Phylogenetic placement refers to a family of tools and methods to analyze, visualize, and interpret the tsunami of metagenomic sequencing data generated by high-throughput sequencing. Compared to alternative (e.g., similarity-based) methods, it puts metabarcoding sequences into a phylogenetic context using a set of known reference sequences and taking evolutionary history into account. Thereby, one can increase the accuracy of metagenomic surveys and eliminate the requirement for having exact or close matches with existing sequence databases.

Phylogenetic placement constitutes a valuable analysis tool per se, but also entails a plethora of downstream tools to interpret its results. A common use case is to analyze species communities obtained from metagenomic sequencing, for example via taxonomic assignment, diversity quantification, sample comparison, and identification of correlations with environmental variables.

In this review, we provide an overview over the methods developed during the first ten years. In particular, the goals of this review are (i) to motivate the usage of phylogenetic placement and illustrate some of its use cases, (ii) to outline the full workflow, from raw sequences to publishable figures, including best practices, (iii) to introduce the most common tools and methods and their capabilities, (iv) to point out common placement pitfalls and misconceptions, (v) to showcase typical placement-based analyses, and how they can help to analyze, visualize, and interpret phylogenetic placement data.

Contact: lczech@carnegiescience.edu and pierre@barbera-bio.info.

Keywords: Phylogenetic Placement; Evolutionary Placement; Phylogenetics; Phylogenetic Trees; Metagenomics; Metabarcoding; Species Community Composition; Species Diversity; Taxonomic Assignment; Taxonomic Classification; Sequence Identification; Review

1. Introduction

Advances in sequencing technologies enable the broad sequencing of genetic material in environmental samples (1, 2), for instance, from water (3–5), soil (6, 7), and air (8), which is known as environmental DNA (eDNA, 9, 10), or from the human body (11–14) and other sources (15–18). Crucially, this enables the ecological survey of a community of organisms in their immediate environment (i.e., in situ), and allows to directly study the genetic composition of species communities (from viruses to megafauna); a field known as metagenomics (19–22).

Metagenomic data typically stem from so-called High-Throughput Sequencing (HTS, 23–25) technologies, such as Next Generation Sequencing (NGS, 26, 27), as well as later generations (28–32). For a sample of biological material, these technologies typically produce thousands to millions or even billions of short genetic sequences (also called “reads”) with a length of some hundred base pairs length each. Over the past decades, decreasing costs and increasing throughput of sequencing technologies have caused an exponential growth in sequencing data (33), which has now passed the peta-scale barrier (34).

A major analysis step in metagenomic studies is to characterize the reads obtained from an environment by means of comparison to reference sequences of known species (35). A straightforward way to accomplish this is to quantify the similarity between the reads and reference sequences. We obtain an indication of possible novelty if the sequence similarity to known species is low (36, 37). However, such approaches do not provide the user with the evolutionary context of the read, and have been found to incorrectly identify sequences (7, 38, 39).

Instead, general phylogenetic methods can be used directly to classify and characterize the reads, providing highly accurate and information-rich results (40–44). However, trying to resolve the phylogenetic relationships between millions of short reads and the given reference sequences represents a significant computational challenge. Furthermore, as most phylogenetic methods require an alignment of sequences, metagenomic data can often not be used directly, as whole-genome reference data might not be available or computationally intractable. Instead, specific marker genes can be targeted (or filtered from the metagenomic data), which are genetic regions that are well-suited for differentiating between species (45). The use of marker genes to identify species is called DNA (meta-) barcoding (9, 46–48); see Section “Types of Query Sequences” for details.

A powerful and increasingly popular class of methods to identify and analyze diverse (meta-)genomic (barcode) data is the so-called phylogenetic placement (or evolutionary placement) of genetic sequences onto a given fixed phylogenetic
reference tree. By placing unknown, anonymous sequences (in this context called query sequences) into the evolutionary context of a tree, these methods allow for the taxonomic assignment of the sequences (i.e., the association of genomic reads to existing species, for example 43, 49, 50). Moreover, they can also provide information on the evolutionary relationships between these query sequences and the reference species/sequences, and thus go beyond simple species identification. Phylogenetic placement has found applications in a variety of situations, such as data cleaning and retention (7), inference of new clades (51, 52), estimation of ecological profiles (53), identification of low-coverage genomes of viral strains (54), phylogenetic analysis of viruses such as SARS-CoV-2 (55, 56), and in clinical studies of microbial diseases (57).

When analyzing the resulting data, there are two complementary interpretations of phylogenetic placement: (1) as a set of individual sequences, placed with respect to the reference phylogeny, e.g., for taxonomic assignment, phylogeographic tracing, or even possible clinical relevance; (2) as a combined distribution of sequences on the tree, characterizing the sampled environment at a given point in time or space to examine the composition of a species community as a whole, for instance as a means of sample ordination and visualization, and association with environmental variables.

In this review, we provide an overview of existing methods to conduct phylogenetic placement, as well as post-analysis methods for visualization and knowledge inference from placement data. We also discuss some practical aspects, such as common pitfalls and misconceptions, as well as caveats and limitations of these methods. We mainly refer to metagenomic input data (or more accurately, metabarcoding data, see below for details) as it represents the most common use case, but also highlight some alternative use cases where phylogenetic placement is employed for other types of sequence data.

2. Phylogenetic Placement

2.1. Overview and Terminology

The modern approach to phylogenetic tree inference is based on molecular sequence data, and uses stochastic models of sequence evolution (58) to infer the tree topology and its branch lengths (59, 60). Note that the computational cost to infer the optimal tree under the given optimality criterion grows super-exponentially in the number of sequences (59). In addition, large trees comprising more than a couple of hundred sequences are often cumbersome to visualize, rendering the approach challenging for current (e.g., metagenomic) large datasets. Furthermore, the lack of phylogenetic signal contained in the short reads of most HTS technology usually does not suffice for a robust tree inference (51, 61–63). Hence, phylogenetic placement emerged from the demand to obtain phylogenetic information about sequence sets that are too large in number and too short in length to infer comprehensive phylogenetic trees (64, 65). In a metagenomic context, a set of sequences obtained from an environment such as water, soil, or the human body, is here called a sample. This is often the data that we intend to place, and might have further metadata associated with it, e.g., environmental factors/variables such as temperature or geo-locations where the sample was taken.

Generally, the input of a phylogenetic placement analysis is a phylogenetic Reference Tree (RT) consisting of sequences spanning the genetic diversity that is expected in the sequences to be placed into the tree. The tree can be rooted or unrooted; in the latter case however, a “virtual” root (or top-level trifurcation) is used in the computation as a fixed point of reference (66). Then, for a single sequence (e.g., a short read), in this context called a Query Sequence (QS), the goal of phylogenetic placement is to determine the branches of the RT to which the QS is most closely evolutionarily related. Note that the RT is kept fixed, that is, the QSs are not inserted as new branches into the tree, but rather “mapped” onto its branches. Hence, the phylogenetic relationships between individual QSs are not resolved.

This is the key insight that makes it possible to efficiently compute the placement of large numbers of QSs. By only determining the evolutionary relationship between the sequences of the RT and each individual QS, the process can be efficiently parallelized, and the required processing time scales linearly in the number of QS. Furthermore, this allows us to consider multiple branches as potential Placement Locations for a given QS, representing uncertainty in the placement, often expressed as a probability (or confidence) of the QS being placed on that branch. This uncertainty might result from weak phylogenetic signal, or might indicate some other issue with the data, as explained later. In maximum-likelihood (ML) based placement (see Section “Maximum Likelihood Placement” for details), these probabilities are computed as the Likelihood Weight Ratio (LWR) resulting from the evaluation of placing the QS attached to an additional (hypothetical) branch into the tree. Hence, for historic reasons, the probability of a placement location (one QS placed on a specific branch) is often called its LWR, and for a given QS, the sum of LWRs over all branches is 1 (equivalent to the total probability). See Table 1 for an overview of different placement tools, and which of the aforementioned quantities they can compute.

In other words, phylogenetic placement can be thought of as an all-to-all mapping from QSs to branches of the RT,
with a probability for each placement location, as shown in Figure 1(d) and Figure 1(e). We can however also interpret each such placement location as if it was an extra branch inserted into the RT, as shown in Figure 1(b) and Figure 1(c). In particular, maximum likelihood placement makes use of its underlying evolutionary model to also estimate the involved branch lengths that are altered through the insertion of a QS, see Figure 1(b) for details. This interpretation highlights the aspect of each individual QS being part of the underlying phylogeny. For example, this allows its taxonomic assignment to that clade of the reference tree where the QS shows the highest accumulated placement probability, as explained later.

**Misconceptions.** In the existing literature, and from our experience in teaching the topic as well as supporting the users of our software, some concepts of phylogenetic placement are not always well explained or understood. Although we have introduced these concepts above already, we briefly address two common misconceptions here, for clarity.

Firstly, a common misconception is that the tree is amended by the QSs, that is, that new branches are added to the RT, and that the phylogenetic relationships of the QSs with each other are hence resolved. This is not the case; instead, the RT is kept fixed, the QSs are only aligned against the reference alignment, but not against each other (in ML placement), and the QSs are mapped only to the existing branches in the RT. This mapping can however be interpreted “as if” the QS was a new terminal node (leaf or tip) of the tree, usually inserted (or “grafted”) into the branch with the most probable placement location, which can be useful in some applications.

Secondly, a further common misconception is that a QS is only placed onto a single branch, or that only the best (most likely) placement location is taken as the result for each placed QS. Instead, each branch is seen as a potential placement location with a certain probability, which sum to one over the tree. It can however be useful to reduce the placement distribution of a QS to only its most probable placement location. Also, for practical reasons, typically not all locations are stored in the resulting file (or even considered in the computation by application of heuristics), as low probability locations can often be discarded to save storage space and downstream processing time; see Section “File Format” for details. Lastly, some placement methods do only output a single best placement, see Table 1.

In summary, phylogenetic placement yields a distribution of potential locations of where a QS could be attached in the RT – but it does not extend the RT by the QS with an actual branch.

**File Format.** Placement data is usually stored in the so-called jplace format (67), which is based on the json format (68, 69). See Figure 2 for an example. It uses a custom augmentation of the Newick format (70) to store the reference tree, where each branch is additionally annotated by a unique edge number, so that placement locations can easily refer to the branches. For each QS (named via the list “n”), the
Fig. 2. Jplace format for phylogenetic placement. The exemplary file consists of a reference “tree” in a custom Newick format that annotates edge numbers in curly brackets, followed by two pqueries, which is the term for combined lists of sequence names and their placement locations. The first pquery contains two placement locations (“p”) for two query sequences (“n”), and the second contains a single location (“p”) for two other sequences including their multiplicities/abundances (“nm”). The order to interpret the values per location is given via the “fields” list, and highlighted by colors here; additional “metadata” and a “version” of the file format can be given. Example adapted from (67).

format then stores a set of possible placement locations (in the list “p”), where each location is described by the values: (1) "edge_num", which identifies the branch of this placement location, (2) "likelihood", which is used by maximum likelihood based placement methods, (3) "like_weight_ratio" (LWR), which denotes the probability (or confidence) of this placement location for the given QS, (4) "distal_length" and (5) "pendant_length", which are the branch lengths involved in the placement of the QS for the given placement location; see Figure 1(b) for an explanation of these lengths.

These five data fields are the standard fields of the Jplace format; further fields can be added as needed. As noted above, typically not all placement locations for a given QS are stored in the file, as low probability placements unnecessarily increase the file size without providing substantial information; in that case, the sum of the stored LWR values might actually be smaller than 1.

The format furthermore allows for multiple names in the "n" list, as well as assigning a “multiplicity” to each such name (by using a list called "nm" instead of "n"). For instance, this allows to only store the placement locations for identical reads once, while keeping track of the original raw abundances of these reads or OTUs. A pair of a "n"/"nm" list and a "p" list is called a “pquery”, and describes a set of placement locations for one or more (identical) QSs. This structure is then repeated for each QS that has been placed.

To our knowledge, the genesis library (71) is the only general purpose toolkit for working with, and manipulating, placement data in Jplace format. It also incorporates many of the downstream visualization and analysis techniques we describe later on. Some other tools that offer basic capability to work with jplace files are BoSSA (72), ggtree (73), and treemo (74), all of which can read jplace files for processing in R.

With the release of several placement tools that do not use the ML framework, see Section “Distance-Based Placement”, the jplace file format (67) may require an update. The standard is written currently (as of version 3) with placement properties such as branch lengths and likelihood scores in mind, which do not translate well to other types of placement algorithms (pers. comm. with S. Mirarab, July 2020). Furthermore, it might be helpful to support sample names, multiple samples per file, and additional per-sample or even per-query annotations and other metadata in the file format. Being based on json, this can already be achieved now by adding these entries ad-hoc, but would lack support by parsers if not properly standardized.

2.2. Types of Query Sequences

In principle, any type of genetic sequence data can be subjected to placement, as long as the reference sequences span the genomic regions where the query sequences originate from. Apart from the availability of suitable reference sequences used to construct a reference tree (see Section “Sequence Selection”), the primary limiting factor is the extent to which a given placement tool supports the data. Currently, the majority of placement tools supports nucleotide (DNA/RNA) and amino acid (protein) data. Many placement methods require query reads to be aligned to the reference, i.e. they need to be homologs.

Metabarcoding and Amplicons. For the above reasons, a common approach to obtain sequences is metabarcoding (9, 46–48). In metabarcoding, one or several marker or barcoding genes, such as 16S (75), 18S (76), ITS, COI, etc. (2, 77–79) are typically chosen to compute the reference alignment, and appropriate primers are selected to enable metabarcoding sequencing of the sample (9). A marker gene should be universally present in the studied organisms, and ideally should only occur once in the genome of each organism (51, 80), i.e., be single-copy. In practice, marker genes often occur multiple times per genome, possibly requiring the need for copy number correction. A marker gene should exhibit sufficient between-species variation to distinguish them from each other, but show low within-species variation (48). Using a metabarcoding approach has several advantages: it targets loci of interest and focuses the sequencing effort there (incidentally also limiting the size of the reference MSA), barcoding genes are typically well suited for phylogenetics (stable regions to aid alignment paired with variable regions to discriminate organisms), and the approach is generally cost-effective. Such approaches use amplicon sequencing (37, 81), wherein only DNA originating from the targeted region is amplified using the Polymerase Chain Reaction (PCR, 82), thus yielding the subsequent sequencing of any remaining DNA fragments from other regions highly improbable. The resulting amplicon sequences have been shown to be well-suited for phylogenetic placement (7, 83).

However, PCR-based amplifications are known to introduce biases in the abundance of the sequencing reads, as some fragments may be copied with a higher likelihood than others (84, 85). Similarly, a further bias that skews abundance results exists as different organisms may have a different number of copies of the targeted gene, ranging from single copies to 15
copies, depending on the organism (86). Some methods exist that attempt to account for copy number bias (87–89) as well as for PCR amplification bias (90, 91).

When an untargeted sequencing approach is chosen instead (such as shotgun metagenomic sequencing), using a broader scope for the reference sequences may be advisable, such as using whole genome data. This might only be feasible for small genomes such as some viruses or mitochondrial DNA. Alternatively, a sensible approach is to filter out any reads that did likely not originate from the genetic regions that constitute the reference alignment. This can be achieved, for example, using hmmsearch from the HMMER-package (92, 93), which allows the user to obtain a list of reads that have an alignment score above a given threshold. Similarly, so-called ans tags (84) represent a shotgun-based alternative to amplicon sequencing.

Recently, placement methods have emerged that do not require the alignment of query sequences to a reference, and some do not even require the references to be aligned against each other (see Section “Distance-Based Placement”). However, establishing that query reads and reference sequences are homologous is still necessary.

Sequencing Technologies. A further consideration is the choice of sequencing technology, with the primary property being the length of the resulting sequencing reads. So far, the vast majority of studies utilizing phylogenetic placement have relied on short-read sequencing technologies such as NGS, using by now well established protocols to perform broad low-cost sequencing (94). However, this approach produces very short (150-400 nucleotide) reads, that typically only cover fragments of a reference gene. For universal single-copy markers, this can limit their applicability to phylogenetics due to the lower information content. However, the approach has been applied successfully to other types of data (95, 96).

More recent sequencing technologies, called third generation sequencing, or long-read sequencing (LRS), yield individual reads that cover entire genes, or even entire genomes (97). While placement was originally developed for short read sequencing, longer read lengths typically increase the phylogenetic signal contained in reads, thus increasing the reliability of phylogenetic methods. Indeed, such sequence data have been shown to overcome this fundamental hurdle to phylogenetically resolving the relationships between query sequences that originally gave rise to phylogenetic placement (43).

An emerging third way to obtain longer reads is to combine short reads into longer so-called Synthetic Long-Reads (SLRs), which have been used successfully to characterize metagenomes (98, 99) and which improve upon short-read metabarcoding approaches for taxonomic classification (43, 100, 101).

Clustering. Once the wet-lab sequencing strategy has been determined, a user eventually obtains a (typically large) set of sequences. After quality control, a potential next step is to consider if, and how, to cluster these raw sequences in order to reduce the amount of data that has to be processed, often at the cost of losing information. Common choices include clustering by similarity threshold (≥ 97%) resulting in Operational Taxonomic Units (OTUs, 102–106), more strictly based on single nucleotide differences resulting in Amplicon Sequencing Variants (ASVs, 107), or more recent alternatives such as SWARM clustering (108). These methods are most commonly used for clustering reads from marker regions, and hence applicable in the placement context; for a comprehensive review of clustering methods, see (109).

If possible, it is recommended to avoid clustering, in order to retain potential phylogenetic signal; this choice however also depends on study design and goals. However, even if sequences are not clustered, we strongly recommend dereplication, that is, removal of exact (strict) duplicates of sequences, to avoid unnecessary redundant computations. For the same reason, sequence dereplication is also useful when pooling the sequences from multiple samples together and placing the resulting set via a single placement run. Tools that offer this capability include USEARCH (103), and VSEARCH (105), as well as the placement-specific CHUNKIFY command in GAPPA (71).

Outgroup Rooting. Finally, an often overlooked source of query sequences are high-quality reference sequences databases. Here, the use-case of placement shifts away from taxonomic assignment: instead such data can be used to attempt an outgroup rooting of an existing tree, using already classified sequences (55, 110, 111). The result of placement, in this case, is a set of suggested branches on which to root the tree, including a probability estimate for each root placement onto each branch (111).

2.3. Reference Sequences, Alignment, and Tree

The phylogenetic reference tree (RT), inferred from a set of reference sequences (RSs) using their alignment (Reference Alignment, RA), is the foundation and scaffold for conducting phylogenetic placement. Ideally, to avoid duplicating work, to ensure high quality, and to provide stable points of reference for comparison between studies, suitable reference trees should be provided by the respective research/organismal communities. First efforts for microbial eukaryotes are on their way (112–115), although some of these are not designed explicitly for phylogenetic placements, but more taxonomic groups will follow. As such, references are however not yet available for all taxonomic groups, we here provide an overview of the process (see also 7, 114, for practical examples).

Sequence Selection. As phylogenetic placement cannot infer evolutionary relationships below the taxonomic level of the reference tree, the first step is the selection of suitable RSs, which should (i) cover the diversity that is expected in the query sequences (QSs), and (ii) be well-established and representative for their respective clades to facilitate meaningful interpretation. In order to capture unexpected diversity and potential outliers, it can be advantageous to include a wider range of sequences as well (7), or to run preliminary tests and filtering (placement- or similarity-based) with a broad reference to ensure that all diversity in the QSs is accounted for.

In many cases, the selection process is (unfortunately) labor-intense, as it requires hand-selecting known sequences from reference databases such as SILVA (116–118), NCBI (119, 120), GREENGENES (121, 122), or RDP (123, 124). This manual process however also often provides the highest quality, and allows to optimally assemble the RSs for a given project. See also (125) for a comparison of these databases.

Important selection criteria are the number of sequences to be selected, as well as their diversity; both of which depend on the study design and goals. Generally, a number of RSs in the order of hundreds to a few thousands has shown to provide enough coverage for most QS datasets, while still
being small enough to properly visualize their phylogeny and to conduct all necessary computations in reasonable time. Often, it is sufficient to include a single species to represent a whole clade (115). Depending on the types of downstream analyses, it can be a disadvantage to select sequences that are too similar to each other (i.e., closely related species, or different strains of the same species), as this can spread the placement distribution across nearby branches. In other words, placements with similar probability in many branches are mostly a consequence of reference alignment regions for which large subtrees contain (almost) identical sequences. This is however expected when conducting taxonomic assignment at species or below-species level, and the reference should be built with the targeted taxonomic resolution in mind.

On the other hand, if the QSs contain enough phylogenetic signal (e.g., when using long reads, whole genome data, or when the target gene has sufficient variability), including multiple representatives of a taxonomic group might allow to obtain more finely resolved placements. For example, in short genomes such as HIV or arthropod mitochondria, where mutations are not concentrated in specific regions but spread all over the genome, reads matching a reference alignment region likely show a decent amount of variation, making placements exploitable (126).

Lastly, the RSs need to at least span the genomic region that the QSs come from. For a more robust inference of the RT however, it can be advantageous to include a larger region with more phylogenetic signal. Theoretically, if one wanted to place shotgun sequences from entire genomes, whole-genome RSs would be needed.

As an alternative to manual selection, the Phylogenetic Automatic Reference Tree (PhAT, 127) is a method that uses reference taxonomy databases to select suitable RSs which represent the diversity of (subsets of) the database. In cases where taxonomic resolution at the species-level does not require expert curation, the PhAT method can provide a basis for rapid data exploration, and help to obtain an overview of the data and its intrinsic diversity.

Reference Alignment Computation. Next, for ML-based tree inference and placement, the RSs need to be aligned against each other to obtain the reference alignment (RA). Typically, this is conducted with de novo multiple sequence alignment tools such as T-Coffee (128), MUSCLE (129), MAFFT (130), and others; see (131–133) for reviews. Recently, MUSCLE v5 introduced an interesting new approach that generates alignment ensembles to capture alignment uncertainty (134, preprint). In the ML framework, the QSs also need to be aligned against the RA, see next section.

Tree Inference. Finally, given the RA, a phylogenetic tree of the RSs is inferred, which is henceforth used as the reference tree (RT); see (135) for a general review on this topic. In theory, any method that yields a fully resolved (bifurcating) tree is applicable, e.g., neighbor joining (136), maximum parsimony (137), or Bayesian inference (60, 138). In practice however, maximum likelihood (ML) tree inference (60, 139) is preferred, in particular when using ML-based placement, as otherwise inconsistencies in the assumed models of sequence evolution can affect placement accuracy. To this end, common software tools include IQ-TREE (140), FASTTREE2 (141), and RAxML (142, 143); see (144) for a review and evaluation of ML-based tree inference tools. An open research question in this context is how to incorporate uncertainty in the tree inference (and in the alignment computation) with phylogenetic placement (134, 145, 146).

Alignment of Query Sequences. For many placement methods, the query sequences need to be aligned against the reference alignment. In principle, de novo alignment methods can be deployed to obtain a comprehensive alignment of both the reference and query sequences. These tools are however not intended for HTS data, and are not well suited for handling the heterogeneity of phylogenetic placement data, with (typically) longer, curated, high-quality reference sequences, and short lower-quality reads (query sequences).

Hence, with the rise of high-throughput sequencing, specialized tools have been developed that extend a given (reference) alignment without fully recomputing the entire alignment. In the context of phylogenetic placement, there are two additional advantages that can be exploited to improve efficiency: (i) query sequences only need to be aligned against the reference, but not against each other (as their phylogenetic relationship is not resolved during placement), and (ii) insertions into the reference that result from aligning a QS against the reference can be omitted as they do not contain any phylogenetic signal for the placement of the QS.

In the simplest case, only the reference alignment and query sequences are required as input. For instance, the hmmalign command of HMMER (92, 93) can align query sequences to the reference alignment using a profile Hidden Markov Model (HMM) built from the reference alignment. Note that the option --m has to be set in order to not insert columns of gaps into the reference. Alternatively, the MAFFT command --addfragments (147) uses an internally constructed guide tree built from a pairwise distance matrix of the reference alignment to aid the alignment process; here, the option --keeplength has to be set to not add columns of gaps to the reference.

Furthermore, the PaPaRa tool (148, 149) can be used that was specifically developed to target phylogenetic placement. It takes the RT as additional input, and uses inferred ancestral sequences at the inner nodes of the tree to improve the alignment process. Here, the option --m has to be set to not insert columns of gaps into the reference. Similarly, PAGAN (150) also utilizes the information in the reference tree, but it does extend the reference alignment with gaps as needed for the query sequence, causing higher computational effort during placement.

Note that typically, read mapping tools such as BOWTIE2 (151) or BWA (152, 153) are not recommended for phylogenetic placement, as they expect low-divergent sequences as input, e.g., from a single species.

2.4. General Purpose Placement Methods

Once initial tasks such as reference tree creation and sequence alignment are completed, the actual placement can commence. There exist several distinct algorithmic approaches for conducting the core part of phylogenetic placement, which we introduce here; see Table 1 for an overview.

Maximum Likelihood Placement. Maximum Likelihood (ML) is a statistically interpretable and robust general inference framework, and one of the most common approaches for phylogenetic tree inference (59, 60, 139). It works by searching through the
Therefore, under the ML framework, we strongly recommend one? Uncertainty: Is there some measure of uncertainty (such as Table 1. General purpose placement tools. This table compares the features of the general purpose (i.e., not use-case specific) phylogenetic placement tools. Columns are as follows. Alignment: Does the tool need the QSs to be aligned against the reference alignment? Multiple: Does the tool produce multiple placement locations per QS, or just a single (best) one? Uncertainty: Is there some measure of uncertainty (such as LWR) assigned to each placement location? Branch Length: Does the tool compute the involved branch lengths at each placement location for each QS?

| Placement Tool | Alignment | Multiple | Uncertainty | Branch Lengths |
|----------------|-----------|----------|-------------|----------------|
| PPLACER        | yes       | yes      | yes         | yes            |
| RAxML-EPA      | yes       | yes      | yes         | yes            |
| EPA-NG         | yes       | yes      | yes         | yes            |
| RAPPAS         | no        | yes      | yes         | no             |
| APPLES         | no        | no       | no          | yes            |
| App-SHAM       | no        | no       | no          | yes            |

super-exponentially large space of potential tree topologies for a given set of sequences (taxa), and computing the phylogenetic likelihood of the sequence data of these taxa being the result of the evolutionary relationships between the taxa as described by each potential tree, while also computing branch lengths of the tree. The result of this inference is the tree topology one is able to find using some heuristic search strategy that best (most likely) “explains” the underlying sequence data. Due to the NP-hardness of the tree search problem, the best tree one can find might not be the globally best one.

To calculate this likelihood, ML methods use statistical models of sequence evolution that describe substitutions between sequences (insertions and deletions are mostly ignored; it is hence also called a substitution model), see (58) for a review. Consequently, the estimated parameters of these models are an inherent property of the resulting phylogenetic tree. The choice of model parameters also directly informs the specific branch lengths of a tree, interpreting a tree under a different set of model parameters thus may lead to inconsistencies. Therefore, under the ML framework, we strongly recommend to use the same substitution model and parameters for tree inference and for phylogenetic placement.

Based on the general ML tree inference framework, ML-based phylogenetic placement works in two steps: First, the QSs are aligned against the RA as described above, and second, using the resulting comprehensive alignment with both reference and query sequences, the QSs are placed on the RT using the maximum likelihood method to evaluate possible placement locations (64, 65, 154).

Standard methods used in ML tree inference use search heuristics to explore some possible tree topologies for a given set of sequences. Instead, for a given QS, ML-based placement only searches through the branches of the reference tree (RT) as potential placement locations for the QS. That is, each branch of the RT is evaluated as a placement location, and branch lengths of the involved branches are optimized, following the same approaches as for de novo tree inference. However, the distal and proximal branch lengths of the placement (see Figure 1(b) for details) are typically re-scaled, so that their sum is equal to the original branch length in the RT. Finally, the phylogenetic likelihood of the tree with the QS amended as a temporary extra taxon is calculated.

For each QS and each branch of the RT, this process yields a likelihood score (which is stored in the jplace format, see Section “File Format”). The Likelihood Weight Ratio (LWR) of a placement location is then computed as the ratio between this likelihood score and the sum over all likelihood scores for the QS across the entire tree (63, 155). These likelihood scores sum to one across all branches, and hence express the confidence (or probability) of the QS being placed on a given branch.

The first two tools to conduct phylogenetic placement in an ML framework were the simultaneously published (as preprints) pplacer (64) and RAxML-EPA (65). Both build on the same general ML concepts, but employ different strategies for improving computational efficiency, e.g., by heuristically limiting the number of evaluated branches (potential placement locations). Additionally, pplacer offers a Bayesian placement mode. The more recent EPA-NG (156) tool combines features from both pplacer and RAxML-EPA, is substantially faster and more scalable on large numbers of cores, and hence is the recommended tool for ML-based placement.

Ancestral-Reconstruction-Based Placement. Recently, multiple methods were introduced that do not rely on aligning query sequences to a reference MSA. The first such group of methods is based on reconstructing ancestral states at interior nodes of the reference tree, again using an ML framework. From these ancestral sequences, k-mers are generated and associated with the branches of the reference tree. Subsequently, phylogenetic placement is performed by comparing the constituent k-mers of a QS with the set of k-mers indexing the reference tree branches, thereby obviating the need for QS alignment. This is the general approach used in both RAPPAS (157) and LSHPLACE (158).

It should be noted that using this procedure, distal and pendant branch lengths of a given RT branch are determined during the association of k-mers with RT branches, meaning that all placements on a given branch have the same fixed location. This means that an additional step to conduct branch length optimization that is not directly offered by RAPPAS or LSHPLACE may be required to obtain more realistic placement branch lengths. RAPPAS however does produce multiple placements per QS and calculates a confidence measure akin to the LWR, yielding a distribution for placing a single QS onto different branches of the tree.

Distance-Based Placement. Finally, the most recent placement approaches utilize methods from distance-based phylogenetic inference.

For example, APPLES (159) is based on the least-squares criterion for tree reconstruction (59). For a given tree, the least-squares method calculates the difference between the pairwise sequence distances and the pairwise patristic distances (i.e., the path lengths between two leaves). A least-squares optimal tree is the tree for which this difference is minimized. In APPLES, this criterion is used to score possible placement locations of a QS on an existing tree, returning the branch which minimizes the between-distances difference. A key advantage of the least-squares approach is its ability to efficiently handle reference trees with hundreds of thousands of leaves, which is currently not computationally feasible using ML methods. Further, the method does not require an alignment of the sequences involved, requiring only a measure of pairwise distance between them. Note however that as these methods still require a reference tree, computing a reference MSA may still be needed, unless the tree is inferred via distance-based methods as well. Consequently, even unassembled sequences,
such as genome skims (160), may be used both as reference and query sequences. Recently, an updated APPLES-2 was published that further improves upon the scalability and accuracy of the tool (161). Note also that APPLES can take as input, but does not require, aligned sequences.

The most recent alignment-free method is App-SPAM (162). It utilizes the concept of a spaced-word, which can be understood as a type of k-mer for which only some characters have to be identical for two subsequences to be considered as having the same k-mer. This relaxed equality definition is informed by a binary pattern, indicating for each site of a spaced word whether it should be taken into account (1) or disregarded (0). Building on this, the tool calculates pairwise distances between a QS and the RSs based on the number of shared spaced-words. Subsequently, the tool identifies the placement branch of a QS as either the terminal branch of the closest RS, or the branch leading to the parental node of the LCA of the two closest RSs, depending on the strength of the signal of the closest RS. Notably, App-SPAM is able to provide both distal and pendant branch lengths for the placements it produces, and does so using an estimated phylogenetic distance (the Jukes-Cantor distance, 163). Note that both APPLES and App-SPAM only produce a single placement per QS and can therefore not offer statistical measures of placement uncertainty such as the LWR.

Generally, distance-based placement methods produce results with lower accuracy compared to ML-based placement, though this gap appears to be narrowing. These newer approaches do however expand the scope of placement to sizes of reference trees, and lengths of reference sequences, that are orders of magnitude larger than what is currently possible with ML methods.

2.5. Application-Specific Placement Methods

Several additional placement methods exist. We provide a survey of these in this section. The placement methods covered in this section set themselves apart through their more specific use-cases, however this does not imply that their scope of use is necessarily limited.

**Viral Data.** A particularly challenging use case for phylogenetic methods is the investigation of viral data, with a highly relevant example coming from the SARS-CoV-2 pandemic. Due to the dense sampling involved in studying such viral outbreaks, differences between individual taxa in a prospective tree may only be due to a very low number of, or even single, mutations. Consequently the amount of phylogenetic signal is generally very low, complicating tree reconstruction (55). Yet, distinguishing between major viral variants and identifying them precisely from a given clinical sample is crucial for epidemiological studies. In this context the UShEr software was introduced that specifically focuses on phylogenetic placement of SARS-CoV-2 sequences (56). In contrast to ML methods, UShEr uses a Maximum Parsimony (MP) approach, and does not operate on the full sequence alignment. This allows the method to focus directly on individual mutations, and consequently only use a fraction of the runtime and memory footprint of conventional ML placement methods. Note that the accuracy of MP-based phylogenetic methods can suffer when one or more lineages in the tree have experienced rapid evolution that results in long branch lengths. In such cases MP may incorrectly determine such lineages to be closely related, an effect termed long branch attraction (164, 165). While this is less of an issue for very closely related sequences such as SARS-CoV-2 or other (but not all) viral data, it may yield the application of such approaches to different types of data more challenging.

**Gene Trees.** In principle, all placement methods aim to provide the location of a QS on a phylogeny that accurately reflects the underlying pattern of speciation, i.e., the species tree. In practice, the reference tree is typically only inferred on a single gene (16S, 18S, ITS, etc.), yielding a gene tree which may substantially differ from the species tree, called gene-tree discordance (166). Alternatively, we may have multiple such gene trees that induce a species tree, and subsequently want to perform query placement onto the species tree via placement onto the constituent gene trees (2). Currently, only two placement methods are able to handle such cases: INSTRAL and DEPP. INSTRAL (167) performs placement of QSs for a species tree induced by a set of gene trees. It does so by first placing into the individual gene trees using existing ML placement methods, then re-inferring the species tree from the extended gene trees. In contrast to this, DEPP (168, preprint) only considers the problem of discordance between a gene tree and its species tree and attempts to account for this during the placement into the species tree. The approach is based on a model of gene tree discordance learned from the data using deep neural networks that yields an embedding of given sequences into a euclidean space. Incidentally, this makes DEPP the first and so-far only phylogenetic placement method to incorporate machine learning. DEPP then uses the pairwise distances that result from the embedding of both reference and query sequences as input to APPLES, which computes the least-squares placement of the QSs.

**Other Use Cases.** Some further tools make application-specific usage of placement. The first pertains to the specific case of samples containing sequences from exactly two organisms, and the task of identifying their respective known reference organisms. The tool MISA was developed with this specific use-case in mind (169).

The second relates to either placing morphological sequences from fossils typically represented by binary characters (presence/absence of a trait) or Ancient DNA (aDNA) sequences. Placing ancient DNA sequences is generally challenging for analysis because of the high degree of degradation due to the age of the DNA molecules, generally shorter read lengths ranging between 50 and 150 base pairs, and post-mortem deamination (170). The PATHPHYNDER tool aims to solve this use-case (171, preprint). Like UShEr, PATHPHYNDER operates on nucleotide variants, focusing on single nucleotide polymorphisms. Furthermore, phylogenetic placement has been used for placement of fossils (172, 173) using morphological data. This approach uses the maximum likelihood framework to use the signal from mixed morphological (binary) and molecular partitions in the underlying MSA.

Lastly, phylogenetic placement has also been proposed as a way to perform OTU clustering. The HMMUFOTU (174) tool implements this specific use-case, along with automated taxonomic assignment (see also Section “Taxonomic Classification and Functional Analysis”). A unique characteristic in comparison to other placement tools is that HMMUFOTU also performs QS alignment and uses this information to pre-select promising placement locations.
2.6. Workflows based on Phylogenetic Placement

Over the last decade, several pipelines have been published that use phylogenetic placement tools as their core method, building on it and using its result in various ways.

Automated Analysis Pipelines. One class of placement pipelines focus on simplifying the overall use of placement methods, typically providing the user with the option to use a pre-computed reference tree, obviating the need for manual selection of reference taxa (154, 175–180). A number of these pipelines also automate the generation of key metrics and downstream analysis steps. Among these pipelines, of particular note is PICRUSt2 (177, 178), which stands out for accounting for 16S copy number correction, and providing the user with a prediction of the functional content of a sample. Similarly, PAPRICA (179) is a pipeline that computes metabolic pathway predictions for bacterial metagenomic sample data.

Divide-and-Conquer Placement. A further key challenge for existing phylogenetic placement tools is scalability with regards to the size of the reference tree. While more recent methods have shown significant improvements in both the memory footprint and execution time required when placing QSs on reference trees on the order of 10^6 reference taxa (see Section “Distance-Based Placement”), such input sizes remain extremely challenging for ML-based placement methods. A number of workflows have been proposed to scale existing placement methods for this use-case by splitting up the reference tree into smaller subtrees on which phylogenetic placement is then performed, creating a divide-and-conquer approach to phylogenetic placement (71, 127, 181–183). These approaches vary primarily in how they select subtrees. SEPP (181) and PPLACERDC (182) generate a subtree based on the topology of the reference tree. SEPP is a general boosting technique in particular for highly diverse reference trees (181, 184). Further, a multi-level placement approach exists (71, 127), which first places onto a broad RT, and then extracts QSs in pre-selected clades of that RT to place them again onto clade-specific high-resolution RTs. Finally, PPLACER-XR (183) selects a set of neighboring reference branches based on similarity to each query sequence, out of which it creates a subtree. Note that in this case, when decomposing the reference tree differently for every query sequence, scalability with regards to the number of query sequences is severely reduced.

A central promise of placement on very large trees is to simplify the curation and engineering tasks involved in creating a reference tree, as here a typical challenge is to decide which taxa to include in the tree. If placement can instead be performed on a tree encompassing an entire database, the curation challenge is circumvented. However, as another common issue with reference tree generation is the inclusion of overly similar reference sequences resulting in unclear or fuzzy placement signal, divide-and-conquer placement approaches may not be sufficient on their own.

Evaluation of Placement Tools. Lastly, PEWO is an extensible testing framework specifically aimed at benchmarking and comparing different phylogenetic placement software (126). It includes a wide range of datasets and thus provides an important resource for identifying which placement tool is best suited for specific use-cases by evaluating the accuracy of existing tools, given some dataset. PEWO does so using a pruning-based evaluation procedure, where a subset of leaves is removed from a reference tree. This subset of sequences is subsequently used as input QSs for placement. The accuracy of a placement is calculated as the number of nodes between the best placement location, and the original location of the QS on the reference tree (called the node distance). This basic approach is used for evaluation in most publications that introduce new placement approaches. Note that the node distance measures two sources of error: error introduced by the placement algorithm, and error introduced by the pruning of the reference tree. In contrast to this, the “delta error” used in the evaluation of APPLES measures the additional error introduced through placement, in addition to the error introduced by the process of altering the reference tree through pruning (159). This new metric is however not yet included in the PEWO workflow. Nevertheless, the usefulness of a comprehensive and standardized testing framework cannot be emphasized enough, as it substantially facilitates further advancement and standardization in the field and the development of novel methods.

3. Visualization and Analysis

As mentioned before, there are two ways to conceptualize phylogenetic placement: (i) as an assignment (or mapping) of individual sequences to the branches of a phylogeny, usually taking the (n-)most likely placement location(s) of each sequence, or (ii) as the distribution of all sequences of a sample across the tree, taking their respective abundances and placement probabilities into account. The former is similar to taxonomic assignment, but with full phylogenetic resolution instead of resolution at the taxonomic levels only, while the latter focuses on, e.g., species communities and their diversity as a whole. In the following we provide an overview of analysis methods that make use of such data.

Abundances and Multiplicities. In both interpretations, an important consideration is whether to take sequence abundances into account. When working with strictly identical sequences, or sequences resulting from some (OTU) clustering, the number of occurrences of each sequence or size of each cluster can be used as additional information for interpreting, e.g., community structure. On the one hand, including their abundances with the placement of each sequence yields information on how prevalent the species of these sequences are; for example, this can provide insight into the key (most abundant) species in environmental samples. On the other hand, dropping abundances and instead considering each sequence once (as a singleton) is more useful for estimating total diversity and taxonomic composition. For example, this way the number of distinct sequences can be regarded as a proxy for the number of species that are present in a sample. Whether to include abundances should hence be decided depending on the type of analysis conducted.

In the jplace format, these abundances can be stored as the so-called “multiplicity” of each placement (67), in the "nm" data field. Unfortunately, the fasta (185) and phylip (186) formats used as input to placement do not natively support abundance annotations, and current placement tools often do not handle them automatically, meaning that the information can be lost. However, the CHUNKIFY workflow (71, 127) mentioned in Section “Clustering” takes abundances into account and annotates them as multiplicities in the resulting jplace file. Furthermore, GAPPA (71) offers a command to
edit the multiplicities as needed, for example setting them post-hoc to the initial sequence abundance determination.

**Visualization.** Prior to more in-depth analyses, a first step in most workflows is a visualization of the immediate results. Following the two interpretations of phylogenetic placement (and hence, depending on the research question at hand), there are several ways to visualize placement results.

First, individual placements can be shown as actual branches attached to the RT, e.g., Figure 1(e). Typically, only the most likely placement location per sequence is used for this, in order to avoid cluttering of the tree; this hence omits the information about uncertainty. This can be conducted by generating trees from placement results, e.g., in **newick** format. Tools to this end are **GAPPA** (71) and **GUPPY**, which is part of **PPLACER** (64). This can subsequently be visualized via standard tree viewing tools (for a review, see 66). Note however that such a visualizations can quickly become overloaded when the number of QSs becomes large.

Second, the LWR distribution of a single sequence can be visualized, to depict the uncertainty in placement across the tree, for example with **ggtree** (73) and **iTOL** (187, 188).

Third, the distribution of all sequences can be visualized directly on the reference tree, for example as shown in Figure 1(e), taking their per-branch probabilities (and potentially their multiplicities/abundances) into account. This gives an overview of all placements, and can for example reveal important clades that received a high fraction of placements, or indicate whether placements are concentrated in a specific region of the tree. These visualizations can directly be generated by **GAPPA** (71) and **iTOL** (187, 188); furthermore, **GUPPY** can produce tree visualizations in the **phyloXML** format (189), which can subsequently be displayed by tree viewer tools such as **ARCHAEOPTERYX** (189).

### 3.1. Placement Quality and Uncertainty Quantification

An important post-analysis aspect is quality control, both in order to assess the suitability of the RT for the given placed sequences (to, e.g., test for missing reference sequences), and in order to assess the placed sequences themselves. Assuming a ‘perfect’ reference tree that exactly represents the diversity of the query sequences, the theoretical expectation is that each sequence gets placed onto a leaf of the tree with an LWR close to 1. Ignoring sequencing errors and other technical issues, deviations from this expectation can be due to several issues.

To this end, plotting the histograms or the distribution of the confidences (LWRs) across all placements can be useful, Figure 3(c). A more involved metric is the so-called Expected Distance between Placement Locations (EDPL, 64), which for a given sequence represents the uncertainty-weighted average distance between all placement locations of that sequence, or in other words, the sum of distances between locations, weighted by their respective probability, see Figure 3(d). The EDPL is a measure of how far the likely placement locations of a sequence are spread out across the tree. It hence can distinguish between local and global uncertainty of the placements, that is, between cases where nearby edges constitute equally good placement locations versus cases where the sequence does not have a clear placement position in the tree (64). These metrics can be explored with **GAPPA** (71) and **GUPPY** (64); see their respective manuals for the available commands.

Examining the distribution of placement statistics, Figure 3(e)-(d), or even the values of individual sequences, can help to identify the causes of problematic placements: (i) Sequences that are spread out across a clade with a flat placement distribution might indicate that too many closely related sequences, such as strains, are included in the RT; the EDPL can be used to quantify this. The query sequence is then likely another variant belonging to this subtree. (ii) Placement towards inner branches of the RT might hint a hard to place query sequence, or at a lack of reference sequence diversity. This occurs if the (putative) ancestor represented by an inner node of the tree is more closely related to the QSs than the extant representatives included in the RT. This can either be the result of missing taxa in the RT, or even because the diversity of the clade is not fully known yet (also known as incomplete taxon sampling), in which case the QS might have originated from a previously undescribed species. (iii) Sequences placed in two distinct clades might indicate technical errors such as the presence of chimeric sequences (192). (iv) Sequences with elevated placement probability in multiple clades (e.g., placements in more than two subtrees) usually result from more severe issues, such as a total lack of suitable reference sequences for the QS, or a severe misalignment of the QS to the reference. This can for instance occur if metagenomic shotgun data has not been properly filtered, such that the genome region that the QS originated from is not included in the underlying MSA. (v) Lastly, long pendant lengths can also occur if a QS does not fit anywhere in the RT, in particular when the RT contains outgroups, which can cause long branch attraction for placed sequences (165).

Quantifying these uncertainties in a meaningful and interpretable way, and distinguishing between their causes, are open research questions. Approaches such as considering the EDPL, flatness of the LWR distribution, pendant lengths relative to the surrounding branch lengths of the RT, might help here, but more work is needed in order to distinguish actual issues from the identification of a new species based on their placement.

### 3.2. Taxonomic Classification and Functional Analysis

By understanding the taxonomic composition of an environment, questions about its species diversity and richness can be answered. Typical metagenomic data analyses hence often include a taxonomic classification of reads with respect to a database of known sequences (193), for example by aggregating relative abundances per taxonomic group. In addition, such a classification based on known data enables to analyze which pathways and functions are present in a sample, and hence to gain insight into the metabolic capabilities of a microbial community.

**Preexisting Tools.** Many tools exist to these ends: **BLAST** (194) and other similarity-based methods were among the early methods, but depend on the threshold settings for various parameters (195), only provide meaningful results if the reference database contains sequences closely related to the queries (7), and the closest hit does often not represent the most closely related species (38, 39). Thus, the advantages of leveraging the power of phylogenetics for taxonomic assignment have long been recognized (196). The classification can be based on de novo construction of a phylogeny (197, 198), which as mentioned is computationally expensive, and tree topologies might change between samples, yielding downstream analyses and
independent comparisons between studies challenging (199). Alternatively, dedicated pipelines for 16S metabarcoding data such as QIIME (200, 201) and mothur (202) are routinely used to conduct taxonomic assignment based on sequence databases and established phylogenies as well as taxonomies; see Section "Sequence Selection" for a list of common databases, and see (203, 204) for comparisons of such pipelines. Other tools for taxonomic assignment and profiling are available, for example based on k-mers, which often use a fixed taxonomy such as the NCBI taxonomy (119, 120) to propose an evolutionary context for query sequences. They hence use a taxonomic tree without branch lengths, which can be an advantage when a fully resolved phylogeny is not available. Tools to this end are for example MEGAN (205), Kraken2 (206, 207), and KAIJU (208), see (209–212) for benchmarks and comparisons. However, these approaches are based on sequence similarity and related approaches, and can therefore be incongruent with the true underlying phylogenetic relationships of the sequences under comparison (213).

Placement-Based Approaches. Phylogenetic placement can be employed to perform an accurate assignment of QSs to taxonomic labels (127), with potentially higher resolution than methods based on manually curated taxonomies (114, 214).
This approach leverages models of sequence evolution (214), and is hence more accurate than similarity-based methods (63). A further advantage over the above pipelines is the ability to use custom reference trees, thus providing a better context for interpreting the data under study. Incongruencies between the taxonomy and the phylogeny can however hinder the assignment, if they are not resolved (215). Furthermore, it is important to note that placement-based methods only work when the query sequences are homologous to the available reference data, hence currently limiting the approach to, e.g., short genomes, metabarcoding or filtered metagenomic data.

A simple approach for taxonomic annotation based on placements is to label each branch of the RT by the most descriptive taxonomic path of its descendants, and to assign each QS to these labels based on its placement locations, potentially weighted by LWRs (127, 216). This is implemented in gappa (71), see Figure 3(b) for an example; a similar visualization of the taxonomic assignment of placements can be conducted with BoSSA (72).

More involved and specialized approaches have also been suggested. PhyloSIFT (214) is a workflow that employs placement for taxonomic classification, using a database of gene families that are particularly well suited for metagenomics. The workflow further includes Edge PCA (introduced in Section “Similarity between Samples”) to assess community structure across samples, and offers Bayesian hypothesis testing for the presence of phylogenetic lineages. The gene-centric taxonomic profiling tool metANNOTATE (217) uses a similar approach to identify organisms within a metagenomic sample that perform a function of interest. To this end, it searches shotgun sequences against the NCBI database (119, 120) first, and then employs placement to classify the reads with respect to genes and pathways of interest. GraftM (199) is a tool for phylogenetic classification of genes of interest in large metagenomic datasets. Its primary application is to characterize sample composition using taxonomic marker genes, which can also target specific populations or functions. The abundance profiling methods TIPP (80) and TIPP2 (218) also use marker genes, and employ the SEPP (181, 184) boosting technique for phylogenetic placement with highly diverse reference trees, which increases classification accuracy when under-represented (novel) genomes are present in the dataset. The more recently introduced TreesAPP tool (219) uses a similar underlying framework, but improves functional and taxonomic annotation by regressing on the evolutionary distances (branch lengths) of the placed sequences, thereby increasing accuracy and reducing false discovery. Lastly, PhyloMAGNET (220) is a workflow for gene-centric metagenome assembly (MAGs) that can determine the presence of taxa and pathways of interest in large short-read datasets. It allows to explore and pre-screen microbial datasets, in order to select good candidate sets for metagenomic assembly.

### 3.3. Diversity Estimates

A goal that is intrinsically connected to taxonomic assignment in studies that involve metagenomic and metabarcoding sequencing is to quantify the diversity within a sample (called α-diversity) and the diversity between samples (called β-diversity). A plethora of methods exists to quantify the diversity of a set of sequences (for an excellent review, see 221). Here, we focus on those approaches that specifically work in conjunction with phylogenetic placement.

Among the α-diversity metrics, Faith’s Phylogenetic Diversity (PD) stands out, both for its widespread use in the literature and its direct use of phylogenetic information (222). More recently, a parameterized generalization of the PD was introduced that is able to interpolate between the classical PD and its abundance weighted formulation (223). Notably, this Balance Weighted Phylogenetic Diversity (BWPD) has been implemented to work directly with the results of phylogenetic placement, using the GUPPY fpd command (64, 214).

To our knowledge, the only other method that computes a measure of α-diversity directly from phylogenetic placement results is SCRAPP (224), which also deploys species delimitation methods (225, 226). In this method, the connection of phylogenetics to diversity is through the concept of a molecular species (227), and quantifying how many such species are contained within a given sample. To facilitate this, SCRAPP resolves the between-QS phylogenetic relationships, resulting in per-reference-branch trees of those QSs that had their most likely placement on that specific branch. Thus, a byproduct of applying this method is a set of phylogenetic trees of the query sequences.

When the goal is to compute a β-diversity measure, a common choice for non-placement based approaches is the so-called Unifrac distance (228, 229), which quantifies the relatedness of two communities that are represented by leaves of a shared phylogenetic tree. Interestingly, the weighted version of the Unifrac distance has been shown to be equivalent to the KR-distance (230), see Section “Similarity between Samples”. As the Unifrac distance is widely used and well understood, this makes the KR-distance a safe choice for calculating between-sample distances, and thus a measure of β-diversity based on phylogenetic placement results.

### 3.4. Placement Distribution

Depending on the research question at hand, and for larger numbers of QSs, it is often more convenient and easier to interpret to look at the overall placement distribution instead of individually placed sequences. This distribution, as shown in Figure 1(e) and Figure 3(a), summarizes an entire sample (or even multiple samples) by adding up the per-branch probabilities (i.e., LWRs) of each placement location of all sequences in the sample(s), ignoring all branch lengths (distal, proximal, and pendant) of the placements. In this context, the accumulated per-branch probabilities are also called the _edge mass_ of a given branch. This terminology is derived from viewing the reference tree as a graph consisting of nodes and edges, and viewing the placements as a mass distribution on that graph. This focuses more on the mathematical aspects of the data, and provides a useful framework for the analysis methods described below.

**Normalization of Absolute Abundances.** High-throughput metagenomic sequence data are inherently compositional (235–237), meaning that the total number of reads from HTS (absolute abundances) are mostly a function of available biological material and the specifics of the sequencing process. In other words, the total number of sequences per sample (often also called library size) is insignificant when comparing samples. This implies that sequence abundances are not comparable across samples, and that they can only be interpreted as proportions relative to each another (238, 239). However, the PCR amplification process is known to introduce biases (84), potentially...
skewing these proportions. For example, the relative abundances of the final amplicons do not necessarily reflect the original ratio of the input gene regions (235, 240); this can be problematic in comparative studies. If these characteristics are not considered in analyses of the data (241), spurious statistical results can occur (242–245); see (234) for further details. For this reason, the estimation of indices such as the species richness is often implemented via so-called rarefaction and rarefaction curves (246), which might however ignore a potentially large amount of the available valid data (247).

Phylogenetic placement of such data hence also needs to take this into account. The total edge masses (e.g., computed as the sum over all LWRs of a sample) are not informative, and merely reflect the total number of placed sequences. A simple strategy, upon which several of the analysis methods introduced below are based, is the normalization of the masses by dividing them by their total sum, effectively turning absolute abundances into relative abundances. This also eliminates the need for rarefaction, as low-abundance sequences only contribute marginally to the data. However, using this ap-
proach can still induce compositional artifacts in the data, as the per-branch probabilities (and hence the edge masses per sequence) have to sum to one for all branches of the tree. In other words, it is conceptually not possible to change the relative edge mass on a branch without also affecting edges masses on other branches.

Transformations of Compositional Data. A statistically advantageous way to circumvent these effects, and resulting misinterpretations of compositional placement data, is to transform the data from per-branch values to per-clade values. This way, individual placement masses in the nearby branches of a clade are transformed into a single value for the entire clade, which expresses a measure of difference (called contrast) of the placement masses within the clade versus the masses in the remainder of the tree. This makes such transformations robust against placement uncertainty in a clade (e.g., due to similar reference sequences), implicitly captures the tree topology, solves the issues of compositional data. From a technical point of view, this transforms the data from a compositional space into an Euclidean coordinate system (248), where the individual dimensions of a data point are unconstrained and independent of each other. This can be achieved by utilizing the reference tree, whose branches imply bi-partitions of the two clades that are split by each branch (238, 249). Instead of working with the per-branch placement masses, the accumulated masses on each side of a branch are contrasted against each other. This yields a view of the data that summarizes all placements in the clades implied by each branch. These transformations are, for example, achieved via two methods that in the existing literature have unfortunately confusingly similar names: imbalances and balances (234).

The edge imbalance (233) is computed on the normalized edge masses of a sample: For each edge, the sum over all masses in the two clades defined by that edge are computed; their difference is then called the imbalance of the edge. The edge balance (232, 238) is computationally similar, but instead of a difference of sums, it is computed as the (isometric) log-ratio of the geometric means of the masses in each clade; the resulting coordinates are called balances (237, 248, 250). Both transformations yield a contrast value for each (inner) branch of the tree, which can then, for example, be used to compare different samples to each other, see Section “Analysis of Multiple Samples”. They differ in the details of their statistical properties, but more work is needed to examine the effects of this on placement analyses (234); in practice, both can be (and are) used to avoid compositional artifacts.

3.5. Analysis of Multiple Samples
In typical metagenomic and metabarcoding studies, more than one sample is sequenced, e.g., from different locations or points in time of an environment. Furthermore, often per-sample metadata is collected as well, such as the pH-value of the soil or the temperature of the water where a sample was collected. These data allow to infer connections between the species community composition of the samples and environmental features. Given a set of samples (and potentially, metadata variables), an important goal is to understand the community structure (251). To this end, fundamental tasks include measuring their similarity (a distance between samples), clustering samples that are similar to each other according to that distance measure, and relating the samples to their environmental variables.

To this end, the methods introduced in this section utilize phylogenetic placement, and assume that the sequences from all samples have been placed onto the same underlying reference tree; they are implemented in GAPPA (71) and partially in GUPPY (64).

Similarity between Samples. A simple first data exploration method consists in computing the Edge Dispersion (232) of a set of samples, which detects branches or clades of the tree that exhibit a high heterogeneity across the samples by visualizing a measure of dispersion (such as the variance) of the per-sample placement mass. The method hence identifies branches and clades “of interest”, where samples differ in the amount of sequences being placed onto these parts of the tree.

The similarity between the placement distributions of two samples can be measured with the phylogenetic Kantorovich-Rubinstein (KR) distance (230, 233), which is an adaptation of the Earth Mover’s distance to phylogenetic placement. The KR distance between two samples is a metric that quantifies by at least how much the normalized mass distribution of one sample has to be moved across the reference tree to obtain the distribution of the other sample. In other words, it is the minimum work needed to solve the transportation problem between the two distributions (transforming one into the other), and is related to the UniFrac distance (228, 229). The distance is symmetrical, and increases the more mass needs to be moved (that is, the more the abundances per branch and clade differ between the two samples), and the larger the respective moving distance is (that is, the greater the phylogenetic distance along the branches of the tree between the clades is). It is hence an intuitive and phylogenetically informed distance metric for placement data, for example to quantify differences in the species composition of two environments.

Edge Principal Component Analysis (Edge PCA) is a method to detect community structure, which can also be employed for sample ordination and visualization (214, 233). Edge PCA identifies lineages of the RT that explain the greatest extent of variation between the sample communities, and is computed via standard Principal Component Analysis on the per-edge imbalances across all samples. The resulting principal components distinguish samples based on differences in abundances within clades of the reference tree. See for example Figure 4(d), where each point corresponds to a sample and is colorized according to a metadata variable of the sample, showing that the ordination discriminates samples according to that variable. Furthermore, as the eigenvectors of each principal component correspond to edges of the tree, these can be visualized on the tree (233, 234), so that those edges and clades of the tree that explain differences between the samples can be identified, e.g., with GUPPY (64) and ARCHAEOPTERYX (189), or with GAPPA (71). Principal components can also be computed from the balances instead of the imbalances (234).

Clustering of Samples. Given a measure of pairwise distance between samples, a fundamental task consists in clustering, that is, finding groups of samples that are similar according to that measure. Squash Clustering (233) is a hierarchical agglomerative clustering method for a set of placement samples, and is based on the KR distance. Its results can be visualized as a clustering tree, where terminal nodes represent samples, each inner node represents the cumulative distribution of all samples below that node (“squashed” samples), and distances along the tree edges are KR distances. We show an example
in Figure 4(e), where each sample (terminal node) is colorized according to associated per-sample metadata variables (features measured for each sample), indicating that the clustering (based on the placement distribution) recovers characteristics of the samples based on that metadata variable.

The clustering hierarchy obtained from Squash Clustering grows with the number of samples, which contains a lot of detail, but can be cumbersome to visualize and interpret for large datasets with many samples. **Phylogenetic k-means** clustering and **Imbalance k-means** clustering (232) are further clustering approaches, which instead yield an assignment of each sample to one of a predefined number of k clusters. **Phylogenetic k-means** uses the KR distance for determining the cluster assignment of the samples, and hence yields results that are consistent with Squash Clustering, while **Imbalance k-means** uses edge imbalances, and hence is consistent with results obtained from Edge PCA. Having the choice over the value k can be beneficial to answer specific questions with a known set of categories of samples (e.g., different body locations where samples were obtained from), but is also considered a downside of k-means clustering. Hence, various suggestions exist in the literature to select an appropriate k that reflects the number of “natural” clusters in the data (252–257). Visualizing the **cluster centroids** obtained from both methods can further help to interpret results by showing the average distributions of all samples in one of the k clusters; see again (234) for details.

**Relationship with Environmental Metadata Variables.** The above methods only implicitly take metadata into account, e.g., by coloring their resulting plots according to a variable. Environmental variables can also be incorporated explicitly in phylogenetic placement analysis, to more directly infer the relationships between the species composition of the samples (e.g., in form of abundances per clade) and the environments these communities live in.

The Edge Correlation (232) visualizes parts of the tree where species abundances (as measured by the accumulated probability mass of each sample) exhibit a strong connection with a metadata variable, see Figure 4(a). It is computed as the per-edge correlation coefficient between the per-sample metadata variable and either the edge masses (highlighting individual edges), or imbalances or balances (highlighting clades) of each sample.

**Placement-Factorization** (232, 234) is a more involved method. It is an adaption of **PhyloFactorization** (258, 259) to phylogenetic placement data. Its goal is to identify branches in the tree along which putative functional traits might have arisen in adaptation to changes in environmental variables. In other words, it can detect clades of the reference tree whose abundances are linked to environmental factors. By “factoring out” the clade with the strongest signal in each step of the algorithm (hence the name of the method), nested dependencies with variables within clades can also be discovered, see Figure 4(b). This factorization of the tree into nested clades can further be used as an ordination tool to visualize how samples are separated by changes along the factors, and as a dimensionality-reduction tool, see Figure 4(c). The method assesses the relationship between per-sample metadata features and the balances computed on the samples; by using Generalized Linear Models, it allows to simultaneously incorporate multiple metadata variables of different types, such as numerical values (pH-value, temperature, latitude/longitude, etc.), binary values (presence/absence patterns, diseased or not), or categorical values (body site that a sample was taken from).

**4. Conclusion and Outlook**

In this review we broadly surveyed the concepts, methods, and software tools that constitute and relate to phylogenetic placement. We have also presented guidelines and best practices for many typical use cases, showcased some common misconceptions and pitfalls, and introduced the most prominent downstream analysis methods. Phylogenetic placement is a versatile approach that is particularly applicable in metagenomics (e.g., for metabarcoding data) and broader eDNA-based ecology studies. It allows for the annotation of sequence data with phylogenetic information, and thereby to investigate the taxonomic content, functional capacity, diversity, and interactions of a community of organisms. Further, it allows for comparing samples from multiple spatial and temporal locations, enabling the analysis of community patterns across time and space, as well as their association with environmental metadata variables.

Despite the growing popularity of phylogenetic placement, there are several methodological and usage aspects that will benefit from further developments.

Currently, significant effort is required to create high-quality reference trees. We believe research effort should focus on simplifying this process, potentially through the design of methods that streamline and automate the commonly involved tasks. For example, while there are some metrics that quantify the quality of an inferred phylogenetic tree (139, 260, 261), there is a lack of metrics to specifically evaluate the suitability of a tree for phylogenetic placement, given some expected input data. Note that the PEWO testing framework (126) (see Section “Workflows based on Phylogenetic Placement”) represents a first step in this direction.

Ideally, reference trees and alignments should be created by, and shared in, research communities that investigate the same group(s) of organisms. This would not only yield obtaining high-quality reference trees trivial, but would also immensely increase the comparability across studies, as well as their reproducibility. Consequently, we would highly encourage such collaborations, and the public sharing of (perhaps even versioned instances of) gold-standard reference trees. Notably, for some environments, first efforts into this direction have already been undertaken (112–115, 262).

Furthermore, as mentioned, there is a lack of established methods that evaluate placement quality in a standardized and meaningful way. In particular, robust metrics are missing to distinguish the case where reference sequences of known species are missing from the tree from the case where the placed data actually contains yet undescribed species. A classification based on the LWR and pendant length of the placement locations might offer a solution here.

Lastly, further work is required to connect environmental metadata to the results of phylogenetic placement. Placement-based spatio-temporal methods are of high interest for addressing research questions in ecology and phylogeography. For example, relating geo-locations of samples to their placement could indicate how species communities differ across space, while creating placement time series could show how community compositions develop and change over time.
We wish to thank the Reviewers for the detailed and constructive comments that helped to improve this manuscript. This work was financially supported by the Carnegie Institution for Science at Stanford, California, USA, the Klaus Tschira Stiftung gGmbH Foundation in Heidelberg, Germany, and the Deutsche Forschungsgemeinschaft (grant DU1319/5-1).

Acknowledgments

Appendices

Competing Interests

The authors declare that they have no competing interests.

Author’s Contributions

LC conceived the review and created the figures. LC and PB drafted the manuscript. All authors conducted literature research, and finalized and approved the manuscript.

References

1. Edwards DJ, Holt KE (2013) Beginner’s guide to comparative bacterial genome analysis using next-generation sequence data. Microbial Informatics and Experimentation 3(1):2.
2. Sunagawa S, et al. (2013) Metagenomic species profiling using universal phylogenetic marker genes. Nature Methods 10(4):119-125.
3. Karsenti E, et al. (2011) A holistic approach to mine Earth-systems biology. PLoS Biology 9(10):e1001219.
4. Giner CR, Forn I, Romac S, Logares R, Vargas CD (2016) Environmental Sequencing Provides Reasonable Estimates of the Relative Abundance of Specific Protists. Applied and Environmental Microbiology 82(15):4757-4766.
5. Lacoursière-Roussel A, Côté G, Leclerc V, Bernatchez L (2016) Quantifying relative fish abundance with eDNA: a promising tool for fisheries management. Journal of Applied Ecology 53(4):1148-1157.
6. Dupont AOãC, Griffiths RI, Bell T, Bass D (2016) Differences in soil micro-eukaryotic communities over soil pH gradients are strongly driven by parasites and saprotrophs. Environmental Microbiology 18(10):4019-4031.
7. Mahé F, et al. (2017) Parasites dominate hyperdiverse soil protist communities in Neotropical rainforests. Nature Ecology & Evolution 1(4):0091.
8. Clare EL, et al. (2022) Measuring biodiversity from DNA in the air. Current Biology.
9. Deiner K, et al. (2017) Metagenomic DNA metabarcoding: A systematic review in methods, monitoring, and applications of global edna. Global Ecology and Conservation 17:00547.
10. Hutterer C, et al. (2012) Structure, function and diversity of the healthy human microbiome. Nature 486(7402):207-214.
11. Methe BA, et al. (2012) A framework for human microbiome research. Nature 484(7402):215-221.
12. Matsen FA (2015) Phylogenetics and the Human Microbiome. Systematic Biology 64(1):626-e41.
13. Wang WL, et al. (2015) Application of metagenomics in the human gut microbiome. Nature Reviews Gastroenterology and Hepatology 21(2):893-814.
14. Hansen B, et al. (2016) Characterization of the bacterial and fungal microbiome in indoor dust and outdoor air samples: a pilot study. Environmental Science: Processes & Impacts 18(6):713-724.
15. Goh J, et al. (2019) The subway microbiome: Seasonal dynamics and direct comparison of air and surface bacterial communities. Microbiology 7(1):1-16.
16. Lorimer J, et al. (2019) Making the microbiome public: Participatory experiments with DNA sequencing in domestic kitchens. Transactions of the Institute of British Geographers 44(3):324-341.
17. El Rabkby MT, Gamal-Eldin S, Amin MA, Aziz RK (2019) Hospital Microbiome Variations As Analyzed by High-Throughput Sequencing. OMICS A Journal of Integrative Biology 23(9):426-438.
18. Thomas T, Gilbert J, Moyer F (2012) Metagenomics - a guide from sampling to data analysis. Microbial Informatics and Experimentation 2(1):3.
19. Czesniewska-Malecka A, Vasselon V, Bouchez A, Kahlert M (2018) Boosting DNA metabarcoding for extremely large phylogenetic trees. ONE 3(1):2.
20. Truong DT, et al. (2015) MetaPhlAn2 for enhanced metagenomic taxonomic profiling. Current Opinion in Microbiology 33:547-558.
21. Keck F, Vasselon V, Rimmel F, Bouchet A, Kahlert M (2018) Boosting DNA metabarcoding for biometric analysis. Frontiers in Genetica 7:101.
22. Bass D, et al. (2018) Clarifying the Relationships between Microsporida and Cryptotrypota. Journal of Eukaryotic Microbiology 65(5):773-782.
23. Keck F, Vasselon V, Rimmel F, Bouchet A, Kahlert M (2018) Boosting DNA metabarcoding for biomonitoring with phylogenetic estimation of operational taxonomic units’ ecological profiles. Molecular Ecology Resources 18(6):1299-1309.
24. Mühlemann B, et al. (2020) Diverse variola virus (smallpox) strains were widespread in northern Europe in the Viking Age. Science 369(6502).
25. Morel B, et al. (2020) Phylogenetic Analysis of SARS-CoV-2 Data Is Difficult. Science 369(6502):809-818.
26. Logares R, et al. (2012) Environmental microbiology through the lens of high-throughput DNA sequencing: Synopsis of current platforms and bioinformatics approaches. Journal of Microbiological Methods 91(1):106-113.
27. Mardis ER (2013) Next-Generation Sequencing Platforms. Annual Review of Analytical Chemistry 6(1):287-303.
28. Page DK, Smokey R, Trellyn A (2011) Sequencing technologies and genome sequencing. Journal of Applied Genetics 52(4):413-435.
29. Niedringhaus TP, Milanova D, Kerby MB, Snyder MP, Barron AE (2011) Landscape of Next-Generation Sequencing Technologies. Analytical Chemistry 83(12):4327-4341.
30. Mignard M, Nilsson M (2014) Fourth-generation sequencing in the cell and the clinic. Genome Medicine 6(4):31.
31. Heath JM, Chain B (2016) The sequence of sequencers: The history of sequencing DNA. Genomics 107(1):1-8.
32. Mardis ER (2017) DNA sequencing technologies: 2006-2016. Nature protocols 12(2):213–218.
33. Muir P, et al. (2016) The real cost of sequencing: calculation cost to keep pace with data generation. Genome Biology 17(1):1-9.
34. Katz K, et al. (2022) The Sequence Read Archive: a decade more of explosive growth. Nucleic Acids Research 50(D1):D287–D290.
35. Desal N, Antonopoulos D, Gilbert JA, Glass EM, Meyer F (2012) From genomes to metagenomics. Current Opinion in Biotechnology 23(1):72–76.
36. Temperton B, Giovanni SJ (2012) Metagenomics: Microbial diversity through a scratched lens. Current Opinion in Microbiology 15(5):605-612.
37. Peabody MA, Van Rossum T, Lo R, Blissner FSL (2015) Evaluation of shotgun metagenomic sequence classification methods using in silico and in vitro simulated communities. BMC Bioinformatics 16:363.
214. Darling AE, et al. (2014) PhyloSift: phylogenetic analysis of genomes and metagenomes. PeerJ 2:e243.

215. Matsen FA, Gallagher A (2012) Reconciling taxonomy and phylogenetic inference: formalism and algorithms for describing discord and inferring taxonomic roots. Algorithms for molecular biology 7:4.

216. Kozlov AM, Zhang J, Yilmaz P, Glückner FO, Stamatakis A (2016) Phylogeny-aware identification and correction of taxonomically mislabeled sequences. Nucleic Acids Research 44(11):5022-5033.

217. Petrenko P, Lobe B, Kurtz DA, Neufeld JD, Doxey AC (2015) MetAnnotate: Function-specific taxonomic profiling and comparison of metagenomes. BMC Biology 13(1):92.

218. Shaw N, Molloy EK, Pep M, Wannow T (2021) TIPP2: metagenomic taxonomic profiling using phylogenetic markers. Bioinformatics.

219. Morgan-Lang C, et al. (2020) TreeSAPP: the Tree-based Sensitive and Accurate Phylogenetic Profile. Bioinformatics 36(18):4706-4713.

220. Schönh ME, Eme L, Ettene T (2013) PhyloMagnet: fast and accurate screening of short-read meta-omics data using gene-centric phylogenetics. Bioinformatics 36(6):1718-1724.

221. Tucker CM, et al. (2017) A guide to phylogenetic metrics for conservation, community ecology and macroecology. Biological Reviews 92(2):698-715.

222. Faith DP (1992) Conservation evaluation and phylogenetic diversity. Biological Conservation 61(1):1-10.

223. McCoy CO, Matsen FA (2013) Abundance-weighted phylogenetic diversity measures distinguish microbial community states and are robust to sampling depth. PeerJ 1:e157.

224. Barbera P, Czech L, Lutterop S, Stamatakis A (2020) SCRAPP: A tool to assess the diversity of microbial samples from phylogenetic placements. Molecular Ecology Resources 21(1):1755-1698.13255.

225. Zhang J, Kapli P, Pavlidis P, Stamatakis A (2013) A general species delimitation method with 40225. Zhang J, Kapli P, Pavlidis P, Stamatakis A (2013) A general species delimitation method with

226. Kozlov AM, Zhang J, Yilmaz P, Glückner FO, Stamatakis A (2016) Phylogeny-aware identification and correction of taxonomically mislabeled sequences. Nucleic Acids Research 44(11):5022-5033.

227. Petrenko P, Lobe B, Kurtz DA, Neufeld JD, Doxey AC (2015) MetAnnotate: Function-specific taxonomic profiling and comparison of metagenomes. BMC Biology 13(1):92.

228. Shah N, Molloy EK, Pep M, Wannow T (2021) TIPP2: metagenomic taxonomic profiling using phylogenetic markers. Bioinformatics.

229. Morgan-Lang C, et al. (2020) TreeSAPP: the Tree-based Sensitive and Accurate Phylogenetic Profile. Bioinformatics 36(18):4706-4713.

230. Evans SN, Matsen FA (2012) The phylogenetic Kantorovich-Rubinstein metric for environmental sequence samples. Journal of the Royal Statistical Society. Series B: Statistical Methodology 74:569-592.

231. Nugent RP, Krohn MA, Hillier SL (1991) Reliability of diagnosing bacterial vaginosis is improved by a standardized method of gram stain interpretation. Journal of Clinical Microbiology 29(2):297-301.

232. Czech L, Stamatakis A (2019) Scalable methods for analyzing and visualizing phylogenetic placement of metagenomic samples. PLOS ONE 14(5):e0217050.

233. Matsen FA, Evans SN (2011) Edge principal components and squash clustering: using the special structure of phylogenetic placement data for sample comparison. PLOS ONE 8(3):1-17.

234. Czech L (2020) Novel Methods for Analyzing and Visualizing Phylogenetic Placements. Ph.D. thesis. Karlsruher Institut für Technologie, Karlsruhe, Germany.

235. Li H (2015) Microbiome, Metagenomics, and High-Dimensional Compositional Data Analysis. Mathematical Geology 47(7):795-828.

236. Gloor GB, Macklaim JM, Pawlowski-Glañh V, Egoozue J (2017) Microbiome Datasets Are Compositional: And This Is Not Optional. Frontiers in Microbiology 8:2224.

237. Quinn TP, Erb L, Richardson MF, Crowley TM (2018) Understanding sequencing data as compositions: an outlook and review. Bioinformatics 34(16):2870-2878.

238. Silverman JD, Washburne AD, Mukherjee S, David LA (2017) A phylogenetic transform enhances analysis of compositional microbiota data. elife 6:e21887.

239. Calle ML (2019) Statistical Analysis of Metagenomics Data. Genomics & Informatics 17(1):1-16.

240. Kanagawa T (2003) Bias and Artifacts in Multitemplate Polymerase Chain Reactions (PCR). Journal of Bioscience and Bioengineering 96(4):317-323.

241. Weiss S, et al. (2017) Normalization and microbial differential abundance strategies depend upon data characteristics. Microbiome 5(1):27.

242. Aitchison J (1986) The statistical analysis of compositional data. Cambridge University Press.

243. Jackson DA (1997) Compositional data in community ecology: The paradigm or peril of modeling and analysis of compositions: an outlook and review. Bioinformatics 34(16):2870-2878.

244. Tsilimigras MCB, Fodor AA (2016) Compositional data analysis of the microbiome: fundamentals, tools, and challenges. Annals of Epidemiology 26(5):330-335.

245. Gloor GB, Macklaim JM, Vu M, Fernandes AD (2016) Compositional uncertainty should not be ignored in high-throughput sequencing data analysis. Austrian Journal of Statistics 45(4):73.

246. Gotelli NJ, Colwell RK (2001) Quantifying biodiversity: procedures and pitfalls in the measurement and comparison of species richness. Ecology Letters 4(4):379-391.

247. McMurdie PJ, Holmes S (2014) Waste Not, Want Not: Why Rarefying Microbiome Data Is Inadmissible. PLoS Computational Biology 10(4):e1003531.

248. Egozue J, Pawlowski-Glahn V (2005) Groups of Parts and Their Balances in Compositional Data Analysis. Mathematical Geology 37(7):795-828.

249. Pawlowski-Glañh V, Egozue J, Tolosaño-Delgado R (2015) Modeling and Analysis of Compositional Data. (John Wiley & Sons, Chichester, UK), p. 272.

250. Egozue J, Pawlowski-Glañh V, Mateu-Figueras G, Barceló-Vidal C (2003) Isometric Logratio Transformations for Compositional Data Analysis. Mathematical Geology 35(3):279-300.

251. Tyson GW, et al. (2004) Community structure and metabolism through reconstruction of microbial genomes from the environment. Nature 428(6978):37-43.

252. Thorndike RL (1953) Who belongs in the family? Psychometrika 18(4):267-276.

253. Rousseau PJ (1987) Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 20:53-65.

254. Bischof H, Leonards A, Seb A (1999) MDL Principle for Robust Vector Quantisation. Pattern Analysis & Applications 2(1):59-72.

255. Pellet D, Moore AW (2000) K-means: Extending K-means with Efficient Estimation of the Number of Clusters. In: ICML. Vol. 1, pp. 727-734.

256. Tibshirani R, Walther G, Hastie T (2001) Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63(2):411-423.

257. Hamerly G, Elkan C (2004) Learning the k in k-means in Advances in Neural Information Processing Systems 16, eds. Thrun S, Saul LK, Schölkopf PB (MIT Press), pp. 281-288.

258. Washburne AD, et al. (2017) Phylogenetic factorization of compositional data yields lineage-level associations in microbiome datasets. PeerJ 5:e2969.

259. Washburne AD, et al. (2019) Phylofactorization: a graph partitioning algorithm to identify phylogenetic scales of ecological data. Ecological Monographs 89(2):e01353.

260. Felsenstein J (1985) Confidence limits on phylogenies: an approach using the bootstrap. Evolution 39(4):783-791.

261. Lamorne F, et al. (2018) Revealing felsenstein’s phylogenetic bootstrap in the era of big data. Nature 556(7702):452-456.

262. Rubinat-Ripoll L (2019) lrubinat/photoreft: a 16s rdna reference tree representing the main groups of picophototrophic eukaryotes and prokaryotes.