METHOD ARTICLE

The National Ecological Observatory Network's soil metagenomes: assembly and basic analysis [version 1; peer review: 2 approved with reservations]

Zoey R. Werbin ¹, Briana Hackos², Michael C. Dietze³, Jennifer M. Bhatnagar¹

¹Department of Biology, Boston University, Boston, MA, 02215, USA

²Department of Mathematics, University of Colorado, Boulder, Boulder, CO, 80309, USA

³Department of Earth & Environment, Boston University, Boston, MA, 02215, USA

V1 First published: 19 Apr 2021, 10:299
<https://doi.org/10.12688/f1000research.51494.1>

Latest published: 23 Mar 2022, 10:299
<https://doi.org/10.12688/f1000research.51494.2>

Abstract

The National Ecological Observatory Network (NEON) annually performs shotgun metagenomic sequencing to sample genes within soils at 47 sites across the United States. NEON serves as a valuable educational resource, thanks to its open data policies and programming tutorials, but there is currently no introductory tutorial for performing analyses with the soil shotgun metagenomic dataset. Here, we describe a workflow for processing raw soil metagenome sequencing reads using the Sunbeam bioinformatics pipeline. The workflow includes cleaning and processing raw reads, taxonomic classification, assembly into contigs, annotation of predicted genes using custom protein databases, and exporting assemblies to the KBase platform for downstream analysis. This workflow is designed to be robust to annual data releases from NEON, and the underlying Snakemake framework can manage complex software dependencies. The workflow presented here aims to increase the accessibility of NEON's shotgun metagenome data, which can provide important clues about soil microbial communities and their ecological roles.

Keywords

metagenomics, microbial ecology, soil microbiome, tutorial, workflow

This article is included in the **Bioinformatics** gateway.

Open Peer Review

Approval Status ? ? ✓

	1	2	3
version 2			
(revision)			✓
23 Mar 2022			view
version 1	?	?	
19 Apr 2021	view	view	
<ol style="list-style-type: none"> 1. Naupaka Zimmerman , University of San Francisco, San Francisco, USA 2. William Nelson , Pacific Northwest National Laboratory, Richland, USA 3. Olubukola Oluranti Babalola , Pacific Northwest National Laboratory, Richland, USA 			

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the **Ecology and Global Change** gateway.

Corresponding author: Zoey R. Werbin (zrwerbin@bu.edu)

Author roles: **Werbin ZR:** Conceptualization, Data Curation, Methodology, Software, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Hackos B:** Data Curation, Methodology, Software, Writing – Original Draft Preparation; **Dietze MC:** Resources, Supervision, Writing – Review & Editing; **Bhatnagar JM:** Formal Analysis, Funding Acquisition, Methodology, Project Administration, Resources, Supervision, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: ZRW is funded by the National Science Foundation (NSF) Graduate Research Fellowship Program (Award #1840990). ZRW, MCD, and JMB are funded by the NSF Macrosystems Biology Program (Award# 1638577). BH is funded by the BU Bioinformatics Research and Interdisciplinary Training Experience (BRITE) NSF-REU program (Award #1949968).

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2021 Werbin ZR *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Werbin ZR, Hackos B, Dietze MC and Bhatnagar JM. **The National Ecological Observatory Network's soil metagenomes: assembly and basic analysis [version 1; peer review: 2 approved with reservations]** F1000Research 2021, 10:299 <https://doi.org/10.12688/f1000research.51494.1>

First published: 19 Apr 2021, 10:299 <https://doi.org/10.12688/f1000research.51494.1>

Introduction

The soil microbiome is responsible for key ecological processes, such as decomposition and nitrogen cycling (Allison *et al.* 2013). One powerful tool for studying the soil microbiome is shotgun metagenomic sequencing, in which all of the genetic material within the DNA extract of a soil sample is sequenced at once, without targeting specific organisms (Quince *et al.* 2017, Pérez-Cobas *et al.* 2020). The largest publicly available sequencing dataset of this type is updated annually by the National Ecological Observatory Network (NEON), which monitors ecological conditions at 47 terrestrial sites spanning 20 ecoclimatic domains across the US and its territories (Keller *et al.* 2018). NEON is funded by the National Science Foundation (NSF), and collects soil samples and releases shotgun metagenomics data annually.

To date, the NEON soil metagenomics data can only be accessed in two formats: as completely raw reads released by NEON, or as processed files through the default protocols of the MG-RAST storage server. Neither format is suitable for most metagenomic analyses, which generally answer scientific questions using custom data processing pipelines that use specific algorithms and targeted reference databases (Ladoukakis *et al.* 2014; Quince *et al.* 2017). To facilitate future scientific analysis, we present a workflow for taking raw sequences and generating a processed dataset that can be linked to other NEON data products, which include soil biogeochemistry, root measurements, or aboveground plant communities.

NEON data is a valuable resource for ecology and bioinformatics, thanks to its open access software, robust documentation, and educational resources (Jones 2020). The pipeline that we present here is designed to complement existing NEON educational resources, such that users without prior bioinformatics experience may use this dataset to learn about microbial communities within the soil. We present code and explanations for each analysis step, including basic quality control (QC), assembling reads into larger genome fragments (“contig” assembly), predicting genes, quantifying gene counts for specific ecological or biogeochemical functions, and exporting to the KBase platform (Arkin *et al.* 2018). We recommend the review by Pérez-Cobas *et al.* (2020) for an overview of software alternatives for each step of this shotgun metagenomics analysis.

Methods

Dataset description

Soil samples are collected annually from 47 NEON sites during peak greenness. Three samples are collected within a NEON plot at a sampling time point. Soil samples are collected up to 30cm below the soil surface, the organic (O) and the mineral (M) horizons (when present) are separated, and subsamples from each horizon are homogenized into one composite sample per horizon. Sample file names include the 4-letter site identifiers, horizons (O or M), and sampling date. Samples are frozen on dry ice until DNA extraction and preparation using the KAPA Hyper Plus kit (Kapa Biosystems). Samples from multiple sites are pooled into sets of 40 or 60 for sequencing, which is conducted on an Illumina NextSeq at the Battelle Memorial Institute (NEON Metagenomics Standard Operating Procedure, v.3). Since there is currently no versioned release of NEON’s metagenomic data, the pipeline described here is designed to be robust to processing new data as it is released from NEON, approximately annually (TOS Science Design for Terrestrial Microbial Diversity, NEON.DOC.000908).

Operation

We assume a Linux operating system and command-line interface. Storage and RAM requirements will depend on the specific analyses performed and the number of samples analyzed. If using shared computing clusters, refer to the Sunbeam manual for [cluster-specific options](#), which are necessary to take full advantage of multi-core processing.

Implementation

Once sequences are downloaded, we use the software [Sunbeam](#) (Clarke *et al.* 2019) to create a bioinformatic pipeline. Sunbeam links a variety of popular bioinformatics tools (e.g. BLAST, MegaHIT, Kraken2, Prodigal), and users can develop and share customized extensions for various purposes. Sunbeam, and its underlying Snakemake framework (Köster *et al.* 2012), are designed to address common problems with software versioning and updating, as well as efficient data re-analysis (i.e. running the minimal tasks necessary to generate updated output files). In addition to Sunbeam’s default steps for cleaning and processing the raw reads, the pipeline below performs taxonomic classification or protein annotation for predicted genes using custom databases.

1. Setup

1.1 Get raw sequence files

1.1a Test sample set [recommended option]: We recommend an initial interactive test of the pipeline with two microbial samples. This will ensure that all necessary software is installed and that file paths are correct. A sample set can be downloaded using the command below:

```
mkdir raw_sequences # create directory for raw sequences

cd raw_sequences # enter directory

wget https://neon-microbial-raw-seq-files.s3.data.neonscience.org/2017/WOOD_002-M-20140925-comp_R1.fastq.gz

wget https://neon-microbial-raw-seq-files.s3.data.neonscience.org/2017/WOOD_002-M-20140925-comp_R2.fastq.gz

wget https://neon-microbial-raw-seq-files.s3.data.neonscience.org/2017/SCBI_012-M-20140915-comp_R1.fastq.gz

wget https://neon-microbial-raw-seq-files.s3.data.neonscience.org/2017/SCBI_012-M-20140915-comp_R2.fastq.gz

cd ../ # return to enclosing directory
```

1.1b Download custom dataset: Use NEON's interactive [Data Portal](#), or to download a specific set of samples that meets your interests. Download links are included in NEON's "Expanded" data packages. For example, you could compare samples from Alaska with those from Puerto Rico, or you could download sites that have accompanying multi-decadal data from the [Long-Term Ecological Research](#) (LTER) program. Samples must have forward and reverse reads and they must be compressed (in.fastq.gz format). Even when compressed, each file may still require multiple GB of storage.

1.2 Install Sunbeam

Full details on Sunbeam installation can be found in the Sunbeam [user guide](#). In short, run the following commands to create a new "analysis" directory and download Sunbeam into that directory:

```
mkdir metagenome_analysis # create directory for analysis

cd metagenome_analysis # enter directory

git clone -b dev https://github.com/sunbeam-labs/sunbeam sunbeam # download development branch

cd sunbeam # enter directory

bash install.sh # run installation script
```

Confirm success of installation (may take 10-15 minutes):

```
bash tests/run_tests.bash
```

If all went well, your screen will say "TESTS SUCCEEDED." A new conda environment should now exist. You can check available environments using:

```
conda env list
```

Activate the Sunbeam environment. This must be run for any Sunbeam commands to work.

```
conda activate sunbeam
```

Next, we tell Sunbeam where the raw sequences are downloaded, by creating a “samples.csv” file that links the forward read files and the reverse read files. If you have not downloaded files to a “raw_sequences” folder (Step 1.1A), change the file path to point to the sequence folder on your own system:

```
cd .. # go to enclosing (analysis) directory

sunbeam list_samples ../raw_sequences >> samples.csv # change this path if your
own raw files are not in "raw_sequences"
```

The last part of setup requires creating a configuration file called “sunbeam_config.yml.” To use the custom configuration that accompanies this workflow run the following command from your analysis directory:

```
wget https://raw.githubusercontent.com/zoey-rw/metagenomes_NEON/main/sunbeam_config.
yml # download configuration file
```

This configuration file is used to set parameters for every part of the analysis (Figure 1).

1.3 Setup troubleshooting and tips

On shared computing clusters, some softwares must be loaded as “modules” before they are used. For instance, to use Miniconda (necessary for every step of this pipeline), this command may work:

```
module load miniconda # may need to specify version
```

Most analyses will run quicker if there are multiple threads available. The custom configuration file, sunbeam_config.yml, assumes you have 8 threads available. This command can check your available threads, though you may not want to use all of them if you share computing resources:

```
echo "CPU threads: $(grep -c processor/proc/cpuinfo) "
```

2. Quality control

In this step, raw sequences are cleaned using the default tools in the Sunbeam pipeline. To remove poor-quality data, or components that are leftover from the sequencing, we use Cutadapt (Martin 2015) and Trimmomatic (Bolger *et al.* 2014). Problematic low-complexity samples are removed using the program Komplexity (Clarke *et al.* 2019). Overall quality of reads is then reported by FastQC (BabrahamBioinformatics, 2018).

Optionally, users may wish to search for and remove sequences that match the PhiX genome (Step 2.1b), which is a common contaminant of Illumina metagenomic data due to its use as a control during sequencing (Mukherjee *et al.* 2015). This contamination was not found in our test samples (Figure 2c), so we proceed without this in Step 2.1a.

2.1a Run quality control without PhiX decontamination [recommended]: To run the quality control step without decontaminating the files, use the following command:

```
sunbeam run -- --configfile./sunbeam_config.yml clean_qc
```

Note: the below command does the same as the above, but produces intermediate outputs for each software (Cutadapt, Trimmomatic, and fastQC). This takes up additional file storage space, but allows you to inspect each output. This is useful for debugging, such as if you suspect that one of these steps is removing more reads than it should.

```
sunbeam run -- --configfile sunbeam_config.yml all_qc
```

```

# General options
all:
  root: /projects/zrwerbin/manuscript_test/metagenome_analysis/
  output_fp: sunbeam_output
  samplelist_fp: samples.csv
  paired_end: true
  download_reads: false
  version: 2.1.1+dev81.gd0e29cd

# Quality control
qc:
  suffix: qc
  # Sample intake
  seq_id_ending: ''
  # Trimmomatic
  threads: 8
  java_heapsize: 512M
  leading: 3
  trailing: 3
  slidingwindow: [4, 15]
  minlen: 36
  adapter_template: $CONDA_PREFIX/share/trimmomatic/adapters/NexteraPE-PE.fa
  # Cutadapt
  fwd_adapters: [CTGAAGCT, TAATGCGC, GTTTCAGTCACGATC]
  rev_adapters: [ACGTCCGT, GTCAGTAC, GTTTCAGTCACGATC]
  # Komplexity
  kz_threshold: 0.55
  # Decontam.py
  pct_id: 0.5
  frac: 0.6
  host_fp: ''

# Taxonomic classifications
classify:
  suffix: classify
  threads: 8
  kraken_db_fp: 'kraken_pluspf'

# Contig assembly
assembly:
  suffix: assembly
  min_length: 1000
  threads: 8

# Contig annotation
annotation:
  suffix: annotation
  min_contig_len: 1000
  circular_kmin: 10
  circular_kmax: 1000
  circular_min_len: 3500

blast:
  threads: 8

blastdbs:
  root_fp: ''
  protein:
  ncyc: 'db/NCyc/NCyc_unique.fasta'
  card: 'db/card/protein_fasta_protein_homolog_model.fasta'

mapping:
  suffix: mapping
  genomes_fp: ''
  samtools_opts: ''
  threads: 8
download:
  suffix: download
  threads: 8

sbx_metaquast:
  threads: 8

sbx_binning:
  threads: 8
  min_contig_length: 1500

maxbin:
  markerset: 107

```

Change to your analysis folder, where the "sunbeam_output" directory will be created (Step 1.2).

Change all 'threads' arguments based on computational power (Step 1.3).

Specifies location of Kraken2 database (Step 3.1).

Specifies location of BLASTp databases (Step 5.1).

These parameters are ignored in our workflow.

Figure 1. Sunbeam configuration file provided for NEON shotgun metagenomics bioinformatic pipeline. Many parameters remain the default values provided in Sunbeam's basic configuration file, while others have been customized for this dataset (e.g. file paths, as well as fwd_adapter, rev_adapter, min_length).

Figure 2. Quality-control reports produced by the *sbx_report* extension, described in Step 3.2A. a) Average quality scores along read positions. b) Counts of read pairs for a subset of samples. c) Proportion of reads retained (blue), discarded as low-quality (light grey), or discarded as PhiX ("Host") contamination (dark grey). No PhiX contamination was observed in the metagenomes from these 2 NEON soil samples.

2.1b Run quality control with PhiX decontamination: To download the PhiX genome, run the following command, which will retrieve the genome from the [Illumina iGenomes](https://www.illumina.com) website, decompress the file, and rename it as a FASTA file within your current directory:

```
wget http://igenomes.illumina.com.s3-website-us-east-1.amazonaws.com/PhiX/Illumina/RTA/PhiX_Illumina_RTA.tar.gz -O - | tar -xz PhiX/Illumina/RTA/Sequence/WholeGenomeFasta/genome.fa

mv PhiX/Illumina/RTA/Sequence/WholeGenomeFasta/genome.fa PhiX/PhiX.fasta
```

In your configuration file, the "host_fp" parameter must point to the folder enclosing the downloaded PhiX genome. The command below will make this change:

```
sed -i "s/host_fp: \"/host_fp: 'PhiX'/" sunbeam_config.yml
```

Next, run the Sunbeam decontamination step, which automatically includes quality control:

```
sunbeam run -- --configfile sunbeam_config.yml all_decontam
```

2.2 Evaluate quality control

Output folders contain log files for each software run within the quality control step. Each sample also has an HTML file produced by FastQC (BabrahamBioinformatics, 2018), which includes visualizations of base quality, sequence lengths, and other checks. More information on interpreting these reports is available on the [FastQC website](https://www.fastqc.org/). By default, the samples with reads that pass quality control will be located in the following directory: `sunbeam_output/qc/decontam/`.

Within our example dataset, average quality scores were high (above 30) throughout sequence reads (Figure 2a). Quality scores represent error rates of base calls (Illumina, 2014). On average, the first few reads tended to be of lowest quality, but otherwise, quality decreases along read length. Quantity of sequences can vary dramatically between samples, with read pair counts ranging from 2 million to 15 million (Figure 2b). This does not necessarily reflect variation in the amount of microbes in the soil - rather, variation can be the result of biases in DNA extraction or sequencing (Pereira *et al.* 2018; Jonsson *et al.* 2016).

3. Taxonomic classification

The taxonomic identity of reads in a metagenome sample can be assigned by comparing predicted proteins or nucleotides to reference databases. This can be performed with short reads (pre-assembly) or with assembled contigs. Both avenues produce similar results for fungal and bacterial sequences (Pearman *et al.* 2020), so we use short reads for compatibility with Sunbeam's default classifier, Kraken2 (Wood *et al.* 2019). Compared to other classification tools, Kraken2 has been shown to perform favorably on soil datasets (Kalantar *et al.* 2020). However, Sunbeam extensions have also been developed for other classifiers, such as Kaiju or MetaPhlAn.

3.1 Classify reads using Kraken2

First, we must download a Kraken2 reference database. You could build your own with specific combinations of organisms, but pre-built databases are updated regularly and shared by the Kraken2 developers. Databases range in size from 100 MB to 90 GB, depending on the genomes included. Most databases are constructed via RefSeq (O'Leary *et al.* 2016), but marker gene databases such as Silva (Quast *et al.* 2012) and RDP (Cole *et al.* 2014) may also be used with Kraken2.

Below, we use the "PlusPF" database, which includes sequences from archaea, bacteria, viral, plasmid, human, UniVec_Core, protozoa & fungi. The full database is 48 GB, but the version capped at 8 GB can be downloaded using this command:

```
wget -c https://genome-idx.s3.amazonaws.com/kraken/k2_pluspf_8gb_20210127.tar.gz -P
kraken_pluspf/ # download database

tar -zxvf kraken_pluspf/k2_pluspf_8gb_20210127.tar.gz -C kraken_pluspf/ # decom-
press database

rm kraken_pluspf/k2_pluspf_8gb_20210127.tar.gz # remove compressed file
```

In your configuration file, the "kraken_db_fp" parameter should point to the folder enclosing the database (Figure 1).

To run the taxonomic classification step:

```
sunbeam run -- --configfile sunbeam_config.yml all_classify
```

3.2a Evaluate taxonomic classification using Sunbeam extension: We can use a Sunbeam extension, *sbx_report*, to inspect results from the classification step. This will provide visual summaries of sequence quality along read position, read decontamination, and relative abundances of taxa from the phylum to the genus level. To download this extension, run:

```
sunbeam extend https://github.com/sunbeam-labs/sbx_report
```

Then run the following to generate HTML reports of read quality and taxonomic classification (Figures 2 and 3):

```
sunbeam run -- --configfile sunbeam_config.yml --use-conda final_report
```

4. Contig assembly

This step takes the cleaned reads and assembles them into longer genome regions called contigs. We assemble reads into contigs to increase sensitivity and accuracy when predicting and annotating genes. Contig assembly has been shown to provide substantial improvements in conjunction with NCycDB in particular (Anwar *et al.* 2019), which we use in Step 5. Contig assembly generally requires more computational power and time than any other step within metagenomic analysis (Quince *et al.* 2017). Using multiple threads (i.e. 16) is recommended, and this may require adding the "--cores 16" argument to the Sunbeam command.

Below, we use the software Megahit (Li *et al.* 2016), which is one of the fastest tools for metagenome assembly. For some samples, this speed may come at the expense of sensitivity, so users are welcome to substitute other software here. One option for this step is *co-assembly* of reads, in which information is shared between reads, which increases sensitivity to low-abundance reads (Sczyrba *et al.* 2017). However, this causes an exponential increase in assembly time and memory usage, possibly taking days or weeks to complete.

Figure 3. Taxonomic abundance reports produced by the *sbx_report* extension, described in Step 3.2A. Heatmap shows phylum-level read abundances for 2 NEON shotgun metagenomics samples.

4.1a Assemble contigs independently [recommended option]: In our configuration file (Figure 1), we have set the minimum length of contigs to 1000bp using the 'min_len' parameter. This value represents the average gene length for prokaryotes (Xu *et al.* 2006).

```
sunbeam run -- --configfile sunbeam_config.yml all_assembly
```

4.1b Co-assemble contigs independently: To take this route, you can use the extension shared by Sunbeam Labs, which carries out co-assembly using Megahit (Li *et al.* 2016).

4.2 Evaluate assembly output

For each sample, basic summaries of the contig assembly are stored in the following directory by default: sunbeam_output/assembly/megahit/. Longer contigs generally represent higher confidence in longer regions of the genome, although misassemblies can occur and lead to long contigs (Sczyrba *et al.* 2017). In the log files, you will find the minimum, maximum, and average contig length, as well as the number of contigs of at least 50bp.

4.2a Optional: evaluate assembly output using metaQUAST: We recommend the tool metaQUAST to perform a more in-depth evaluation assembly, such as summaries of contig length distributions (Figure 4), detection of misassemblies and errors, or comparison with reference databases to estimate the abundance of unknown species (Mikheenko *et al.* 2016). To download the metaQUAST program (as part of QUAST), run the following lines:

```
wget https://sourceforge.net/projects/quast/files/latest/download # download newest version
tar -xzf download # decompress file
```

Worst
Median
Best

☒ Show heatmap

Statistics without reference	SCBI_012_M_20140915_comp_cont...	WOOD_002_M_20140925_comp_cont...
# contigs	279	1051
# contigs (>= 0 bp)	279	1051
# contigs (>= 1000 bp)	279	1051
# contigs (>= 5000 bp)	8	2
# contigs (>= 10000 bp)	3	0
# contigs (>= 25000 bp)	1	0
# contigs (>= 50000 bp)	0	0
Largest contig	37 288	6167
Total length	449 012	1 338 887
Total length (>= 0 bp)	449 012	1 338 887
Total length (>= 1000 bp)	449 012	1 338 887
Total length (>= 5000 bp)	97 053	11 531
Total length (>= 10000 bp)	58 937	0
Total length (>= 25000 bp)	37 288	0
Total length (>= 50000 bp)	0	0
N50	1337	1204
N75	1111	1081
L50	79	434
L75	172	729
GC (%)	58.33	62.58
Mismatches		
# N's	0	0
# N's per 100 kbp	0	0

Figure 4. Output statistics from metaQUAST, summarizing contig lengths per sample. To produce similar statistics without downloading reference genomes, run metaQUAST with the "--max-ref-num" parameter set to 0.

To run the metaQUAST program on a sample or set of samples, specify the directory of input samples and output location like this (note: version number of QUAST may differ):

```
python ./quast-5.0.2/metaquast.py -o metaquast_output/sunbeam_output/assembly/contigs/
*.fa --max-ref-num 0
```

Section 2.4 of the metaQUAST manual discusses which reference genomes or databases are downloaded by default.

5. Annotation

The annotation step of the pipeline carries out BLAST searches on assembled contigs. Sunbeam will automatically use BLASTn for nucleotide databases, while BLASTx and BLASTp will be used for protein databases. Before protein databases are searched, the location of Open Reading Frames (ORFs) are predicted using the software Prodigal (Hyatt *et al.*, 2010).

Gene presence does not necessarily mean that the genes are transcribed or active; however, due to the metabolically expensive nature of maintaining genomic pathways (Lynch, 2006), there is potentially meaningful correspondence between gene presence and functional potential (Pérez-Cobas *et al.* 2020). Below, we demonstrate preparation of two BLAST protein databases that may be scientifically relevant for soil metagenomics.

Downloading the Comprehensive Antibiotic Resistance Database (CARD): CARD (Alcock *et al.* 2020) is a curated reference database of DNA sequences and proteins, designed to identify mutations and mechanisms of resistance to antibiotics, which can develop as a result of poor human stewardship (Brown & Wright 2016). However, antibiotic resistance can also be an ecological signifier of fungal-bacterial competition for nutrients (Bahram *et al.* 2018). We use the homolog protein genes to construct our reference database.

```
wget https://card.mcmaster.ca/download/0/broadstreet-v3.1.0.tar.bz2 -P db/card/ #
download into new directory

cd db/card/ # enter download directory

tar -xf broadstreet-v3.1.0.tar.bz2/protein_fasta_protein_homolog_model.fasta #
extract file

cd ../../ # return to analysis directory
```

Next, we convert to BLASTp database for use within our pipeline:

```
makeblastdb -in db/card/protein_fasta_protein_homolog_model.fasta -title card_
protein -dbtype 'prot' -hash_index # convert to BLASTp database
```

Downloading NCycDB: NCycDB categorizes genes into pathways that represent transformations such as nitrification, denitrification, and anammox. NCycDB was compiled from other sources, including COG, eggNOG, KEGG and the SEED (Tu *et al.* 2019). The NCycDB must be downloaded from Github and converted into a BLAST protein database. From the analysis directory, run the following commands to download the database, decompress the file, and change the file suffix:

```
svn export https://github.com/qichaol1984/NCyc/trunk/data/db/NCyc && gunzip db/
NCyc/NCyc_100.faa.gz
```

This database has duplicate sequences that can introduce problems later on. We can remove duplicates using the following commands, which utilize the programs BLAST and cd-hit:

```
mv db/NCyc/NCyc_100.faa db/NCyc/NCyc_100.fasta # change file extension
cd-hit -i db/NCyc/NCyc_100.fasta -o db/NCyc/NCyc_unique.fasta -c 1 -t 1 # remove duplicate
sequences
```

Next, we convert to BLASTp database for use within our pipeline:

```
makeblastdb -in db/NCyc/NCyc_unique.fasta -parse_seqids -title NCyc_unique --
dbtype prot -hash_index # convert to BLASTp database
```

In your configuration file, the “root_fp” and “protein” parameters should point to the BLAST database directory and file names (Figure 1). See the Sunbeam documentation for examples of configuration files that include nucleotide databases.

5.1 Run annotation

To run the annotation step:

```
sunbeam run -- --configfile sunbeam_config.yml all_annotate
```

6. Annotation post-processing

A suite of tools have been published for working with the BLASTxml outputs from Step 5. Python scripts can be used to convert BLASTxml to a CSV format; for examples, see the [Github repository](#) associated with this manuscript.

Once we have the read counts of genes associated with specific functions, we can compare results across samples. Gene counts should first be normalized to account for variation in sequencing depths (Pereira *et al.* 2018). One widely-used method is relative-log expression (RLE), which calculates scaling factors based on the geometric mean of gene

Figure 5. Log₁₀ normalized counts from a BLASTp search of Open Reading Frames (ORFs) within contigs from two shotgun metagenomic samples. Contigs were assembled using Megahit (Li *et al.* 2016), and ORFs were predicted using Prodigal (Hyatt *et al.* 2010). These samples are a subset of the full NEON shotgun metagenomics dataset (NEON DP1.10107.001). A) BLASTp hits for a search against the Comprehensive Antibiotic Resistance Database (CARD) (Alcock *et al.* 2020). Tetracycline resistance genes are defined as CARD entries with the word “tetracycline” in their description and “tet” in their name. B) BLASTp hits for a search against NCycDB (Tu *et al.* 2019). Genes are subset to those belonging to “Organic degradation and synthesis” pathways.

abundances across samples. RLE can be implemented using the DESeq R package (Love *et al.* 2014), and can be used to identify genes that are differentially abundant between groups (such as sites, or soil horizons).

For our two test samples, we can plot the outputs from each BLASTp search (Figure 5). Among antibiotic resistance genes, we can look at trends for specific types of antibiotics. Tetracycline resistance, for example, has become widespread in soil bacteria, possibly linked to intensive farming (Schmitt *et al.* 2006). For a subset of tetracycline-resistance genes, normalized abundances appear higher in the sample from the NEON's WOOD site (Figure 5A). For our nitrogen-cycling genes, we can subset to those associated with organic synthesis and degradation. For these genes, we see a similar pattern, with higher normalized abundances in the sample from the WOOD sites (Figure 5B). However, the SCBI sample had a lower sequencing depth overall (Figure 2B), which can prevent the detection of low-abundance genes.

7. Exporting to KBase for binning

The outputs from this pipeline can be further analyzed using the KBase platform, developed by the U.S. Department of Energy for microbiome analysis (Arkin *et al.* 2018). KBase links hundreds of different software tools using an online interface, which allows users to create “Narratives” for specific data analysis projects. Individual files can be uploaded to KBase directly, or they can be transferred in batches using Globus Online (Foster 2011).

For example, a KBase Narrative (Figure 6) could be used to create Metagenome-Assembled Genomes (MAGs). Because MAGs are created directly from contigs, rather than from microbes grown in an experimental setting, they often have no cultured relatives, representing a hidden source of genetic diversity in the microbiome (Nayfach *et al.* 2020). KBase

DATA Import FASTA File as Assembly from Staging Area Success

DATA Import FASTQ/SRA File as Reads from Staging Area Success

MaxBin2 Bin Contigs using MaxBin2 - v2.2.4 Success
Group assembled metagenomic contigs into lineages (Bins) using depth-of-coverage, nucleotide composition, and marker

JGI MetaBAT2 Contig Binning - v1.7 Success
Bin metagenomic contigs

CONCOCT Bin Contigs using CONCOCT - v1.1 Success
Group assembled metagenomic contigs into lineages (Bins) using depth-of-coverage and nucleotide composition

DAS-Tool Optimize Bacterial or Archaeal Binned Contigs using DAS Tool - v1.1.2 Success
Optimize bacterial or archaeal genome bins using a dereplication, aggregation and scoring strategy

CheckM Assess Genome Quality with CheckM - v1.0.18
Runs the CheckM lineage workflow to assess the genome quality of isolates, single cells, or genome bins from metagenome

Reset Finished with **success** 43m 7s ago View Configure Info Job Status Result

Report

View report in separate window

[CheckM Plot](#) | **CheckM TABLE**

Bin Name	Marker Lineage	# Genomes	# Markers	# Marker Sets	0	1	2	3	4	5+	Completeness	Contamination
bin_001	k__Bacteria	901	171	117	117	51	3	0	0	0	26.88	0.66

Figure 6. Creating and evaluating Metagenome-Assembled Genomes (MAGs) using the KBase Narrative interface (Arkin *et al.* 2018). First, quality-controlled sequencing reads and assembled contigs are imported using upload modules. Then, contigs are binned into putative genomes (or “bins”) using MaxBin2 (Wu *et al.* 2016), MetaBAT2 (Kang *et al.* 2019), and CONCOCT (Alneberg *et al.* 2014). Finally, DAS Tool (Sieber *et al.* 2018) is used to choose the highest-quality bins.

includes a variety of tools for creating MAGs, each using different algorithms, and outputs from multiple tools can be synthesized using a program called DAS Tool (Sieber *et al.* 2018). For each putative genome, or “bin,” summary statistics are produced that estimate the completeness and possible contamination of the genome, using a set of genes that are expected to be “single-copy” within a genome (Sieber *et al.* 2018).

In our example Narrative, we combine the output from three tools, MaxBin2 (Wu *et al.* 2016), MetaBAT2 (Kang *et al.* 2019), and CONCOCT (Alneberg *et al.* 2014). As inputs, we use the contigs assembled in Step 4 of this pipeline, as well as the quality-controlled sequencing reads from Step 2, for the sample WOOD_002-M-20140925-COMP. For this sample, DAS Tool produces one genome, *bin.001*, which is less than 27% complete. Bins can be further refined manually, and genomes that are more than 90% complete with less than 5% contamination may be good candidates for submission to public databases (Bowers *et al.* 2017). High-quality MAGs can uncover entirely new lineages in the microbial tree of life (Nayfach *et al.* 2020).

Troubleshooting, tips and tricks

For any rule, if not all files are processed, the step can be repeated using the `--unlock` and `--rerun-incomplete` parameters, i.e.:

```
sunbeam run -- --configfile sunbeam_config.yml clean_qc --rerun-incomplete --unlock
sunbeam run -- --configfile sunbeam_config.yml clean_qc --rerun-incomplete
```

To customize or expand on the workflow above, it is helpful to know the basic logic of Snakemake, which is the underlying framework for the Sunbeam pipeline. Snakemake relies on a series of rules, which specify input files, output files, and any necessary commands. When a rule is called, Snakemake works backwards from the output files to decide if any input files are missing or outdated, and tries to re-run rules as needed. If you want to add an extension to Sunbeam, a full guide is available in the [Sunbeam documentation](#).

To scale up to a larger dataset, a significant amount of computational power will be necessary, ideally with 8 or more cores for parallel computation. For those without access to institutional high-performance clusters, the scientific computing platform CyVerse (Merchant *et al.* 2016) offers free computational and storage resources. Note that intermediate files are generated for multiple steps, which can multiply the amount of storage needed for each metagenomic sample. Deleting these intermediate files when a step has completed will reduce the storage requirements.

Data availability

Raw metagenomics sequencing data is published as DP1.10107.001 from the National Ecological Observatory Network (<https://data.neonscience.org/data-products/explore>). All other data is previously published and cited throughout the paper.

Software availability

Bioconductor packages available at <https://www.bioconductor.org/>. CRAN packages available at <https://cran.r-project.org/>. Sunbeam software available at <https://sunbeam.readthedocs.io>.

Scripts to download NEON raw data, as well as process final BLASTxml files, are hosted at https://github.com/zoey-rw/metagenomes_NEON.

Archived scripts as at time of publication: <http://doi.org/10.5281/zenodo.4589528> (Werbin 2021).

License: Creative Commons Zero v1.0 Universal.

Acknowledgements

This material is based in part upon work supported by the National Science Foundation through the National Ecological Observatory Network, which is operated under cooperative agreement by Battelle Memorial Institute. We also thank Michael Silverstein at Boston University for assistance with Python scripting.

References

- Alcock BP, Raphenya AR, Lau TTY, *et al.*: **CARD 2020: Antibiotic resistome surveillance with the comprehensive antibiotic resistance database.** *Nucleic Acids Research*. 2020.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Allison SD, Lu Y, Weihe C, *et al.*: **Microbial abundance and composition influence litter decomposition response to environmental change.** *Ecology*. 2013; **94**(3): 714–725.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Alneberg J, Bjarnason BS, De Bruijn I, *et al.*: **Binning metagenomic contigs by coverage and composition.** *Nat Methods*. 2014; **11**(11): 1144–1146.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Andrews S, Krueger F, Seconds-Pichon A, *et al.*: **FastQC. A quality control tool for high throughput sequence data.** Babraham Bioinformatics. Babraham Institute; 2015.
- Anwar MZ, Lanzen A, Bang-Andreasen T, *et al.*: **To assemble or not to resemble-A validated Comparative Metatranscriptomics Workflow (CoMW).** *GigaScience*. 2019; **8**(8): 1–10.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Arkin AP, Cottingham RW, Henry CS, *et al.*: **KBase: The United States department of energy systems biology knowledgebase.** *Nat Biotechnol*. 2018; **36**(7): 566–569.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Bahram M, Hildebrand F, Forslund SK, Anderson JL, Soudzilovskaia NA, Bodegom PM, *et al.*: **Structure and function of the global topsoil microbiome.** *Nature [Internet]*. 2018; **560**(7717): 233–237.
[Publisher Full Text](#)
- Banerji A, Jahne M, Herrmann M, *et al.*: **Bringing Community Ecology to Bear on the Issue of Antimicrobial Resistance.** *Front Microbiol*. Frontiers Media S.A.; 2019; **10**: p. 15.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Bolger AM, Lohse M, Usadel B: **Trimmomatic: A flexible trimmer for Illumina sequence data.** *Bioinformatics*. 2014.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Breitwieser FP, Salzberg SL: **Pavian: Interactive analysis of metagenomics data for microbiome studies and pathogen identification.** *Bioinformatics*. 2020; **36**(4): 1303–1304.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Brown ED, Wright GD: **Antibacterial drug discovery in the resistance era.** *Nature*. 2016; **529**(7586): 336–343.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Clarke EL, Taylor LJ, Zhao C, *et al.*: **Sunbeam: An extensible pipeline for analyzing metagenomic sequencing experiments.** *Microbiome*. 2019; **7**(1): 1–13.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Cole JR, Wang Q, Fish JA, *et al.*: **Ribosomal Database Project: Data and tools for high throughput rRNA analysis.** *Nucleic Acids Res*. 2014.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Donovan PD, Gonzalez G, Higgins DG, *et al.*: **Identification of fungi in shotgun metagenomics datasets.** *PLoS One*. 2018; **13**(2): 1–16.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Edwards J, Edwards R: **Fastq-pair: efficient synchronization of paired-end fastq files.** *BioRxiv*. 2019; 552885.
[Publisher Full Text](#)
- Felix M, Jablonski KP, Letcher B, *et al.*: **Sustainable data analysis with Snakemake.** 2020. 1–16.
[Publisher Full Text](#)
- Foster I: **Globus online: Accelerating and democratizing science through cloud-based services.** *IEEE Internet Computing*. 2011; **15**(3): 70–73.
[Publisher Full Text](#)
- Hyatt D, Chen GL, LoCascio PF, *et al.*: **Prodigal: Prokaryotic gene recognition and translation initiation site identification.** *BMC Bioinformatics*. 2010.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Illumina: **Quality Scores.** *Technical Note: Informatics*. 2014: 1–2.
[Reference Source](#)
- Illumina: **iGenomes.** n.d..Retrieved October 12, 2020
[Reference Source](#)
- Jones M: **NEON Educational Resources for Online Teaching.** *NEON Observatory Blog*. 2020.
- Jonsson V, Österlund T, Nerman O, *et al.*: **Variability in Metagenomic Count Data and Its Influence on the Identification of Differentially Abundant Genes.** *J Comput Biol*. 2017; **24**(4): 311–326.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Kalantar K, Carvalho T, de Bourcy C, *et al.*: **IDseq – An Open Source Cloud-based Pipeline and Analysis Service for Metagenomic Pathogen Detection and Monitoring.** *April*, 1–14. 2020.
[Publisher Full Text](#)
- Kang DD, Li F, Kirton E, *et al.*: **MetaBAT 2: An adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies.** *PeerJ*. 2019; **2019**(7): 1–13.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Keller M, Schimel DS, Hargrove WW, *et al.*: **A continental strategy for the National Ecological Observatory Network.** *Front Ecol Environ*. 2008; **6**(5): 282–284.
[Publisher Full Text](#)
- Köster J, Rahmann S: **Snakemake-a scalable bioinformatics workflow engine.** *Bioinformatics*. 2012; **28**(19): 2520–2522.
- Ladoukakis E, Kolis FN, Chatziioannou AA: **Integrative workflows for metagenomic analysis.** *Front Cell Dev Biol*. 2014; **2**(NOV): 1–11.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Li D, Luo R, Liu CM, *et al.*: **MEGAHIT v1.0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices.** *In Methods*. 2016.
[Publisher Full Text](#)
- Love MI, Huber W, Anders S: **Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2.** *Genome Biol*. 2014; **15**(12): 1–21.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Lynch M: **Streamlining and simplification of microbial genome architecture.** *Annu Rev Microbiol*. 2006; **60**: 327–349.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Martin M: **Cutadapt removes adapter sequences from high-throughput sequencing reads.** *EMBnet*. 2010.
[Publisher Full Text](#)
- Mikheenko A, Saveliev V, Gurevich A: **MetaQUAST: Evaluation of metagenome assemblies.** *Bioinformatics*. 2016; **32**(7): 1088–1090.
- Mukherjee S, Huntemann M, Ivanova N, *et al.*: **Large-scale contamination of microbial isolate genomes by illumina Phix control.** *Stand Genomic Sci*. 2015; **10**(APRIL2015), 1–4.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Nasko DJ, Koren S, Phillippy AM, *et al.*: **RefSeq database growth influences the accuracy of k-mer-based lowest common ancestor species identification.** *Genome Biol*. 2018; **19**(1): 165.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- National Ecological Observatory Network: **Soil shotgun metagenomes (DP1.10107.001) RELEASE-2021.** Feb 8, 2021.
[Reference Source](#)
- Nayfach S, Roux S, Seshadri R, *et al.*: **A genomic catalog of Earth's microbiomes.** *Nat Biotechnol*. 2020.
[PubMed Abstract](#) | [Publisher Full Text](#)
- O'Leary NA, Wright MW, Brister JR, *et al.*: **Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation.** *Nucleic Acids Res*. 2016.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Pearman WS, Freed NE, Silander OK: **Testing the advantages and disadvantages of short- And long-read eukaryotic metagenomics using simulated reads.** *BMC Bioinformatics*. 2020; **21**(1): 1–15.
- Pérez-Cobas AE, Gomez-Valero L, Buchrieser C: **Metagenomic approaches in microbial ecology: an update on whole-genome and marker gene sequencing analyses.** *Microb Genom*. 2020; **6**(8).
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Pérez-Cobas AE, Gomez-Valero L, Buchrieser C: **Metagenomic approaches in microbial ecology: an update on genome and marker gene sequencing analyses.** *Microb Genom*. 2020; **6**(8).
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Quast C, Pruesse E, Yilmaz P, *et al.*: **The SILVA ribosomal RNA gene database project: Improved data processing and web-based tools.** *Nucleic Acids Res*. 2013; **41**(D1): 590–596.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Quince C, Walker AW, Simpson JT, *et al.*: **Shotgun metagenomics, from sampling to analysis.** *Nat Biotechnol*. 2017; **35**(9): 833–844.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Schmitt H, Stooß K, Hamscher G, *et al.*: **Tetracyclines and tetracycline resistance in agricultural soils: Microcosm and field studies.** *Microb Ecol*. 2006; **51**(3): 267–276.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Sczyrba A, Hofmann P, Belmann P, *et al.*: **Critical Assessment of Metagenome Interpretation - A benchmark of metagenomics software.** *Nat. Methods*. 2017; **14**(11): 1063–1071.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Sieber CMK, Probst AJ, Sharrar A, *et al.*: **Recovery of genomes from metagenomes via a dereplication, aggregation and scoring strategy.**

Nat Microbiol. 2018; **3**(7): 836–843.

[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)

Tu Q, Lin L, Cheng L, *et al.*: **NCycDB: A curated integrative database for fast and accurate metagenomic profiling of nitrogen cycling genes.** *Bioinformatics.* 2019; **35**(6): 1040–1048.

[PubMed Abstract](#) | [Publisher Full Text](#)

US Long Term Ecological Research Network: **LTER Sites.** n.d..Retrieved October 13, 2020.

[Reference Source](#)

Wang M, Doak TG, Ye Y: **Subtractive assembly for comparative metagenomics, and its application to type 2 diabetes metagenomes.** *Genome Biol.* 2015; **16**(1).

[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)

Waring BG, Averill C, Hawkes CV: **Differences in fungal and bacterial physiology alter soil carbon and nitrogen cycling: Insights from meta-analysis and theoretical models.** *Ecol Lett.* 2013; **16**(7): 887–894.

[PubMed Abstract](#) | [Publisher Full Text](#)

Weder N, Zhang H, Jensen K, *et al.*: **c.** *J Am Acad Child Adol Psych.* 2014; **53**(4): 163–178.

[Publisher Full Text](#)

Werbin Z: **zoey-rw/metagenomes_NEON: Adding license (Version v1.0.1).** *Zenodo.* 2021, March 8.

[Publisher Full Text](#)

Wood DE, Lu J, Langmead B: **Improved metagenomic analysis with Kraken 2.** *Genome Biol.* 2019.

[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)

Wu YW, Simmons BA, Singer SW: **MaxBin 2.0: An automated binning algorithm to recover genomes from multiple metagenomic datasets.** *Bioinformatics.* 2016; **32**(4): 605–607.

[PubMed Abstract](#) | [Publisher Full Text](#)

Xu L, Chen H, Hu X, *et al.*: **Average gene length is highly conserved in prokaryotes and eukaryotes and diverges only between the two kingdoms.** *Mol Biol Evol.* 2006; **23**(6): 1107–1108.

[PubMed Abstract](#) | [Publisher Full Text](#)

Open Peer Review

Current Peer Review Status: ? ?

Version 1

Reviewer Report 19 July 2021

<https://doi.org/10.5256/f1000research.54670.r84561>

© 2021 Nelson W. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

William Nelson

Biological Sciences Division, Pacific Northwest National Laboratory, Richland, WA, USA

Rationale:

My main question is who is the audience for this pipeline? Is this intended to be used by students to learn some metagenomic analysis and how the NEON data set can be interrogated? Or is this intended to be used by researchers, in which case I think the downstream annotation and analysis components are somewhat thin. Is this officially recognized by NEON as a standard pipeline that will enable comparison between analyses? I don't wish to sound dismissive, but this reads like a Yet-Another-Metagenomics-Pipeline paper, which on one hand is fine - there's nothing technically or scientifically wrong with it - but this would be a more impactful report if the purpose behind it was more strongly presented.

Description:

There is nothing wrong with the description of the various steps, but the descriptions are superficial. There is little discussion of why the methods were chosen and what their strengths and weaknesses are.

Replication:

The code blocks are great, but the formatting rendered incorrectly in my browser (Firefox) - newlines were not present, making it hard to interpret what the actual commands are. Also, I tried to follow along with those commands on our institutional computing cluster and got stuck on the installation of sunbeam. I was able to install sunbeam on my desktop server, but the test of the install failed. I went ahead and tried to follow the analysis anyway, but ran into multiple problems. Just a caveat that providing the commands doesn't ensure replicability.

A few other comments:

End of Dataset description: "TOS Science Design for Terrestrial Microbial Diversity, NEON.DOC.000908" - What is this?

The comment about miniconda, "this command may work", is likely to be confusing. Might be best just to say that anaconda is required and to talk to local IT about its availability and how to use it.

The transition between section 1.2 and 2 should make it clearer that section 1.2 was describing constructing the configuration file and sections 2 through 5 are describing the individual steps that make up the sunbeam pipeline. As it reads now, it could be interpreted that the QC step is subsequent to the sunbeam run.

Is section 4.1b missing a code block?

I did not understand what you meant by "We use the homolog protein genes to construct our reference database." in section 5.

The Bowers 2017 reference appears to be missing from the bibliography.

Is the rationale for developing the new method (or application) clearly explained?

Partly

Is the description of the method technically sound?

Partly

Are sufficient details provided to allow replication of the method development and its use by others?

Partly

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: I have 20 years experience performing microbial genomic and metagenomic analysis, including assembly, binning and annotation.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 22 Nov 2021

Zoey Werbin

- *My main question is who is the audience for this pipeline? Is this intended to be used by students to learn some metagenomic analysis and how the NEON data set can be interrogated? Or is this intended to be used by researchers, in which case I think the*

downstream annotation and analysis components are somewhat thin. Is this officially recognized by NEON as a standard pipeline that will enable comparison between analyses? I don't wish to sound dismissive, but this reads like a Yet-Another-Metagenomics-Pipeline paper, which on one hand is fine - there's nothing technically or scientifically wrong with it - but this would be a more impactful report if the purpose behind it was more strongly presented.

Thank you for identifying these deficiencies within the manuscript. Our intended audience is both students and researchers working with NEON soil metagenomes. We have stated this explicitly in the last paragraph of the Introduction to the article, and strengthened each section of the paper to increase its value to these groups. Specifically, we have added subsections titled "Background and Rationale" and "Considerations for NEON data" to each analysis section. We plan to submit this revised manuscript for inclusion as a NEON community resource.

- *There is nothing wrong with the description of the various steps, but the descriptions are superficial. There is little discussion of why the methods were chosen and what their strengths and weaknesses are.*

Each step has now been supplemented with descriptions of our preferred methods as well as the strengths and weaknesses of alternative methods (in "Background and Rationale"). We describe which methods have or have not been benchmarked or optimized for soil metagenomes, specifically, as well as their usefulness for the NEON dataset, given the properties of the data (in "Considerations for NEON data").

- *The code blocks are great, but the formatting rendered incorrectly in my browser (Firefox) - newlines were not present, making it hard to interpret what the actual commands are. Also, I tried to follow along with those commands on our institutional computing cluster and got stuck on the installation of sunbeam. I was able to install sunbeam on my desktop server, but the test of the install failed. I went ahead and tried to follow the analysis anyway, but ran into multiple problems. Just a caveat that providing the commands doesn't ensure replicability.*

Great points. In response to this and to the comments of Reviewer #1, we have adjusted our specific bioinformatic methods to address Sunbeam installation issues. We now recommend the stable branch of the metaGEM pipeline, which has run successfully in multiple Linux environments. The code blocks have all been shortened to improve readability and cross-browser formatting.

- *End of Dataset description: "TOS Science Design for Terrestrial Microbial Diversity, NEON.DOC.000908" - What is this?*

The citation for this sampling protocol document has been changed to "Stanish & Parnell, 2018", with the full protocol version information within the Works Cited.

- *The comment about miniconda, "this command may work", is likely to be confusing. Might be best just to say that anaconda is required and to talk to local IT about its availability and how to use it.*

The sentence on miniconda requirements has been revised to point readers to their system administrators.

- *The transition between section 1.2 and 2 should make it clearer that section 1.2 was describing constructing the configuration file and sections 2 through 5 are describing the individual steps that make up the sunbeam pipeline. As it reads now, it could be interpreted that the QC step is subsequent to the sunbeam run.*

This recommendation is no longer relevant, given our shift in methods and manuscript

organization.

- *Is section 4.1b missing a code block?*

This section is no longer present, given our shift in methods and manuscript organization.

- *I did not understand what you meant by "We use the homolog protein genes to construct our reference database." in section 5.*

This section is no longer present, given our shift in methods and manuscript organization.

- *The Bowers 2017 reference appears to be missing from the bibliography.*

This reference has been added to the bibliography.

Competing Interests: No competing interests were disclosed.

Reviewer Report 28 June 2021

<https://doi.org/10.5256/f1000research.54670.r83581>

© 2021 Zimmerman N. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Naupaka Zimmerman

Department of Biology, University of San Francisco, San Francisco, CA, USA

This is a timely and valuable contribution that has the potential to aid in the use of NEON data by a wider audience. The core approach (using Sunbeam, a snakemake pipeline, to analyze NEON metagenomics data) seems like a good one, and will offer advantages to users who are not yet comfortable enough to develop their own such pipeline from scratch.

While in general the approach is a good one and the need for the tool is real and well-articulated by the authors, there are a number of aspects that could be improved to maximize the value of this contribution. I will outline a few here, but I was unable to complete the full pipeline in my testing using the example data specified in the manuscript, and so I am not able to comment on all aspects of the pipeline at this time. I would be happy to do another view and assessment after hearing from the authors.

I outline some suggestions below:

In the last paragraph of the introduction, I would encourage the authors to revise this sentence: "The pipeline that we present here is designed to complement existing NEON educational resources, such that users without prior bioinformatics experience may use this dataset to learn about microbial communities within the soil." The background skills that are necessary to successfully understand and implement the approach outlined here is not trivial and I don't think it's exactly best suited for someone "without prior bioinformatics experience". I think such a user would more likely need a graphical interface that did not presume comfort with the *nix command line etc. I think the approach outlined here is a valuable contribution because it targets users who may have some comfort with programmatic and command-line approaches, but does

not yet have the skill to develop a flexible pipeline themselves.

In the methods section, first paragraph, I think I would revise to be more careful with tenses. In some cases the collection protocols will remain mostly unchanged (e.g. I don't think NEON is planning to add any core sites), but other things may change (the kits that they use, the sequencing depth or sequencer used, etc. Since NEON is a 30 year project, it might help the manuscript's longevity if this paragraph were worded to reflect possible future methodological changes.

I might encourage a mention or a suggestion that users use tmux or screen to run pipelines like this is they are connected to a remote server over something like ssh. If the connection drops during a many hours long pipeline, it can be quite frustrating.

In step 1.2, why do you suggest the use of the develop branch of Sunbeam? Isn't that more likely to include breaking changes that will be overly challenging for the target audience? Perhaps this could be adjusted to use a stable branch or version, and the text could highlight the develop branch alternative for those willing to trade troubleshooting time in exchange for quicker access to more advanced features.

For downloading the config file, it might be better to pull from an archival version of the file instead of the github version, or at the least include a version at a specific commit and not just the main branch, so that it remains stable. Otherwise either the code could break, or the authors would need to continually update the configuration to track with software changes.

In my testing of the approach in the manuscript, I am unable to get past the tests that occur after the installation of Sunbeam (``bash tests/run_tests.bash``). The tests repeatedly fail with segmentation faults during either the megahit or kraken steps. This is on an Ubuntu 20.04 machine with lots of RAM/disk space/cores. I am not sure where the issue is, and I would consider myself reasonably able to troubleshoot such problems, so I am concerned that similar problems might arise and be too challenging for the target audience/user. I would be happy to work with the authors in more detail to resolve this problem (share log files, etc). I shall share them via a comment when I am able to.

Overall, I think this is a valuable contribution that fills a need in the community and uses a good approach to do so. However, in its current form, I cannot successfully run the example code, even on the recommended sample files, and so I have concerns with the brittleness of the approach outlined. I'd encourage the authors to do some additional testing on other machines and settings, and/or build some more resilience into the installation walkthrough so that the average target user is able to make use of this contribution.

Is the rationale for developing the new method (or application) clearly explained?

Yes

Is the description of the method technically sound?

Partly

Are sufficient details provided to allow replication of the method development and its use by others?

Partly

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Environmental microbial ecology, including specific experience in bioinformatics and pipelines, and several years of experience working with large NEON sequencing datasets.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 22 Nov 2021

Zoey Werbin

Original reviewer comments are italicized.

- *This is a timely and valuable contribution that has the potential to aid in the use of NEON data by a wider audience. The core approach (using Sunbeam, a snakemake pipeline, to analyze NEON metagenomics data) seems like a good one, and will offer advantages to users who are not yet comfortable enough to develop their own such pipeline from scratch. While in general the approach is a good one and the need for the tool is real and well-articulated by the authors, there are a number of aspects that could be improved to maximize the value of this contribution. I will outline a few here, but I was unable to complete the full pipeline in my testing using the example data specified in the manuscript, and so I am not able to comment on all aspects of the pipeline at this time. I would be happy to do another view and assessment after hearing from the authors.*

Thank you for highlighting the issues with the reproducibility of the pipeline we outlined. Due to the referenced issues with installing software, we have switched to a similar Snakemake pipeline (metaGEM) that has been tested on various computing systems. We describe this new pipeline in the "Implementation" section of the revised manuscript.

- *In the last paragraph of the introduction, I would encourage the authors to revise this sentence: "The pipeline that we present here is designed to complement existing NEON educational resources, such that users without prior bioinformatics experience may use this dataset to learn about microbial communities within the soil." The background skills that are necessary to successfully understand and implement the approach outlined here*

*is not trivial and I don't think it's exactly best suited for someone "without prior bioinformatics experience". I think such a user would more likely need a graphical interface that did not presume comfort with the *nix command line etc. I think the approach outlined here is a valuable contribution because it targets users who may have some comfort with programmatic and command-line approaches, but does not yet have the skill to develop a flexible pipeline themselves.*

This sentence has been revised to reflect that our audience is those with basic bioinformatics experience. Further, each section of the manuscript has been expanded to include a thorough description of the rationale for various decisions in the subsections "Background and Rationale" and "Considerations for NEON data", so that this can be a more useful introductory guide to soil metagenomics.

- *In the methods section, first paragraph, I think I would revise to be more careful with tenses. In some cases the collection protocols will remain mostly unchanged (e.g. I don't think NEON is planning to add any core sites), but other things may change (the kits that they use, the sequencing depth or sequencer used, etc. Since NEON is a 30 year project, it might help the manuscript's longevity if this paragraph were worded to reflect possible future methodological changes.*

Tenses in the "Dataset description" section have been modified to reflect that the reported sampling and sequencing protocols are accurate as of 2021. We state that this bioinformatics protocol is intended for short-read data specifically, and that NEON protocols may shift in the future.

- *I might encourage a mention or a suggestion that users use tmux or screen to run pipelines like this is they are connected to a remote server over something like ssh. If the connection drops during a many hours long pipeline, it can be quite frustrating.*

We now reference tmux and screen in Implementation, within the sub-section "Local vs cluster analysis".

- *In step 1.2, why do you suggest the use of the develop branch of Sunbeam? Isn't that more likely to include breaking changes that will be overly challenging for the target audience? Perhaps this could be adjusted to use a stable branch or version, and the text could highlight the develop branch alternative for those willing to trade troubleshooting time in exchange for quicker access to more advanced features.*

Due to our shift in methods, we no longer use either the develop or stable branch of Sunbeam. At the time of writing, however, the develop branch had implemented a potential fix for the segmentation fault errors, but it did not resolve errors on all operating systems. We hope the local and cluster options for running the metaGEM pipeline will also help with reducing troubleshooting time.

- *For downloading the config file, it might be better to pull from an archival version of the file instead of the github version, or at the least include a version at a specific commit and not just the main branch, so that it remains stable. Otherwise either the code could break,*

or the authors would need to continually update the configuration to track with software changes.

With our shift from Sunbeam to metaGEM, we decided to remove the example configuration file. The configuration file that comes installed with metaGEM primarily needs file paths to be modified by the user, whereas most parameters can be left as-is. Throughout the text, we've bolded sentences that instruct the user to modify the configuration filepaths.

- *In my testing of the approach in the manuscript, I am unable to get past the tests that occur after the installation of Sunbeam ('bash tests/run_tests.bash'). The tests repeatedly fail with segmentation faults during either the megahit or kraken steps. This is on an Ubuntu 20.04 machine with lots of RAM/disk space/cores. I am not sure where the issue is, and I would consider myself reasonably able to troubleshoot such problems, so I am concerned that similar problems might arise and be too challenging for the target audience/user. I would be happy to work with the authors in more detail to resolve this problem (share log files, etc). I shall share them via a comment when I am able to.*

Overall, I think this is a valuable contribution that fills a need in the community and uses a good approach to do so. However, in its current form, I cannot successfully run the example code, even on the recommended sample files, and so I have concerns with the brittleness of the approach outlined. I'd encourage the authors to do some additional testing on other machines and settings, and/or build some more resilience into the installation walkthrough so that the average target user is able to make use of this contribution.

These are excellent points and led to a dramatic shift in the focus and implementation of this analysis pipeline. The main text of the manuscript now focuses on the various options available to users for each step of soil metagenomic analysis, and describes issues specific to soil ecology and the NEON dataset specifically. The code at the end of each section is now an example of how these decisions may be implemented via specific tools. For this revision, we have communicated with the developers of the tools mentioned (metaGEM and Toolchest) and are confident that these tools will maintain resilience in the coming years. We hope this sufficiently addresses problems of brittleness.

Competing Interests: No competing interests were disclosed.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research