A Galaxy-based QIIME2 pipeline for determining phylogenetic biodiversity metrics from fungal ITS data

---

Master Research Project Report, Biology, Leiden University

Written by: Ciel Huisman (1850954)

Specialization: Biodiversity and Sustainability

Internship at Naturalis Biodiversity Center at Darwinweg 2, 2333 CR Leiden

Research Group: Understanding Evolution

Supervisor: Dr. R.A. Vos

Duration: 02-01-2024 until 14-03-2025 (32 EC)

Contact person raw data: Dr. R. A. Vos

Article style: BMC Bioinformatics

Table of Contents

Abstract 3

Keywords 3

Background 4

Methods 10

Programming of workflow 10

Coding implementation 10

Snakemake 10

Data analysis platform and plugins 11

Galaxy 11

QIIME2 11

Hardware and operating system 12

Metabarcoding data 12

Implementation 12

Overview 12

Galaxy sub-workflow 14

Workflow step by step 15

Step 1: Pasting data and manifest file into Galaxy 15

Step 2: Importing sequences into QIIME2 16

Step 3: Denoising of sequences 16

Step 4: Alignment and phylogeny building 16

Step 5: Determination of diversity metrics 17

Step 6: Exporting data from QIIME2 (and Galaxy) 18

Step 7: Visualization of phylogenetic diversity 18

Results 18

Runtime and plots 18

Discussion 20

Features of the pipeline 20

Automated data analysis 20

Organization and flexibility 21

Visualization options 21

Current issues and possible improvements 22

Bioblend inputs 22

Logistical problems 22

Rarefaction without resampling 23
Future work 23
Multiple runs & data provenance 23
CI/CD 24
Upscaling: NDOR 24
Conclusion 24
Availability and requirements 25
List of abbreviations 25
Acknowledgements 26
References 26

Abstract

Humankind has caused a stark decline in biodiversity over the past 50 years, which has led to increasing demands from legislation to conserve and restore ecosystems and their services to humans. Conservation and restoration are only possible with proper biodiversity monitoring, but traditional methods are slow and insufficient. The rise of DNA-metabarcoding has made it possible to detect species through environmental DNA (eDNA). This allows for the easier and faster determination of diversity metrics. As this is particularly challenging for the fungal kingdom – using the barcode ITS – the Naturalis eDentity project is building a reference tree to map detected species even before they are identified to still understand their biodiversity patterns without complete species identification. However, it is a complicated process to go from DNA sequence to phylogenetic alpha and beta diversity metrics, and bioinformatic pipelines are still being developed for this purpose. This thesis explores such a pipeline written in Python and R, using the Snakemake framework, the Bioblend package, the Galaxy webserver, and QIIME2 data analysis plugins. It was tested with ITS sequence data from seven locations in The Netherlands, which took four hours and fifteen minutes. It yielded one phylogenetic alpha diversity heatmap, and two plots of phylogenetic beta diversity: a dendrogram and a PCOA plot. Such automation had clear advantages over slower and more error-prone use of the GUI. It improved visualization by relying on R packages with more customizable options instead of the available QIIME2 plugins. There were minor issues with the use of bioblend, the continuity of the workflow, and its reproducibility, but the workflow's organization is flexible enough to fix them in due time. Much future work is still needed to meet the goals of the eDentity project, such as the implementation of multiple runs, data provenance, CI/CD, and cloud integration. However, this kind of workflow has shown clear potential for eDNA monitoring, thereby aiding in bending the curve of biodiversity loss thanks to improved monitoring.

Keywords

DNA-metabarcoding, eDNA, Fungal ITS, Galaxy, Phylogenetic diversity, Python, QIIME2, R, Snakemake

Background

Worldwide biodiversity is deteriorating, a process that has accelerated in the past 50 years and has dire consequences for both nature and humans. It encompasses the extinction of species and threatens nature's vital contributions to humanity through ecosystem services and functions, such as clean water and pollination (IPBES, 2019). The Kunming Montreal Global Biodiversity Framework was created during the COP-15 Convention on Biological Diversity to combat biodiversity loss, which described its vision that "by 2050, biodiversity is valued, conserved, restored and wisely used, maintaining ecosystem services, sustaining a healthy planet and delivering benefits essential for all people" (CBD, 2022). Subsequent political levels follow these global agreements: the EU Restoration Plan wants to restore its ecosystems, habitats and species, as currently, 81% of its nature is in a bad state (European Commission, 2022).

According to these recent legislations, biodiversity needs to be monitored to conserve and restore it. Firstly, to properly determine the severity of the problem – the decline in biodiversity over time – the current state of affairs needs to be known on whatever scale relevant for future implementation of legislation (Taylor & Gemmell, 2016). Secondly, the drivers of this biodiversity loss must be identified, which means a plan can be made to reduce human impacts and address the decline (Jaureguiberry et al., 2022). Thirdly, the effects of these plans must be determined to evaluate progress. This could also mean comparing land uses or nature management strategies between areas to find the best approach for a particular conservation or restoration goal (Valentini et al., 2016; Willerslev et al., 2014). Both governments and industries benefit from this, as the latter are increasingly asked to report on and compensate for their environmental impact (e.g. CSRD; European Parliament, 2022).

Traditional monitoring methods rely on the expertise of biodiversity researchers that determine species presence and all kinds of biodiversity metrics (alpha, beta, or gamma; see Box 1), but this has drawbacks. The manual identification of all species in a particular ecosystem is a time-intensive process, especially with the great array of taxonomic groups that need to be surveyed by their respective experts (Hammond, 1992; D. Hawksworth et al., 1995). Another downside of this method is the unintended bias that arises when some groups are understudied compared to others, meaning these groups get overlooked in conservation and restoration efforts (Chen, 2021; Tensen, 2018; Dorey, 2018). Some biodiversity cannot or hardly be detected by the naked eye, like micro-organisms, genetic diversity, and cryptic species (Bickford et al., 2007). This means traditional biodiversity research cannot cope with recent legislation's demands to monitor the full breadth of biological diversity efficiently and accurately.

Identification of species has been drastically simplified by a technique called DNA barcoding, in which small parts of the genome of organisms ('markers') are read as a proverbial barcode that is unique for roughly every species (Hebert et al., 2003; see also Box 2 at the end). The sampling effort has since been further reduced by the development of Next Generation Sequencing, in which even far and few between pieces of DNA could be amplified and their barcodes read, or 'sequenced', to serve as a proxy for biodiversity. The next step, Third Generation Sequencing, does not need amplification and can sequence DNA in real-time (Hu et al., 2021; Schadt et al., 2010). This is useful, as all organisms 'contaminate' their surroundings with environmental DNA (eDNA; see Barnes & Turner, 2016; Taberlet et al., 2012). To illustrate: a teaspoon of soil could contain the DNA of thousands of species, either from whole (micro-)organisms, animal feces, seeds, or even shed skin cells (see Ruppert et al. (2019) for extensive coverage of biomonitoring uses). With this method of eDNA metabarcoding, it is possible to identify species and their relative abundance with less effort, cost, and bias than traditional methods (Deiner et al., 2017).

Box 1: How to measure biodiversity?

In the context of this thesis, we will only consider alpha and beta diversity. See Magurran (2021) for more information on measuring alpha, beta, and gamma diversity.

Alpha diversity

Alpha diversity is the biodiversity of a certain location or sample. The most straightforward way to measure alpha diversity is to count the number of species present, i.e. species richness. This is a form of 'taxonomic diversity'. However, species richness has been questioned as a useful biodiversity metric, as it has low information density and fails to adequately capture biodiversity in the field and simulated community experiments (Lyashevska & Farnsworth, 2012; Wilsey et al., 2005).

Aside from species richness, numerous indices have been proposed that also express species abundance, such as 'evenness' (Smith & Wilson, 1996). Some use both species richness and abundance, like the Shannon and Simpson Indices, often in the context of Hill (1973). These metrics better describe biodiversity, but they have drawbacks regarding sensitivity to changes and are misleading in their responses to them (e.g. Santini et al., 2017). Other indices focus on 'functional diversity' instead, which stresses the distribution of one or more traits and can help understand ecosystem functioning and services (Cadotte et al., 2011; Laureto et al., 2015). An issue of using functional diversity is that traits are difficult or impossible to observe from micro-organisms and (e)DNA sequences directly.

Another crucial issue when choosing a biodiversity metric for conservation and restoration is capturing the full breadth of biodiversity across the Tree of Life. The 'phylogenetic diversity' (PD) metric coined by Faith (1992) is applicable to DNA data and incorporates both species richness and phylogenetic information. PD is based on the genetic distance of all species in the sample, whose evolutionary relationship with each other is visualized in a phylogenetic tree. A phylogenetic tree usually has species on its tips, but can also be based on taxa or whole families (Figure 2).

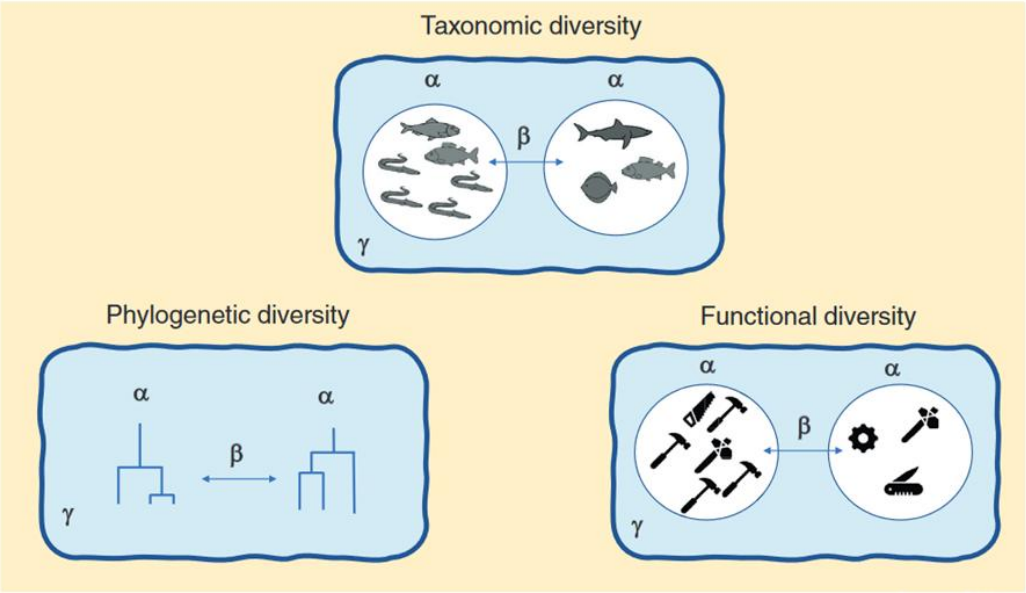


Figure 1: Differences between alpha, beta, and gamma diversity types. Source: Current Biology.

As for the actual calculation of this metric: the branch lengths of such a tree can be made proportional to the evolutionary distance based on shared (morphological) features or on substitutional distance in DNA mutations. In this case, Faith's alpha PD can be calculated by summing up all branch lengths from root to tip. In Figure 2, this corresponds to counting the amount of feature changes, or 'ticks', in the tree, meaning the PD of this sample is 36. Note that the phylogenetic diversity increases when species are less related, as they share fewer branches and add more to the total branch length.

Since then, many metrics have been introduced using this concept or similar notions of phylogeny: incorporating average branch lengths (Warwick & Clarke, 1995), community assembly processes (Hodkinson et al., 2002), species abundance information (Helmus et al., 2007), and phylogenetic abundance distributions (Cadotte et al., 2010). An advantage of PD and related indices is the valuation it can provide for underrepresented societal benefits, such as genetic diversity for medicinal compounds found in plants and marine organisms, and the 'insurance value' of biodiversity for future generations (IPBES, 2019).

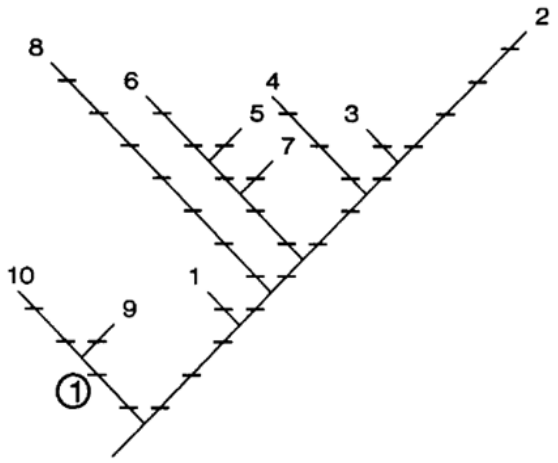


Figure 2: Phylogenetic tree of ten different, arbitrary species (1-10). Each 'tick' on the tree represents the development of a feature in that specific lineage and can be represented as adding one unit of branch length to the tree. Source: Faith (1992).

Beta diversity

So far, we have discussed alpha diversity as either taxonomic, functional, or phylogenetic. When these metrics of a sample or area need to be compared with another sample or area—either spatially or temporally—this is called beta diversity. Beta diversity represents compositional differences in species richness, functional traits, or overlap in phylogenetic relationships and is necessary to effectively inform biodiversity conservation and restoration (Socolar et al., 2016).

Just like alpha diversity, beta diversity knows many indices. The Jaccard and Bray Curtis indices are measures of (dis)similarity on a taxonomic level, and functional beta diversity may be conveyed in distance or dissimilarity matrices of traits (e.g. Villéger et al., 2013). Phylogenetic beta diversity is based on PD and is most commonly expressed as the unique branch length of a sample compared to the total branch length. This is the Unifrac metric, which can be unweighted or weighted to diminish the influence of low abundance features (Lozupone et al., 2007; Lozupone & Knight, 2005), adjusted for variance (Chang et al., 2011), and generalized to better detect abundance

changes in moderately rare species (Chen et al., 2012). An important advantage of Unifrac is its correlation with functional diversity, where species richness, abundance and other phylogenetic beta indices do not show this (Swenson, 2011).

Similarly to PD, Unifrac can be calculated by looking at branch lengths that are proportional to evolutionary change. Only now there are two samples, and we are interested in how unique they are in comparison to each other. This means we can sum the lengths of the unique branches of both and divide this by the total branch length. Hence, it is possible to express the 'distance' between any two samples with Unifrac, which can be put in a distance matrix and visualized to see which samples are most different. In Figure 3, the Unifrac is equal to the black horizontal parts of the tree (excluding the root), as they are unique to either sample A or B (= approximately 12cm). This does not consider the abundance of the species in each sample. A more advanced measure would be to 'weight' the unique branch lengths to the abundance, as happens in the determination of different weighted Unifrac metrics.

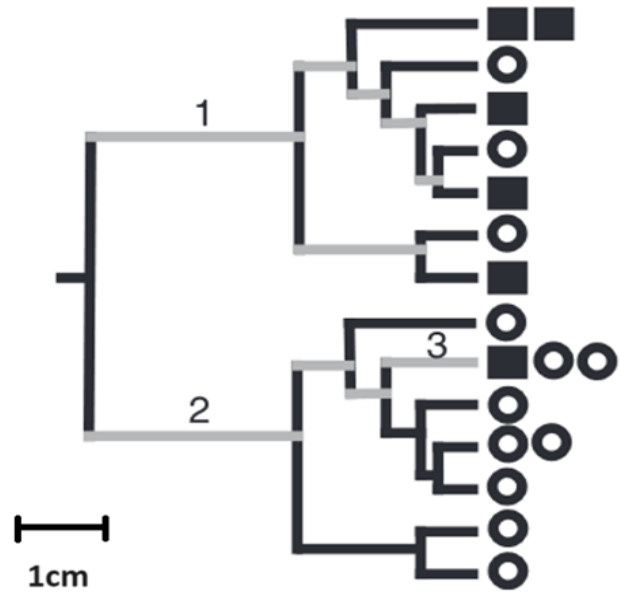


Figure 3: Representation of unweighted Unifrac calculation. Black horizontal branches are unique to either sample A (squares) or sample B (circles) and thus are included in the calculation. Source: Lozupone et al. (2007).

For these reasons, the Naturalis Biodiversity Center has chosen DNA metabarcoding as a cornerstone for their eDentity project, which builds an infrastructure to monitor Dutch biodiversity at a molecular level. It will consist of an automated sample processor for specimens and DNA samples, a DNA laboratory for extraction, a DNA sample storage, data processing pipelines, a reference database for species classification, and a Biocloud to store the data and its patterns. This would enable over 300,000 samples per year to be processed for science, policy, and industry, and the first sampling would start in the summer of 2025. This information will also be used for future reference: an example might be an unknown species that can be documented until discovery or reusing information about the effects of management and restoration policies for similar areas (Naturalis; unpublished grant proposal, 2023).

A critical component of this infrastructure lies in data processing: how to go from bits of eDNA to useful information on biodiversity. It is especially challenging to infer such things for the fungi kingdom, as only 3-8% of an estimated 2.2 to 3.8 million species are currently described (Hawksworth & Lücking,

2017), even though they play a crucial role in ecosystems (Brussaard, 1997; Fitter et al., 2005). Naturalis has been building phylogenetic trees based on ITS data (see Box 2) to aid in taxonomic identification and researching fungi diversity. These efforts aim to eventually produce a reference tree usable even without known taxonomical annotations and with minimal computational time. One example of recent research is the development of a method to first create a minimal 'backbone' tree – based on the UNITE database for fungal diversity (Nilsson et al., 2019) – to decrease the computational complexity of such a task (Carton & Romeijn; BSc. Thesis, 2022). Another similar study performed phylogenetic placement using BLAST combined with maximum likelihood and Bayesian methods (ten Haaf; BSc. Thesis, 2023). The latter study also included some investigations on diversity, but this was not the main focus and did not use phylogenetic diversity metrics.

As discussed in Box 1, using these metrics is crucial to understanding the multifaceted nature of biodiversity. Conveniently, the indices from traditional monitoring can also infer diversity from DNA sequences. However, this means that specifically designed bioinformatic pipelines are needed, as it is a complicated process from raw eDNA data to determination of such metrics (again, see Box 2). Naturalis has a web-based platform hosted on the Galaxy webserver to facilitate the creation of such pipelines by combining different 'tools' through a graphical user interface and an application programming interface. Currently, the Databricks-based web service 'nbitk' is also in development, and it is a Jupiter Notebook to employ Galaxy for phylogenetic placement and validation through BLAST.

However, this is only part of the pipeline from sequence to biodiversity, and Naturalis is still looking for the optimal way to design the final data analysis framework for eDentity. This thesis therefore aims to create a DNA metabarcoding pipeline to automate the whole process from ITS data to plots on alpha and beta diversity. It is a proof of concept to build such a workflow written in Python and R, inside a 'Snakemake' framework, with the 'bioblend' API to access Galaxy, and using microbiome analysis tools from 'QIIME2' to do so.

Box 2: How to read (a lot of) DNA?

Genes and barcodes

To effectively read DNA for taxonomic assignment, the DNA sequences the assignment is based on should at least be variable – undergoing evolution over time – but not to the point that members from the same species have wildly different sequences. Some genes that are in this category are 'housekeeping' genes, whose major sequence disruption would be fatal to the organism, but which have minor variations per species. Another requirement is reliable and universal amplification of this DNA via PCR (Polymerase Chain Reaction), so it can be easily detected by the sequencer. When the flanking regions of the barcode are highly conserved across the group you are sequencing, PCR can use them to 'find' this barcode effectively.

The first barcode that satisfied these conditions and was used for this purpose was 16S ribosomal RNA. Today, the 16S ribosomal RNA marker is very popular in microbiology and microbiome research. However, other markers were chosen for other groups, such as the nuclear ITS rDNA marker for fungi, the chloroplast genes rbcL and matK for plants, and the mitochondrial COI gene for animals (Coissac et al., 2012).

This thesis relies on data amplified by the ITS marker, which is widely used in fungal metabarcoding studies (Nilsson et al., 2019). ITS (or Internal Transcribed Spacer) is a marker located in the nucleus of

the cell and resides in ribosomal or rDNA clusters that are repeated in tandem for up to thousands of times. The sequence is flanked by the '18S' and '28S', and interrupted by the '5.8S' region, which are highly conserved building blocks of the ribosome and thus can be easily recognized in the PCR process (Figure 4). The ITS regions do not play a role in the production of ribosomes: the regions are excised from the final ribosomal structure.

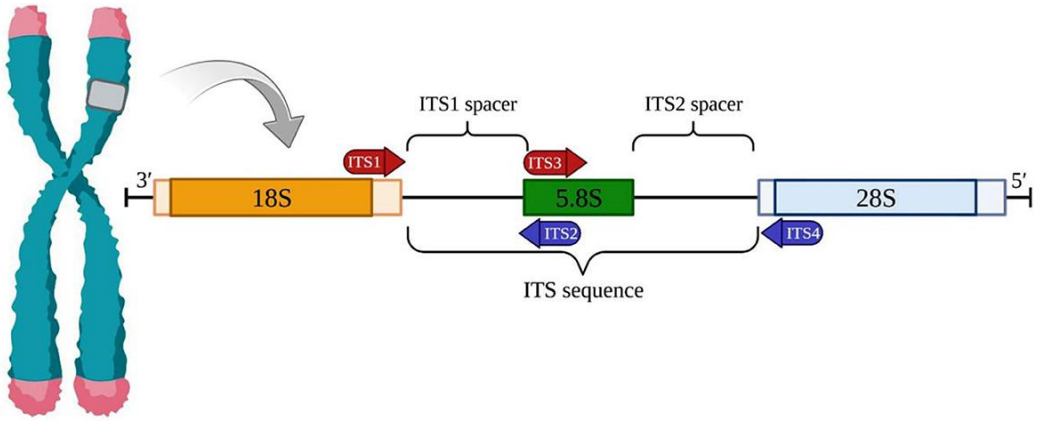


Figure 4: Schematic overview of the ITS region, consisting of ITS1 and ITS2. The arrows with ITS1-ITS4 indicate the 'primer' sequences where the polymerase anchors itself to amplify the ITS rDNA. Red and blue denote forward and reverse primers, which depend on the position of the DNA molecule's 3' and 5' end. Source: Fathy et al. 2023.

Pipelines

However, things like species classification and diversity metrics cannot be inferred from raw sequence data, especially not with increasing amounts of data. Many steps are needed to clean and prepare all the data before taxonomies can be assigned, and even more steps to build a phylogeny and calculate diversity. These individual steps can be performed by bioinformatic tools or packages, sometimes knit together in a 'pipeline' (Figure 5). Common examples of pipelines are QIIME 2 (Bolyen et al., 2018), OBITools (Boyer et al., 2016), and VSEARCH (Rognes et al., 2016), which span some or all of the processes from raw sequence to the end result relevant to the research question.

More specifically, the tools and/or pipeline should first be suited to the type of sequence data. Each sequencer delivers its own type of data format, which is relevant to steps such as demultiplexing; making sense of the labeling used to distinguish samples produced in the same sequencing run. Also, the clustering of sequences to the lowest level can be done in Amplicon Sequence Variants (ASVs) or Operational Taxonomic Units (OTUs), which require different steps. Thirdly, some tools are designed to the markers of one group of organisms, such as ITS, and for example, extract only those DNA pieces for more accurate results (ITS express; Rivers et al., 2018) or they remove mitochondrial pseudogenes that influence COI classification negatively (VTAM; González et al., 2023). Finally, it depends on whether the end result should be a species list, a phylogenetic tree for one area, or a plot of the beta diversity between two or more sites. See also Hakimzadeh et al. (2023) for an overview of past and present pipelines.

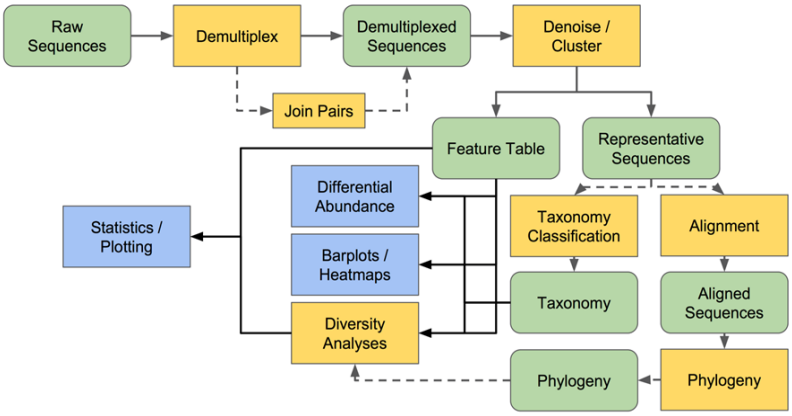


Figure 5: Overview of a DNA metabarcoding pipeline from raw sequence data to biodiversity metric visualizations. Green rectangles represent data, yellow represents operations on said data, and blue represents visualizations. Source: QIIME2 documentation, https://docs.qiime2.org/2024.10/tutorials/overview/.

Methods

Programming of workflow

Coding implementation

This workflow was written mainly in Python (3.10.13) using the IDE PyCharm (2023.3.3). The remaining parts were written in R (4.1.2) using RStudio (2022.07.1 +554). A virtual environment had to be created with various packages from both programming languages for this workflow to function correctly. This was achieved with the 'conda' package and environment manager (24.7.1). This conda environment, called 'barcode-phylogenetic-diversity', contained channels and packages described in the environment.yml file found in the repository. Using conda specifically allows automated unit testing through CI/CD (Continuous Integration and Continuous Delivery) in cloud services such as AWS (Amazon Web Services).

Snakemake

This workflow was programmed using the data analysis framework Snakemake (Köster & Rahmann, 2012; Mölder et al., 2021). Snakemake implements rules for specific actions that define how output files should be created from input files. These can be created through shell commands or scripts in Python or other programming languages, like R. Since the output of one rule can be used as input for another, Snakemake also automatically determines dependencies between the rules.

Snakemake also contains additional features like wildcards, helper functions, wrappers, benchmarking and logging that can be implemented for the different rules. Another essential characteristic is scalability: a workflow can be run on a single core, a cluster, or in the cloud with little modification. This makes it particularly suitable in the light of the eDentity project.

Data analysis platform and plugins

Galaxy

Galaxy is a free, open-source data analysis environment (The Galaxy Community, 2024). Its primary focus is on the analyses of DNA-related data. It is also used for creating and exchanging workflows, education and training, and publishing data analysis tools. This study has utilized the 24.1 Galaxy Release (June 2024) and Naturalis' own Galaxy environment. Galaxy Naturalis runs on the commercial cloud of Amazon Web Services (AWS) and is mainly focused on genomic and phylogenetic analyses.

Inside Galaxy, there are several features of note for the workflow at hand. Firstly, there is the concept of 'histories', which can be created to keep track of different projects and can be assigned different names. Switching from one history to the next in the Galaxy environment is possible. Secondly, inside these histories, users can upload one or more 'dataset' objects, which have their own information to be accessed, such as their provenance, history content API ID (i.e. dataset ID), and Unique Universal ID (UUID). Lastly, the uploaded datasets can be fed to so-called 'tools', plugins Galaxy can use to perform operations on said datasets. The workflow from this thesis solely used plugins from QIIME2 for its analysis, which will be discussed in the next section. All available tools can be found and documented in the Galaxy Toolshed (https://toolshed.g2.bx.psu.edu/).

The histories, datasets, and tools inside Galaxy can be accessed via a graphical user interface (GUI), command line interface (CLI), or application programming interface (API). The GUI is accessed via a URL that can be opened in a web browser (https://galaxy.naturalis.nl/) and requires a username and password with the right credentials. For the CLI and API, only an API key is sufficient, which can be obtained through the GUI under the 'User' heading at the top of the page (User > Preferences > Manage API key).

The workflow build for this thesis operates solely through an API: the Python package 'bioblend' (Sloggett et al., 2013). This package facilitated the interaction with the Galaxy Server, also called the Galaxy Instance. It allowed the creation of a 'GalaxyInstance' client, which then could perform various built-in functions such as creating histories, uploading datasets, and running tools. BioBlend was installed into the virtual environment through the 'bioconda' channel, which itself relies on conda.

QIIME2

QIIME2 is the successor to QIIME, which stands for 'Quantitative Insights Into Microbial Ecology'. Both are open-source bioinformatics pipelines that allow microbiome analysis from raw sequence data to taxonomy, phylogeny, diversity metrics and visualizations. Important features of QIIME2 include data provenance tracking, its system for semantic types, a large number of plugins, and support for various interfaces (from CLI to API and GUI).

Data provenance is tracked by introducing QIIME-specific file formats that include the data and metadata and keeping records of all the steps needed to produce the file's contents. When importing data into QIIME2, it first needs to be converted to a 'qiime zipped artifact' (.qza). This artifact can then undergo a number of operations called 'methods' to transform or analyze the data within while keeping the same extension. An artifact cannot be visualized. In contrast, a 'visualization' can be visualized and is created by a visualizer method that turns a .qza file into a .qzv file (qiime zipped visualization). These can be viewed in a browser (View QIIME2).

Aside from distinguishing artifacts and visualizations, QIIME2 files have semantic types that differ according to the original data type and/or the methods that produced the file. The most common semantic types are 'FeatureData[Sequence]' and 'FeatureTable[Frequency], which are both artifacts. The former represents a file with sequences that can be used mainly for taxonomic assignment and

phylogeny building. At the same time, the latter is a table consisting of sequence abundance information useful for calculating diversity metrics. Visualization methods usually bring forth their own semantic types.

As for the actual methods that produce artifacts and visualizations with their own semantic types: there are a lot. Some of these are simple operations, such as transforming a table, but others are more complicated tools, which perform steps like demultiplexing or taxonomic assignment. QIIME2 has its own tools implemented, but it also supports plugins from various sources that perform their own operations, such as denoising, clustering, and aligning sequences. Examples of plugins include DADA2, VSEARCH, and Fasttree. For more information about QIIME2, see the QIIME2 documentation online (https://docs.qiime2.org/2024.10/).

Hardware and operating system

This project was run on a Lenovo Thinkpad E14 with an 11th-generation Intel(R) Core(TM) i7-1165G7 and 16 GB of RAM. Since Windows was the operating system, a WSL, Ubuntu 22.04.3 LTS, was installed to run the analysis.

Metabarcoding data

The data used in this research are demultiplexed ITS sequences from eDNA soil sampling collected in seven locations (samples) in the Netherlands. The format is Phred33 FASTQ from Illumina paired-end sequencing, which means each sample has a forward and reverse pair. These fourteen files were zipped to reduce storage, producing the extension .fastq.gz.

Implementation

Overview

The workflow's overall inputs are DNA metabarcoding samples (data directory), while the endpoints are one visualization of alpha and two visualizations of beta diversity: a heatmap, a dendrogram, and a PCOA plot (results/div-metrics directory). To accomplish these results, the snakefile consists of twelve rules that perform different operations on the data. The directed acyclic graph (DAG) below (Figure 6) provides an overview of the twelve rules and their connections.

The backbone of the workflow constitutes eight out of twelve rules and is executed by running the *run_tools.py* script in the bin directory. Running all tools like this in the Galaxy environment would require three types of input: the name of the tool to obtain from the Toolshed, the dataset(s) the tool needs to be run on, and the parameters for the tool. The input of most rules is a file with a tool name and one or more files with a dataset ID, which reside in the 'results/galaxy-ids' directory. The parameters are set by the *parameters.py* script from the workflow directory, which is imported as a module and can be modified to change the parameters. After running the tool, each of these eight rules produces another dataset inside Galaxy and saves the ID of this dataset to a new file. This way, the process can be repeated for the next rule, and so on.

The other rules exist merely to facilitate this process. The 'get_tool_names' and the 'paste_manifest' rule have their own Python scripts and are designed to produce the tool names from the tool list (*workflow/tools.txt)* and upload the data into Galaxy, respectively. The 'extract_data' rule exists to use the dataset IDs of the alpha and beta diversity products to extract them from Galaxy, after which the

'visualize' rule can create the desired figures from them using R. See also Table 1 for the exact correspondence between scripts, snakefile rules, and qiime2 tools in this research. Also, please consult the documentation (see Results section) for information on how to execute the workflow.

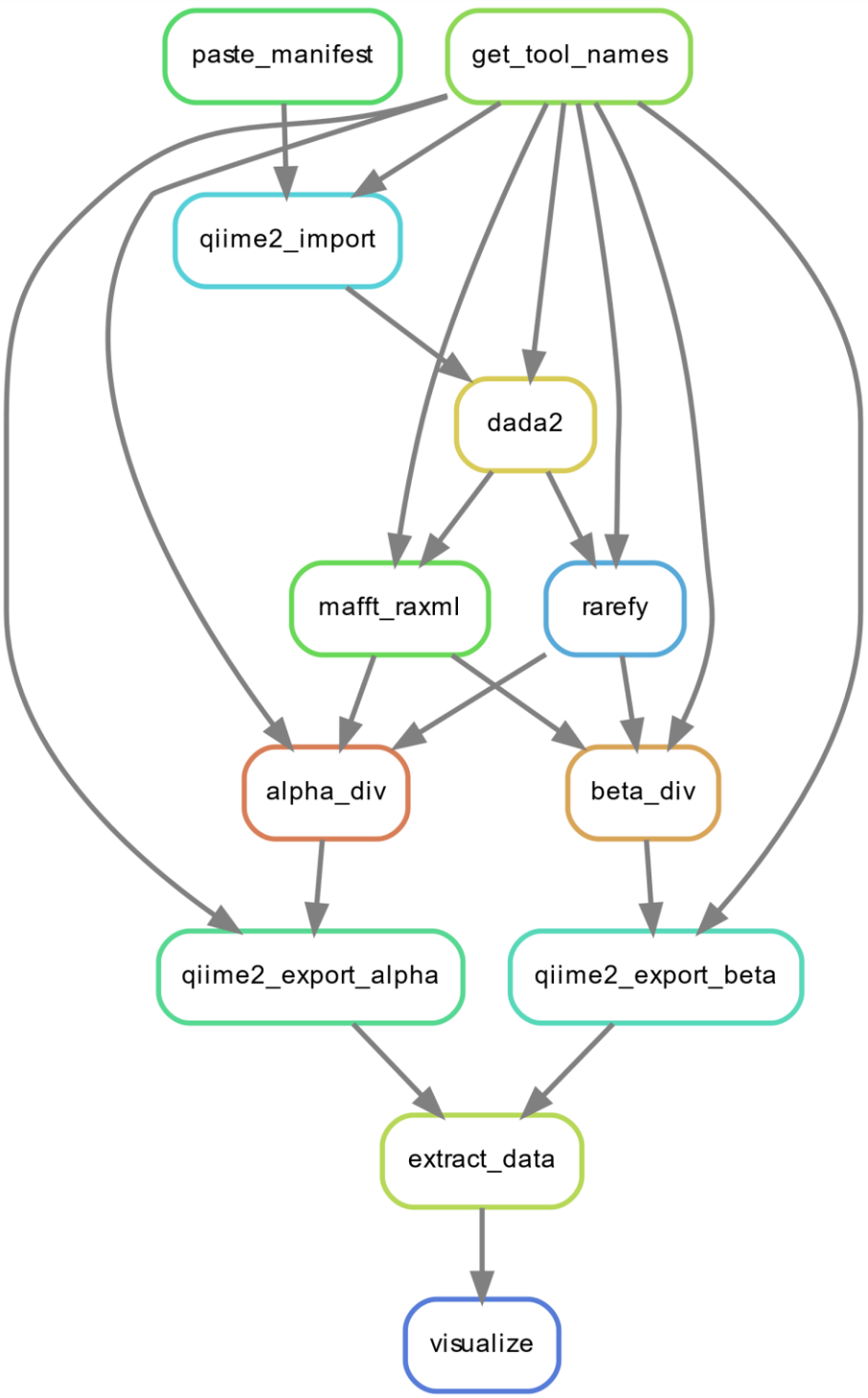


Figure 6: DAG of the snakefile rules and their dependencies.

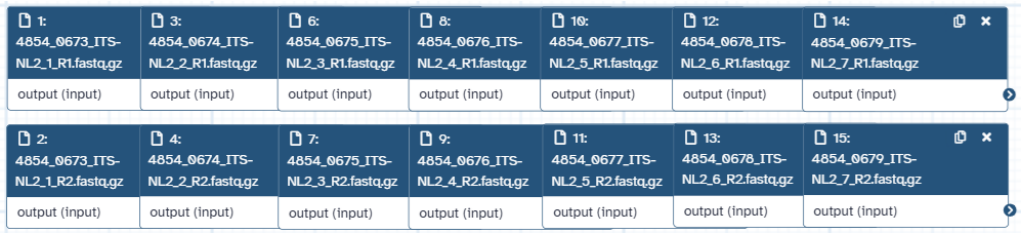
Table 1: An overview of the scripts, snakefile rules, and QIIME2 tool names described in this study. All rules and tool names belonging to run_tools.py (+ parameters.py) *fully take place inside the Galaxy environment.*

Script (bin)	**Snakefile rule**	**QIIME2 tool name**
get_tool_names.py	get_tool_names	-
paste_and_manifest.py	paste_manifest	-
run_tools.py (+ parameters.py)	qiime2_import dada2 mafft_raxml rarefy alpha_div beta_div qiime2_export_alpha qiime2_export_beta	qiime2 tools import qiime2 dada2 denoise-paired qiime2 phylogeny align-to-tree-mafft-raxml qiime2 feature-table rarefy qiime2 diversity alpha-phylogenetic qiime2 diversity beta-phylogenetic qiime2 tools export qiime2 tools export
galaxy_extract.py	extract_data	-
visualize.R	visualize	-

Galaxy sub-workflow

The workflow inside Galaxy describes the tool outputs and their connections in more detail (Figure 7). It starts with the 'qiime2 tools import' and ends with the 'qiime2 tools export' when all operations are finished. The QIIME2 tools in between vary in the amount of input and output datasets, and not all outputs are used for another tool.

Most parameters for the tools were not set explicitly, which meant they were left on their default settings. Consult the Toolshed or Galaxy GUI for more information about these settings. In Table 1 below, the 'required' parameters used for this thesis can be found.



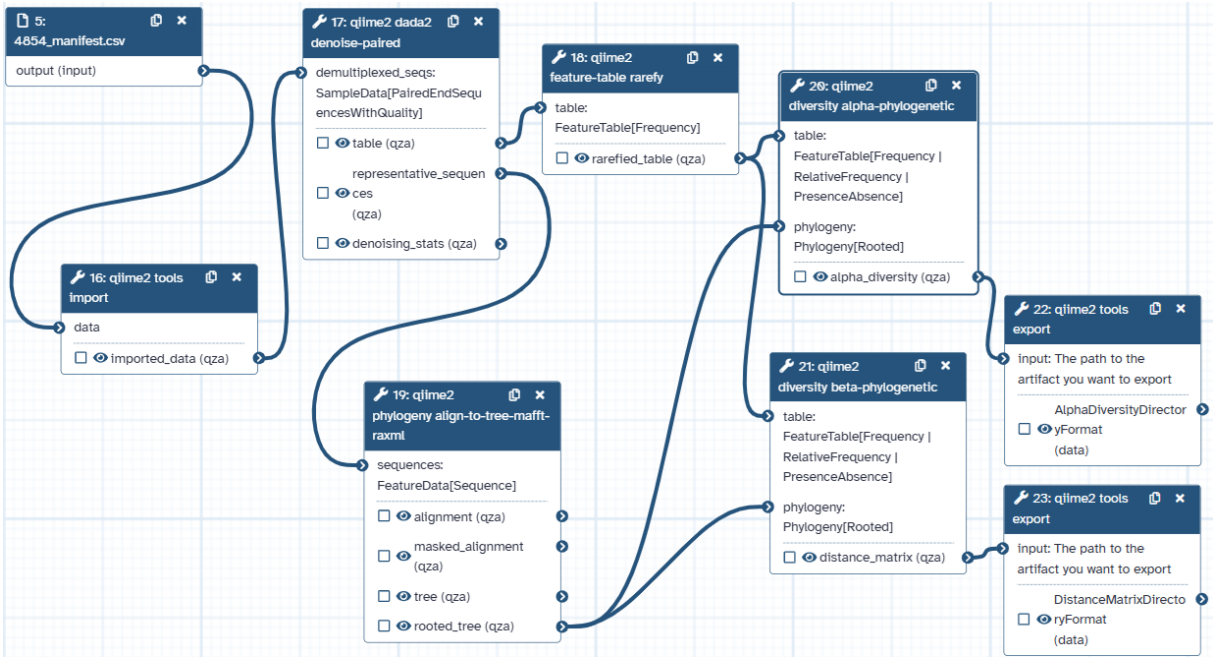


Figure 7: Screenshots of the sub-workflow as seen within Galaxy. Above are the pasted datasets used to test the workflow. The links between the inputs and outputs of tools can be found below.

Table 2: QIIME2 tool names and their corresponding parameters. Only required parameters are shown. The sampling depth parameter was based on the denoising statistics from a test run of qiime2 dada2 denoise-paired.

Tool name	**Parameters**
qiime2 tools import	'format': PairedEndFastqManifestPhred33 'type': SampleData__ob__PairedEndSequencesWithQuality__cb__
qiime2 dada2 denoise-paired	'trunc_len_f': 0 'trunc_len_r': 0
qiime2 phylogeny align-to-tree-mafft-raxml	'substitution_model': GTRGAMMA 'raxml_version': Standard
qiime2 feature-table rarefy	'sampling_depth': 51992
qiime2 diversity alpha-phylogenetic	'metric': faith_pd
qiime2 diversity beta-phylogenetic	'metric': weighted_unifrac
qiime2 tools export (alpha diversity)	'type_peek': SampleData__ob__AlphaDiversity__cb__ 'fmt_peek': AlphaDiversityDirectoryFormat
qiime2 tools export (beta diversity)	'type_peek': DistanceMatrix 'fmt_peek': DistanceMatrixDirectoryFormat

Workflow step by step

Step 1: Pasting data and manifest file into Galaxy

The first step of this workflow addresses the need to get the FASTQ ITS sequences from DNA metabarcoding into the Galaxy environment for further analysis. In Galaxy, this is called 'pasting' the data, which was done after creating the 'barcode-phylogenetic-diversity' history. These steps were accomplished by the 'paste_manifest' rule in the snakefile, which utilized the *paste_and_manifest.py* script inside the 'bin' directory to do so. This resulted in fourteen pasted data entries or seven forward-reverse pairings.

At the same time, this script would create a so-called 'manifest file' (data directory) to import the data into qiime2 (see next step). This manifest file (*manifest.csv*) is essential to map the sample IDs, the location of the pasted data (absolute file path), and the direction of the reads to facilitate the import of the type of data used here. Afterwards, the created file was also pasted in Galaxy as a 'PairedEndFastqManifestPhred33' artifact. Its dataset ID was saved to refer to the manifest file while importing the data.

Step 2: Importing sequences into QIIME2

For convenience, the names of all tools to use in this workflow can be supplied in one text file: *tools.txt* in the workflow directory. The tool names were then obtained by the 'get_tool_names' rule, which called upon the *get_tool_names.py* script (bin). The script stripped the tool names from this file and saved them as separate text files.

The first tool name relevant here, 'qiime2 tools import', was the one to import the data into QIIME2 and provided the first input for the snakefile rule 'qiime2_import'. The next input consisted of the appropriate dataset ID, which corresponded to the manifest file of the previous step. This allowed the conversion of the sequences from .fastq.gz files into .qza files usable by QIIME2. This method also assigned the proper semantics according to the input data: it created one object of 'SampleData[PairedEndSequencesWithQuality]' containing all fourteen entries. While doing so, it saved the dataset ID of this object for the next step of denoising.

Step 3: Denoising of sequences

After import, the data was not yet suitable for phylogeny building or the determination of diversity metrics, as it needed to be denoised first. This preprocessing was performed by the 'dada2' rule in the snakefile with the tool 'dada2 denoise-paired', which is a plugin from DADA2. DADA2 (Divisive Amplicon Denoising Algorithm) performs multiple steps: filtering on quality score, dereplication, denoising, chimera filtering, and merging of paired reads. The denoising step is designed to disentangle the variation in DNA sequences from variation arising from amplicon sequencing errors. It does so by calculating the probability that the abundance of a certain read is consistent with an error model. This model describes the distribution of expected reads around a particular sequence and considers quality scores. Suppose the probability is low enough ($p < 0.05$). In that case, the lowest p-value is assigned the next original 'sequence', and the cycle continues until no significant p-values are found (Callahan et al., 2016).

After supplying the tool name ('dada2 denoise-paired') and the imported dataset ID, running this tool resulted in three output files. It yielded one 'SampleData[DADA2Stats]' object that contained information about the dataset and denoising performed, one 'FeatureData[Sequence]' object with denoised sequences, and one FeatureTable[Frequency] file with sequence abundance information of said sequences. The first one was used to determine the right sampling depth for the rarefaction step, and the id of the latter two was used for phylogeny building and rarefaction, respectively.

Step 4: Alignment and phylogeny building

The next step is taken by the rule 'mafft_raxml', which utilizes the id of the FeatureData[Sequence] object from the previous step. The tool 'align-to-tree-mafft-raxml' encompasses the whole process of going from sequence data to a phylogeny, which is four different steps.

The first step is alignment, which uses the 'mafft' part of the plugin. MAFFT stands for Multiple Alignment Fast Fourier Transform; an algorithm that identifies homologous regions between sequences to align them. It is based on an amino acid substitution model that incorporates information about the phisico-chemical properties (i.e. volume and polarity) of the two amino acids in question. It also uses a Fourier Transformation to decrease computational time. For more elaborate information on MAFFT and its different versions, see Katoh (2002) and Katoh & Standley (2013).

The aligned sequences, now dubbed FeatureData[AlignedSequence], are used in the next step: masking. This step removes DNA sites that may be too noisy or uninformative, which has been shown to be beneficial for phylogenetic inference and systemic bias correction (Castresana, 2000; Cummins & McInerney, 2011). However, this step does not change the semantics of the artifact.

Thirdly, the phylogenetic relationships between the sequences are inferred from their alignment with the help of the 'raxml' section of the 'align-to-tree-mafft-raxml' plugin. RAxML is an algorithm for this purpose and its most recent version has been described by Stamatakis (2014), but see also Stamatakis et al. (2005). The algorithm builds a phylogenetic tree by optimizing the likelihood of said tree. It starts with an initial parsimony tree, then performs rearrangements on the subtrees, after which the nearest branches are optimized, and the likelihood of the tree is calculated. When the final tree is selected, it is stored in a Phylogeny[Unrooted] object.

The last operation concerns changing the semantics from Phylogeny[Unrooted] to Phylogeny[Rooted], which is done by rooting the tree. This means answering the question: Where in the tree is the most recent common ancestor of these species located? The plugin uses the 'midpoint root' method to accomplish this, and the tree is rooted on its longest branch. This indicates the direction of evolution. In the next step, only the ID of this rooted tree was used to determine diversity metrics.

Step 5: Determination of diversity metrics

Now that the 'dada2 denoise-paired' has produced a FeatureTable[Frequency] artifact this could be employed to extract information about biodiversity. However, not all samples have the same amount of sequences, which could lead to bias (Hughes & Hellmann, 2005; Sanders, 1968). Therefore, the tool 'qiime2 feature-table rarefy' was applied first (rule 'rarefy'), meaning the feature table is rarefied according to a certain sampling depth. The sampling depth encompassed the number of sequences randomly drafted per sample, with samples containing fewer sequences being dropped. This number should be chosen to optimize the coverage of both samples and the number of sequences per sample, and it should be derived from the DADA2 denoising statistics (SampleData[DADA2Stats]). Rarefaction itself has no consequences for semantics.

Secondly, the now rarefied FeatureTable[Frequency] and the Phylogeny[Rooted] artifacts were chosen as inputs for the next rule: 'alpha_div'. As the name suggests, this rule calculated the alpha diversity of the samples, and it called upon the qiime2 tool 'diversity alpha-phylogenetic'. This calculation works similarly to the one explained in Box 1 and produces a SampleData[AlphaDiversity] object, which, in this case, contains the Faith PD.

Lastly, the 'diversity beta-phylogenetic' tool calculated the beta diversity for the 'beta_div' rule, which produced a 'weighted Unifrac' metric. Again, this was done according to the unique branch length, as described in Box 1. This operation expressed the differences between the samples in a distance matrix, which were stored in a file with the semantic DistanceMatrix.

Step 6: Exporting data from QIIME2 (and Galaxy)

As mentioned before, QIIME2 has certain visualization possibilities, but these were deemed unsatisfactory for this pipeline's desired outcome. For one, no direct method exists to create a heatmap of the alpha diversity. Another issue is the absence of visualization options for dendrograms with samples for tips. As described in the next section, R packages were chosen instead for this purpose.

This meant the products of 'diversity alpha-phylogenetic' and 'diversity beta-phylogenetic' needed to be exported from QIIME2 for further visualization. This happened with the rules 'export_alpha' and 'export_beta'. To do so, both SampleData[AlphaDiversity] and DistanceMatrix IDs were fed to the tool 'qiime2 tools export', which converted their artifacts from .qza files into .tsv files usable in other environments. Different parameter settings had to be supplied while running the same tool for the various products. Another rule, 'extract_data', downloaded all .tsv files from Galaxy according to their IDs and saved them in the results/div-metrics directory.

Step 7: Visualization of phylogenetic diversity

The R package 'ggplot2' visualized the alpha diversity with a colored heatmap to distinguish samples based on the height of their Faith PD. The file was saved as a .png file in the results/div-metrics directory.

Another package was necessary to create the .png of the dendrogram with samples as tips to see which samples are more closely related: ggdendro. The PCOA plot was made for a more general overview of relatedness.

Results

Runtime and plots

Please consult the Github repository for documentation of the entire workflow: https://github.com/naturalis/barcode-phylogenetic-diversity. The runtime of the whole workflow was about four hours and fifteen minutes, but note that this may vary with different amounts of sequence data and different parameters. The resulting heatmap, dendrogram, and PCOA can be found in the figures below (Figure 8, Figure 9, and Figure 10).

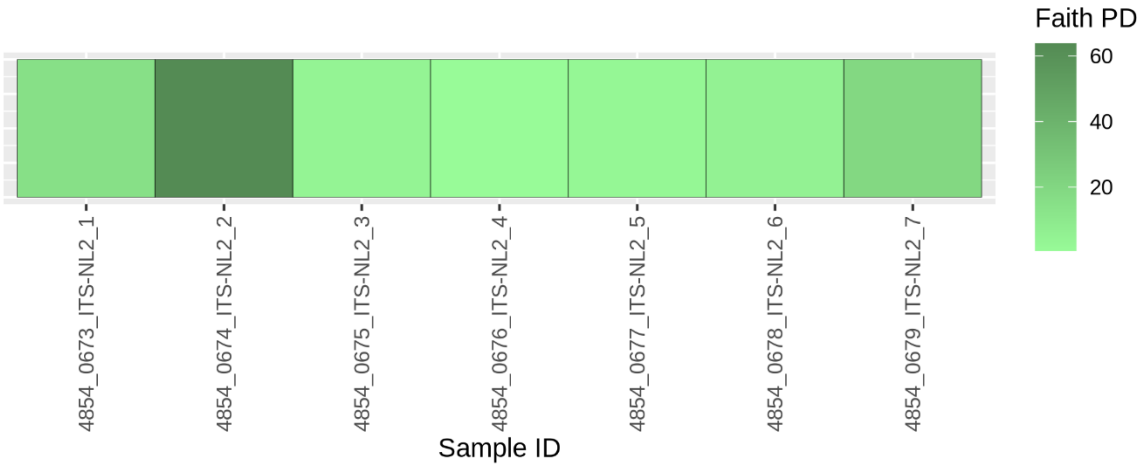


Figure 8: Heatmap of the alpha diversity (Faith PD) of the seven samples described in this study.

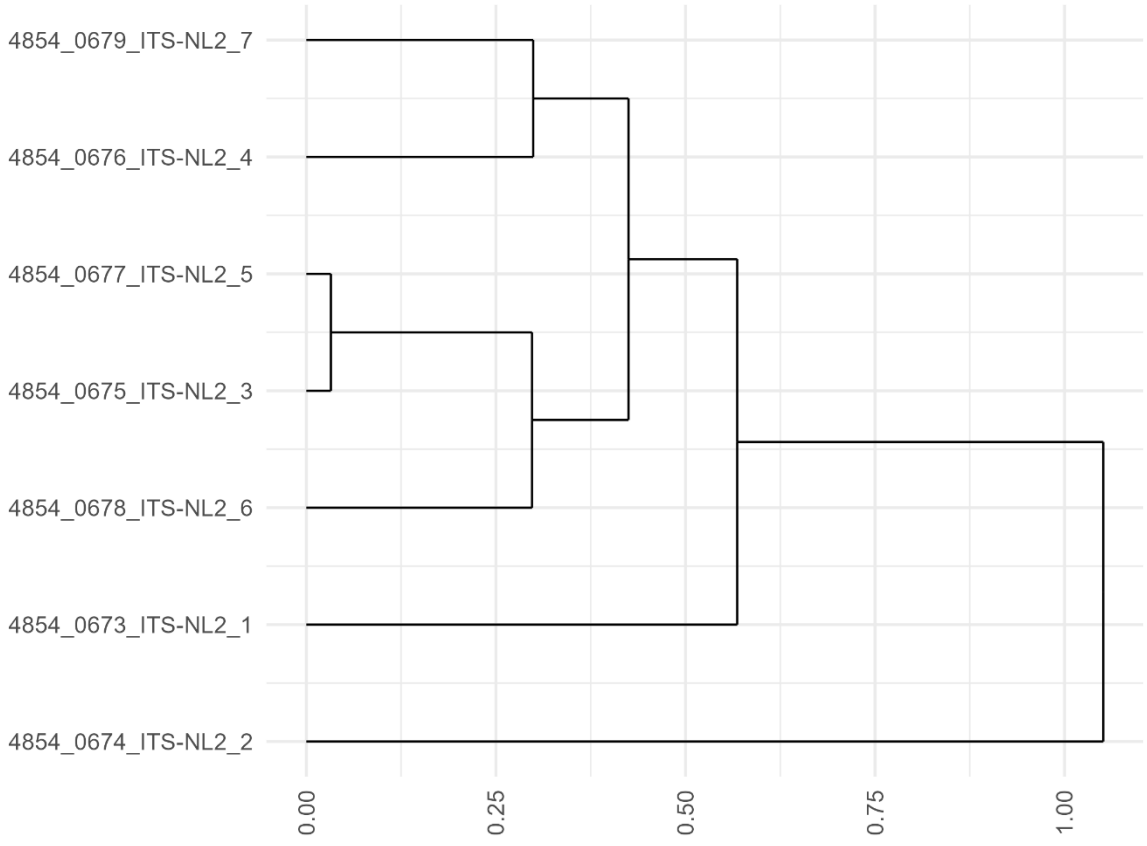


Figure 9: Dendrogram of the beta diversity (weighted Unifrac) of the seven samples described in this study.

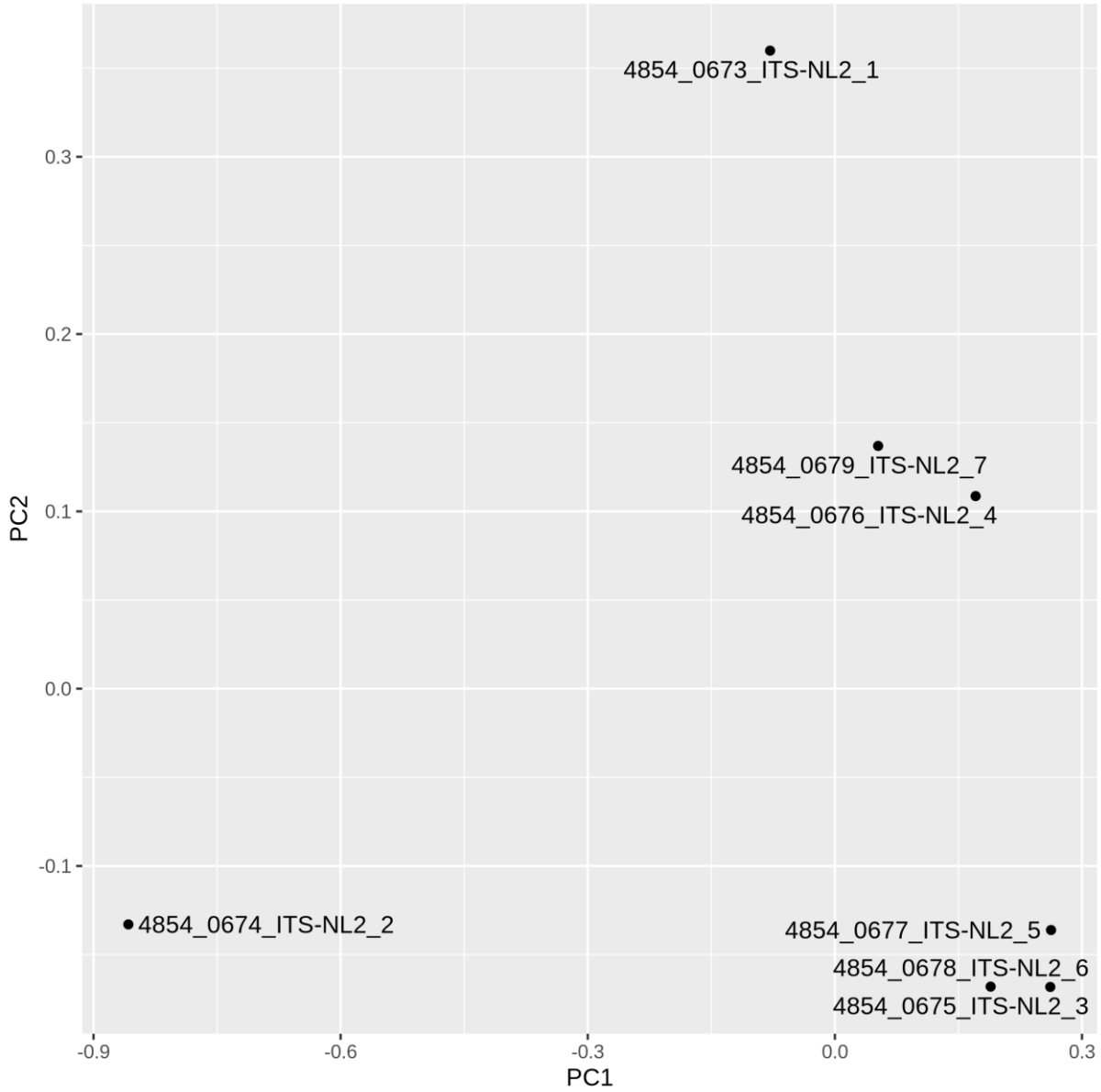


Figure 10: PCOA plot of the beta diversity (weighted Unifrac) of the seven samples described in this study.

Discussion

Features of the pipeline

Automated data analysis

An important feature of this data analysis workflow is its automation of an entire pipeline to reduce errors. Manual execution inside the GUI is relatively error-prone, leaving room for the user to make mistakes by selecting the wrong but similar datasets or parameters before using a tool. Even when using the CLI, this might happen unintentionally when inputting the wrong file name, for example. Error reduction is therefore a major advantage of employing an automated workflow. It is still possible

to make mistakes in choosing tools, setting parameters, and making snakefile rules, but this needs to happen consistently over all the involved variables and files to escape the error flagging system. The logging from the Python scripts implemented in Snakemake also makes it easier to spot the exact location of the problem.

Automation not only reduces errors in a single run but also increases reproducibility and allows reuse of new datasets. Reproducibility is also a crucial component of data analysis because this grants the opportunity to verify certain results. Reusability in itself is essential, too, as this could broaden the application of the methods used and further scientific progress. Again, this is not possible with manual input, as consistency over multiple runs cannot be guaranteed. This workflow instead allows better comparison between the results of different runs and biodiversity patterns to be meaningfully studied. The same or different datasets can be run with the same settings, and this requires little modification (see also in the next subsection). Especially relevant for eDentity is upscaling to larger and more datasets, which will be discussed in the section 'Future work'.

The third advantage lies in the low runtime of the workflow. Performing the analysis inside the GUI and waiting for different steps to complete inevitably takes time, both in overall run time and from the user themselves. This is reduced by automation. As described, it takes 4h15 for 475MB of data to be visualized into phylogenetic diversity plots. The user only needs to spend time preparing the run (however, see the next section). Similar to reproducibility and reusability, low run time also conveniences upscaling.

Organization and flexibility

The other side of the coin for this workflow is its relative flexibility, which results from its organization. As explained, the pipeline can be modified without having to adjust the bin scripts. If it is necessary to change certain parts of the workflow – because of an updated or alternative tool, for example – the user can do so by adjusting the scripts in the src directory. This mainly results from the *run_tools.py* script being designed for varying numbers of input and output files. The same is possible for the parameters (however, see also the next section), which can be changed to suit a specific study or goal. All this ensures an amount of flexibility is retained, while still operating a highly automated workflow.

Visualization options

Some issues with the QIIME2 visualizations were addressed by the workflow. Using the alternatives offered by R packages increased the accessibility and usability of the phylogenetic diversity plots. This was deemed necessary, as the QIIME2 bound visualization format prevented using the .qzv visualizations outside of the QIIME2 View website. The latter necessitated the use of a browser and was inconvenient for an API-based workflow. By using the R packages (ggplot2, ggdendro) instead, the .png images created could be viewed immediately by clicking on the files.

The second improvement was the added visualization and customization options, mainly derived from ggplot2 and ggdendro. As discussed, QIIME2 could not straightforwardly produce an alpha diversity

heatmap and a more sophisticated dendrogram for beta diversity. The R alternatives, therefore, could give the user a clear overview of the alpha biodiversity and directly identify the nearest neighbors of different samples based on the beta diversity. QIIME2 did offer a PCOA-like plot – the emperor plot – and had its own customization options in QIIME2 View, but this did not allow point labelling, font size changes, or fixing of overlapping text, for example. This was also addressed by the R packages implementation, resulting in an improved overview of sample clustering.

Current issues and possible improvements

Bioblend inputs

Using BioBlend made the implementation easier, as the Galaxy API on itself is relatively low-level and may not have been suitable for this purpose. Creating clients and applying methods to perform the tasks inside Galaxy were satisfactory, and were mostly flexible and user-friendly. However, there was one exception: the parameter builder. Before running the tools, the 'inputs().set()' function needs to create an InputsBuilder object from the parameters in a specific nesting and order. This can be found by eyeballing the XML script of the tool inside the Galaxy Toolshed. Another possibility is utilizing the ToolClient-related 'build()' function and extracting the 'state_inputs' as stated in the documentation, then manually adding the parameters and their nesting. Either way, this requires additional work, which cannot be automated, and as of the writing of this thesis, there is no alternative to this procedure.

Logistical problems

Two main logistical issues were related to choosing parameters for truncation and sampling depth. The first one concerns forced suboptimal truncation parameters for DADA2. The truncation parameters describe after which base pair the quality score of the average sequence is considered too low for further analysis. For both forward ('trunc_len_f') and reverse ('trunc_len_r') sequences, truncation can be set to discard the bases after the input number position. According to the QIIME2 documentation, these required parameters can be derived from the demultiplexing statistics that result from QIIME2's demultiplexing tools. This is not possible for the data used here because it is already demultiplexed, so the data was not truncated.

Improving the denoising through the quality scores would be possible, but this would require extra effort to hardcode it into the workflow. An example would be making an extra Python script and snakefile rule to extract the information and automatically decide the truncation, plus adjustments to the 'dada2' snakefile rule, the *run_tools.py* script and *parameters.py*. Another option would be an integration with some visualization the user has to view, after which they supply their own estimate to an input prompt that becomes a parameter. Whether this would be a necessary or acceptable solution depends on the relative denoising improvement, and if this aligns with the goals for this project, so this should be studied first.

A similar problem arose with the sampling depth parameter of rarefaction, although the information for the parameter setting was actually available. As explained in the Implementation section, the sampling depth should be determined from the DADA2 denoising statistics. Therefore, it may require disruption of the workflow to manually check the appropriate sampling depth, a test run, or another setup as proposed for the truncation parameter. However, the lowest number of sequences per sample could be extracted from an exported version of the SampleData[DADA2Stats] table and used as a default. It would still need a new script and some minor modifications to the aforementioned scripts, but it would cease to disrupt the workflow and ensure a bare minimum of rarefaction.

Rarefaction without resampling

Most QIIME2 tools offered many parameters, although the determination of diversity metrics lacked reproducibility and accuracy. The lack of resampling methods for the rarefaction step leads to different results for identical workflow runs, as there is no option for choosing the random seed. If the feature table was resampled instead, the alpha and beta diversity could be calculated for many iterations and averaged, making the process more accurate and reproducible. There is a QIIME2 plugin in development implementing both resampled rarefaction and bootstrapping ('q2-boots'; unpublished Raspet et al., 2024), which could be integrated into the workflow by swapping it with the current 'qiime2 feature table rarefy'. As discussed, the scripts involved are flexible enough to do so.

Future work

Multiple runs & data provenance

Performing multiple runs was not within the scope of this thesis, as it was a proof of concept pipeline. However, it would be a crucial next step for the workflow's development to suit the needs of the eDentity project, for which many samples need to be processed. In this stage of development, it is impossible to perform multiple runs without overwriting, as the snakefile puts all results in one directory. A solution for this could be introducing new directories inside the results section for each snakemake run based on the date and time of execution. This way, the documentation would also improve. Deleting the history after every run would also be highly recommended in this context.

Clear documentation on when the code was executed is useful and necessary, but this workflow does not yet allow for documentation on data provenance outside of QIIME2 or Galaxy. However, provenance tracking is essential when dealing with multiple runs based on different data, tools, or parameters. Provenance is currently lost when exporting the alpha and beta diversity QZA objects as TSV files. Future work should thus include this option, for example, using the 'provenance_lib' package for Python. Additional snakefile rules are then needed to extract the QZA files alongside the TSV files and another script for provenance extraction. When this is implemented, each run will be indistinguishable, and each result richly annotated with provenance data. This is in accordance with the FAIR (Findable, Accessible, Interoperable, Reusable) principles of scientific data sharing.

CI/CD

Even though the previous sections have elaborated on the workflow's possible improvements, there has been no discussion yet about how to test those and future modifications of the code. Currently, this is done manually, and while this is acceptable for this thesis, it would not do in the context of the eDentity project. Proper CI/CD would therefore be both convenient and necessary. A Github Actions workflow would ensure automated unit testing after any push or pull request, as a YAML file gets executed anytime this occurs. The code inside this file would set up the environment with the *environment.yml*, run unit tests inside the conda environment, and then run the whole pipeline. For example, the unit testing in question could be performed with the Python-based 'pytest' framework. Integrating this into the workflow would improve automation, provide early error detection, and allow for more convenient collaboration on the project.

Another advantage is that the workflow could be run in the cloud more easily. This is of particular importance when upscaling such a pipeline to process more ITS data, as is the goal of the eDentity project (also, see next section). To do so, the above-mentioned Github Actions workflow could be supplemented with code uploading the pipeline to the cloud after successful unit testing and running it in that location. This is possible with the AWS CLI ('awscli' package) facilitating this interaction, given the user supplies an 'AWS access key ID' and 'AWS secret access key' with the right credentials and the S3 bucket name to store the pipeline in. Afterwards, a Snakemake job can be submitted to AWS Batch and be run with the right 'Batch job queue' and 'Batch job definition'. This means that the results are also stored in the cloud and no longer take up storage elsewhere.

Upscaling: NDOR

The final part of this section ties into the cloud implementation of the section above. It would be useful not only to run the analysis in the cloud but also to use eDentity's large amounts of metabarcoding data stored in the cloud. Galaxy Naturalis has a tool available to download data from NDOR on the same AWS cloud to accomplish this. This step could be woven into the workflow with some minor modifications and an extra snakefile rule at the file's beginning. Involving an HPC cluster would then facilitate the 'barcode-phylogenetic-diversity' pipeline to handle large amounts of data and completely operate in the cloud.

Conclusion

Recent legislation and increasing amounts of eDNA data have made automated biodiversity monitoring indispensable, but the bioinformatic infrastructures needed are still under construction. This project attempted to create such a pipeline from fungal ITS metabarcoding data to visualizations about alpha and beta diversity. The automated workflow offers many advantages over the GUI while

retaining the flexibility to remedy some of the current issues and implement the next steps in pipeline development. It also improved visualization with its own R script to circumvent the suboptimal QIIME2 plots. However, much work is still needed to use this workflow in the context of Naturalis' eDentity project, mainly concerning unit testing and integration into the cloud. In conclusion, this thesis has been a successful proof of concept for using Snakemake, Galaxy, and QIIME2 tools to automate metabarcoding processing.

Availability and requirements

Project name:	barcode-phylogenetic-diversity
Project home page:	https://github.com/naturalis/barcode-phylogenetic-diversity
Operating system(s):	Linux
Programming language:	Python, R
Other requirements:	See *environment.yml*
License:	Apache-2.0
Any restrictions to use:	None

List of abbreviations

API – Application Programming Interface

ASV – Amplicon Sequence Variant

AWS – Amazon Web Services

CI/CD – Continuous Integration / Continuous Delivery

CLI – Command Line Interface

DADA – Divisive Amplicon Denoising Algorithm

DAG – Directed Acyclic Graph

eDNA – Environmental DNA

GUI – Graphical User Interface

HPC – High Performance Computing

ITS – Internal Transcribed Spacer

MAFFT – Multiple Alignment Fast Fourier Transformation

OTU – Operational Taxonomic Unit

PCOA – Principal Coordinate Analysis

PCR – Polymerase Chain Reaction

PD – Phylogenetic Diversity

RAxML – Randomized Axelarated Maximum Likelihood

QIIME – Quantitative Insights Into Microbial Evolution

QZA – Qiime Zipped Artifact

QZV – Qiime Zipped Visualization

UUID – Unique Universal ID

Acknowledgements

I would like to thank Dr. R.A. Vos for his guidance during this project and his patience for my developing programming skills. Secondly, I wish to thank Joppe for sharing some of his code about manifest files with me. Thirdly, I want to thank ChatGPT for answering the questions I was too embarrassed to ask anyone else.

On a more personal note, my gratitude goes out to the people who believed in me when I did not even believe in myself. Many thanks to my mentor Marco and my friend Rowyan in particular. And thank you, Cor, for making me laugh during rainy days at the museum. I know you would have been proud to know I finally finished this. And, as always, thank you H.K.

References

Barnes, M. A., & Turner, C. R. (2016). The ecology of environmental DNA and implications for conservation genetics. *Conservation Genetics*, *17*(1), 1–17. https://doi.org/10.1007/s10592-015-0775-4

Bickford, D., Lohman, D. J., Sodhi, N. S., Ng, P. K. L., Meier, R., Winker, K., Ingram, K. K., & Das, I. (2007). Cryptic species as a window on diversity and conservation. *Trends in Ecology & Evolution*, *22*(3), 148–155. https://doi.org/10.1016/j.tree.2006.11.004

Bolyen, E., Rideout, J. R., Dillon, M. R., Bokulich, N. A., Abnet, 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. (2018). *QIIME 2: Reproducible, interactive, scalable, and extensible microbiome data science* [Preprint]. PeerJ Preprints. https://doi.org/10.7287/peerj.preprints.27295v2

Boyer, F., Mercier, C., Bonin, A., Le Bras, Y., Taberlet, P., & Coissac, E. (2016). obitools: A unix-inspired software package for DNA metabarcoding. *Molecular Ecology Resources*, *16*(1), 176–182.

Cadotte, M. W., Carscadden, K., & Mirotchnick, N. (2011). Beyond species: Functional diversity and the maintenance of ecological processes and services. *Journal of Applied Ecology*, *48*(5), 1079–1087. https://doi.org/10.1111/j.1365-2664.2011.02048.x

Cadotte, M. W., Jonathan Davies, T., Regetz, J., Kembel, S. W., Cleland, E., & Oakley, T. H. (2010). Phylogenetic diversity metrics for ecological communities: Integrating species richness, abundance and evolutionary history. *Ecology Letters*, *13*(1), 96–105. https://doi.org/10.1111/j.1461-0248.2009.01405.x

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

Carton, C., & Romeijn, L. (2022). *Building a phylogeny for the fungal kingdom with ITS data.* Leiden Institute of Advanced Computer Science (LIACS).

Castresana, J. (2000). Selection of Conserved Blocks from Multiple Alignments for Their Use in Phylogenetic Analysis. *Molecular Biology and Evolution*, *17*(4), 540–552. https://doi.org/10.1093/oxfordjournals.molbev.a026334

CBD. (2022). *15/4. Kunming-Montreal Global Biodiversity Framework*.

Chang, Q., Luan, Y., & Sun, F. (2011). Variance adjusted weighted UniFrac: A powerful beta diversity measure for comparing communities based on phylogeny. *BMC Bioinformatics*, *12*(1), 118. https://doi.org/10.1186/1471-2105-12-118

Chen, E. Y.-S. (2021). Often Overlooked: Understanding and Meeting the Current Challenges of Marine Invertebrate Conservation. *Frontiers in Marine Science*, *8*. https://doi.org/10.3389/fmars.2021.690704

Chen, J., Bittinger, K., Charlson, E. S., Hoffmann, C., Lewis, J., Wu, G. D., Collman, R. G., Bushman, F. D., & Li, H. (2012). Associating microbiome composition with environmental covariates using generalized UniFrac distances. *Bioinformatics*, *28*(16), 2106–2113. https://doi.org/10.1093/bioinformatics/bts342

Coissac, E., Riaz, T., & Puillandre, N. (2012). Bioinformatic challenges for DNA metabarcoding of plants and animals. *Molecular Ecology*, *21*(8), 1834–1847. https://doi.org/10.1111/j.1365-294X.2012.05550.x

Cummins, C. A., & McInerney, J. O. (2011). A Method for Inferring the Rate of Evolution of Homologous Characters that Can Potentially Improve Phylogenetic Inference, Resolve Deep Divergence and Correct Systematic Biases. *Systematic Biology*, *60*(6), 833–844. https://doi.org/10.1093/sysbio/syr064

Deiner, K., Bik, H. M., Mächler, E., Seymour, M., Lacoursière-Roussel, A., Altermatt, F., Creer, S., Bista, I., Lodge, D. M., de Vere, N., Pfrender, M. E., & Bernatchez, L. (2017). Environmental DNA metabarcoding: Transforming how we survey animal and plant communities. *Molecular Ecology*, *26*(21), 5872–5895. https://doi.org/10.1111/mec.14350

DIRECTIVE (EU) 2022/2464 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL, Official Journal of the European Union (2022). https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32022L2464

European Commission. (2022). *Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on nature restoration*. https://eur-lex.europa.eu/resource.html?uri=cellar:f5586441-f5e1-11ec-b976-01aa75ed71a1.0001.02/DOC_1&format=PDF

Faith, D. P. (1992). Conservation evaluation and phylogenetic diversity. *Biological Conservation*, *61*(1), 1–10. https://doi.org/10.1016/0006-3207(92)91201-3

González, A., Dubut, V., Corse, E., Mekdad, R., Dechatre, T., Castet, U., Hebert, R., & Meglécz, E. (2023). VTAM: A robust pipeline for validating metabarcoding data using controls. *Computational and Structural Biotechnology Journal*, *21*, 1151–1156. https://doi.org/10.1016/j.csbj.2023.01.034

Hakimzadeh, A., Abdala Asbun, A., Albanese, D., Bernard, M., Buchner, D., Callahan, B., Caporaso, J. G., Curd, E., Djemiel, C., Brandström Durling, M., Elbrecht, V., Gold, Z., Gweon, H. S., Hajibabaei, M., Hildebrand, F., Mikryukov, V., Normandeau, E., Özkurt, E., M. Palmer, J., … Anslan, S. (2023). A pile of pipelines: An overview of the bioinformatics software for metabarcoding data analyses. *Molecular Ecology Resources*, 1755-0998.13847. https://doi.org/10.1111/1755-0998.13847

Hammond, P. (1992). Species Inventory. In B. Groombridge (Ed.), *Global Biodiversity: Status of the Earth's Living Resources* (pp. 17–39). Springer Netherlands. https://doi.org/10.1007/978-94-011-2282-5_4

Hawksworth, D., Kalin-Arroyo, M., Hammond, P., Ricklefs, R., Samways, M., Aguirre-Hudson, B., Dadd, M., Groombridge, B., Hodges, J., Jenkins, M., Mengesha, M., Grant, W., Latham, R., Lewinsohn, T., Lodge, D., Platnick, N., Wright, D., Crowe, T., & Stace, C. (1995). *Global Biodiversity Assessment Chapter 3: Magnitude and Distribution of Biodiversity* [Dataset]. https://www.researchgate.net/profile/William-Grant-12/publication/258883560_GBA1995-Ch_3-part_1/links/0c96052951e315a54b000000/GBA1995-Ch-3-part-1.pdf

Hawksworth, D. L., & Lücking, R. (2017). Fungal Diversity Revisited: 2.2 to 3.8 Million Species. *Microbiology Spectrum*, *5*(4), 10.1128/microbiolspec.funk-0052–2016. https://doi.org/10.1128/microbiolspec.funk-0052-2016

Hebert, P. D. N., Cywinska, A., Ball, S. L., & deWaard, J. R. (2003). Biological identifications through DNA barcodes. *Proceedings of the Royal Society B: Biological Sciences*, *270*(1512), 313–321. https://doi.org/10.1098/rspb.2002.2218

Helmus, M. R., Bland, T. J., Williams, C. K., & Ives, A. R. (2007). Phylogenetic Measures of Biodiversity. *The American Naturalist*, *169*(3), E68–E83. https://doi.org/10.1086/511334

Hodkinson, I. D., Webb, N. R., & Coulson, S. J. (2002). Primary community assembly on land – the missing stages: Why are the heterotrophic organisms always there first? *Journal of Ecology*, *90*(3), 569–577. https://doi.org/10.1046/j.1365-2745.2002.00696.x

Hu, T., Chitnis, N., Monos, D., & Dinh, A. (2021). Next-generation sequencing technologies: An overview. *Human Immunology*, *82*(11), 801–811. https://doi.org/10.1016/j.humimm.2021.02.012

Hughes, J. B., & Hellmann, J. J. (2005). The Application of Rarefaction Techniques to Molecular Inventories of Microbial Diversity. In *Methods in Enzymology* (Vol. 397, pp. 292–308). Academic Press. https://doi.org/10.1016/S0076-6879(05)97017-1

IPBES. (2019). *Global assessment report on biodiversity and ecosystem services of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services* (Version 1). [object Object]. https://doi.org/10.5281/ZENODO.5657041

Jaureguiberry, P., Titeux, N., Wiemers, M., Bowler, D. E., Coscieme, L., Golden, A. S., Guerra, C. A., Jacob, U., Takahashi, Y., Settele, J., Díaz, S., Molnár, Z., & Purvis, A. (2022). The direct drivers of recent global anthropogenic biodiversity loss. *Science Advances*, *8*(45), eabm9982. https://doi.org/10.1126/sciadv.abm9982

Katoh, K. (2002). MAFFT: A novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Research*, *30*(14), 3059–3066. https://doi.org/10.1093/nar/gkf436

Katoh, K., & Standley, D. M. (2013). MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability. *Molecular Biology and Evolution*, *30*(4), 772–780. https://doi.org/10.1093/molbev/mst010

Köster, J., & Rahmann, S. (2012). Snakemake—A scalable bioinformatics workflow engine. *Bioinformatics*, *28*(19), 2520–2522. https://doi.org/10.1093/bioinformatics/bts480

Laureto, L. M. O., Cianciaruso, M. V., & Samia, D. S. M. (2015). Functional diversity: An overview of its history and applicability. *Natureza & Conservação*, *13*(2), 112–116. https://doi.org/10.1016/j.ncon.2015.11.001

Lozupone, C. A., Hamady, M., Kelley, S. T., & Knight, R. (2007). Quantitative and Qualitative β Diversity Measures Lead to Different Insights into Factors That Structure Microbial Communities. *Applied and Environmental Microbiology*, *73*(5), 1576–1585. https://doi.org/10.1128/AEM.01996-06

Lozupone, C., & Knight, R. (2005). UniFrac: A New Phylogenetic Method for Comparing Microbial Communities. *Applied and Environmental Microbiology*, *71*(12), 8228–8235. https://doi.org/10.1128/AEM.71.12.8228-8235.2005

Lyashevska, O., & Farnsworth, K. D. (2012). How many dimensions of biodiversity do we need? *Ecological Indicators*, *18*, 485–492. https://doi.org/10.1016/j.ecolind.2011.12.016

Magurran, A. E. (2021). Measuring biological diversity. *Current Biology*, *31*(19), R1174–R1177. https://doi.org/10.1016/j.cub.2021.07.049

Mölder, F., Jablonski, K. P., Letcher, B., Hall, M. B., Tomkins-Tinch, C. H., Sochat, V., Forster, J., Lee, S., Twardziok, S. O., Kanitz, A., Wilm, A., Holtgrewe, M., Rahmann, S., Nahnsen, S., & Köster, J. (2021). *Sustainable data analysis with Snakemake* (10:33). F1000Research. https://doi.org/10.12688/f1000research.29032.1

Naturalis. (2023). *eDentity: A DNA metabarcoding facility for rapid identification and monitoring of biodiversity*. Unpublished grant request.

Nilsson, R. H., Anslan, S., Bahram, M., Wurzbacher, C., Baldrian, P., & Tedersoo, L. (2019). Mycobiome diversity: High-throughput sequencing and identification of fungi. *Nature Reviews Microbiology*, *17*(2), 95–109. https://doi.org/10.1038/s41579-018-0116-y

Rivers, A. R., Weber, K. C., Gardner, T. G., Liu, S., & Armstrong, S. D. (2018). ITSxpress: Software to rapidly trim internally transcribed spacer sequences with quality scores for marker gene analysis. *F1000Research*, *7*.

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

Ruppert, K. M., Kline, R. J., & Rahman, M. S. (2019). Past, present, and future perspectives of environmental DNA (eDNA) metabarcoding: A systematic review in methods, monitoring,

and applications of global eDNA. *Global Ecology and Conservation*, *17*, e00547. https://doi.org/10.1016/j.gecco.2019.e00547

Sanders, H. L. (1968). Marine Benthic Diversity: A Comparative Study. *The American Naturalist*, *102*(925), 243–282. https://doi.org/10.1086/282541

Santini, L., Belmaker, J., Costello, M. J., Pereira, H. M., Rossberg, A. G., Schipper, A. M., Ceauşu, S., Dornelas, M., Hilbers, J. P., & Hortal, J. (2017). Assessing the suitability of diversity metrics to detect biodiversity change. *Biological Conservation*, *213*, 341–350.

Schadt, E. E., Turner, S., & Kasarskis, A. (2010). A window into third-generation sequencing. *Human Molecular Genetics*, *19*(R2), R227–R240. https://doi.org/10.1093/hmg/ddq416

Sloggett, C., Goonasekera, N., & Afgan, E. (2013). BioBlend: Automating pipeline analyses within Galaxy and CloudMan. *Bioinformatics*, *29*(13), 1685–1686. https://doi.org/10.1093/bioinformatics/btt199

Socolar, J. B., Gilroy, J. J., Kunin, W. E., & Edwards, D. P. (2016). How Should Beta-Diversity Inform Biodiversity Conservation? *Trends in Ecology & Evolution*, *31*(1), 67–80. https://doi.org/10.1016/j.tree.2015.11.005

Stamatakis, A. (2014). RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, *30*(9), 1312–1313. https://doi.org/10.1093/bioinformatics/btu033

Stamatakis, A., Ludwig, T., & Meier, H. (2005). RAxML-III: A fast program for maximum likelihood-based inference of large phylogenetic trees. *Bioinformatics*, *21*(4), 456–463. https://doi.org/10.1093/bioinformatics/bti191

Swenson, N. G. (2011). Phylogenetic Beta Diversity Metrics, Trait Evolution and Inferring the Functional Beta Diversity of Communities. *PLoS ONE*, *6*(6), e21264. https://doi.org/10.1371/journal.pone.0021264

Taberlet, P., Coissac, E., Pompanon, F., Brochmann, C., & Willerslev, E. (2012). Towards next-generation biodiversity assessment using DNA metabarcoding. *Molecular Ecology*, *21*(8), 2045–2050. https://doi.org/10.1111/j.1365-294X.2012.05470.x

Taylor, H. R., & Gemmell, N. J. (2016). Emerging Technologies to Conserve Biodiversity: Further Opportunities via Genomics. Response to Pimm et al. *Trends in Ecology & Evolution*, *31*(3), 171–172. https://doi.org/10.1016/j.tree.2016.01.002

ten Haaf, L. (2023). *Localized Information Comparison and Analysis for MycoDiversity Database* [Leiden Institute of Advanced Computer Science (LIACS)]. LIACS Thesis Repository. https://theses.liacs.nl/2741

Tensen, L. (2018). Biases in wildlife and conservation research, using felids and canids as a case study. *Global Ecology and Conservation*, *15*, e00423. https://doi.org/10.1016/j.gecco.2018.e00423

The Galaxy Community. (2024). The Galaxy platform for accessible, reproducible, and collaborative data analyses: 2024 update. *Nucleic Acids Research*, *52*(W1), W83–W94. https://doi.org/10.1093/nar/gkae410

Valentini, A., Taberlet, P., Miaud, C., Civade, R., Herder, J., Thomsen, P. F., Bellemain, E., Besnard, A., Coissac, E., Boyer, F., Gaboriaud, C., Jean, P., Poulet, N., Roset, N., Copp, G. H., Geniez, P., Pont, D., Argillier, C., Baudoin, J., … Dejean, T. (2016). Next-generation monitoring of aquatic biodiversity using environmental DNA metabarcoding. *Molecular Ecology*, *25*(4), 929–942. https://doi.org/10.1111/mec.13428

Villéger, S., Grenouillet, G., & Brosse, S. (2013). Decomposing functional β-diversity reveals that low functional β-diversity is driven by low functional turnover in European fish assemblages:

Decomposing functional β-diversity. *Global Ecology and Biogeography*, *22*(6), 671–681. https://doi.org/10.1111/geb.12021

Warwick, R., & Clarke, K. (1995). New “biodiversity” measures reveal a decrease in taxonomic distinctness with increasing stress. *Marine Ecology Progress Series*, *129*, 301–305. https://doi.org/10.3354/meps129301

Willerslev, E., Davison, J., Moora, M., Zobel, M., Coissac, E., Edwards, M. E., Lorenzen, E. D., Vestergård, M., Gussarova, G., Haile, J., Craine, J., Gielly, L., Boessenkool, S., Epp, L. S., Pearman, P. B., Cheddadi, R., Murray, D., Bråthen, K. A., Yoccoz, N., … Taberlet, P. (2014). Fifty thousand years of Arctic vegetation and megafaunal diet. *Nature*, *506*(7486), 47–51. https://doi.org/10.1038/nature12921

Wilsey, B., Chalcraft, D., Bowles, C., & Willig, M. (2005). Relationships among indices suggest that richness ia an incomplete surrogate for grass biodiversity. *Reports Ecology*, *86*, 1178–1184. https://doi.org/10.1890/04-0394