Received: 19 September 2023 | Accepted: 12 August 2024

DOI: 10.1111/2041-210X.14434

Methods in Ecology and Evolution

APPLICATION

SimpleMetaPipeline: Breaking the bioinformatics bottleneck in metabarcoding

Jake Williams[1,2] | Nathalie Pettorelli[2] | Rosalie Dowell[1,2] | Kenneth Macdonald[3] | Christopher Meyer[3] | Margaux Steyaert[1,2] | Sarah Tweedt[3] | Emma Ransome[1]

[1]Department of Life Sciences, Imperial College London, Ascot, UK

[2]Institute of Zoology, Zoological Society of London, London, UK

[3]National Museum of Natural History, Smithsonian Institution, Washington, District of Columbia, USA

Correspondence
Jake Williams
Email: jakewilliams844@gmail.com

Funding information
Natural Environment Research Council, Grant/Award Number: NE/R012229/1

Handling Editor: Andrew Mahon

Abstract

1. The democratisation of next-generation sequencing has vastly increased the availability of sequencing data from metabarcoding. However, to effectively prepare these metabarcoding data for subsequent analysis, researchers must consistently apply several different bioinformatic tools—including those which denoise reads, cluster sequences and assign taxonomic identities. This often creates a bioinformatics bottleneck in workflows for non-specialists due to obstacles around: (a) integrating different tools, (b) the inability to easily modify and rerun bioinformatic pipelines involving non-scripted ('point-and-click') elements and (c) the multiple outputs that may be required of a single dataset (e.g. amplicon sequence variants [ASVs] and operational taxonomic units [OTUs]), which often results in users running pipelines multiple times.
2. Here, we introduce SimpleMetaPipeline, an open-source bioinformatics pipeline implemented in R, which addresses these obstacles. SimpleMetaPipeline integrates the most robust and commonly used existing bioinformatic tools in a single reproducible pipeline, with a streamlined choice of parameters, to generate a sequence data table containing alternative clustering and assignment options. SimpleMetaPipeline accepts demultiplexed paired-end and single reads from multiple sequencing runs.
3. We describe the pipeline and demonstrate how alternative annotations enable the easy implementation of multi-algorithm agreement tests to strengthen inferences.
4. SimpleMetaPipeline represents a valuable addition to the existing library of pipelines, providing easy and reproducible bioinformatics, including a range of commonly desired clustering and assignment options, such as OTUs and ASVs.

KEYWORDS
amplicon sequence variants, bioinformatics pipeline, eDNA, metabarcoding, next-generation sequencing, R

1 | INTRODUCTION

There is a growing interest in applying next-generation sequencing to a wide range of ecological questions. Metabarcoding or marker gene amplicon sequencing can now rapidly deliver an in-depth and complementary perspective on ecological communities to that provided by traditional biomonitoring (Porath-Krause et al., 2022). The declining cost of these approaches has resulted in increasing adoption across ecological specialisms, thus generating vast amounts of raw sequencing data (Kodama et al., 2012). This includes published data, which can often be utilised to answer questions quite different from those the original authors intended if the data are published accessibly (Shea et al., 2023), and if it can be readily reanalysed.

However, there is a bottleneck for non-specialists using these approaches for high-throughput environmental monitoring at the bioinformatics step, which is required to convert raw sequencing data into annotated community matrices that can be used in analysis (Porath-Krause et al., 2022). This bioinformatics bottleneck is due to challenges non-specialists encounter in two related areas: (i) Ease-of-use (Bolyen et al., 2019), that is, the extent to which the integration of different tools with a variety of native formats is facilitated; and (ii) reproducibility (Sandve et al., 2013; Powers & Hampton, 2019; Wratten et al., 2021), in general, the ease and reliability with which one can re-generate identical results from raw data, and in this case, specifically the ability to easily modify and rerun bioinformatic pipelines using non-scripted ('point-and-click') elements.

Existing pipelines currently tend to trade-off ease-of-use against reproducibility. They either provide GUIs and other point-and-click solutions to increase users' accessibility (see e.g. mifish (Sato et al., 2018); q2galaxy (Bolyen et al., 2019) and APSCALE (Buchner et al., 2022)), thereby limiting reproducibility. Alternatively command line pipelines can enhance reproducibility but often require computing skills beyond those of the general user—for example, mothur (Schloss & Westcott, 2011), DADA2 (Callahan et al., 2016), other QIIME2 interfaces (Bolyen et al., 2019) or stitching a bespoke combination of tools together in a bash script. It should be noted that in the case of QIIME2 extensive documentation and an active user community and forum provide an excellent learning opportunity for new users.

To our knowledge, none of the existing pipelines enable the easy and efficient generation of alternative sequence annotations (i.e. annotations which provide alternative answers to the same 'question', such as alternative sequence clusters or alternative taxonomic assignments). Herein we define annotations as any information generated about a sequence, including with which other sequences from the dataset they form clusters, and any taxa to which they can be assigned. Examples of such alternative annotations include the concurrent generation of both amplicon sequence Variants (ASVs, also known as exact sequence variants [ESVs]), and operational taxonomic units (OTUs) or taxonomic assignments from multiple assignment algorithms. Alternative annotations are important as it is now common practice for metabarcoding studies to present results for both ASVs and OTUs as a way to explore the influence of taxonomic resolution on their results (Antich et al., 2021). Furthermore, the taxonomic assignment of sequences is a source of uncertainty in metabarcoding studies as all methods have their strengths and weaknesses (Hleap et al., 2021), and comparing the assignments from multiple assignment algorithms is one way to address this. This need for alternative annotations can exacerbate the bottleneck challenges if they are produced by running raw data through bioinformatic pipelines multiple times. Even slight differences, accidentally introduced, could make results incomparable. This problem is avoidable if identical commands are run within identical computing environments with identical random seeds (necessary if algorithms have probabilistic components as many bioinformatic tools do). But achieving this manually requires computational knowledge, can be time consuming, and is subject to user error that can be extremely difficult to trace (Grüning et al., 2018; Mangul et al., 2019).

Here, we present SimpleMetaPipeline, an easy-to-use, entirely scripted bioinformatics pipeline producing alternative annotations. It is open-source, implemented in R and combines well-established, existing and peer-reviewed bioinformatics tools. Implementing the pipeline in R helps make the source code more accessible to users, given the widespread use of R in ecology, and is appropriate given that multiple bioinformatic tools are native to R (DADA2, LULU and IDTAXA). Scripted pipelines in R are highly shareable, maintainable and reusable provided good scientific work flow practices are followed (Djaffardjy et al., 2023). SimpleMetaPipeline requires a single short R script, defining all parameters, to be run alongside a correctly formatted directory of raw fastq files, including as many Illumina sequencing runs as desired. From this, the pipeline applies existing bioinformatic tools (e.g. DADA2) to reproducibly generate a sequence data table containing denoised ASVs as rows, and columns containing all alternative clustering and assignment annotations.

SimpleMetaPipeline is novel in two important ways. First, it is clear, simple and easy to use, requiring only a single R script to be run, and has guaranteed reproducibility from this single R script, where other pipelines focus on either ease-of-use or reproducibility. Second, it utilises an underlying sequence data table structure to efficiently handle alternative annotations. Specifically, SimpleMetaPipeline retains all bioinformatic annotations produced in an accessible form in the output. This has the added benefit of enabling testing for agreement between the alternative annotations of multiple algorithms, providing new opportunities to improve inferences from next-generation sequencing data.

2 | OVERVIEW AND WORKFLOW

SimpleMetaPipeline integrates bioinformatics tools to trim, denoise, cluster and taxonomically assign raw, demultiplexed, input amplicon datasets from multiple Illumina sequencing runs. These tools include: Cutadapt v3.5 (trimming; Martin, 2011), DADA2 v1.24.0 (denoising; Callahan et al., 2016), VSEARCH v.2.4.1 (clustering; Rognes et al., 2016), Swarm v3.1 (clustering; Mahé et al., 2015), LULU v0.1.0 (clustering; Frøslev et al., 2017), DECIPHER v2.24.0 (taxonomy assignment with the IDTAXA function; Murali et al., 2018) and BLAST v.2.9.0–2 (taxonomy assignment; Altschul et al., 1990). Please see the Supporting Information 'SimpleMetaPipeline Algorithms and Parameters' for full details of parameter choices.

Briefly, the pipeline starts by using DADA2's robust error estimation to generate a reliable list of all ASVs present and their frequencies across samples (Callahan et al., 2016). All subsequent tools in the pipeline are then applied to these ASVs, and their standard outputs are captured. First, LULU is used to annotate each ASV with the 'curated ASV' to which it belongs (Frøslev et al., 2017). LULU curation uses sequence similarity and distribution to cluster sequences together, these clusters are thus sometimes referred to as 'distribution-based OTUs' (Frøslev et al., 2017). Second, either VSEARCH or Swarm (according to user-specified preference) is used to annotate each ASV with the OTU to which it belongs (note that these are similarity-based OTUs specifically; Mahé et al., 2015; Rognes et al., 2016); then (in the only step not applying directly to ASVs) LULU is applied to 'curate' these OTUs, and each ASV is then annotated with the 'curated OTU' to which it belongs (Frøslev et al., 2017). Even in this case information is recorded for each ASV independently. Thus, there are always three types of clusters produced by SimpleMetaPipeline, depending on the option chosen these will either be LULU, VSEARCH and VSEARCH+LULU; or LULU, Swarm and Swarm+LULU. (SimpleMetaPipeline is not designed to compare VSEARCH and Swarm clusters within a single pipeline run).

Finally, if desired, the pipeline will assign taxonomy to ASVs. IDTAXA can be used to annotate each ASV with a k-mer-based taxonomic assignment (Murali et al., 2018). BLAST can be used to annotate each ASV with a similarity-based taxonomic assignment (Altschul et al., 1990). This creates a range of information about each ASV, including both the assignments themselves and various metrics quantifying the degree of uncertainty associated with these assignments. We provide a workflow diagram to illustrate the input data required, steps in the pipeline and outputs (Figure 1).

Bioinformatic tools were chosen for their complementarity from among those that have been robustly tested, benchmarked and peer-reviewed. Crucially, no preference was given to tools based on their native format. Combining DADA2, VSEARCH, Swarm and LULU in a single pipeline provides all of the most commonly used complementary sequence and clustering annotations simultaneously (e.g. Antich et al., 2021; Brandt et al., 2021). These algorithms are complementary in the sense that they each use sequencing information in slightly different ways to estimate clusters. Briefly, VSEARCH uses a global sequence similarity threshold (Mahé et al., 2015); SWARM iteratively adds sequences to clusters using a small local similarity threshold and abundance information (Rognes et al., 2016); LULU combines a sequence similarity threshold with a co-occurrence threshold (Frøslev et al., 2017). As noted including these alternative annotations in the final output enables uncertainties associated with taxonomic resolution and choice of clustering algorithm(s) to be assessed in analysis without rerunning bioinformatics. IDTAXA and BLAST were combined as they determine taxonomic assignment of sequences in radically different, but widely accepted and well-justified ways, with BLAST tending to minimise under-classifications and IDTAXA minimising over-classifications (Altschul et al., 1990; Murali et al., 2018). Comparing the two assignments can thus increase the confidence in an assignment (if a conservative approach

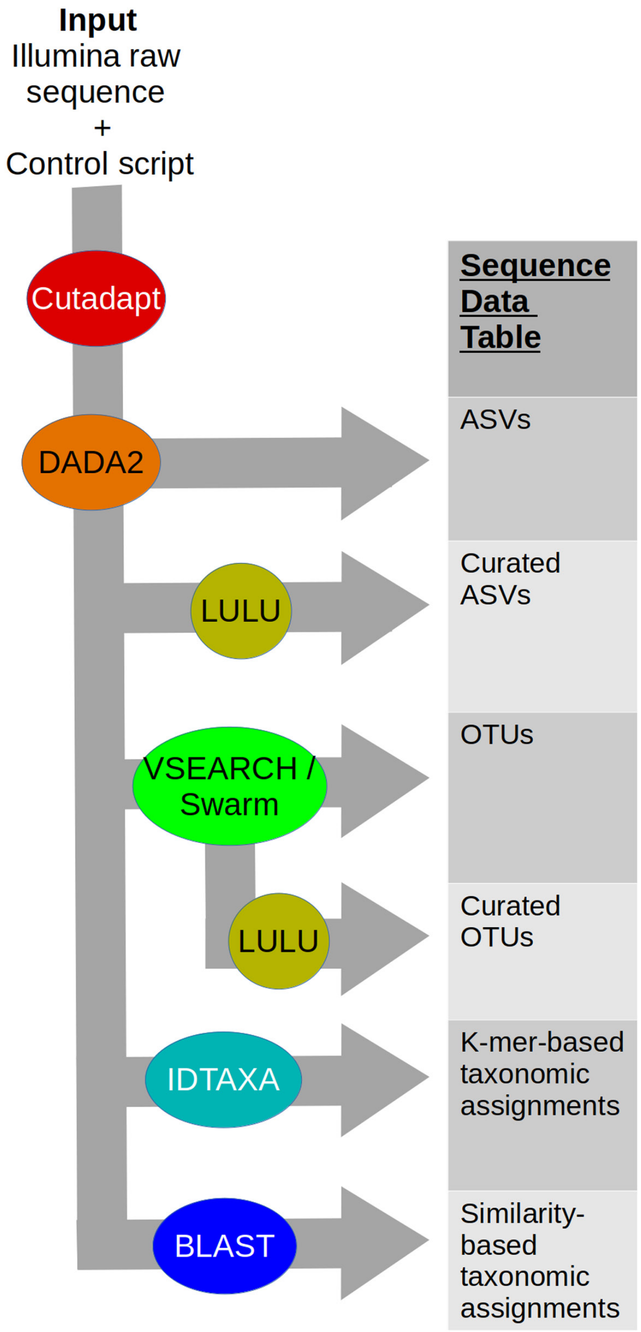


FIGURE 1 Diagram of the SimpleMetaPipeline workflow. Ovals represent the different steps in the pipeline and the order in which they occur—either in series or in parallel. The table on the right represents the format of the output 'Sequence Data Table' (as shown in Table 1) in simplified graphical form. Arrows indicate the step in the pipeline where each set of information in the Sequence Data Table is generated.

TABLE 1 An example of sequence data table format if both IDTAXA and BLAST assignment options are selected.

		Example row content				
Source of output	Column description	Example row 1	Example row 2	Example row 3	Example row 4	Example row 5
DADA2	ASV	ASV1	ASV2	ASV3	ASV4	ASV5
	Sequence	TACG…	ATTT…	GTAC…	CCTT…	AAAT…
	Sample 1	11	0	4	589	98
	…	…	…	…	…	…
	Sample n	34	55	0	0	7
LULU	Curated ASV	ASV1	ASV1	ASV2	ASV2	ASV2
	Curated ASV Representative Sequence	1	0	0	0	1
VSEARCH/Swarm	OTU	OTU1	OTU1	OTU1	OTU2	OTU2
	OTU Representative Sequence	1	0	0	0	1
VSEARCH/Swarm + LULU	Curated OTU	OTU1	OTU1	OTU1	OTU1	OTU1
	Curated OTU Representative Sequence	1	0	0	0	0
IDTAXA	Rank 1	Taxa1	Taxa2	Taxa1	Taxa3	Taxa3
	…	…	…	…	…	…
	Rank n	Taxa4	NA	Taxa5	Taxa6	Taxa7
	Rank 1 Confidence	100	43	78	81	83
	…	…	…	…	…	…
	Rank n Confidence	46	0	46	55	63
BLAST	Blast Percent Identical	98	77	89	88	92
	Blast evalue	0	0	0	0	0
	Blast Query Coverage	99	100	100	97	58
	Rank 1	Taxa1	Taxa2	Taxa1	Taxa3	Taxa3
	…	…	…	…	…	…
	Rank n	Taxa4	NA	Taxa5	Taxa6	Taxa7

Note: Note that this table is transposed to aid presentation. Column names, as output from the pipeline, are abbreviated and do not include spaces.

is taken where agreement between algorithms is required) or help understand the degree of uncertainty (e.g. by calculating the proportion of ASVs in a cluster which received the same assignment from both algorithms at a given rank).

The pipeline includes a conda environment definition that installs each of the tools mentioned previously, along with R version 4.2 (R Core Team, 2022) and the following R packages: SeqinR v4.2–16 (Charif & Lobry, 2007), ShortRead v1.54.0 (Morgan et al., 2009), gridExtra v.2.3 (Auguie & Antonov, 2017), ggplot2 v.3.4.0 (Wickham, 2011) and dplyr v1.1.1 (Wickham et al., 2023). SimpleMetaPipeline source code is available for UNIX/Linux and macOS environments and is archived on Zenodo (Williams et al., 2024a). The development version can be accessed on GitHub at https://github.com/J-Cos/SimpleMetaPipeline, where installation instructions are available.

A supporting R package is also provided, which can quickly and reproducibly generate a variety of standardised annotated community matrices from the alternative annotations stored in a sequence data table (e.g. matrices reflecting OTUs or ASVs, or including taxonomic assignments produced by IDTAXA or those produced by BLAST). This division of functionality between pipeline and package is thus crucial to enabling efficient handling of alternative annotations. Specifically, the package generates 'phyloseq objects', derived from the Phyloseq R package commonly used in the analysis of metabarcoding data (McMurdie & Holmes, 2013). The package source code is archived on Zenodo (Williams et al., 2024b), and the development version can be accessed on GitHub at https://github.com/J-Cos/SimpleMetaPackage, where installation instructions are available.

3 | INPUT DATA PREPARATION AND PARAMETER CHOICES

3.1 | Control scripts

SimpleMetaPipeline requires running a single R script, known as a control script. An example control script is provided in the codebase with sensible defaults (or guidance where a sensible default value is impossible) for all adjustable parameters. The example also includes detailed descriptions of what each adjustable parameter controls and links to underlying tools where applicable. Where parameters for underlying tools do not appear in the control script they are not adjustable and the default values are used.

3.2 | Demultiplexed fastq directory

SimpleMetaPipeline accepts demultiplexed paired-end or single read fastq or fastq.gz files, with each R1/R2 pair or single read file named by sample. These files can be generated from any marker gene amplicon, and SimpleMetaPipeline has been tested with COI gene, 18S rRNA gene, 16S rRNA gene, ITS rRNA gene, 23s rRNA gene and 12s rRNA gene marker datasets. The fastq files from each Illumina sequencing run should be stored in separate directories. This is important as it allows DADA2 denoising to learn error rates for each sequencing run independently (Callahan et al., 2016). In some cases, samples may appear multiple times across a batch of sequencing runs (as commonly occurs in multi-run experiments to address low quality or failed sequencing of certain samples). SimpleMetaPipeline can handle this scenario as a unique sequencing run identifier is automatically appended to each sample name, allowing decisions about how to handle these duplicates to be made downstream, without needing to rerun bioinformatics.

3.3 | Taxonomic assignment

An appropriate IDTAXA classifier and/or BLAST database, generated from any reference library one wishes to use, will need to be provided alongside the fastq files if sequence classification is required. Details of how to generate IDTAXA classifiers and BLAST databases are provided by each of these tools respectively (Altschul et al., 1990; Murali et al., 2018).

4 | OUTPUTS

4.1 | Sequence data table

SimpleMetaPipeline outputs a sequence data table with ASVs as rows and information on each ASV generated by the pipeline as columns. Columns contain ASV annotations themselves—for example OTU2 or *Taxa3*—and useful information about these annotations (Table 1). This information includes a variety of assignment certainty measures provided by the underlying algorithms: sequence similarity and e-value from the BLAST algorithm and assignment confidences from the IDTAXA algorithm, as well as TRUE/FALSE values showing whether ASVs were identified as representative sequences of their clusters. An example of pipeline output is included in the Supporting Information.

4.2 | Diagnostic outputs

SimpleMetaPipeline generates additional outputs that enable the inspection of performance of different steps in the pipeline. These diagnostic outputs include a set of tables displaying: (1) a count of all primer sequences removed by cutadapt; (2) the number of dereplicated sequences in each sample at each DADA2 step (input, filtering, denoising, merging and chimera removal); (3) the distribution of ASV lengths (number of bases) and (4) the number of clusters produced under each clustering approach. Further, standard diagnostic figures are provided from DADA2 (quality profiles and error plots) and IDTAXA (taxonomic assignment plot).

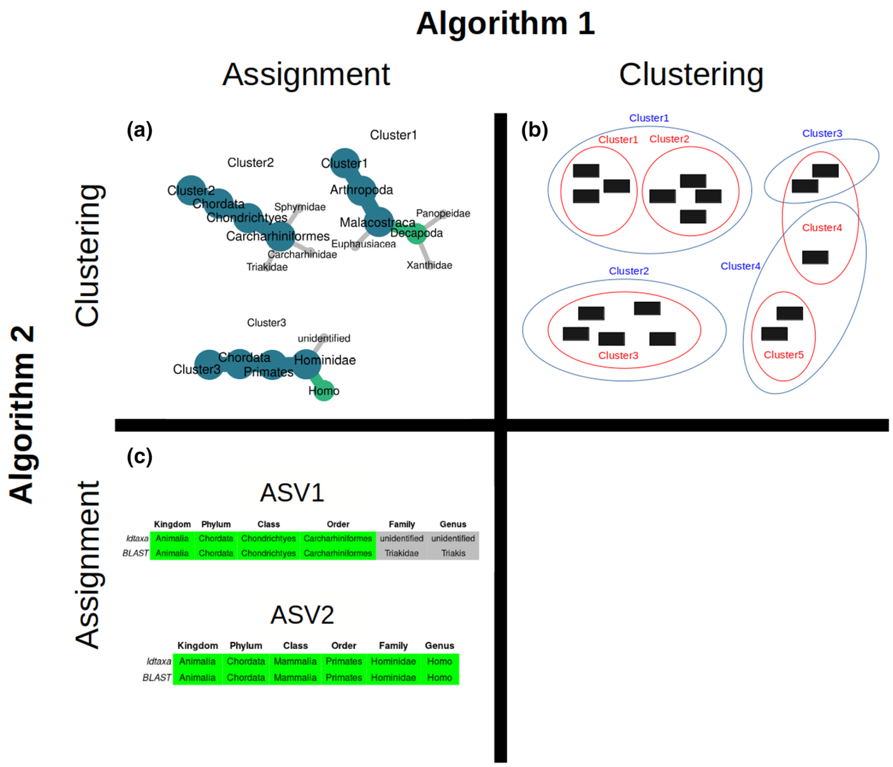
5 | EXAMPLES AND BENCHMARKING

Sequence data tables, as output by SimpleMetaPipeline, enable easy comparison between clustering and assignment methods. This allows testing for multi-algorithm agreement to better understand uncertainties in annotations. Such tests can be conducted for agreement between (1) clustering algorithms, (2) assignment algorithms and (3) clustering and assignment algorithms (Figure 2). The concept of multi-algorithm agreement tests is that the different annotations given to ASVs by the robust and widely used, yet methodologically distinct, algorithms deployed in SimpleMetaPipeline each contain information about the biology of the ASV.

In the case of two clustering algorithms, there is no straightforward rule that can be applied to require agreement. However, the variation between clustering algorithms can be used to interrogate clusters of interest to understand their potential relationship to other clusters and internal sequence diversity. In the case of two assignment algorithms, SimpleMetaPackage enables the application of the conservative rule of, for each sequence at each taxonomic rank, only accepting a taxonomic assignment agreed upon by both algorithms. In the case of agreement between clustering and assignment algorithms (e.g. testing whether all sequences in a cluster receive the same assignment), SimpleMetaPackage enables phyloseq objects to be generated with clusters receiving taxonomic assignments only if the proportion of their reads receiving that annotation is above a user specified threshold. For example, if this threshold is set to 85% for a given rank then, for each cluster at that taxonomic rank, an assignment is only accepted if at least 85% of reads from that cluster have received the assignment at that rank.

5.2 | Benchmarking speed and memory

Run times and resource requirements for multi-step bioinformatic processing of metabarcoding data vary depending on marker genes, sequencing depth and the number of sequencing runs processed together. If algorithms, bioinformatic parameters and reference databases are also adjustable, as in the case of SimpleMetaPipeline, then this variation is further increased. We do not attempt to exhaustively benchmark how all combinations of these variables influence run times and resource requirements. However, by benchmarking pipeline performance in processing published datasets we provide real world examples of what users can expect.



ASV1

	Kingdom	Phylum	Class	Order	Family	Genus
Idtaxa	Animalia	Chordata	Chondrichtyes	Carcharhiniformes	unidentified	unidentified
BLAST	Animalia	Chordata	Chondrichtyes	Carcharhiniformes	Triakidae	Triakis

ASV2

	Kingdom	Phylum	Class	Order	Family	Genus
Idtaxa	Animalia	Chordata	Mammalia	Primates	Hominidae	Homo
BLAST	Animalia	Chordata	Mammalia	Primates	Hominidae	Homo

FIGURE 2 Varieties of multi-algorithm agreement. Only two-way algorithm agreements are visualised, three-way and four-way algorithm agreement tests are also possible by combining the two-way varieties visualised here. (a) Agreement between assignment and clustering algorithms. Three clusters are shown, with the proportion of component ASVs assigned to each taxa at each rank visualised, with taxonomic assignments in large blue circles representing those received by all component ASVs. For example, Cluster1 contains three ASVs all assigned to the phylum Arthropoda and class Malacostraca, but they are assigned to different orders (Decapoda and Euphausiaceae). A conservative approach would therefore be to assign the cluster to the class Malacostraca but leave it unidentified at lower ranks. (b) Agreement between clustering algorithms. Two alternative clustering outputs are shown (red and blue ovals containing ASVs represented by black bars). For example, the blue Cluster1 contains two red clusters containing three and four ASVs each. In this case, agreement and disagreement between clustering algorithms provides additional information to interrogate the internal structure of, or potential relationships between, specific clusters of interest. (c) Agreement between assignment methods. Two ASVs are shown, each receiving an assignment from both IDTAXA and BLAST. ASV1 receives diverging assignments at lower ranks (family and genus), while ASV2 receives the same assignment from both algorithms at all ranks. A conservative approach would therefore assign ASV1 to the Order Charchariniformes but leave it unidentified at lower ranks.

We conducted all benchmark runs on a laptop with a 4-core CPU and 32GB of RAM. All benchmark runs included all SimpleMetaPipeline steps, including taxonomic assignment and made use of different reference databases appropriate to the marker gene. See Table 2 and Supporting Information for full details. In the case of single Illumina MiSeq runs a relatively shallowly sequenced COI dataset (total raw reads = ca. 11 million; samples = 20) completed in 3.5 h, whereas a more deeply sequenced 23S rRNA dataset (total raw reads = ca. 22 million; samples = 20) completed in 11.5 h. Multiple MiSeq runs take substantially longer, for a given depth of sequencing, due to the previously noted requirement that DADA2 learns the error rate for each MiSeq run separately (Callahan et al., 2016). A dataset of four shallowly sequenced 18S rRNA gene MiSeq runs (total raw reads = ca. 15 million; samples = 238), where the sequences were merged before publication substantially speeding up the DADA2 step while reducing its reliability, completed in 13.5 h. Finally, a dataset of three shallowly sequenced 16S rRNA gene MiSeq runs (total raw reads = ca. 27 million; samples = 110) completed in 32 h. These figures are intended to provide an indication of orders of magnitude, while making clear that exact results will vary depending on the variables mentioned previously.

The performance of the pipeline is largely dependent on the underlying algorithms that compose it and different algorithms within the pipeline scale differently as the number of input sequences increases. The time required for denoising with DADA2 and assignment with BLAST and IDTAXA scales roughly linearly, but the time required for clustering with LULU, VSEARCH and Swarm scales exponentially. Further, the memory requirements can become large when large numbers of MiSeq runs (>10 runs) are processed together (LULU) or a large taxonomic classifier (>1 GB) is used (IDTAXA) thus requiring the use of a high performance computing cluster. All algorithms used are parallelised, thus enabling big data applications and substantial speed improvements from the use of additional cores if running the pipeline on a high performance cluster.

TABLE 2 SimpleMetaPipeline benchmarking results for published datasets, including taxonomic classification.

Maker gene	Publication	Reference database	# MiSeq runs	FASTQ type	# FASTQ files	Total reads (nearest million)	Amplicon length range	Total size of input (GB)	Time required (hours)
23S rRNA	Williams, Pettorelli, Hartmann, et al. (2024)	µgreen-db (Djemiel et al., 2020)	1	Paired-end	40	22 million	350–370	9.9	11.5
COI	Steyaert et al. (2020)	MIDORI2 (Leray et al., 2022)	1	Paired-end	40	11 million	280–360	5.4	3.5
18S rRNA	DiBattista et al. (2020)	SILVA (Quast et al., 2012)	4	Pre-merged	238	15 million	320–430	1.6	13.5
16S rRNA	Williams, Pettorelli, Hartmann, et al. (2024)	GTDB (Parks et al., 2022)	3	Paired-end	220	27 million	240–270	4.1	32

Note: All benchmarks performed on a laptop with a 4 core CPU and 32GB of RAM. All files were input in fastq.gz format. See Appendix for ControlScripts used in each benchmark run.

6 | CONCLUDING REMARKS

SimpleMetaPipeline provides a novel and accessible tool that generates robust bioinformatic outputs and usable annotated community matrices from raw metabarcoding data. It will be particularly useful for workers with a knowledge of R but a limited background in bioinformatics (a common combination in ecology) and where: (a) multiple sequencing runs need to be compared, as in large projects and meta-analyses; (b) there is uncertainty about what outputs are required; or (c) there is an established need for multiple alternative annotations, such as ASVs and OTUs. It thus represents a valuable open-source addition to the existing library of pipelines, helping democratise bioinformatics in ecology.

AUTHOR CONTRIBUTIONS

Jake Williams, Nathalie Pettorelli, Christopher Meyer and Emma Ransome conceived the project; all authors developed the method; Jake Williams wrote the code; and Jake Williams led the writing with help from Nathalie Pettorelli and Emma Ransome. All authors assisted with editing and approved for publication. The development of this application did not involve local data collection.

ACKNOWLEDGEMENTS

The authors have nothing to report.

CONFLICT OF INTEREST STATEMENT

The authors know of no conflicts of interest associated with the submitted work.

PEER REVIEW

The peer review history for this article is available at https://www.webofscience.com/api/gateway/wos/peer-review/10.1111/2041-210X.14434.

DATA AVAILABILITY STATEMENT

No original data are used in this manuscript. SimpleMetaPipeline and SimpleMetaPackage are available at https://github.com/J-Cos/SimpleMetaPipeline and https://github.com/J-Cos/SimpleMetaPackage. The code for each is also archived on Zenodo at https://zenodo.org/doi/10.5281/zenodo.7740558 (Williams, Pettorelli, Dowell, et al., 2024a); and https://zenodo.org/doi/10.5281/zenodo.7990341 (Williams, Pettorelli, Dowell, et al., 2024b).

DATA/CODE FOR PEER REVIEW STATEMENT

No data are used in this manuscript. Code for both SimpleMetaPipeline and SimpleMetaPackage are archived on Zenodo and development versions are available at GitHub repositories, these links have been removed for peer review, but will be reinserted for publication. For the purposes of peer review, zip files of each repository have been provided. All code is anonymised.

ORCID

Jake Williams https://orcid.org/0009-0000-5068-1295
Nathalie Pettorelli https://orcid.org/0000-0002-1594-6208

Rosalie Dowell https://orcid.org/0000-0002-1518-3909
Kenneth Macdonald https://orcid.org/0000-0002-0923-2460
Christopher Meyer https://orcid.org/0000-0003-2501-7952
Margaux Steyaert https://orcid.org/0000-0002-7330-035X
Sarah Tweedt https://orcid.org/0000-0003-2114-4997
Emma Ransome https://orcid.org/0000-0002-5720-1570

REFERENCES

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*, *215*(3), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2

Antich, A., Palacin, C., Wangensteen, O. S., & Turon, X. (2021). To denoise or to cluster, that is not the question: Optimizing pipelines for COI metabarcoding and metaphylogeography. *BMC Bioinformatics*, *22*(1), 177. https://doi.org/10.1186/s12859-021-04115-6

Auguie, B., & Antonov, A. (2017). Package "gridExtra", Miscellaneous Functions for "Grid" Graphics, 1–24.

Bolyen, E., Rideout, J. R., Dillon, M. R., Bokulich, N. A., Abnet, C. C., al-Ghalith, G. A., Alexander, H., Alm, E. J., Arumugam, M., Asnicar, F., Bai, Y., Bisanz, J. E., Bittinger, K., Brejnrod, A., Brislawn, C. J., Brown, C. T., Callahan, B. J., Caraballo-Rodríguez, A. M., Chase, J., … Caporaso, J. G. (2019). Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nature Biotechnology*, *37*(8), 852–857.

Brandt, M. I., Trouche, B., Quintric, L., Günther, B., Wincker, P., Poulain, J., & Arnaud-Haond, S. (2021). Bioinformatic pipelines combining denoising and clustering tools allow for more comprehensive prokaryotic and eukaryotic metabarcoding. *Molecular Ecology Resources*, *21*(6), 1904–1921. https://doi.org/10.1111/1755-0998.13398

Buchner, D., Macher, T. H., & Leese, F. (2022). APSCALE: Advanced pipeline for simple yet comprehensive analyses of DNA metabarcoding data. *Bioinformatics (Oxford, England)*, *38*(20), 4817–4819. https://doi.org/10.1093/bioinformatics/btac588

Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J. A., & Holmes, S. P. (2016). DADA2: High-resolution sample inference from Illumina amplicon data. *Nature Methods*, *13*(7), 581–583. https://doi.org/10.1038/nmeth.3869

Charif, D., & Lobry, J. R. (2007). SeqinR 1.0-2: A contributed package to the R project for statistical computing devoted to biological sequences retrieval and analysis. In U. Bastolla, M. Porto, H. E. Roman, & M. Vendruscolo (Eds.), *Structural approaches to sequence evolution. Biological and medical physics, biomedical engineering* (pp. 207–232). Springer. https://doi.org/10.1007/978-3-540-35306-5_10

DiBattista, J. D., Reimer, J. D., Stat, M., Masucci, G. D., Biondi, P., de Brauwer, M., Wilkinson, S. P., Chariton, A. A., & Bunce, M. (2020). Environmental DNA can act as a biodiversity barometer of anthropogenic pressures in coastal ecosystems. *Scientific Reports*, *10*(1), 8365.

Djaffardjy, M., Marchment, G., Sebe, C., Blanchet, R., Belhajjame, K., Gaignard, A., Lemoine, F., & Cohen-Boulakia, S. (2023). Developing and reusing bioinformatics data analysis pipelines using scientific workflow systems. *Computational and Structural Biotechnology Journal*, *21*, 2075–2085. https://doi.org/10.1016/j.csbj.2023.03.003

Djemiel, C., Plassard, D., Terrat, S., Crouzet, O., Sauze, J., Mondy, S., Nowak, V., Wingate, L., Ogée, J., & Maron, P. A. (2020). µgreen-db: A reference database for the 23S rRNA gene of eukaryotic plastids and cyanobacteria. *Scientific Reports*, *10*(1), 5915.

Frøslev, T. G., Kjøller, R., Bruun, H. H., Ejrnæs, R., Brunbjerg, A. K., Pietroni, C., & Hansen, A. J. (2017). Algorithm for post-clustering curation of DNA amplicon data yields reliable biodiversity estimates. *Nature Communications*, *8*(1), 1188. https://doi.org/10.1038/s41467-017-01312-x

Grüning, B., Chilton, J., Köster, J., Dale, R., Soranzo, N., van den Beek, M., Goecks, J., Backofen, R., Nekrutenko, A., & Taylor, J. (2018). Practical computational reproducibility in the life sciences. *Cell Systems*, *6*(6), 631–635. https://doi.org/10.1016/j.cels.2018.03.014

Hleap, J. S., Littlefair, J. E., Steinke, D., Hebert, P. D. N., & Cristescu, M. E. (2021). Assessment of current taxonomic assignment strategies for metabarcoding eukaryotes. *Molecular Ecology Resources*, *21*(7), 2190–2203. https://doi.org/10.1111/1755-0998.13407

Kodama, Y., Shumway, M., & Leinonen, R. (2012). The sequence read archive: Explosive growth of sequencing data. *Nucleic Acids Research*, *40*(D1), D54–D56. https://doi.org/10.1093/nar/gkr854

Leray, M., Knowlton, N., & Machida, R. J. (2022). MIDORI2: A collection of quality controlled, preformatted, and regularly updated reference databases for taxonomic assignment of eukaryotic mitochondrial sequences. *Environmental DNA*, *4*(4), 894–907.

Mahé, F., Rognes, T., Quince, C., de Vargas, C., & Dunthorn, M. (2015). Swarmv2: Highly-scalable and high-resolution amplicon clustering. *PeerJ*, *2015*(12), e1420. https://doi.org/10.7717/peerj.1420

Mangul, S., Mosqueiro, T., Abdill, R. J., Duong, D., Mitchell, K., Sarwal, V., Hill, B., Brito, J., Littman, R. J., Statz, B., Lam, A. K. M., Dayama, G., Grieneisen, L., Martin, L. S., Flint, J., Eskin, E., & Blekhman, R. (2019). Challenges and recommendations to improve the installability and archival stability of omics computational tools. *PLoS Biology*, *17*(6), e3000333. https://doi.org/10.1371/journal.pbio.3000333

Martin, M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal*, *17*(1), 10. https://doi.org/10.14806/ej.17.1.200

McMurdie, P. J., & Holmes, S. (2013). Phyloseq: An R package for reproducible interactive analysis and graphics of microbiome census data. *PLoS One*, *8*(4), e61217. https://doi.org/10.1371/journal.pone.0061217

Morgan, M., Anders, S., Lawrence, M., Aboyoun, P., Pagès, H., & Gentleman, R. (2009). ShortRead: A bioconductor package for input, quality assessment and exploration of high-throughput sequence data. *Bioinformatics*, *25*(19), 2607–2608.

Murali, A., Bhargava, A., & Wright, E. S. (2018). IDTAXA: A novel approach for accurate taxonomic classification of microbiome sequences. *Microbiome*, *6*(1), 140. https://doi.org/10.1186/s40168-018-0521-5

Parks, D. H., Chuvochina, M., Rinke, C., Mussig, A. J., Chaumeil, P. A., & Hugenholtz, P. (2022). GTDB: An ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Research*, *50*(D1), D785–D794. https://doi.org/10.1093/nar/gkab776

Porath-Krause, A., Strauss, A. T., Henning, J. A., Seabloom, E. W., & Borer, E. T. (2022). Pitfalls and pointers: An accessible guide to marker gene amplicon sequencing in ecological applications. *Methods in Ecology and Evolution*, *13*(2), 266–277. https://doi.org/10.1111/2041-210X.13764

Powers, S. M., & Hampton, S. E. (2019). Open science, reproducibility, and transparency in ecology. *Ecological Applications*, *29*(1), e01822. https://doi.org/10.1002/eap.1822

Quast, C., Pruesse, E., Yilmaz, P., Gerken, J., Schweer, T., Yarza, P., Peplies, J., & Glöckner, F. O. (2012). The SILVA ribosomal RNA gene database project: Improved data processing and web-based tools. *Nucleic Acids Research*, *41*(D1), D590–D596. https://doi.org/10.1093/nar/gks1219

R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Rognes, T., Flouri, T., Nichols, B., Quince, C., & Mahé, F. (2016). VSEARCH: A versatile open source tool for metagenomics. *PeerJ*, *2016*(10), e2584. https://doi.org/10.7717/peerj.2584

Sandve, G. K., Nekrutenko, A., Taylor, J., & Hovig, E. (2013). Ten simple rules for reproducible computational research. *PLoS Computational*

Biology, *9*(10), e1003285. https://doi.org/10.1371/journal.pcbi.1003285

Sato, Y., Miya, M., Fukunaga, T., Sado, T., & Iwasaki, W. (2018). MitoFish and mifish pipeline: A mitochondrial genome database of fish with an analysis pipeline for environmental DNA metabarcoding. *Molecular Biology and Evolution*, *35*(6), 1553–1555. https://doi.org/10.1093/molbev/msy074

Schloss, P. D., & Westcott, S. L. (2011). Assessing and improving methods used in OTU-based approaches for 16S rRNA gene sequence analysis. *Applied and Environmental Microbiology*, *77*, 3219–3226. https://doi.org/10.1128/aem.02810-10

Shea, M. M., Kuppermann, J., Rogers, M. P., Smith, D. S., Edwards, P., & Boehm, A. B. (2023). Systematic review of marine environmental DNA metabarcoding studies: Toward best practices for data usability and accessibility. *PeerJ*, *11*, e14993. https://doi.org/10.7717/peerj.14993

Steyaert, M., Priestley, V., Osborne, O., Herraiz, A., Arnold, R., & Savolainen, V. (2020). Advances in metabarcoding techniques bring us closer to reliable monitoring of the marine benthos. *Journal of Applied Ecology*, *57*(11), 2234–2245. https://doi.org/10.1111/1365-2664.13729

Wickham, H. (2011). ggplot2. *Wiley Interdisciplinary Reviews: Computational Statistics*, *3*(2), 180–185. https://doi.org/10.1002/wics.147

Wickham, H., François, R., Henry, L., & Müller, K. (2023). dplyr: A grammar of data manipulation. https://dplyr.tidyverse.org

Williams, J., Pettorelli, N., Dowell, R., Macdonald, K., Meyer, C., Steyaert, M., Tweedt, S., & Ransome, E. (2024a). SimpleMetaPipeline. https://zenodo.org/doi/10.5281/zenodo.7740558

Williams, J., Pettorelli, N., Dowell, R., Macdonald, K., Meyer, C., Steyaert, M., Tweedt, S., & Ransome, E. (2024b). SimpleMetaPackage. https://zenodo.org/doi/10.5281/zenodo.7990341

Williams, J., Pettorelli, N., Hartmann, A. C., Quinn, R. A., Plaisance, L., O'Mahoney, M., Meyer, C. P., Fabricius, K. E., Knowlton, N., & Ransome, E. (2024). Decline of a distinct coral reef holobiont community under ocean acidification. *Microbiome*, *12*(1), 75. https://doi.org/10.1186/s40168-023-01683-y

Wratten, L., Wilm, A., & Göke, J. (2021). Reproducible, scalable, and shareable analysis pipelines with bioinformatics workflow managers. *Nature Methods*, *18*(10), 1161–1168. https://doi.org/10.1038/s41592-021-01254-9

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

Supporting Information S1. SimpleMetaPipeline algorithms and parameters.

How to cite this article: Williams, J., Pettorelli, N., Dowell, R., Macdonald, K., Meyer, C., Steyaert, M., Tweedt, S., & Ransome, E. (2024). SimpleMetaPipeline: Breaking the bioinformatics bottleneck in metabarcoding. *Methods in Ecology and Evolution*, *15*, 1949–1957. https://doi.org/10.1111/2041-210X.14434