathForOutputs}/feature-table.biom" -o "${pathForOutputs}/feature-table.tsv" --to-tsv

# Add the taxonomy data ';' separated (what -sc-separated stands for) to the feature table
biom add-metadata -i "${pathForOutputs}/feature-table.biom" -o "${pathForOutputs}/feature-table-with-taxonomy.biom" --observation-metadata-fp "${pathForOutputs}/taxonomy.tsv" --sc-separated taxonomy
# Conver the feature table with the taxonomy added to be a .tsv file
biom convert -i "${pathForOutputs}/feature-table-with-taxonomy.biom" -o "${pathForOutputs}/feature-table-with-taxonomy.tsv" --to-tsv --header-key taxonomy
```

After the feature table with the taxonomic assignments is constructed, it is combined with information from the sequencing metadata file, the DNA extraction table, the DNA sequence of each ASV, and the number of sequences represented by each ASV (obtained from the number of non-chimeric sequences provided as input to DADA2) to provide all the information required for the microbial community composition data product ingest. As part of this process, taxonomic assignments are standardized and formatted for NEON ingest using the following rules:

Taxonomic classifications are made to the lowest possible level; any ASVs that are not able to be classified using the reference database are listed as 'Unclassified' at the Domain level.

For the generation of taxonomic names in the 'scientificName' column of the ingest table, if the ASV is identifiable to the species level, then the bionomial name (genus + species) is represented in the 'scientificName' field; If the ASV is only identifiable to a taxonomic level above species, then the entry in the 'scientificName' field is the lowest classified taxonomic level plus 'sp.'. For example, a fungal ASV that is classifiable to the genus *Russula* will be represented by 'Russula sp.'; a bacterial ASV that is classifiable only to the Kingdom level will be 'Bacteria sp.'; Unclassifiable samples will be 'Unclassified sp.'.

When these processing and analysis steps are run within the Pachyderm pipeline manager, the final step is to extract the most useful outputs of the pipeline from the output directories of the different pipeline processing steps (managed through Pachyderm's pfs file system), and to aggregate them all in one directory per unique combination of marker gene type and sequence run ID that is outside of the pfs file system.

QA/QC Procedure:
There are not additional QA/QC tests run on the data other than those mentioned above to trim sequences using Cutadapt, and to computationally identify and attempt to resolve sequencing errors using DADA2.

6 UNCERTAINTY

Although for simplicity, the taxonomic attributions given in the community composition data product represent the most confidently assigned taxonomic attribution for each ASV as determined by the VSEARCH taxonomic attribution algorithm using the specified reference databases, these attributions have varying levels of confidence. These confidence estimates are generated by VSEARCH as part of this data analysis pipeline but are not reported in the final data product. The accuracy of confidence estimates will vary with different taxonomic groups and with different versions of the reference database, and therefore are of limited direct value in interpreting taxonomic results. However, the data user should remain aware of this taxonomic uncertainty that is inherent in any taxonomic classification method. As part of this data product, we provide the full sequence for each ASV identified in the processed NEON samples, and encourage data users to run their own independent taxonomic classification using a relevant reference database in order to confirm or refine the taxonomic attributions of ASVs as necessary. Data users should also remain aware of additional uncertainty that is introduced during the sequence de-noising step with DADA2, although this uncertainty is not easily quantifiable, and is not considered when taxonomic attributions are generated.

7 FUTURE PLANS AND MODIFICATIONS

The taxonomic databases used for producing taxonomic assignments of ASVs may change in the future as additional biological sequence variation is recorded in reference databases. This is expected to allow more detailed taxonomic assignments to be possible, but may not retain backward compatibility with past results analyzed with different databases. In addition, the analysis platform (currently QIIME2) or the underlying analysis programs (currently Cutadapt, DADA2, and VSEARCH) may change in the future as other methods are introduced that may provide substantial analytical benefits. In these cases, there would also not be backward compatibility with past results. If future analytical changes are made that result in a fundamental change to the data, a new data product revision will be made and users will be notified of the change.

8 BIBLIOGRAPHY

Bolyen E, Rideout JR, Dillon MR, Bokulich NA, Abnet CC, Al-Ghalith GA, Alexander H, Alm EJ, Arumugam M, Asnicar F, Bai Y. 2019. Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nature Biotechnology* 37, 852–857. https://doi.org/10.1038/s41587-019-0209-9

Callahan BJ, McMurdie PJ, Rosen MJ, Han AW, Johnson AJ, Holmes SP. 2016. DADA2: High-resolution sample inference from Illumina amplicon data. *Nature Methods* 13(7):581-583. doi:10.1038/nmeth.3869

Rognes T, Flouri T, Nichols B, Quince C, Mahé F. 2016. VSEARCH: a versatile open source tool for metagenomics. *PeerJ* 4, e2584