METAWORKS: A flexible, scalable bioinformatic pipeline for multi-marker biodiversity assessments

Teresita M. Porter[1*], Mehrdad Hajibabaei[1]

[1] Centre for Biodiversity Genomics @ Biodiversity Institute of Ontario & Department of Integrative Biology, University of Guelph, Guelph, ON, CANADA

* Corresponding author:

T.M. Porter

terrimporter@gmail.com

Abstract

Background: Multi-marker metabarcoding is increasingly being used to generate biodiversity information across different domains of life from microbes to fungi to animals such as in ecological and environmental studies. Current popular bioinformatic pipelines support microbial and fungal marker analysis, while ad hoc methods are used to process animal metabarcode markers from the same study. The purpose of this paper is to introduce MetaWorks, a 'meta'barcode pipeline that does 'the works' and supports the bioinformatic processing of various metabarcoding markers including rRNA and their spacers as well as protein coding loci.

Results: MetaWorks provides a Conda environment to quickly gather most of the programs and dependencies for the pipeline. MetaWorks is automated using Snakemake to ensure reproducibility and scalability. We have supplemented existing RDP-trained classifiers for SSU (prokaryotes), ITS (fungi), and LSU (fungi) with trained classifiers for COI (eukaryotes), rbcL (diatoms or eukaryotes), SSU (diatoms or eukaryotes), and 12S (fish). MetaWorks can process rRNA genes, but it can also properly handle ITS spacers by trimming flanking conserved rRNA gene regions, as well as handle protein coding genes by removing obvious pseudogenes.

Conclusions: As far as we are aware, MetaWorks is the first flexible multi-marker metabarcode pipeline that can accommodate rRNA genes, spacer, and protein coding markers in the same pipeline. This is ideal for large-scale, multi-marker studies to provide a harmonized processing environment, pipeline, and taxonomic assignment approach. Updates to MetaWorks will be made as needed to reflect advances in the

underlying programs, reference databases, or hidden Markov model (HMM) profiles for pseudogene filtering. Future developments will include support for additional metabarcode markers, RDP trained reference databases, and HMM profiles for pseudogene filtering.

Keywords

Metabarcoding, conda, snakemake, COI, rbcL, rRNA gene, ITS

Background

Marker gene sequencing, metabarcoding, or metasystematics are interrelated techniques that involves extracting environmental DNA from bulk samples such as soil, water, or individuals collected from traps without having to isolate any individual specimens followed by enrichment of a signature DNA region to identify biological community composition using bioinformatics [1–3]. In different fields of study, from microbial ecology to animal biodiversity studies, different signature DNA regions are targeted for their ability to identify target taxa. For example, in prokaryotes, the 16S rRNA region is often used for genus level taxonomic assignments [4, 5]. In animals, plants, and fungi, COI, rbcL, or ITS are commonly targeted to identify metabarcode reads to the species rank, respectively, based on the availability of reference sequences [6–9]. Other markers commonly used for phylogenetic analyses are also popular targets, such as SSU for eukaryotes, arbuscular mycorrhizal fungi, and diatoms; or 12S for fish [10–15]. For each of these markers, reference sequences are often housed in

their own separate databases where they can be analyzed by custom- or built-in tools [16–22].

Existing well-developed and popular pipelines such as QIIME and MOTHUR were initially developed to support the microbial ecology community [23–25]. Methods incorporated into these pipelines largely support the analysis of 16S rRNA genes, but they do also support the analysis of popular fungal markers such as ITS and LSU [5, 19, 26, 27]. QIIME also supports a data flow to properly isolate the ITS spacer regions from conserved flanking regions [28]. In our study of biodiversity genomics, we also had a need for a pipeline that could handle other metabarcode regions such as COI, rbcL, and 12S. For protein-coding markers, we wanted the option to filter for pseudogenes as an additional way to remove these non-target sequences from the dataset. Pseudogenes are duplicated copies of a gene that are not functional, may be truncated, under relaxed selection pressure means these sequences may accumulate insertions and deletions that may introduce premature stop codons or frame shifts. This may be problematic for biodiversity studies that use metabarcoding if these pseudogenes are amplified and result in inflated richness estimates. Also, since sequencing technologies are often changing, with datasets getting larger, and active development of metabarcode sequence handling programs are often updated, we also wanted a pipeline with the flexibility to update underlying programs frequently at the start of every new project.

For analyzing rRNA genes such as 16S, 18S, and ITS, there exist numerous projects that offer the ability to identify metabarcodes based on a comparison to sequences classified using morphology-based taxonomy, phylogenetic relatedness, or some mixture of both and these are supported by the major pipelines [16, 17, 19, 20].

For analyzing animal or plant markers, such as COI and rbcL, the BOLD database has a number of in-house tools that are appropriate for analyzing single sequences and is best used as a curation tool [18] but the underlying database is not fully available in a format that existing popular pipelines can use as is. We also identified a need for a pipeline that would use a proven taxonomic assignment method, as opposed to a sequence similarity-based method like BLAST, to identify our metabarcodes and reduce false positive rates [29–31]. Thus, we also needed RDP-trained reference sequence datasets to support the assignment of 18S, 12S, COI, and rbcL datasets, with options to support the identification of target taxa such as diatoms or fish [29, 30, 32]. Another important consideration when working with protein coding genes, is the ability to easily filter out obvious pseudogenes, such as the nuclear encoded mitochondrial sequences (NuMTs) co-sequenced with COI primers during PCR [33, 34].

For large scale or multi-year projects such as ecological and biomonitoring programmes, e.g. Baird and Hajibabaei, 2012, the question of version control and reproducibility is of utmost concern [35]. Coordinating analyses across different labs focusing on different target taxa or utilizing different markers can be complicated, with researchers each using their own bioinformatic pipelines as appropriate. Here is where we also felt the need to use tools to ensure environment as well as pipeline reproducibility. Though the target taxa or markers may differ across labs, we saw a need for a unifying bioinformatic pipeline that could offer a degree of standardization, consistency, and reproducibility.

As multi-marker studies are carried out on phylogenetically divergent taxa, such as in biodiversity or trophic studies, there is a need for more generic pipelines where

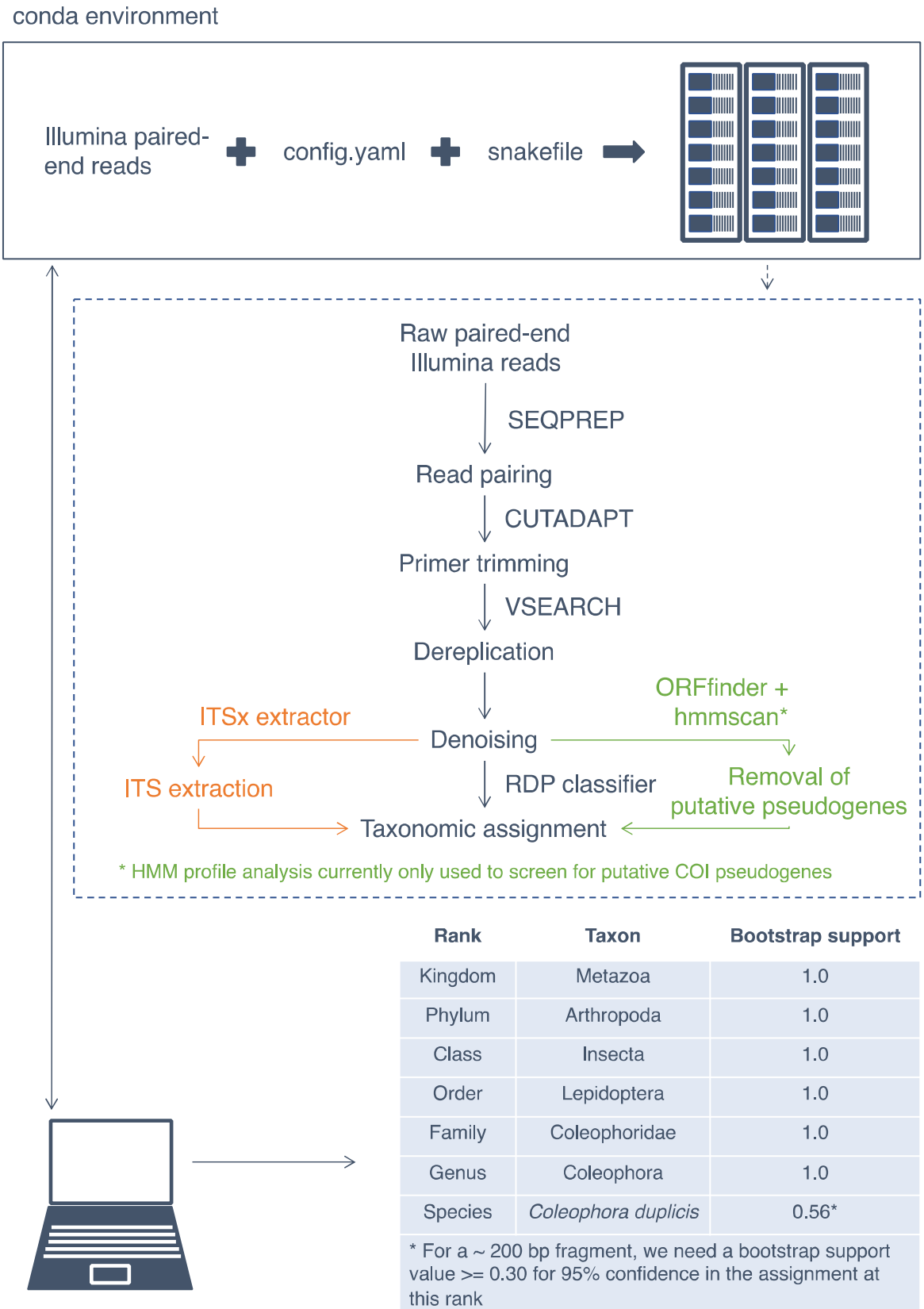
different markers can be analyzed using similar dataflows with 3rd party programs instead of being limited to database-specific pipelines and tools [36, 37]. Our objective was to develop a flexible bioinformatic pipeline suitable for processing multi-marker metabarcode datasets generated using paired-end Illumina sequencing with the following considerations: 1) reproducibility with respect to the computational environment used as well as the pipeline itself, 2) scalability to leverage multi-core processors to speed up the analysis of large datasets, 3) support the use widely used rRNA gene, spacer, or protein-coding markers. To support the last consideration, we have curated multiple reference sets to ensure consistently defined taxonomic lineages, trained the RDP naive Bayesian classifier to make rapid and accurate flexible-rank taxonomic assignments. Our pipeline also supports the processing of protein-coding markers, by implemented steps to help remove obvious pseudogenes. Finally, as collaborators in several large-scale metabarcoding projects involving various stakeholders, this pipeline has evolved to address the practical needs of various projects from graduate student research to national biomonitoring programmes.

Implementation

MetaWorks is a multi-marker ‘meta’-barcode pipeline that does ‘the works’ by supporting the bioinformatic processing of popular markers including rRNA genes, spacers, and protein coding genes. The standard pipeline is shown in Figure 1, as well as accommodations for processing ITS spacers, such as the removal of flanking rRNA gene sequences using the ITSx program [38], and protein coding genes, such as pseudogene filtering. More detailed data flows used to process different markers are

shown in the supplement (Fig S1 - S4). This pipeline is meant to be run in a Linux environment at the command-line. The most recent version is available from https://github.com/terrimporter/MetaWorks.

Figure 1. MetaWorks v1 pipeline overview. The pipeline can be run in a conda environment, providing raw paired-end Illumina reads, a configuration file, and a snakefile. Snakemake can be directed to run parallel jobs across many CPUs in a high performance computing environment. A summary of the steps carried out in the pipeline are shown in the dashed box. The standard pipeline is shown along with the variation needed to process ITS sequences (orange) and the variations needed to screen out putative pseudogenes (green). We note with the asterisk that the pseudogene removal step is currently different for rbcL and COI: for rbcL, longest ORF lengths are screened for outliers; whereas for COI, longest ORFs are further subjected to hidden Markov model (HMM) profile analysis and HMM scores are used to screen for outliers. For each exact sequence variant (ESV), for each sample, read counts and taxonomic assignments are provided along with bootstrap support values. An example of the taxonomic assignment output is shown in the table.



Rank	Taxon	Bootstrap support
Kingdom	Metazoa	1.0
Phylum	Arthropoda	1.0
Class	Insecta	1.0
Order	Lepidoptera	1.0
Family	Coleophoridae	1.0
Genus	Coleophora	1.0
Species	*Coleophora duplicis*	0.56*

* For a ~ 200 bp fragment, we need a bootstrap support value >= 0.30 for 95% confidence in the assignment at this rank

148

The first feature of MetaWorks is the use of a conda environment [39]. We provide a conda environment file (environment.yml) and when activated ensures that most of the required software programs and their dependencies are available for the pipeline to call and ensures a consistent processing environment is created for all users who run the pipeline. As the pipeline is updated, the environment may also be updated to contain the newest versions of the underlying programs. Using the provided environment also ensures that the correct version of each program is used. If anaconda isn't already installed on the user's system, a stripped-down version of anaconda, 'miniconda' is available from https://docs.conda.io/en/latest/miniconda.html . We provided instructions on how to install and use conda on the MetaWorks README page on GitHub https://github.com/terrimporter/MetaWorks.

Unfortunately, not all of the programs we use in our pipelines are currently available as conda packages and if they are not already available on the user's system, they will need to be installed separately. For example, the Ribosomal Database Project (RDP) classifier v2.12 is available from https://sourceforge.net/projects/rdp-classifier/ and is used to make taxonomic assignments [5]. The RDP classifier uses a naive Bayesian method to taxonomically assign unknown query sequences as well as provide a measure of statistical support for each assignment at each rank. We have previously described and compared how this method works compared to the top BLAST hit method [30]. In that paper, we showed how the classifier is faster than the top BLAST hit method and helps to reduce the rate of false-positive assignments. In studies where

erroneously identifying a metabarcode sequence as a potential invasive species or pathogen could lead to alarm, reducing the false-positive assignment rate is critical. An additional program is needed if pseudogene filtering of protein-coding genes is carried out. The NCBI ORFfinder is a translation program used to identify open reading frames (ORFs) and is used here help screen out putative pseudogenes and is available from https://www.ncbi.nlm.nih.gov/orffinder/. Instructions on how these programs can be downloaded and installed are also provided in the MetaWorks README file on GitHub.

The pipeline itself is automated using Snakemake [40]. Snakemake requires three sets of files to run: 1) raw paired-end Illumina sequence files, 2) a configuration file that specifies file paths, as well as program and pipeline settings, and, 3) a snakefile that runs each step of the bioinformatic pipeline according to the settings in the configuration file. One advantage of using Snakemake over a simple shell script to automate the pipeline, is the ability to resume a pipeline from where it left off if a problem is encountered. Another feature is that Snakemake supports the use of multiple parallel jobs and this is ideal for high performance computing environments where many cores are available to speed up the analysis of large datasets. The MetaWorks pipeline is also versioned and regularly updated on GitHub.

Snakemake also requires a configuration file. In this file, the user can specify the primers used in the study and adjust the settings for the major steps of the pipeline (described below). The user needs to enter the complete paths to where the raw data is stored and provide the path to the RDP classifier [5]. The RDP classifier already provides trained prokaryote SSU (16S), fungal LSU (28S), and fungal ITS (Warcup/UNITE) datasets [5, 19, 26, 27]. To use a custom trained dataset, such as

SSU (diatoms or eukaryotes), rbcL (diatoms or eukaryotes), or 12S (fish) the trained classifier first needs to be downloaded from GitHub and the path to this reference dataset needs to be provided in the configuration file. For most users, only the configuration file will need to be edited and customized and the snakefile itself does not need to be changed.

The standard MetaWorks pipeline is ideal for processing rRNA gene markers. Raw, paired-end Illumina reads are paired using SEQPREP [41]. Default settings are used with the option to change the Phred quality score cutoff (default, 20) or the minimum overlap (default, 25 bp). Primers are removed by aligning the primer sequence to the sequence and removing this sequence region using CUTADAPT in two separate steps, the first to remove the forward primer region, then successfully trimmed sequences are trimmed to remove the reverse primer region [42]. Default settings are used with the ability to customize the primer sequences (use N's instead of I's if applicable), specify the minimum sequence retained after trimming (default, 150 bp), adjust the Phred quality threshold at the 5' and 3' ends of the sequence (default, 20, 20), and to set the maximum number of N's in the primer sequence region (default, 3).

At this stage, the sequences for each sample are still in their own individual files. To run a global analysis, each of these sequence files concatenated into a single new file. These primer-trimmed reads are now ready for further processing in VSEARCH [43]. VSEARCH is an open-source program that can be used as an alternative to USEARCH [44]. USEARCH is proprietary software appropriate for processing marker gene sequences / metabarcoding reads with a free 32-bit version limited to using 4Gb memory or less. The 64-bit version can use all the available memory on a user's

system and requires a paid license. For small datasets, USEARCH can be used as is with no issues, but for larger datasets that require more memory, VSEARCH may be a better alternative. As a result, we use VSEARCH in this pipeline but keep in mind that the algorithms used for dereplication, denoising, chimera-removal, and tracking read counts are based on those originally developed in USEARCH [45, 46].

Primer-trimmed reads are dereplicated using the 'derep_fulllength' command in VSEARCH while tracking read numbers in each sequence cluster for downstream steps. This step retains just the unique sequences. At this step it is possible to have two sequences that differ in length but may be identical in the overlapping region. Sequences in the output file are ordered by decreasing cluster size. Dereplicated reads are then denoised using the 'unoise3' algorithm. At this step, sequences with predicted errors are corrected and rare reads are removed. The parameter to set the minimum number of reads per cluster is set to 3 to remove just singletons and doubletons, but this can be modified in the configuration file. The output can be thought of as operational taxonomic units (OTUs) clustered with 100% sequence similarity, but they have also been denoised and we refer to these as exact sequence variants (ESVs). Similar sets of denoised reads have been previously described at amplicon sequence variants (ASVs) when using the DADA2 pipeline or zero-radius OTUs (ZOTUs) in the original USEARCH pipeline [45, 47]. In USEARCH, chimera removal is incorporated in the unoise3 function, but in VSEARCH, chimera-removal needs to be run separately using the 'uchime3_denovo' command. To map read counts to the newly generated denoised ESVs, we use the 'search_exact' method. The 'search_exact' method is

preferred over the ‘usearch_global’ with the ‘id 1.0’ parameter because it is faster and optimized to find exact matches.

When processing ITS sequences, we use the ITSx extractor to remove any rRNA gene regions (SSU, 5.8S, or LSU) [38]. ITS is the standard fungal barcode marker and one of the supplementary barcodes for plants [6, 48, 49, 8]. The ITS1 region is the targeted region in the Earth Microbiome Project and the ITS2 region is targeted by others including the Metagenomics-based ecosystem biomonitoring project (Ecobiomics) and the ITS2 database V [4, 21, 50]. Our current pipeline is set to retain the ITS2 region but can be edited to target the ITS1 region in the configuration file.

Each denoised ESV is taxonomically assigned using the Ribosomal Database Project (RDP) classifier [5]. The RDP classifier uses a naive Bayesian method for taxonomic assignment and requires both reference sequences as well as a clearly defined taxonomy file as input for training. Although the classifier was originally developed to classify prokaryote 16S rRNA gene sequences using the Bergey’s taxonomy, we have retrained the classifier for a number of other rRNA gene and protein coding markers including COI (eukaryotes), rbcL (diatom and eukaryote), SSU (diatom and eukaryote), and 12S (fish) (Table 1) [30, 32]. Each of the trained reference sets should be downloaded, as needed, for MetaWorks processing. Note that the FASTA formatted files used for training the RDP classifier are also available for each of these reference sets and that they could alternatively be used for custom BLAST database creation if needed for comparison purposes against the top BLAST hit method.

Table 1. RDP-trained reference sets that can be used with MetaWorks v1.

| Marker | Target taxa | Classifier availability | Number of | Number of | Source data |

			included sequences	included taxa at all ranks (species)	
COI	Eukaryotes	https://github.com/terrimporter/CO1Classifier	1,221,528	154,351 (114,687)	BOLD [18], INSDC [51]
rbcL	Diatoms	https://github.com/terrimporter/rbcLdiatomClassifier	3,504	1,432 (1,023)	R-Syst::diatom [22]
rbcL	Eukaryotes	https://github.com/terrimporter/rbcLClassifier	164,454	65,742 (53,344)	INSDC
12S	Fish	https://github.com/terrimporter/12SfishClassifier	2,853	4,751 (2,833)	MitoFish [52]
SSU (18S)	Diatoms	https://github.com/terrimporter/SSUdiatomClassifier	2,962	1,198 (828)	R-Syst::diatom
SSU (18S)	Eukaryotes	https://github.com/terrimporter/18SClassifier	42,301	7,504 (5,440 genera)	SILVA [17]
SSU (16S)	Prokaryotes	Built-in to the RDP classifier*	13,212	3,247 (2,506 genera)	RDP [5]
ITS	Fungi (Warcup)	Built-in to the RDP classifier	17,878	10,621 (8,551)	[27]
ITS	Fungi (UNITE)	Built-in to the RDP classifier	145,019	23,222 (20,337)	[19]
LSU	Fungi	Built-in to the RDP classifier	11,442	2,633 (1,895)	[26]

For protein coding genes, we attempt to remove obvious pseudogenes. For rbcL, we translate our denoised ESVs using ORFfinder [53]. We retain the longest open reading frame (ORF) and screen for ORFs with outlier lengths. Outliers are identified based on their nucleotide ORF sequence length. ORFs with a sequence length less than the 25th percentile - (1.5 x interquartile range (IQR)) are removed as short outliers. ORFs with a sequence length greater than the 75th percentile length + (1.5 x IQR) are removed as long outliers. An extra step was used to filter out potential COI pseudogenes. Denoised ESVs were translated as described above and the longest ORFs were retained. Then we used hidden Markov model (HMM) profile analysis using hmmscan to compare the amino acid ORFs to an HMM profile using the program HMMER available from http://hmmer.org/ . The COI HMM profile was built

using COI sequences mined from the BOLD data releases as previously described in [Porter and Hajibabaei, in prep]. Amino acid ORFs with short outlier HMMER scores were filtered out of the dataset as putative pseudogenes (or genuine sequences with PCR/sequencing errors).

The final results file is a comma separated file suitable for import into R for data analysis. The output contains ESVs for each sample, read counts, ESV/ORF sequences along with the taxonomic assignment. Bootstrap support values are also provided for the taxonomic assignments at each rank. These bootstrap support values can be used to filter for assignments where assignments are likely to be correct 95-99% confidence, assuming the query is present in the reference sequence database. A guide for cutoff values is provided for each custom-trained classifier in the GitHub README pages (Table 1). For the built-in reference sets in the RDP classifier, an 80% cutoff is recommended [5] unless the query sequence is shorter than 250 bp in which case a cutoff of 50% is recommended on their website at https://rdp.cme.msu.edu/classifier/classifier.jsp. In addition, we also provide statistics for the output at each major bioinformatic step (number of sequences, as well as max, min, mean, median, and mode lengths). Log files are also retained for each major step run in VSEARCH. If more than one amplicon was pooled prior to sample indexing, the pipeline can be run multiple times updating the configuration file as needed.

Discussion

MetaWorks is meant to be run at the command line in a conda environment [39]. The use of a conda environment facilitates quickly obtaining most of the programs and

dependencies needed to run the pipeline. The choice of Snakemake to automate the pipeline was meant to ensure reproducibility and provide scalability when run in a high-performance computing environment [40]. The pipeline was built with flexibility in mind, to support the bioinformatic processing of multiple microbial, fungal, or animal metabarcode markers. There is no need to gather separate OTU x sample tables or FASTA files for downstream analyses. The final results file contains taxonomic assignments with bootstrap support values, for each ESV, from each sample, along with read counts. The output is in a comma separated file, easily imported into R, a popular open-source software environment for data analysis and visualizations [54]. At this step, ESV x sample tables can be regenerated or ESV/ORF sequences can be extracted for further analysis.

MetaWorks is a flexible, scalable bioinformatic pipeline that will process raw paired-end Illumina reads for any study that uses a metabarcoding approach. We envision that MetaWorks will fill a need in multi-marker metabarcoding studies that target taxa from multiple different domains of life, to provide a unified processing environment, pipeline, and taxonomic assignment approach for each marker from ribosomal RNA genes, spacers, or protein coding genes. QIIME 2 is perhaps the most popular and comprehensive platform for such work, but to date, focuses on processing mainly prokaryote and fungal datasets [25]. As of yet, MetaWorks is the only bioinformatic pipeline that can handle rRNA genes but that also integrates special processing steps to handle ITS spacers as well as filter out obvious pseudogenes in protein coding markers such as COI. For example, we have a pseudogene filtering protocol for filtering out COI pseudogenes based on HMM profile analysis. This is made

possible due to the large amount of high-quality COI reference data available from the BOLD data releases [18]. Where a comprehensive hmm profile is lacking, we provide a more general pseudogene filtering pipeline based on simple translation, retention of longest ORFs, and removal of ORFs with outlier lengths.

There has been a lot of activity with respect to building new bioinformatic tools to handle COI metabarcodes. Recent work, such as the BOLDigger program, has attempted to make the BOLD identification engine more suitable for identifying large batches of COI metabaracodes, but submission size is limited to 100 queries at a time so as not to exceed the limits set by the BOLD server [55]. Unfortunately, point-and-click type interfaces are not always easy to integrate into custom pipelines, especially when dealing with large batches of sequences from high throughput sequencing platforms, so a command-line interface would be welcome. Another python package called ‘Alfie’ calculates k-mer frequencies and classifies COI metabarcode sequences to the kingdom rank using a machine learning method [56]. A new program, called NUMTdumper, has been developed as a stand-alone program meant to be incorporated into bioinformatic pipelines [57]. NUMTdumper provides a method to screen for NuMTs based on read counts while acknowledging the trade-offs between removing all possible NuMTs while erroneously removing genuine reads. An R package called ‘coil’ has also recently been developed that will place COI barcode and metabarcode sequences in frame using profile HMM analysis [58]. In MetaWorks, we incorporate two traditional approaches to identifying and removing obvious pseudogenes, one based on nucleotide to open reading frame translation, and a second based on translation combined with

HMM profile analysis where an HMM profile is available, and we incorporate these steps into the bioinformatic pipeline.

Our work on database curation, taxonomic assignment methods, and pipelines continues. MetaWorks is currently used in the national biomonitoring project STREAM (https://stream-dna.com/) and in various other research projects where we advance the pipeline and its components regularly as needed. Updates to the underlying RDP-trained classifiers described here are added to GitHub in an ongoing basis. New versions of MetaWorks will be released when the underlying software packages make major changes or when an existing reference database is updated or a new reference database or hmm profile is created for pseudogene filtering.

Conclusions

MetaWorks is a fusion of several different marker-specific pipelines that we have developed over the years. As far as we are aware, MetaWorks is the first multi-marker metabarcode pipeline that not only handles rRNA genes but implements the steps to rigorously analyze ITS spacers and protein coding genes. In large-scale studies that use multiple metabarcode markers to target taxa across multiple domains of life, MetaWorks can be used to provide a harmonized computational environment, pipeline, and rigorous taxonomic assignment method. This flexible pipeline is open-source and can be modified to support additional metabarcode markers, RDP trained reference databases, or hmm profiles for pseudogene filtering.

Availability and implementation

The MetaWorks pipeline is available on GitHub at https://github.com/terrimporter/MetaWorks as are the RDP classifier-formatted reference sets and underlying FASTA files presented in Table 1 at https://github.com/terrimporter .

Acknowledgements

References

1. Pace NR. A Molecular View of Microbial Diversity and the Biosphere. Science. 1997;276:734–40.

2. Hajibabaei M. The golden age of DNA metasystematics. Trends in genetics. 2012;28:535–537.

3. Taberlet P, Coissac E, Pompanon F, Brochmann C, Willerslev E. Towards next-generation biodiversity assessment using DNA metabarcoding. Molecular ecology. 2012;21:2045–2050.

4. Gilbert JA, Jansson JK, Knight R. The Earth Microbiome project: successes and aspirations. BMC biology. 2014;12:69.

5. Wang Q, Garrity GM, Tiedje JM, Cole JR. Naive Bayesian Classifier for Rapid Assignment of rRNA Sequences into the New Bacterial Taxonomy. Applied and Environmental Microbiology. 2007;73:5261–7.

6. CBOL Plant Working Group, Hollingsworth PM, Forrest LL, Spouge JL, Hajibabaei M, Ratnasingham S, et al. A DNA barcode for land plants. Proceedings of the National Academy of Sciences. 2009;106:12794–12797.

7. Hebert PDN, Cywinska A, Ball SL, deWaard JR. Biological identifications through DNA barcodes. Proceedings of the Royal Society B: Biological Sciences. 2003;270:313–21.

8. Schoch CL, Seifert KA, Huhndorf S, Robert V, Spouge JL, Levesque CA, et al. Nuclear ribosomal internal transcribed spacer (ITS) region as a universal DNA barcode marker for Fungi. Proceedings of the National Academy of Sciences. 2012;109:6241–6.

9. Bruns TD, White TJ, Taylor JW. Fungal Molecular Systematics. Annual Review of Ecology and Systematics. 1991;22:525–64.

10. Schüßler A. Glomales SSUrRNA gene diversity. New Phytologist. 1999;144:205–7.

11. James TY, Kauff F, Schoch CL, Matheny PB, Hofstetter V, Cox CJ, et al. Reconstructing the early evolution of Fungi using a six-gene phylogeny. Nature. 2006;443:818–22.

12. Hibbett DS, Binder M, Bischoff JF, Blackwell M, Cannon PF, Eriksson OE, et al. A higher-level phylogenetic classification of the Fungi. Mycological Research. 2007;111:509–47.

13. Zimmermann J, Abarca N, Enk N, Skibbe O, Kusber W-H, Jahn R. Taxonomic Reference Libraries for Environmental Barcoding: A Best Practice Example from Diatom Research. PLoS ONE. 2014;9:e108793.

14. Sato Y, Miya M, Fukunaga T, Sado T, Iwasaki W. MitoFish and MiFish Pipeline: A Mitochondrial Genome Database of Fish with an Analysis Pipeline for Environmental DNA Metabarcoding. Molecular Biology and Evolution. 2018;35:1553–5.

15. Ahmed M, Back MA, Prior T, Karssen G, Lawson R, Adams I, et al. Metabarcoding of soil nematodes: the importance of taxonomic coverage and availability of reference sequences in choosing suitable marker(s). MBMG. 2019;3:e36408.

16. DeSantis TZ, Hugenholtz P, Larsen N, Rojas M, Brodie EL, Keller K, et al. Greengenes, a Chimera-Checked 16S rRNA Gene Database and Workbench Compatible with ARB. Applied and Environmental Microbiology. 2006;72:5069–72.

17. Pruesse E, Quast C, Knittel K, Fuchs BM, Ludwig W, Peplies J, et al. SILVA: a comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB. Nucleic Acids Research. 2007;35:7188–96.

18. Ratnasingham S, Hebert PD. BOLD: The Barcode of Life Data System (http://www. barcodinglife. org). Molecular ecology notes. 2007;7:355–364.

19. Abarenkov K, Nilsson RH, Larsson K-H, Alexander IJ, Eberhardt U, Erland S, et al. The UNITE database for molecular identification of fungi – recent updates and future perspectives. New Phytologist. 2010;186:281–285.

20. Cole JR, Wang Q, Fish JA, Chai B, McGarrell DM, Sun Y, et al. Ribosomal Database Project: data and tools for high throughput rRNA analysis. Nucleic Acids Research. 2014;42:D633–42.

21. Ankenbrand MJ, Keller A, Wolf M, Schultz J, Förster F. ITS2 Database V: Twice as Much. Molecular Biology and Evolution. 2015;32:3030–2.

22. Rimet F, Chaumeil P, Keck F, Kermarrec L, Vasselon V, Kahlert M, et al. R-Syst::diatom: a barcode database for diatoms and freshwater biomonitoring - data sources and curation procedure. INRA Report. 2015.

23. Schloss PD, Westcott SL, Ryabin T, Hall JR, Hartmann M, Hollister EB, et al. Introducing mothur: Open-Source, Platform-Independent, Community-Supported Software for Describing and Comparing Microbial Communities. Applied and Environmental Microbiology. 2009;75:7537–41.

24. Caporaso JG, Kuczynski J, Stombaugh J, Bittinger K, Bushman FD, Costello EK, et al. QIIME allows analysis of highthroughput community sequencing data. Nature Methods. 2010;7:335–6.

25. Bolyen E, Rideout JR, Dillon MR, Bokulich NA, Abnet C, Al-Ghalith GA, et al. Reproducible, interactive, scalable, and extensible microbiome data science using QIIME 2. Nature Biotechnology. 2019;37:852–7.

26. Liu K-L, Porras-Alfaro A, Kuske CR, Eichorst SA, Xie G. Accurate, Rapid Taxonomic Classification of Fungal Large-Subunit rRNA Genes. Appl Environ Microbiol. 2012;78:1523–33.

442 27. Deshpande V, Wang Q, Greenfield P, Charleston M, Porras-Alfaro A, Kuske CR, et al. Fungal
443 identification using a Bayesian classifier and the Warcup training set of internal transcribed
444 spacer sequences. Mycologia. 2016;108:1–5.

445 28. Rivers AR, Weber KC, Gardner TG, Liu S, Armstrong SD. ITSxpress: Software to rapidly trim
446 internally transcribed spacer sequences with quality scores for marker gene analysis. F1000Res.
447 2018;7:1418.

448 29. Porter TM, Gibson JF, Shokralla S, Baird DJ, Golding GB, Hajibabaei M. Rapid and accurate
449 taxonomic classification of insect (class Insecta) cytochrome c oxidase subunit 1 (COI) DNA
450 barcode sequences using a naïve Bayesian classifier. Mol Ecol Resour. 2014;14:929–42.

451 30. Porter TM, Hajibabaei M. Automated high throughput animal CO1 metabarcode
452 classification. Scientific Reports. 2018;8:4226.

453 31. Virgilio M, Backeljau T, Nevado B, De Meyer M. Comparative performances of DNA
454 barcoding across insect orders. BMC bioinformatics. 2010;11:206.

455 32. Maitland VC, Robinson CV, Porter TM, Hajibabaei M. Freshwater diatom biomonitoring
456 through benthic kick-net metabarcoding. BioRxiv.
457 2020;:http://biorxiv.org/lookup/doi/10.1101/2020.05.25.115089.

458 33. Song H, Buhay JE, Whiting MF, Crandall KA. Many species in one: DNA barcoding
459 overestimates the number of species when nuclear mitochondrial pseudogenes are
460 coamplified. PNAS. 2008;105:13486–91.

461 34. Moulton MJ, Song H, Whiting MF. Assessing the effects of primer specificity on eliminating
462 numt coamplification in DNA barcoding: a case study from Orthoptera (Arthropoda: Insecta):
463 DNA BARCODING. Molecular Ecology Resources. 2010;10:615–27.

464 35. Baird DJ, Hajibabaei M. Biomonitoring 2.0: a new paradigm in ecosystem assessment made
465 possible by next-generation DNA sequencing. Molecular ecology. 2012;21:2039–2044.

466 36. Drummond AJ, Newcomb RD, Buckley TR, Xie D, Dopheide A, Potter BC, et al. Evaluating a
467 multigene environmental DNA approach for biodiversity assessment. GigaSci. 2015;4:46.

468 37. Adamowicz SJ, Boatwright JS, Chain F, Fisher BL, Hogg ID, Leese F, et al. Trends in DNA
469 barcoding and metabarcoding. Genome. 2019;62:v–viii.

470 38. Bengtsson-Palme J, Ryberg M, Hartmann M, Branco S, Wang Z, Godhe A, et al. Improved
471 software detection and extraction of ITS1 and ITS2 from ribosomal ITS sequences of fungi and
472 other eukaryotes for analysis of environmental sequencing data. Methods in Ecology and
473 Evolution. 2013;4:914–9.

474 39. Anaconda. Anaconda Software Distribution. 2016. https://anaconda.com.

475 40. Koster J, Rahmann S. Snakemake--a scalable bioinformatics workflow engine.
476 Bioinformatics. 2012;28:2520–2.

477 41. St. John J. SeqPrep. 2016. https://github.com/jstjohn/SeqPrep/releases.

478 42. Martin M. Cutadapt removes adapter sequences from high-throughput sequencing reads.
479 EMBnet journal. 2011;17:pp–10.

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

482 44. Edgar RC. Search and clustering orders of magnitude faster than BLAST. Bioinformatics.
483 2010;26:2460–1.

484 45. Edgar RC. UNOISE2: improved error-correction for Illumina 16S and ITS amplicon
485 sequencing. bioRxiv. 2016. doi:10.1101/081257.

486 46. Edgar R. UCHIME2: improved chimera prediction for amplicon sequencing. bioRxiv.
487 2016;:074252.

488 47. Callahan BJ, McMurdie PJ, Holmes SP. Exact sequence variants should replace operational
489 taxonomic units in marker-gene data analysis. The ISME Journal. 2017;11:2639–43.

490 48. China Plant BOL Group, Li D-Z, Gao L-M, Li H-T, Wang H, Ge X-J, et al. Comparative analysis
491 of a large dataset indicates that internal transcribed spacer (ITS) should be incorporated into
492 the core barcode for seed plants. Proceedings of the National Academy of Sciences.
493 2011;108:19641–6.

494 49. Hollingsworth PM. Refining the DNA barcode for land plants. Proceedings of the National
495 Academy of Sciences. 2011;108:19451–2.

496 50. Edge TA, Baird DJ, Bilodeau G, Gagné N, Greer C, Konkin D, et al. The Ecobiomics project:
497 Advancing metagenomics assessment of soil health and freshwater quality in Canada. Science
498 of The Total Environment. 2020;710:135906.

499 51. Cochrane G, Karsch-Mizrachi I, Takagi T, Sequence Database Collaboration IN. The
500 International Nucleotide Sequence Database Collaboration. Nucleic Acids Research.
501 2016;44:D48–50.

502 52. Iwasaki W, Fukunaga T, Isagozawa R, Yamada K, Maeda Y, Satoh TP, et al. MitoFish and
503 MitoAnnotator: A Mitochondrial Genome Database of Fish with an Accurate and Automatic
504 Annotation Pipeline. Molecular Biology and Evolution. 2013;30:2531–40.

505 53. Wheeler DL. Database resources of the National Center for Biotechnology. Nucleic Acids
506 Research. 2003;31:28–33.

54. R Core Team. R: A Language and Environment for Statistical Computing. 2017. https://www.R-project.org/.

55. Buchner D, Leese F. BOLDigger – a Python package to identify and organise sequences with the Barcode of Life Data systems. MBMG. 2020;4:e53535.

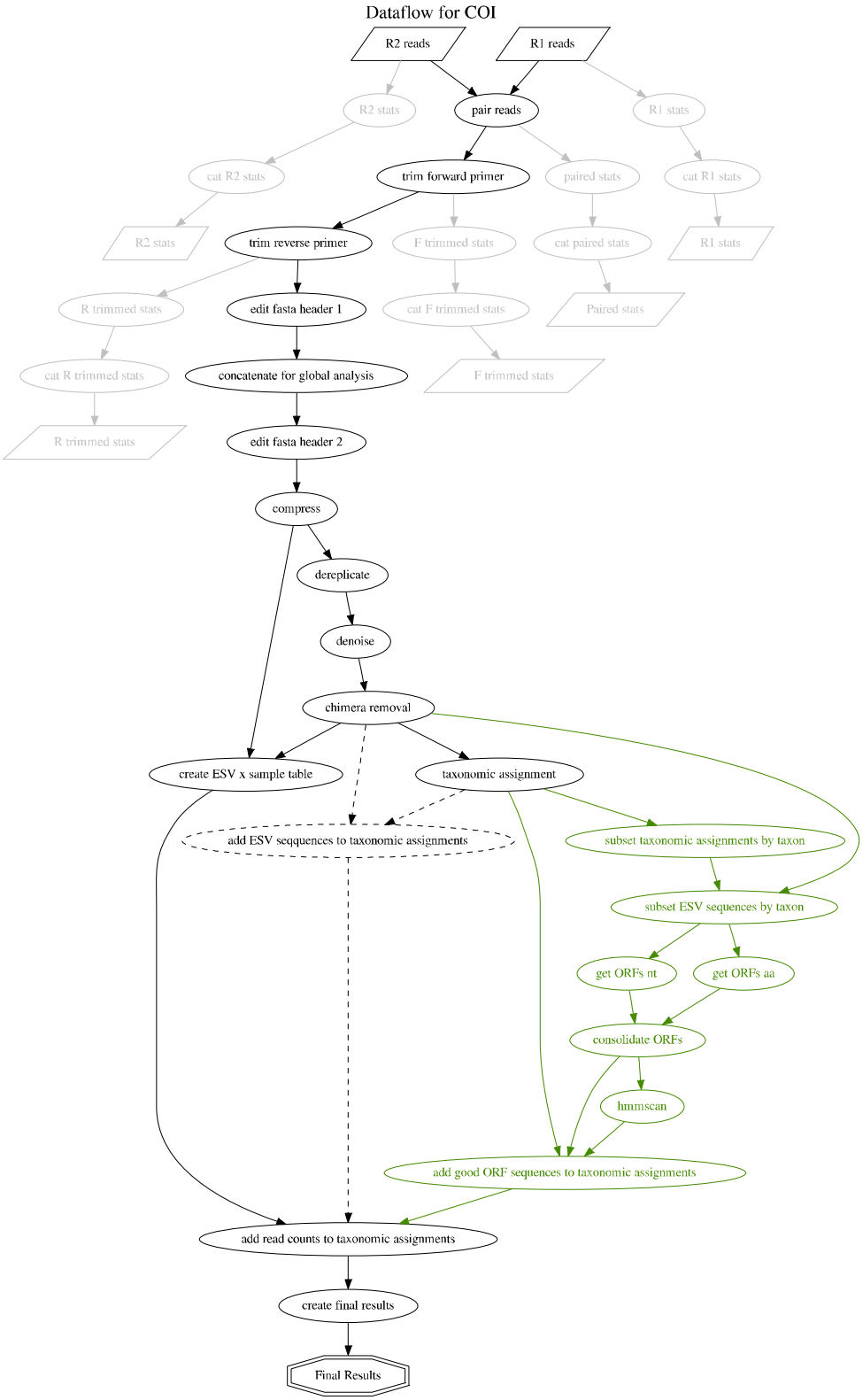
56. Nugent CM, Adamowicz SJ. Alignment-free identification of COI DNA barcode data with the Python package Alfie. preprint. Bioinformatics; 2020. doi:10.1101/2020.06.29.177634.

57. Andújar C, Creedy TJ, Arribas P, López H, Salces-Castellano A, Pérez-Delgado A, et al. NUMT dumping: validated removal of nuclear pseudogenes from mitochondrial metabarcode data. preprint. Evolutionary Biology; 2020. doi:10.1101/2020.06.17.157347.

58. Nugent CM, Elliott TA, Ratnasingham S, Adamowicz SJ. coil: an R package for cytochrome C oxidase I (COI) DNA barcode data cleaning, translation, and error evaluation. bioRxiv. 2019;:35.

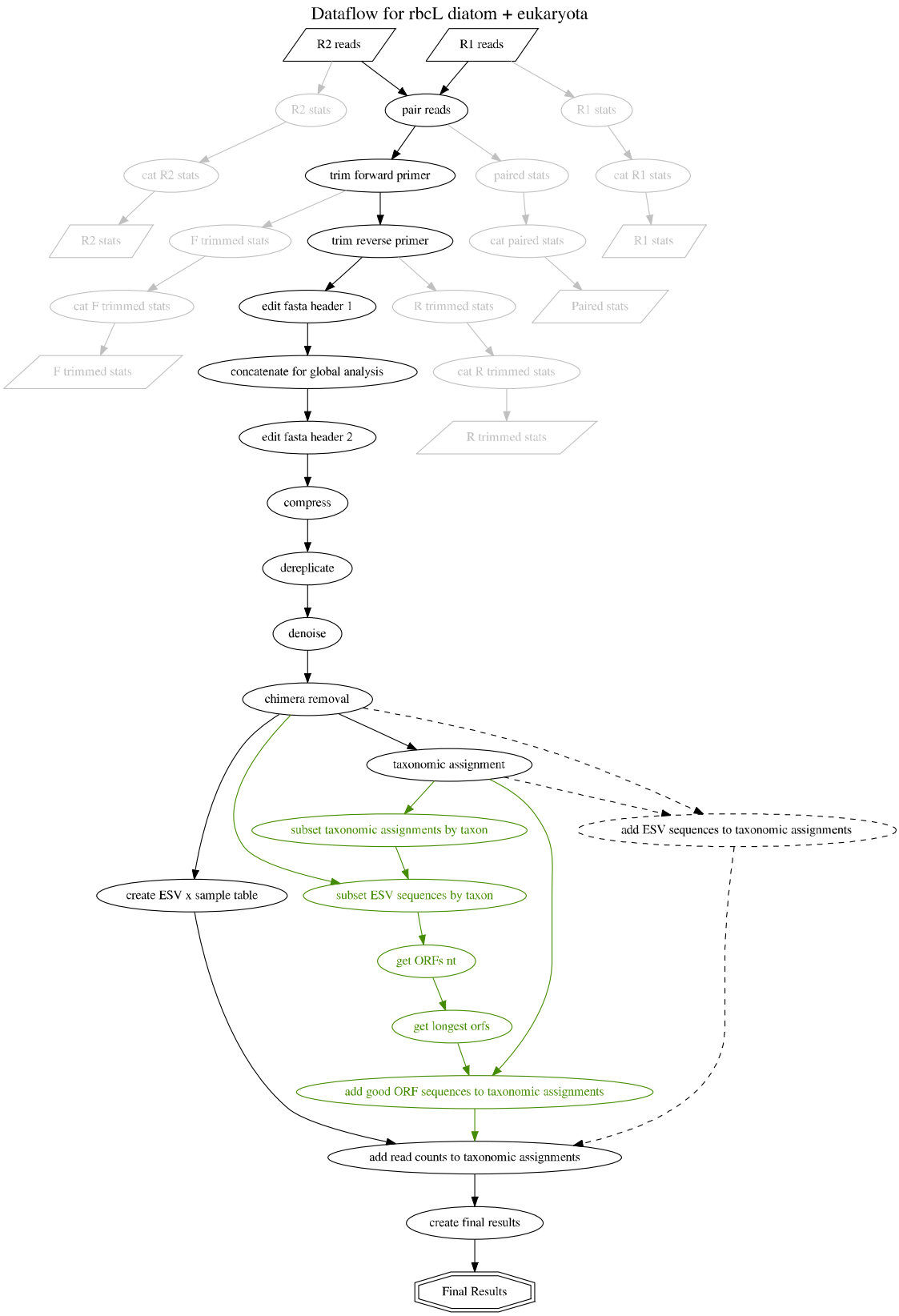
Supplementary Material

Figure S1. Dataflow for COI mtDNA metabarcodes. Input and output files are shown as parallelograms. Snakemake rules or processes are shown as ovals. The final results file contains ESVs, for each sample, as well as ESV/ORF sequences, read counts, as well as taxonomic assignments with bootstrap support values. The main dataflow is shown in black, if pseudogene filtering is selected these steps are shown in green, if pseudogene filtering is not selected the dashed steps are performed. The generation of various statistical reports are shown in grey.


Dataflow for COI
R2 reads
R1 reads
R2 stats
pair reads
R1 stats
cat R2 stats
trim forward primer
paired stats
cat R1 stats
R2 stats
trim reverse primer
F trimmed stats
cat paired stats
R1 stats
R trimmed stats
edit fasta header 1
cat F trimmed stats
Paired stats
cat R trimmed stats
concatenate for global analysis
F trimmed stats
R trimmed stats
edit fasta header 2
compress
dereplicate
denoise
chimera removal
create ESV x sample table
taxonomic assignment
add ESV seqquences to taxonomic assignments
subset taxonomic assignments by taxon
subset ESV sequences by taxon
get ORFs nt
get ORFs aa
consolidate ORFs
hmmscan
add good ORF sequences to taxonomic assignments
add read counts to taxonomic assignments
create final results
Final Results

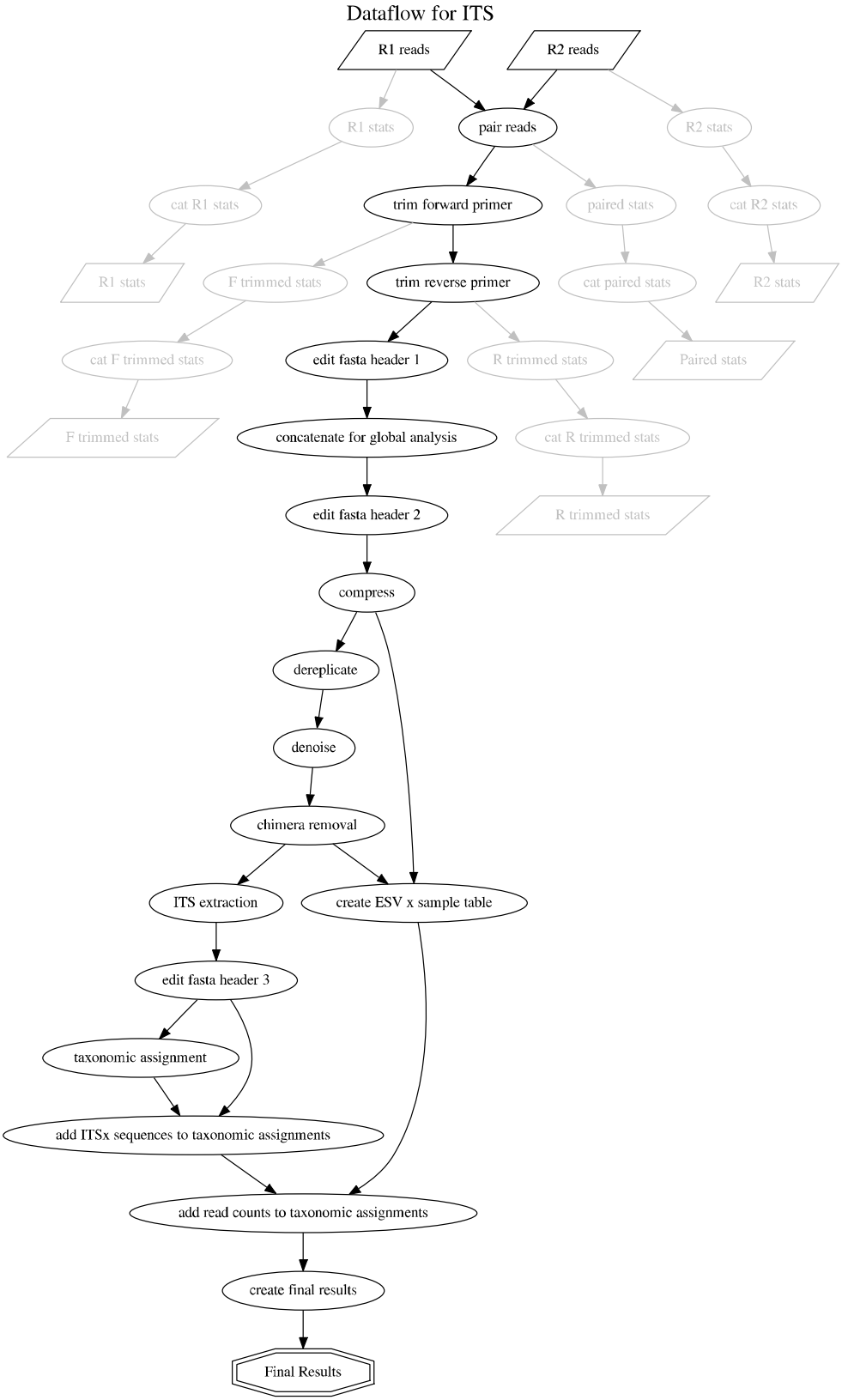
530

Figure S2. Dataflow for rbcL mtDNA metabarcodes. Input and output files are shown as parallelograms. Snakemake rules or processes are shown as ovals. The final results file contains ESVs, for each sample, as well as ESV/ORF sequences, read counts, as well as taxonomic assignments with bootstrap support values. The main dataflow is shown in black, if pseudogene filtering is selected these steps are shown in green, if pseudogene filtering is not selected the dashed steps are performed. The generation of various statistical reports are shown in grey. The data flow is essentially the same whether the rbcL eukaryota or diatom-specific classifiers are used.


Dataflow for rbcL diatom + eukaryota
R2 reads
R1 reads
R2 stats
pair reads
R1 stats
cat R2 stats
trim forward primer
paired stats
cat R1 stats
R2 stats
F trimmed stats
trim reverse primer
cat paired stats
R1 stats
cat F trimmed stats
edit fasta header 1
R trimmed stats
Paired stats
F trimmed stats
concatenate for global analysis
cat R trimmed stats
edit fasta header 2
R trimmed stats
compress
dereplicate
denoise
chimera removal
taxonomic assignment
subset taxonomic assignments by taxon
add ESV sequences to taxonomic assignments
create ESV x sample table
subset ESV sequences by taxon
get ORFs nt
get longest orfs
add good ORF sequences to taxonomic assignments
add read counts to taxonomic assignments
create final results
Final Results

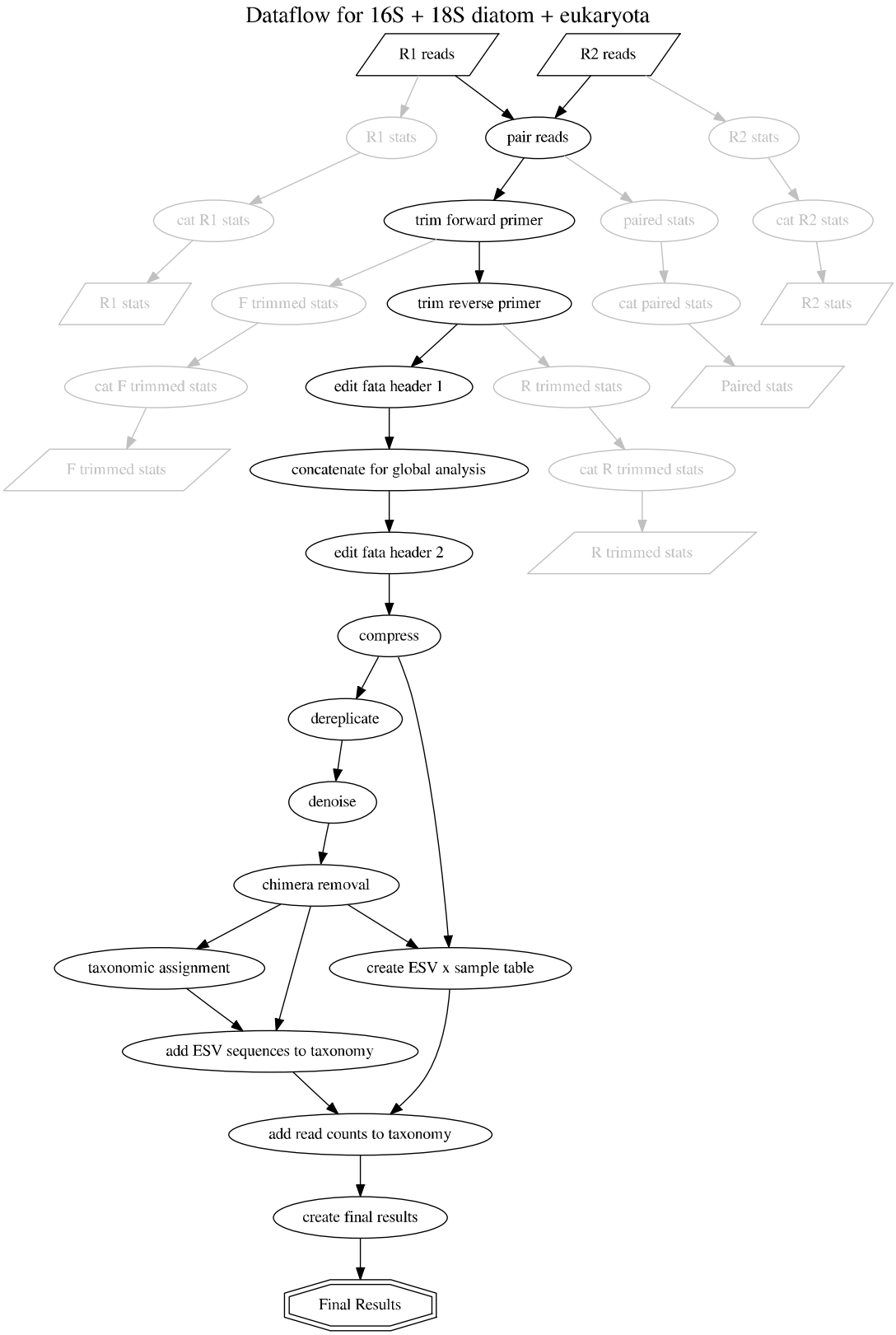
539

Figure S3. Dataflow for ITS metabarcodes. Input and output files are shown as parallelograms. Snakemake rules or processes are shown as ovals. The final results file contains ESVs, for each sample, as well as ESV/ORF sequences, read counts, as well as taxonomic assignments with bootstrap support values. The generation of various statistical reports are shown in grey.


Dataflow for ITS
R1 reads
R2 reads
R1 stats
pair reads
R2 stats
cat R1 stats
trim forward primer
paired stats
cat R2 stats
R1 stats
F trimmed stats
trim reverse primer
cat paired stats
R2 stats
cat F trimmed stats
edit fasta header 1
R trimmed stats
Paired stats
F trimmed stats
concatenate for global analysis
cat R trimmed stats
edit fasta header 2
R trimmed stats
compress
dereplicate
denoise
chimera removal
ITS extraction
create ESV x sample table
edit fasta header 3
taxonomic assignment
add ITSx sequences to taxonomic assignments
add read counts to taxonomic assignments
create final results
Final Results

545

Figure S3. Dataflow for rRNA gene metabarcodes. Input and output files are shown as parallelograms. Snakemake rules or processes are shown as ovals. The final results file contains ESVs, for each sample, as well as ESV/ORF sequences, read counts, as well as taxonomic assignments with bootstrap support values. The generation of various statistical reports are shown in grey. The dataflow is essentially the same whether the RDP classifier built-in 16S reference set is used, or the custom-trained 18S eukaryote or diatom-specific classifiers are used.


Dataflow for 16S + 18S diatom + eukaryota
R1 reads
R2 reads
R1 stats
pair reads
R2 stats
cat R1 stats
trim forward primer
paired stats
cat R2 stats
R1 stats
F trimmed stats
trim reverse primer
cat paired stats
R2 stats
cat F trimmed stats
edit fata header 1
R trimmed stats
Paired stats
F trimmed stats
concatenate for global analysis
cat R trimmed stats
edit fata header 2
R trimmed stats
compress
dereplicate
denoise
chimera removal
taxonomic assignment
create ESV x sample table
add ESV sequences to taxonomy
add read counts to taxonomy
create final results
Final Results

553