Novel Methods for Post-Processing Evolutionary Data Using Machine Learning Techniques

Dissertation
by

Lucas Czech

Karlsruhe Institute of Technology (KIT),
Karlsruhe, Germany

Heidelberg Institute for Theoretical Studies (HITS),
Heidelberg, Germany

First Reviewer:	Prof. Dr. Alexandros Stamatakis
Second Reviewer:	Prof. Dr. Emmanuel Müller

September 30, 2018

Zusammenfassung

Auf deutsch...

Abstract

In English...

Hiermit erkläre ich, dass ich diese Arbeit selbständig angefertigt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht habe. Ich habe die Satzung des Karlsruher Institutes für Technologie (KIT) zur Sicherung guter wissenschaftlicher Praxis beachtet.

Karlsruhe, September 30, 2018

. . . . . .

(**Lucas Czech**)

Acknowledgment

"Evolution forged the entirety of [...] life on this planet using only one tool – the mistake."

— Dr. Robert Ford (Anthony Hopkins),
Westworld, Season 1: The Original

TODO: see alexey and andre

falsification, science "irrt sich vorwaerts"... mein beitrag, aber nicht ohne die hilfe vieler anderer

There are many people who helped and supported me in different ways during the conduction of this dissertation.

First, I would like to thank my advisor Professor Dr. Alexandros Panayotis Stamatakis,

Many thanks also to my second advisor Professor Dr. Emmanuel Müller for his support and advice.

also, other reviewers and examiners of my thesis and work.

To the team of the Exelixis Lab— namely ... — also many thanks, you made .... andre, paschalia, tomas, diego, jiajie, kassian

mark, emily, jaime, fred, micah, cedric, colomban, pelin

Finally, I want to thank my parents Peter and Maria and my sister Judith, who not only constantly supported me during this thesis, but through all of my years of study in Karlsruhe, Heidelberg and around the world.

This work was financially supported by the **Klaus Tschira Stiftung gGmbH** in Heidelberg, Germany. HITS, KIT

We thank **S. Srinivasan** and **E. Matsen** for providing the Bacterial Vaginosis dataset [225] and for helping us understanding their methods and implementations. We also thank **M. Dunthorn**, **L. Rubinat-Ripoll**, **C. Berney**, **L. Guidi**, **G. Lentendu**, **A. Kozlov**, and **P. Barbera** for their feedback on our methods and help with this manuscript.

We thank **S. Srinivasan** and **E. Matsen** for providing the Bacterial Vaginosis dataset [225]; **M. Dunthorn**, **G. Lentendu**, and **A. Kozlov** for their feedback on our methods and this manuscript.

thanks also to the (mostly) anonymous reviewers of our publications, who pointed out flaws to us and thus helped to make them better.

phil collins peter gabriel

richard dawkins, ohne den ich nicht auf das thema aufmerksam geworden waere und daher nicht mit der phd stelle angefangen haette.

not to forget: https://xkcd.com/1706/

or maybe: https://www.xkcd.com/1840/

or better: https://xkcd.com/1605/

Contents

List of Figures

List of Tables

List of Acronyms

RT reference tree

RA reference alignment

QS query sequence

BT backbone tree

CT clade tree

LWR likelihood weight ratio

PhAT Phylogenetic Automatic (Reference) Tree

PCA Principal Components Analysis

MDS Multidimensional scaling

BV Bacterial Vaginosis

NTS Neotropical Soils

TO Tara Oceans

HMP Human Microbiome Project

1. Introduction

TODO: check that the url and access date of all online sources are present in bibliography! TODO: I used a few public domain images from wikipedia as sources, and modified them as needed. make sure that this is okay. TODO: search for all abbreviations used and add them to the acro list. also, check Pierre's MA, and Alexey's and Andre's Diss for needed acronyms!

TODO: list of acronyms! see andre, add MB/GB, PCA, BV, TO, HMP, etc

1.1 Motivation

1.2 Objective and Contribution

swarm code contrib [147, 148]

full list of publications is available in D

mention genesis and gappa, their implementation chapter in the appendix, their paper?!

mention http://github.com/lczech/placement-methods-paper for the result files of two of the papers

1.3 Structure and Overview

2. Foundations

This chapter is partially based on the peer-reviewed publications:

- **Lucas Czech**, Pierre Barbera, and Alexandros Stamatakis. "Methods for Automatic Reference Trees and Multilevel Phylogenetic Placement." *Bioinformatics*, 2018.
- **Lucas Czech** and Alexandros Stamatakis. "Scalable Methods for Post-Processing, Visualizing, and Analyzing Phylogenetic Placements." *PLOS One*, 2018.

TODO: **Contributions:** Lucas Czech... Pierre Barbera... Alexandros Stamatakis... and...

TODO: In this chapter, we introduce...

2.1 Evolution and Genetics

Life on Earth is at least 3.77 billion years old [56], and is continuously evolving due to *natural selection* [46]. Driven by *variation*, biological populations diversify through successive generations, leading to the origination of new species. This continuous process is called *evolution* [91]. Heritable characteristics are passed down from parent to offspring, with occasional random mutations leading to variation. Thus, some organisms are better adapted to their environment than others, and have more reproductive success. There is hence a natural selection for advantageous mutations, which can then spread through generations.

The characteristics and traits of an organism are carried by, and inherited via, *deoxyribonucleic acid* (DNA). DNA is the molecule that encodes the genetic information needed for the functioning of all living organisms. It is structured in form of

a double helix [253], and built from two strands of molecules called *nucleotides*. The nucleotides build the backbone of the double helix, and connect the two strands via opposing pairs of *nucleobases*, see Figure 2.1(a). The redundant structure of pairs of nucleobases gives stability to the DNA molecule, and also serves as a mechanism of error correction when reading the genetic information during cellular processes. In DNA, there are four distinct nucleobases: adenine (`A`), cytosine (`C`), guanine (`G`), and thymine (`T`), where `A` pairs with `T`, and `C` pairs with `G`, respectively, as shown in Figure 2.1(b).

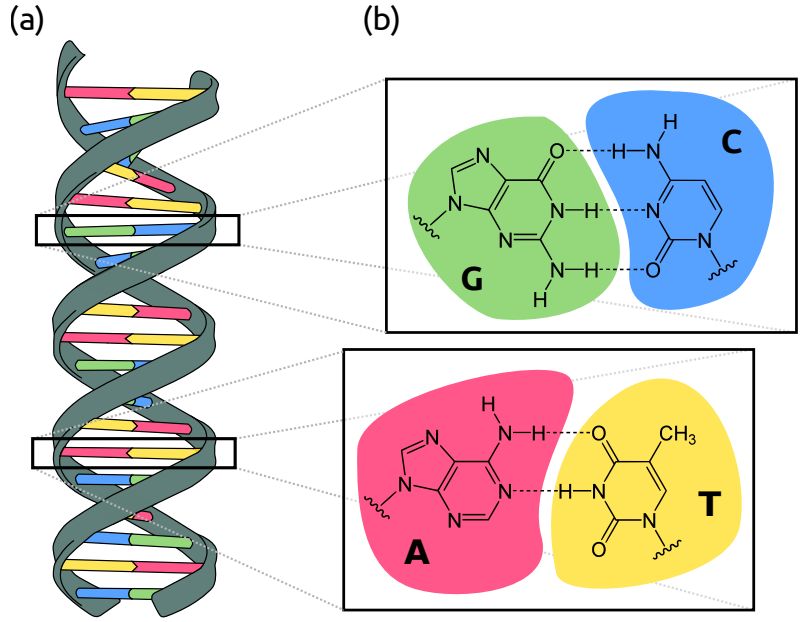

Figure 2.1: DNA double helix and nucleobases. (a) The double helix structure of DNA, with the backbone in gray, connected by pairs of nucleobases. (b) The atomic structure of the four nucleobases, and their connection to each other. Source and license: see [41], image derived from [165, 224, 267, 268].

The sequence of nucleobases along the strands of DNA is what encodes the genetic instructions used by all known living organisms. Parts of the DNA encode for proteins, which perform a plethora of different functions within organisms. Proteins consist of long chains of amino acid residues, and are assembled in a process called protein (bio)synthesis. This is described by the central dogma of molecular biology [37, 38]: First, DNA is *transcribed* into the intermediary ribonucleic acid (RNA), which is then *translated* into the actual protein.

In each step of this process, the molecular alphabet used to encode information differs. While DNA uses the four nucleobases as described above, in RNA, the nucleobase uracil (`U`) is used instead of thymine (`T`). Proteins on the other hand are (mostly) build from a set of 20 standard amino acids. The set of rules used by the molecular machinery for translating nucleobases into amino acids is called the *genetic code*: In a DNA sequence, three consecutive nucleobases are needed to encode one amino acid [219].

The entirety of the genetic material of an organism, that is, its complete DNA sequence, is called its *genome*. A *gene* is a sequence which codes for a molecule that has a particular function, such as a protein [77]. DNA and genes are the basic units of heredity. They are what is varying across generations, and what is selected for in the process of natural selection [47]. The study of genes, variation and heredity is called *genetics* [86].

2.2 Sequence Analysis

All life on this planet is related to each other and descends from a common ancestor. Still, it is remarkable that the basic molecular principles and mechanisms of life—DNA, amino acids, and the genetic code—are virtually identical for all living organisms. This implies that by understanding and comparing the genetic information encoded in the genetic sequences of different organisms, one can understand the diversification patterns of evolution.

2.2.1 Genome Sequencing

Prior to analyzing the DNA of an organism, the physical order of nucleobases in the DNA molecule has to be determined. That is, the DNA has to be "read" and stored in a human-accessible text format, typically a computer file. This technical process is called DNA *sequencing*.

For many decades, the main technique for this purpose was Sanger sequencing [208, 209]. It is labor- and time-intensive, but through improvement and automation, costs were constantly reduced. Eventually, this allowed for large-scale efforts, such as the Human Genome Project [243], which sequenced the whole human genome of more than three billion nucleobases. Sanger sequencing allows to determine long parts of the sequence at once (> 500 nucleobases), which then have to be assembled to build the final sequence.

In the last decades, a variety of novel high-throughput DNA sequencing technologiesß have been developed [83, 191, 201]. In particular, *Next Generation Sequencing* (NGS) [137, 152] has revolutionized biology by transforming it into a data-driven and compute-intense discipline [67]. The costs of these technologies are decreasing faster than Moore's law [257]. This leads to a "tsunami" of sequence data, which poses a challenge for computational methods working with these data. Compared to Sanger sequencing, NGS technologies are generally cheaper and faster [167, 249], but come at the price of introducing more errors in the sequence output, or only being able to determine shorter parts of the sequence at once – both of which constitute a challenge for the subsequent assembly of the final sequence.

The result of DNA sequencing is a textual representation of the order of nucleobases. Although this representation ignores the physical and chemical properties of the respective molecules, it is helpful in many applications, and allows to leverage existing algorithms. Each contiguous sequence coming from the sequencing machine is called a *read*. Because of the pairing of nucleobases, both DNA strands can be sequenced,

which provides a means of error correction. Such data are typically stored in the `fastq` file format [34]. These so-called paired-end reads are then merged to form a final sequence representation of one strand – that is, a sequence of the characters `A`, `C`, `G`, and `T`. These data are stored in formats such as the `fasta` file format [189]. Due to the pairing of nucleobases, the length of a DNA sequence is measured in *base pairs* (abbreviated bp): 1 bp represents one character in the file. These files are then the input for computational methods for working with DNA sequences.

2.2.2 Metagenomics

Sanger sequencing requires careful preparation of the genetic material, and is thus best used for sequencing single organisms. There are however many (microbial) organisms that cannot be cultured in a Petri dish, and are hence hard to sequence with this technique. Apart from being cheaper, Next Generation Sequencing machines however "digest" all genetic material presented to them. They thus allow for studying microbial samples directly extracted from their environment [66, 176, 233]. This enables to study environments such as water [78, 85, 110], soil [61, 149], the human body [101, 154, 166, 252], and many others. Each sample from such an environment then represents a geographical location, a body site, a point in time, etc. The DNA of all organisms being present in a sample is sequenced, resulting in a large number of reads per sample. These reads are anonymous, as it is unclear to which organism they originally belonged. The study of these data, that is, genetic material from environmental samples, is called *metagenomics* [184].

A first step in metagenomic studies is often to characterize the reads obtained from an environment in terms of *reference sequences* of known species. Reads that are similar to (parts of) reference sequences can be assigned to them, while reads with low similarity to known sequences might indicate novel, undescribed species [238]. Key tasks in metagenomic studies are the identification and classification of the anonymous reads ("Who is there?"), and their functional annotation ("What are they doing?") [53]. Both are introduced in the following.

Functional annotation [229] is the prediction of gene functions of the reads, and the inference of metabolic capacity of microbial communities [28]. As the proteins that are needed in the pathways of such functions can be encoded by genes across the genome, whole-genome sequencing is necessary to capture all genes of interest. For example, in shotgun sequencing [5, 226], the DNA is fragmented into small pieces within the size range that the used sequencing technology can handle, typically a few hundred bp long. This allows to sequence all genetic material contained in a sample. Thus, the resulting reads originate from different parts of the genomes of their organisms, which can then be functionally annotated [80]. This however necessitates to use whole-genome reference sequences in order to be able to assign reads to known species and functions. Typical databases of reference sequences however lack many of the protein sequences from the microbial species present in a sample, mostly because of organisms that cannot be cultured [28].

For the task of identification and classification of reads however, whole genome references are not needed. Instead, specific *marker genes* can be used, which are

regions of genes that are particularly suited for delineating between different species [200]. The method of using marker genes to identify species is called *DNA barcoding* [52, 95, 211]. The choice of genes to use as marker is important, and depends on the types of organisms to be studied. The used marker genes should ideally be present in all of the organisms of interest, short enough to be sequenced with current technology, and have enough variation between species to distinguish between them, but have low variation within species [120].

In many metagenomic studies of *bacteria* and *eukaryota*, the 16S [255] and 18S [168] rRNA regions are used as marker genes, respectively [259, 260]. These regions belong to the small subunit (SSU) of the ribosomal ribonucleic acid (rRNA), which is an essential component of the ribosome. The ribosome is a molecular machinery that is responsible for protein synthesis (translation) in all living organisms. Often, prior to sequencing, these regions are amplified by many orders of magnitude, using polymerase chain reaction (PCR) to create copies of these regions [11]. The resulting reads are then de-replicated again, which results in sequences called *amplicons*. While the PCR amplification process is known to introduce bias [28, 138], this inexpensive method is commonly used in practice, particularly for the 16S and 18S rDNA regions. A recent alternative to using PCR for obtaining reads from these regions are $_{\text{mi}}$tags [138]. In this approach, shotgun sequencing is used to get reads from the whole genomes of the organisms in a sample. These reads are then filtered to only contain reads from the 16S region (for *bacteria*), which capture the diversity of the sample without bias.

Because of the ubiquity of the 16S and 18S rDNA regions in organisms and, consequently, in sequencing studies, many databases provide reference sequences for these marker regions. The reads or amplicons obtained from an environmental sample can then be employed to estimate the microbial diversity of the organisms in the sample by comparison against known species.

2.2.3 Sequence Alignment

Organisms that evolved from a (not too distant) common ancestor share genetic information. Regions of their DNA that have a shared ancestry are called *homologous* regions [117]. This homology is typically inferred from sequence similarity. However, due to mutations, differences in the sequences can occur. There are three main types of sequence mutations that can occur in evolution: a *substitution* is the exchange of a nucleobase for another; a *insertion* adds one or more extra nucleotides into the sequence; a *deletion* removes one or more nucleotides from the sequence. The latter two types of mutations change the length of the sequence; a mutation that is either one of them is called an *indel*.

Because of indels, sequences have to be aligned to each other in order to compare their homologous regions. That is, gap characters (-) have to be added to the sequences such that homologous characters in the sequence get aligned. This results in an $n \times m$ matrix, where n is the number of sequences (rows), and m is the number of homologous characters (columns), called *sites*. This matrix is called a *multiple*

sequence alignment (MSA), or simply an *alignment*. Figure 2.2 shows an example of the alignment process and the resulting MSA.

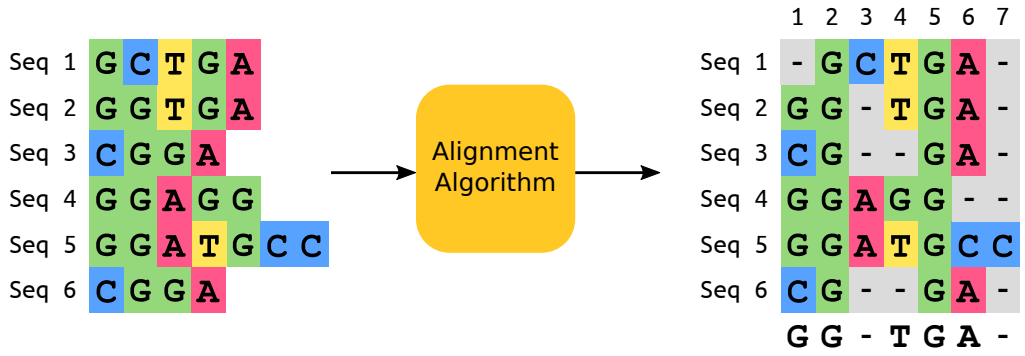

Figure 2.2: Multiple Sequence Alignment. The left hand side shows a set of six sequences. Using an alignment algorithm, gaps are inserted into these sequences at presumed indel positions. The right hand side shows the result of this process, where homologous characters at the sites of the multiple sequence alignment (MSA) are aligned to each other. Below the MSA, the majority rule consensus sequence is shown, see Section 2.2.4.

Sequence alignment can be understood as an optimization problem under a given optimality criterion. On the one hand, *global alignments* attempt to align every character in every sequence, which is most useful for similar sequences of roughly equal size. For example, the Needleman-Wunsch algorithm [177] is a general global alignment technique based on dynamic programming. On the other hand, *local alignments* are better suited for dissimilar sequences which might contain similar regions within a larger sequence context. The Smith-Waterman algorithm [220] is a general local alignment technique using the same dynamic programming scheme, which additionally allows to start and end at any place in the sequence. As both algorithms have their particular use cases [193], hybrid methods have also been developed [29]. Furthermore, heuristic approaches such as BLAST [3] and USEARCH [65] can calculate millions of near-optimal alignments in reasonable time.

These algorithms are efficient for the pairwise alignment of two sequences. Calculating an MSA however has been shown to be NP-hard [107, 251]. Thus, for most empirical data sets, other approaches and heuristics are needed [239]. Tools such as CLUSTAL [97], MUSCLE [64], and MAFFT [111] can calculate near-optimal multiple sequence alignments for many thousands of sequences.

A special use case for aligning sequences arises in metagenomic studies, where environmental sequences are often compared to a set of known reference sequences. In these studies, one often first calculates an MSA of the reference sequences, and then successively aligns the environmental sequences against this MSA. This is because calculating an MSA for millions or billions of sequences from scratch is too expensive even for modern tools. Hence, specialized algorithms for this use case have been developed, such as PaPaRa [14, 15] and hmmalign, which is a subprogram of the HMMER suite [62, 63].

2.2.4 Consensus Sequences

When working with a number of related but not identical sequences, it is often convenient to "summarize" homologous characters in form of a *consensus sequence*. Such a sequence is typically calculated based on the relative frequencies of the characters per alignment site. It then represents typical features and motifs of the input set of sequences.

The most straight forward method is to construct *majority rule consensus* sequences [49, 161], where each site is represented by the most frequent character at that site. Figure 2.2 shows an example below the MSA on the right hand side. In order to also include information from the less frequent characters at a site in the consensus sequence, *ambiguity characters* can be used [102]. They allow to denote multiple alternative nucleobases as a single character. For example, if the nucleobases `A` and `G` are similarly frequent at a site, this site is represented by the ambiguity character `R`. See Table A.1 for the full list of character representations.

Using ambiguity characters allows for more involved consensus methods. For example, *threshold consensus* sequences [48, 49] include the most frequent characters that are needed to achieve some given frequency threshold per site, and represent these characters by their ambiguity character. Furthermore, many methods based on fixed thresholds have been proposed, such as Cavener's method [31, 32]; see [49] for a critical comparison.

It is theoretically also possible to directly use the relative frequencies of characters per site in the mathematical frameworks of many downstream analysis methods. This would allow to leverage all of the information contained in the input set of sequences. However, to our knowledge, there is no convention or file format to store such information, and consequently, no way of forwarding this information to the respective tools.

2.3 The Tree of Life

The shared evolutionary history of life gives rise to a branching pattern, where new *lineages* split from a common ancestor. This branching pattern forms a tree-like structure, which classifies organisms in a hierarchy based on common descent.

While this *tree of life* is an expedient and, hence, pervasive model [171], it ignores certain biological and evolutionary events. A strict hierarchy does not allow for reticulate events, such as hybridization [145], genetic recombination [96], or horizontal gene transfer [59, 182, 203]. Although approaches such as networks have been proposed to model these events [100], the simplicity of a hierarchy or tree structure still has proven to be useful in classifying and naming organisms, and understanding their evolutionary relationships.

2.3.1 Taxonomy and Nomenclature

Early attempts at classifying organisms date back to the Greek philosopher Aristotle, who used observable attributes to divide living things into groups [127]. This

approach as well as the efforts of later centuries were non-uniform and inconsistent. The basis for the modern system of classification was established by Swedish botanist Carl Linnaeus in the mid-18th century [57]. He proposed a *nomenclature*, that is, a naming system for organisms, as well as a *taxonomy*, that is, a rank-based classification of organisms [132, 133].

A taxonomic group of organisms is called a *taxon* (plural: *taxa*). Each taxon is associated with a *taxonomic rank*, which can subsume other ranks, thus forming a hierarchy of higher and lower ranks. A taxonomic rank represents the relative level of a group of organisms in the taxonomy. The principal ranks in modern use are *domain*, *kingdom*, *phylum*, *class*, *order*, *family*, *genus*, and *species*, see Figure 2.3. If needed, further ranks can be included in between (such as *sub-genus*), or more refined lower levels be added (such as *strain*, which is a further distinction within a species).

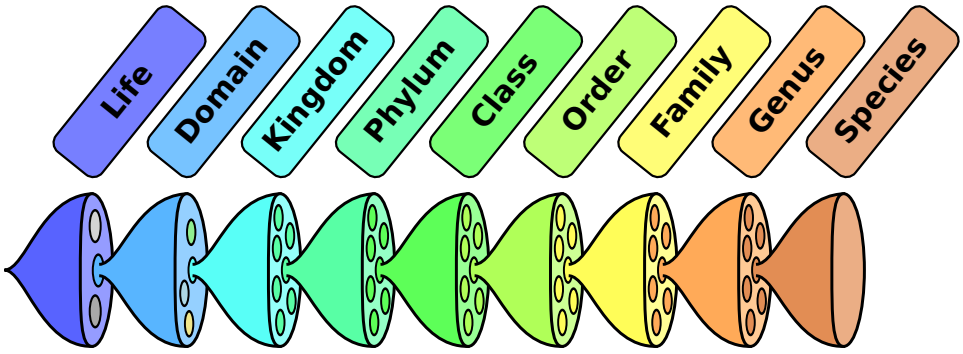

Figure 2.3: Biological classification into taxonomic ranks. The figure depicts a typical set of nested taxonomic ranks [260], which form a hierarchy with increasingly deeper levels towards the right. Source: Image derived from [90].

In order to scientifically name the groups of organisms (taxa) in a taxonomy, the *binomial nomenclature* as introduced by Linnaeus is still prevalent to this day. It uses two terms, often of Latin origin, which respectively specify the taxonomic ranks *genus* and *species* that an organism belongs to, for example *Homo sapiens*.

While early classifications used *phenotypes*, that is, observable characteristics or traits of organisms, modern approaches to taxonomy take genetic information into account [163]. For example, the three-domain system [259, 260] resolves the oldest evolutionary relationships, that is, the highest taxonomic levels, based on 16S rRNA data. Although this classification has been challenged [30, 89, 162], it is widely used. It divides cellular life forms into the three domains *bacteria*, *archaea*, and *eukaryota*; the latter is further separated into kingdoms, which include the kingdoms *fungi*, *plants*, and *animals*.

2.3.2 Phylogenetic Trees

The classification of organisms into a taxonomy is based on (subjective) dissimilarity and thus arbitrary: The number of organisms that are grouped into a taxon at a

given rank can vary, and the separation into discrete ranks does not reflect the gradual nature of evolution [79]. A more involved approach that can resolve these issues is *phylogenetics*, which is the study of the evolutionary history and relationships of biological entities (individuals, species, populations).

The evolutionary relationships of such entities are called their *phylogeny*. As the true phylogeny of a set of taxa is unknowable, it has to be inferred from data that is available. A phylogenetic analysis uses inference methods that evaluate observed heritable traits in order to resolve the phylogeny under a given model of evolution of these traits. While phenotypes can be used, modern phylogenetic inference is mostly based on DNA data. The result of a phylogenetic analysis is a *phylogenetic tree*, also called an *evolutionary tree*, or—synonymously—a *phylogeny*. Figure 2.4 shows two examples of phylogenetic trees.

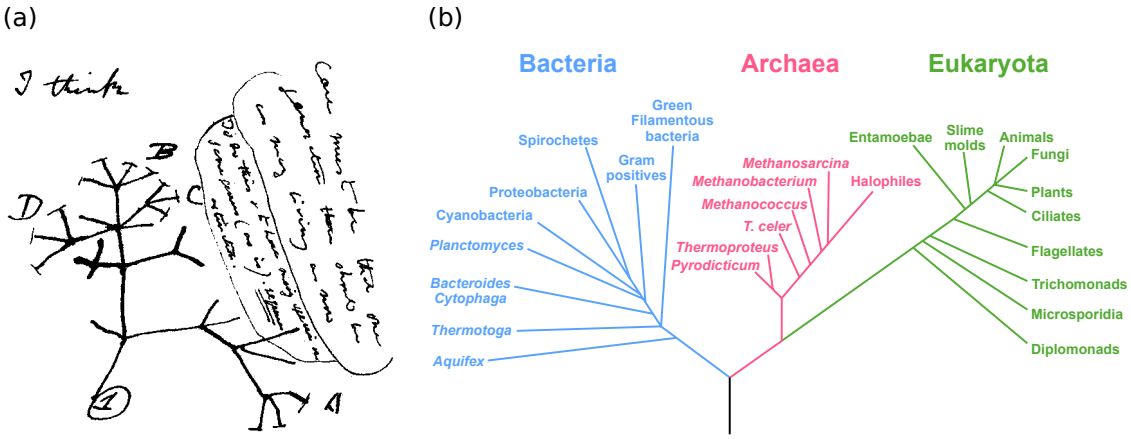

Figure 2.4: Exemplary phylogenetic trees. (a) In 1837, Charles Darwin sketched his first evolutionary tree below the words "I think" in his notebook on "Transmutation of Species". Source: Image derived from [45]. (b) A modern phylogenetic tree showing the three-domain system [259, 260], which emphasizes the separation of *bacteria*, *archaea*, and *eukaryota* based on 16S rRNA genes. The black branch at the bottom represents the speculative last universal common ancestor of all living organisms. Source: Image derived from [75].

Properties of Trees

The leaf nodes, or *tips*, of the tree represent living (*extant*) biological entities such as species, strains, individual organisms, or even cells of a multicellular organism. Thus, the tips are often referred to as the *taxa* of the tree, which is meant as a generic term that includes all the above entities. The tips are often named according to the entities they represent (e. g., species names); such a tree is called a *labeled* tree. The inner nodes on the other hand are usually anonymous and represent speciation events, where two novel lineages arose from a putative common ancestor. The branching pattern of a phylogenetic tree thus reveals the evolutionary history of its taxa.

The edges of the tree, also called its *branches*, can have associated *branch lengths*, which represent the evolutionary time between the two adjacent nodes, for example, measured as the average change in nucleobases between their respective sequences. The unique path between any two nodes thus can be interpreted as a measure of evolutionary relatedness of the taxa represented by the nodes.

A phylogenetic tree is *rooted* if it is a directed tree that has a unique *root node*, which corresponds to the putative common ancestor of the other nodes in the tree. See Figure 2.4(b) for an example. As evolution is a processes that happens over time, from a biological point of view, every tree has a root. However, most models of DNA evolution are time-reversible, meaning that the direction of change in the sequences cannot be inferred from the data under such models, see Section 2.4.2 for details. Thus, tree inference methods can also yield *unrooted* trees without direction and without a root node. In these methods, for computational reasons, often a *virtual root* is used, which is a hypothetical additional node placed on a branch of the tree. For tasks such as traversing a tree, but also in order to store a tree in a file, unrooted trees usually have a distinguished, but arbitrary, "starting" node called a *top-level trifurcation*. An unrooted tree can be rooted a posteriori, for example by using an *outgroup* of taxa that are closely related to the group of taxa of interest (the *ingroup*), but not part of it. Then, a root node can be placed on the branch that separates the outgroup from the ingroup.

An inner node that has exactly three neighboring nodes is called a *bifurcation* or a *bifurcating* node. In rooted trees, these neighbors are the the parent and the two children of a node, hence the name. An inner node with more neighbors is called a *multifurcation* or a *multifurcating* node. This naming also applies to the whole tree: A tree, where each inner node (with the exception of the root node in a rooted tree) is bifurcating, is also called a bifurcating tree. Otherwise (if there is at least one multifurcating node), it is a multifurcating tree. Note that in evolution, an actual multifurcation event is highly unlikely, as it corresponds to the simultaneous formation of more than two new lineages from a single ancestral lineage. Multifurcating trees are for example used when relationships cannot be properly resolved from the existing data, or to summarize a set of otherwise contradicting trees.

Each edge of the tree induces a *bipartition* or *split* of the taxa of the tree into two groups, one on each side of the edge. Splits of edges that are adjacent to tip nodes are *trivial*, as they appear in every possible topology of a given set of taxa. Therefore, the *non-trivial* splits are mostly of interest. The set of bipartitions induced by the edges of a tree uniquely defines the tree topology; for example, the tree in Figure 2.5(a) is described by the set of bipartitions $B = \{(\,\mathsf{ABC}|\mathsf{DE}\,), (\,\mathsf{AB}|\mathsf{CDE}\,)\}$.

For two trees T_1 and T_2 with the same taxa, but differing topologies, their sets of bipartitions B_1 and B_2 can be used to define distance metrics between them. The *Robinson-Foulds* (RF) distance [202], or symmetric difference metric, for instance is defined as the number of bipartitions that are unique to either of the trees:

$$\mathrm{RF}(T_1, T_2) = 1/2 \cdot (\,|B_1 \cup B_2| - |B_1 \cap B_2|\,) \tag{2.1}$$

The factor of 1/2 is derived from the fact that we use all bipartitions here, instead of only the non-trivial ones. This unweighted, absolute distance is often used when comparing phylogenies. There also exists weighted and relative variants, as well as a variant called *branch score* [123], which further takes the branch lengths of the trees into account.

A set of taxa is *monophyletic*, if there is a bipartition of the tree that splits these taxa from all other taxa of the tree. In a rooted tree, the node at the end of that edge is then the putative common ancestor of these taxa. Furthermore, a monophyletic set of taxa is called a *clade* of the tree; in other words, a clade is a subtree that is separated from the rest of the tree by one edge. For example, the taxa A, B, and C in Figure 2.5 are monophyletic—that is, they form a clade of the tree. Lastly, a non-monophyletic set of taxa is *paraphyletic*.

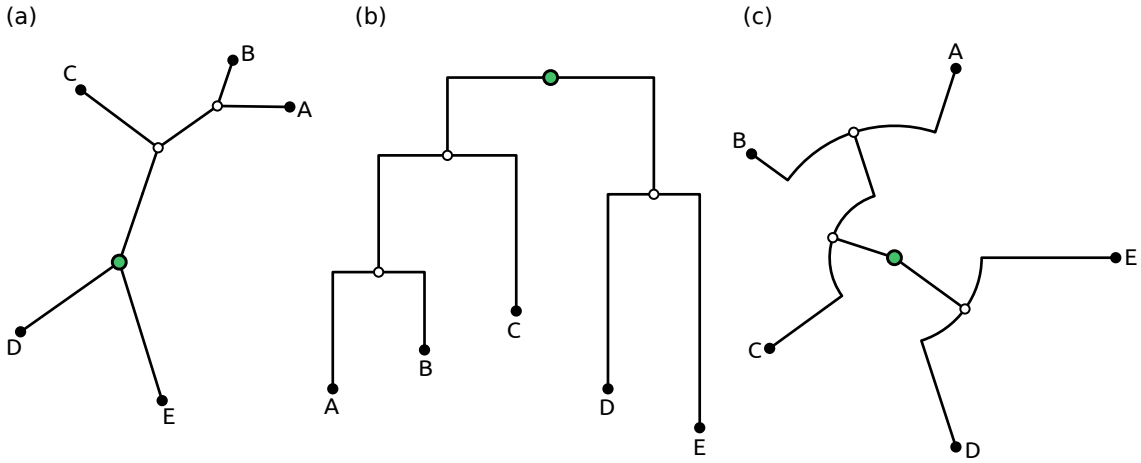

Figure 2.5: Types of phylogenetic trees. Here, we show three different types of labeled, bifurcating trees. Tip nodes are marked with black dots, inner nodes with white dots, and the top-level trifurcation or root node with a larger green dot. (a) An unrooted tree with five taxa. One node is arbitrarily selected as top-level trifurcation. (b) The same tree topology, but rooted on the inner branch that splits the taxa D and E from the other taxa. The tree is drawn in rectangular style, where vertical lines correspond to branch lengths. The horizontal lines are simply used to distribute the taxa, and have no biological interpretation. (c) The same tree again, but this time drawn in circular style. Here, radial lines correspond to branch lengths, while arcs only serve drawing purposes.

Practical Aspects of Trees

While the topology of the tree is what models the evolutionary relationships, there are several ways of visualizing or drawing that information. Figure 2.5 shows some examples, which all visualize the same topology (except for the rooting). The figure also summarizes some of the terms and concepts introduced above. Different drawing styles each have their advantages. For example, in a rectangular tree (Figure 2.5(b)),

branch lengths are easier to read and compare, while a circular tree (Figure 2.5(c)) can fit more taxa in the same drawing area.

Taxonomy and phylogeny serve a related, but different purpose: While the former is a system of classification, the latter reveals evolutionary history. However, there is a correspondence between a taxonomy and a rooted phylogeny: Inner nodes of the tree constitute older evolutionary relationships, which are represented by the higher ranks of the taxonomy. Figure 2.4(b) shows such a correspondence for the three domains of life. It is however possible that the taxa at one rank of the taxonomy are not monophyletic in the tree. In this case, the two are *incongruent*.

The most common file format for storing phylogenetic trees is the `Newick` format [6], which uses parentheses and commas to specify the nesting structure of the tree, and allows to store node labels and branch lengths. The `NEXUS` format [144] is a container format for biological data, and internally also relies on the `Newick` format for storing trees. Furthermore, the `phyloXML` format [93] is an `XML` based format that allows to store arbitrary data at the nodes and edges of the tree.

2.3.3 Tree Inference

A phylogeny can be inferred from data that has per-taxon traits which are homologous, that is, which have evolved from the same traits in the common ancestor and are thus comparable [72, 265]. While historically these traits were mostly phenotypes (bone shapes and sizes, metabolism, etc.), the focus has since shifted towards molecular data such as DNA and amino acid sequences, as their *phylogenetic signal* is generally more abundant and less biased [98]. Most often, a multiple sequence alignment is used, whose homologous sites represent the traits of the taxa. To determine the degree of relatedness between taxa, mathematical models of trait evolution are employed.

The general concept of tree inference is then to put closely related taxa close to each other in the phylogeny. Hence, a tree inference can be thought of as an optimization problem, which searches for the best tree given an optimality criterion. However, the space of all possible tree topologies is too large for an exhaustive brute-force search for virtually all empirical datasets. For a given number of taxa n, the number of distinct tree topologies N is given as $N(n) = \prod_{i=3}^{n} (2i - 5)$, which grows over-exponentially fast [72]. There are thus different approaches and heuristics to conduct tree search.

Distance based methods such as *Unweighted Pair Group Method with Arithmetic Mean* (UPGMA) [221] and *Neighbor Joining* [207] use a pairwise distance matrix between sequences, and thus do not necessarily need an alignment. *Maximum Parsimony* [210] uses an optimality criterion that is based on Occam's razor, that is, it yields the tree that explains the observed tip sequences (taxa) with the minimal number of substitutions (mutations). The *Maximum Likelihood* (ML) method [71] employs statistical techniques in order to evaluate the probability of a given phylogenetic tree with respect to a given alignment, and successively search the tree space for the most likely tree, see Section 2.4. Furthermore, *Bayesian Inference* also relies

on the evaluation of tree probability [265], and uses Bayes' theorem to calculate the posterior distribution of the relevant evolutionary processes; it thus can incorporate prior empirical knowledge into the process.

Typical software tools for inferring ML trees include IQ-TREE [179], FastTree [195], and RAxML [228], while Bayesian inference can for example be conduced using tools such as BEAST [232] or MrBayes [205].

2.4 Maximum Likelihood Tree Inference

In the context of this work, we are mostly interested in Maximum Likelihood (ML) tree inference. It uses a probabilistic framework in which the (phylogenetic) likelihood

$$\mathcal{L}(\text{ MSA } \mid T, \bar{b}, M, \bar{\theta}\ ) \tag{2.2}$$

is evaluated that an observed MSA is the outcome of an evolutionary history described by a phylogenetic tree with topology T and branch lengths $\bar{b}$, under a given model of trait evolution M with parameters $\bar{\theta}$. For a fixed model M (see Section 2.4.2), the likelihood can be expressed as a function of T, $\bar{b}$ and $\bar{\theta}$, which is known as the *phylogenetic likelihood function* (PLF).

2.4.1 Tree Search

By maximizing the PLF using maximum likelihood estimation, the parameter values (including tree topology) are found which best explain the observed data. This process is called *tree search*. Typically, the estimates are obtained in an iterative process, which alternates between two phases until a (potentially local) optimum is found:

1. Optimizing the tree topology T, given the branch lengths $\bar{b}$ and the model parameters $\bar{\theta}$.
2. Optimizing the branch lengths of the given tree topology, as well as the model parameters.

Finding the most likely tree topology is a discrete optimization problem, which has been shown to be NP-hard under the ML criterion [33]. Furthermore, the evaluation of the PLF is computationally expensive, as it involves many floating point operations, see Section 2.4.4. A general heuristic for the tree search that avoids an exhaustive evaluation of the tree space is thus as follows. First, a starting tree is generated, either randomly, or using methods such as Neighbor Joining or Maximum Parsimony. Then, the likelihood of the tree is successively improved by applying topological rearrangements (*moves*) of its taxa and clades. For instance, in *greedy hill-climbing* [228], only those moves are applied (*accepted*) that immediately improve the PLF.

For a fixed tree topology T, the maximum likelihood estimates of the branch lengths $\bar{b}$ and the model parameters $\bar{\theta}$ are usually obtained with general-purpose numerical optimization methods. Since the derivatives of the PLF can be easily computed, the Newton-Raphson method [270] is often used for optimizing the branch lengths, see Section 2.4.5. Model parameters are commonly optimized with Brent's method [27] or with the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [74].

2.4.2 Models of Molecular Sequence Evolution

So far, we assumed to have a model M for describing the evolution of the traits that are used for inferring the tree. For sequence data, such a model yields an estimate of the evolutionary distance between the sequences of different taxa. Because the inference assumes homologous traits, the only mutations that are typically considered in aligned sequences are substitutions.

Markov Chain Model

Most commonly, a continuous-time Markov chain (MC) is used to describe the evolution of a single site within a set of aligned sequences [76]. For DNA data, the MC has four states `A`, `C`, `G`, and `T`, which correspond to the nucleobases, and transitions between the states correspond to their substitutions, see Figure 2.6. While the MC model ignores aspects such as natural selection and the molecular mechanisms of evolution, it describes the relative rate of changes in a way that allows multiple substitutions to occur along the same branch (`T` → `A` → `G`).

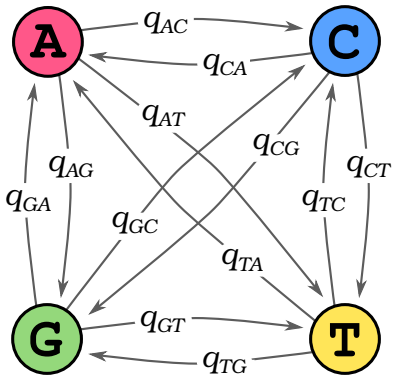

Figure 2.6: Markov chain model of nucleotide substitutions. The evolution of characters at a site in an alignment can be modeled as a Markov chain (MC). The states of the MC for DNA data are the four nucleobases `A`, `C`, `G`, and `T`. The model allows transitions with rates q_{ij} with $i, j \in \{ A, C, G, T \}, i \neq j$ between all states, which correspond to substitutions of the nucleobases.

The process of state transitions is defined by the substitution rate matrix

$$Q = \begin{pmatrix} -q_A & q_{AC} & q_{AG} & q_{AT} \\ q_{CA} & -q_C & q_{CG} & q_{CT} \\ q_{GA} & q_{GC} & -q_G & q_{GT} \\ q_{TA} & q_{TC} & q_{TG} & -q_T \end{pmatrix} \tag{2.3}$$

$$-q_i = -\sum_{j \neq i} q_{ij},\ i, j \in \{ A, C, G, T \}$$

where the elements q_{ij} are the *instantaneous transition rates* from state i to state j. The rows of the Q-matrix have the requirement to sum to 0, by which the diagonal elements q_i are defined.

The expected number of substitutions at an alignment site between two nodes of the tree is expressed as the branch length b between the nodes, and is a measure of evolutionary time t between them. Under the MC model, evolutionary time and branch length are proportional to each other with the *evolutionary rate* r being their proportionality factor: $t = r \cdot b$. Then, for a given time t, the *transition probabilities* $p_{ij}(t)$ between states in a stationary process are obtained by exponentiating the Q-matrix [266]. These probabilities are specified by the matrix

$$P(t) = e^{Qt} \tag{2.4}$$

For positive transition rates $q_{ij} > 0, \forall i \neq j$, if the process runs long enough, the Markov chain eventually reaches the unique *stationary* distribution $\Pi = (\pi_A, \pi_C, \pi_G, \pi_T)$, with π_i being the proportion of time spent in state i. If the Markov process reached equilibrium after running long enough, it can be interpreted as having generated the sequences of the MSA. In that case, Π is the *equilibrium base composition* of the MSA, and π_i are the *equilibrium* or *stationary base frequencies* of the MSA.

Time-Reversible Models

As mentioned before, most models of DNA evolution assume a *time-reversible* Markov process, which means that $\pi_i q_{ij} = \pi_j q_{ji}\ \forall\, i \neq j$. This assumption is biologically not meaningful, as evolution is a process in time, and thus does have a direction. It however allows for simplified calculations: The Q-matrix of a time-reversible model can be formulated as the product of a symmetric rate matrix $R = \{r_{i \leftrightarrow j}\}$ and a diagonal matrix with the stationary base frequencies:

$$Q = R \cdot diag(\pi_i) = \begin{pmatrix} -q_A & r_{A \leftrightarrow C} \cdot \pi_C & r_{A \leftrightarrow G} \cdot \pi_G & r_{A \leftrightarrow T} \cdot \pi_T \\ r_{A \leftrightarrow C} \cdot \pi_A & -q_C & r_{C \leftrightarrow G} \cdot \pi_G & r_{C \leftrightarrow T} \cdot \pi_T \\ r_{A \leftrightarrow G} \cdot \pi_A & r_{C \leftrightarrow G} \cdot \pi_C & -q_G & r_{G \leftrightarrow T} \cdot \pi_T \\ r_{A \leftrightarrow T} \cdot \pi_A & r_{C \leftrightarrow T} \cdot \pi_C & r_{G \leftrightarrow T} \cdot \pi_G & -q_T \end{pmatrix} \tag{2.5}$$

The matrix describes the most general model of DNA evolution, where all 6 substitution rates $r_{i \leftrightarrow j}, i \neq j$ and all 4 base frequencies π_i can be different. This model is called the Generalized Time-Reversible (GTR) model [236]. As the sum of the base frequencies must be 1, and as the substitution rates are usually normalized by requiring that $r_{G \leftrightarrow T} = 1.0$, the GTR model has a total of 8 free parameters (that is, 3 base frequencies, and 5 substitution rates). The base frequencies can also be estimated from the character frequencies in the given MSA, in which case they are called the *empirical* base frequencies.

There are also more restrictive models, which have fewer free parameters, and are thus more robust if data for estimating them is sparse, at the expense of descriptiveness. The Jukes-Cantor model (JC69) [106] has no free parameter and assumes equal substitution rates $r_{i \leftrightarrow j} = 1$, $i \neq j$ and equal base frequencies $\pi_i = 1/4$. The K80 model [114] adds a free parameter κ, which describes the ratio between two types of substitutions that are not equally likely to occur in evolution: $r_{A \leftrightarrow C} = r_{G \leftrightarrow T} = \kappa \cdot r_{A \leftrightarrow G} = \kappa \cdot r_{A \leftrightarrow T} = \kappa \cdot r_{C \leftrightarrow G} = \kappa \cdot r_{C \leftrightarrow T}$. The F81 model [71] instead extends the JC69 by allowing different base frequencies: $\pi_A \neq \pi_C \neq \pi_G \neq \pi_T$. The HKY85 model [94] combines the K80 model and the F81 model, and hence has 4 free parameters. Further models have also been proposed, which offer compromises between the number of free parameters and the expressiveness of the model [266].

The state space of the Markov process becomes significantly larger for protein data, as it needs to comprise 20 standard amino acids instead of 4 nucleobases. Hence, the GTR model for protein data has $(400 - 20)/2 - 1 + 19 = 208$ free parameters. Typical amino acid alignments do not contain enough data to reliably estimate these parameters, and thus easily lead to over-fitting. Thus, so-called *empirical* amino acid models are commonly used, which have substitution rates and equilibrium base frequencies that were pre-estimated on large collections of reference alignments. Among others, some popular models include DAYHOFF [50], WAG [258], and LG [124].

2.4.3 Further Aspects of Tree Inference

Evolution is an incredibly complex process with intricate details. Many more models and methods have thus been proposed to refine tree inference [266], of which we here briefly introduce a few.

Rate Heterogeneity

The models of sequence evolution described above make the simplifying assumption that the alignment sites evolve independently and are identically distributed. However, certain regions of DNA or amino acid sequences are under higher evolutionary pressure than others, for example if they describe important molecular functions that are conserved in their evolutionary history. It is thus expected that some alignment sites evolve faster than others. That is, the evolutionary rate r of these sites is not constant across the alignment. In the context of phylogenetic inference, several models of *rate heterogeneity among sites* have been proposed to account for this, some of which are described in the following.

A simple model is the *proportion of invariable sites*, where the likelihood of an alignment site is influenced by a parameter $p \in [0, 1]$ that describes the proportion of sites that are assumed to be identical (*invariable*) across all taxa. The more elaborate Γ model [263] postulates a shape parameter $\alpha > 0$ which models the rate heterogeneity as a gamma distribution $\Gamma(\alpha)$. The distribution shape ranges from exponential-like ($\alpha < 1$, high rate heterogeneity) to normal-like ($\alpha > 10$, low rate heterogeneity). Thus, by optimizing the single free parameter α, different unimodal rate heterogeneity profiles can be approximated. The CAT or *per-site rates* model [227] is a compute- and memory-efficient approximation of the Γ model, which explicitly assigns one of K rate categories to each alignment site instead of using a distribution of rates. Lastly, the FreeRate model [264] allows for multimodal distributions by using K rate categories and respective weights, which can approximate any distribution at the cost of having to estimate these free parameters.

Alignment Partitioning

Apart from the evolutionary rate r, also the substitution patterns among the sites of an MSA can differ. In order to account for this, the MSA can be split into different *partitions*, where each such partition is assigned its own model of evolution. For example, as three nucleobases code for one amino acid in regions that encode for proteins (see Section 2.1), three partition can be used, each modeling the evolution of the first, second, and third nucleobase of each amino acid. Furthermore, large multi-gene MSAs can be split into partitions corresponding to individual genes, which might be under different evolutionary pressure each.

Constrained Trees

The tree search (see Section 2.4.1) can (theoretically) yield any topology from the vast space of possible trees. It is however often serviceable to run a *constrained* tree search, for example to include prior knowledge about the taxa, to maintain congruence with a given taxonomy, or because some other constraints are required. Such a constraint can for example be specified by enforcing certain bipartitions to be retained in the tree, that is, splits of the taxa that must be separated from each other in the tree. As a bipartition is induced by a branch in the tree, this is equivalent to starting the tree search with a multifurcating tree, and then resolving these multifurcations without changing the other parts of the tree. A constrained search yields a *constrained tree*.

2.4.4 Likelihood Computation

We here introduce the basic computational aspects of the Maximum Likelihood method. For a more exhaustive description of the topic, see [266]. We assume a fixed tree topology T, fix branch lengths $\bar{b}$, as well as a given model of sequence evolution M with parameters $\bar{\theta}$. That is, we do not cover the tree search itself here, but describe how to compute the likelihood $\mathcal{L}$ (Equation 2.2) for a given MSA under these conditions.

A central point of the ML method is to account for the unknown states at the inner nodes of the tree. That is, the total likelihood is obtained by summing over the probabilities of every possible state of the inner nodes, which can be efficiently computed by the *Felsenstein pruning algorithm* (FPA) [71]. It traverses the tree in post-order fashion, that is from the tips towards the (virtual) root, and recursively calculates a so-called *conditional likelihood vector* (CLV) at each inner node.

In a sense, a CLV summarizes the subtree below its corresponding node. For every alignment site and every state, it describes the *conditional likelihood* of the node to be in that state at that site, given the subtree topology and its branch lengths, for the respective subset of the alignment (tip sequences). We here assume a set N of states, that is, the sequences consist of characters $c \in N$, for example $N = \{ \texttt{A}, \texttt{C}, \texttt{G}, \texttt{T} \}$. Furthermore, for simplicity, we do not consider alignment partitioning or rate heterogeneity among sites here, and thus use a fixed evolutionary rate r. Then, a CLV contains $|N|$ elements per alignment site, each describing the conditional likelihood of being in the corresponding state. For DNA data, these are CL(A), CL(C), CL(G), and CL(T).

Felsenstein Pruning Algorithm

In order to start the recursion of the FPA, first, the CLVs at the tip nodes have to be initialized. In principle, these can be the actual likelihoods of observing the characters $c \in N$ at the corresponding site. However, this uncertainty is rarely available in empirical data. Thus, tip nodes are usually initialized with "pseudo-CLVs", where for instance a nucleobase A in the alignment yields CL(A) = 1, and CL(C) = CL(G) = CL(T) = 0.

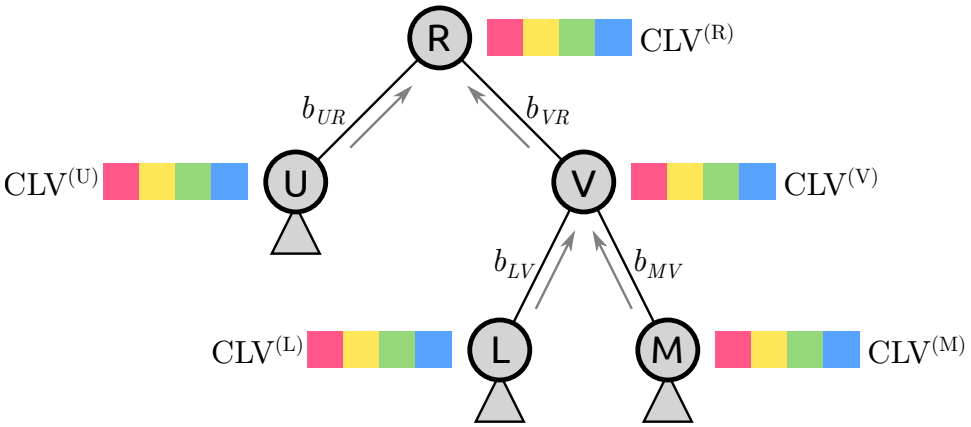

Figure 2.7: Felsenstein pruning algorithm. An exemplary tree topology with a (virtual) root R, an inner node V, and three other nodes U, L, and M, and branch lengths between nodes. Potential subtrees are marked by triangles below nodes. Each node has a CLV assigned to it, which "summarizes" the subtree below it. A CLV stores a conditional likelihood for every alignment site and for every state. Here, for simplicity, we show the CLVs for one site and for four states, which for instance represent the likelihood of that site to be in either state of the four nucleobases.

After the tip CLVs have been initialized, the algorithm traverses up the tree, see Figure 2.7. This can be thought of as moving along the branches towards a parent node, which induces the possibility of state transitions. This step is thus where the model M is employed (see Section 2.4.2). As we assumed a fixed evolutionary rate r, we can infer the time t between two nodes from the branch length b between them: $t = r \cdot b$. Then, for a given branch, we can compute the probability $p_{ij}(t)$ of a transition from state i to state j after moving along the branch, see Equation 2.4. Note that the probability p_{ij} depends on the branch length, meaning that for every branch (and every update in its branch length during optimization, see Section 2.4.5), a separate P-matrix has to be computed from the Q-matrix of the model.

At a parent node whose children have been computed, we can now apply the recursion step of the algorithm. For instance, in the topology shown in Figure 2.7, the CLV of an inner node V can be computed given its children L and M. For the computation, the CLVs of the two child nodes, as well as the transition probabilities p_{ij} for the branch lengths b_{LV} and b_{MV} of the branches towards the parent are needed. Then, a single entry of the CLV, that is, the conditional likelihood of node V to be in state c at site s, is

$$\mathrm{CLV}_{s,c}^{(V)} = \left(\sum_{j \in N} p_{cj}(r \cdot b_{LV}) \cdot \mathrm{CLV}_{s,j}^{(L)} \right) \left(\sum_{k \in N} p_{ck}(r \cdot b_{MV}) \cdot \mathrm{CLV}_{s,k}^{(M)} \right) \tag{2.6}$$

The equation can be interpreted as follows: The product of transition probability and conditional likelihood represents a change from state c to another state in N. By summing this product for all states, all possible inner states are accounted for. Finally, the product of the two sums is the new conditional likelihood of the node being in state c at site s, given the evolutionary history of its children and their subtrees.

By repeating the computation for every state $c \in N$ and every site s of the alignment, the complete CLV for node V is computed. The recursion is then applied to all nodes upwards the tree, until all CLVs are computed.

Likelihood Evaluation at the Root

Once all CLVs are computed, the overall likelihood $\mathcal{L}$ can be computed from the CLV of the root node. Given the root node R as shown in Figure 2.7, the total *per-site likelihood* $\mathcal{L}_s$ of an alignment site s is accumulated from the conditional likelihoods of all states, taking their respective base frequencies π_i into account:

$$\mathcal{L}_s = \sum_{i \in N} \pi_i \cdot \mathrm{CLV}_{s,i}^{(R)} \tag{2.7}$$

Due to the time reversibility of the model, for unrooted trees, a *virtual* root can be used, that is, an additional node that is presumed to be present on a branch of the tree. If for example the node R in Figure 2.7 represents a virtual root, the

two branches between nodes U and V are actually one branch with branch length $b_{UV} = b_{UR} + b_{VR}$. Then, an alternative way of computing the per-site likelihood is what we here call the (per-site) *edge likelihood*. Instead of using the CLV at the virtual root R, we can use the CLVs of U and V, and the corresponding branch length b_{UV} for the computation. In that case, state transitions along the branch have to be additionally accounted for in the computation. The per-site edge likelihood of an alignment site s can then be computed as

$$\mathcal{L}_s = \sum_{i \in N} \sum_{j \in N} \pi_i \cdot \mathrm{CLV}_{s,i}^{(U)} \cdot p_{ij}(r \cdot b_{UV}) \cdot \mathrm{CLV}_{s,j}^{(V)} \tag{2.8}$$

This way of calculating the edge likelihood $\mathcal{L}_s$ works for any two adjacent nodes, if their respective CLVs have been computed to represent the two subtrees induced by the branch between the nodes.

Finally, the overall likelihood $\mathcal{L}(\mathrm{MSA} \mid T, \bar{b}, M, \bar{\theta})$ for the entire MSA can be computed. For mathematical simplicity, the sites are generally assumed to evolve independently, although this is not expected to be the case from a biological perspective. Then, for an alignment with m sites, the overall likelihood is simply the product of the per-site or edge likelihoods:

$$\mathcal{L} = \prod_{s=1}^{m} \mathcal{L}_s \tag{2.9}$$

For computational reasons, and to avoid numerical underflow, in practice, the logarithm of the likelihood (*log-likelihood*) is typically used. Furthermore, as identical sites yield exactly the same likelihood, such sites are often compressed and the respective likelihood is multiplied with an according *weight*.

2.4.5 Branch Length Optimization

Another important aspect of the tree search is the optimization of the branch lengths of the tree. That is, for a given tree topology T, and a fixed model M with parameters $\bar{\theta}$, we want to compute the branch lengths $\bar{b}$ that maximize the likelihood $\mathcal{L}(\mathrm{MSA} \mid T, \bar{b}, M, \bar{\theta})$. This procedure is called *branch length optimization* (BLO), and typically uses the Newton-Raphson method [270], as mentioned in Section 2.4.1.

We consider to optimize a single branch length b. In order to maximize $\mathcal{L}$, we need to find the root of the first derivative $\mathcal{L}'$. The Newton-Raphson method takes an initial value for b and then iteratively approximates values that lead closer to the root:

$$b_{n+1} = b_n - \frac{\mathcal{L}'}{\mathcal{L}''} \tag{2.10}$$

Note that the derivatives $\mathcal{L}'$ and $\mathcal{L}''$ can be obtained analytically [266], and have to be computed in every iteration. The algorithm stops when the change in b between two iterations is below a threshold, that is, when then optimization *converges*. This procedure is repeated for all branch lengths $\bar{b}$ in the tree.

2.5 Phylogenetic Placement

In studies of sequence data, one of the most common tasks is a phylogenetic analysis of the data, that is, to infer the evolutionary context of the sequences. However, since the amount of sequence data produced in typical metagenomic studies can be quite substantial, computational challenges and bottlenecks arise [214]. In particular, both calculating an MSA and inferring a phylogeny are NP-hard [33, 107], and thus impractical or infeasible for large datasets. Furthermore, metagenomic reads are often short, and hence lack phylogenetic signal to robustly infer a tree and to properly resolve their relationships.

Thus, *phylogenetic placement*, also called *evolutionary placement*, has been developed for conducting phylogenetic analyses of such data [174, 250], as implemented in tools such as PPLACER [158], RAxML-EPA [16], and EPA-NG [10]. Instead of resolving the phylogeny of a set of metagenomic sequences, phylogenetic placement treats each sequence, called a *query sequence* (QS), separately. It evaluates how these QSs relate to an existing *reference tree* (RT) based on known reference sequences. For each QS, it computes the probabilities of *placing* the sequence on the branches of the RT, thereby classifying them into a phylogenetic context of related sequences, without the need to resolve relationships between the QSs.

2.5.1 Pipeline and Computation

In the most common use case, the QSs are reads or amplicons from environmental samples. Most often barcoding regions or marker genes such as 16S or 18S are used (see Section 2.2.2), but there also exist studies that use different, or even a set of, maker genes [233]. Furthermore, other types of sequences such as $_{\text{mi}}$tags [138] can be used.

The RT and the reference sequences it represents are typically assembled by the user so that they capture the expected species diversity in the samples. We however proposed an automated approach for assembling suitable sets of reference sequences [42], which we describe in Chapter 3. Distinct samples from one study are typically placed on the same underlying RT in order to facilitate comparisons between the samples, see Section 2.5.5.

We here assume to have given a set of suitable reference sequences, their alignment, and an RT inferred from them. As phylogenetic placement used a maximum likelihood criterion, the RT has to be strictly bifurcating. Prior to the placement, the QSs need to be aligned against the reference alignment of the RT by programs such as PaPaRa [14, 15] or HMMALIGN [62, 63], see also Section 2.2.3. The input to phylogenetic placement are then the reference tree (RT), its underlying alignment, and the aligned query sequences (QSs).

Computation for one Query Sequence

The placement is conducted for each QS separately, always using the same fixed RT as a starting point. Each branch of the tree is evaluated as a potential *placement*

location of the QS, which indicates how likely the branch is to be the ancestor of the QS. In Figure 2.8, the procedure for one QS and one branch (between nodes D and P) is shown: The sequence is inserted as a new tip node Q on the branch, connected to it by a new *pendant* branch and a new node C. This splits the original branch into two parts, called the *distal* and the *proximal* branch, respectively, which are named according to the direction of the root of the tree. Note that the tree can also be unrooted, in which case a top-level trifurcation is typically used as root.

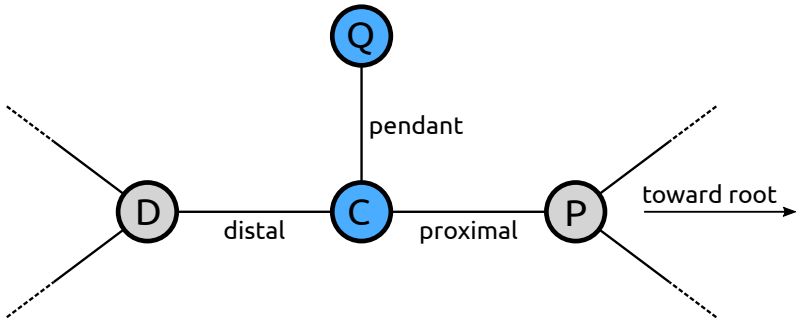

Figure 2.8: Terminology of a phylogenetic placement. The nodes D and P belong to the reference tree (RT). When placing a query sequence (QS), the branch between them is split into two parts by a new node C, which serves as the attachment point for another new node Q that represents the QS. The *pendant* branch leads to Q. The original branch is split into the *proximal* branch, which leads towards the root of the RT, and the *distal* branch, which leads away from the root.

In the next step, the branch lengths of the tree are optimized, as explained in Section 2.4.5. After the optimization, the sum of the lengths of the distal and proximal branches is not necessarily equal to the original branch length between D and P. Thus, typically, the two lengths are proportionally rescaled to maintain this equality.

Lastly, the likelihood of the tree with the newly attached sequence is evaluated as explained in Section 2.4.4. Note that the likelihood computation uses the MSA (extended by the query sequence), the tree topology and branch lengths, as well as the model and its parameters as before. The model of nucleotide evolution should be the same that was used when inferring the tree, see Section 2.4.2. After this, the newly created nodes on the branch are removed again, thus restoring the original reference tree.

The above procedure is repeated for every branch i of the tree T, yielding a set of likelihood scores $\mathcal{L}(i)$ for each possible placement location. In other words, for each branch of the tree, the process yields a so-called *placement* of the QS, that is, an optimized position on the branch, along with a likelihood score for the whole tree. The likelihood scores for a QS are then transformed into probabilities, which quantify the uncertainty of placing the sequence on the respective branch [231, 250].

The probability of placing a QS on a branch q is called the *likelihood weight ratio* (LWR), and is calculated relative to the other branches i of the tree T as

$$\mathrm{LWR}(q) = \frac{\mathcal{L}(q)}{\sum_{i \in T} \mathcal{L}(i)} \tag{2.11}$$

By construction, the sum of the LWRs of all branches for a single QS is 1.0. It can thus be interpreted as a probability distribution over the branches of the tree, as shown in Figure 2.9. For most use cases, the LWRs and the respective placement on the branches are the most important values, while pendant lengths are rarely used in downstream analyses. However, long pendant lengths can indicate that the RT is missing sequences that are closely related to the QS.

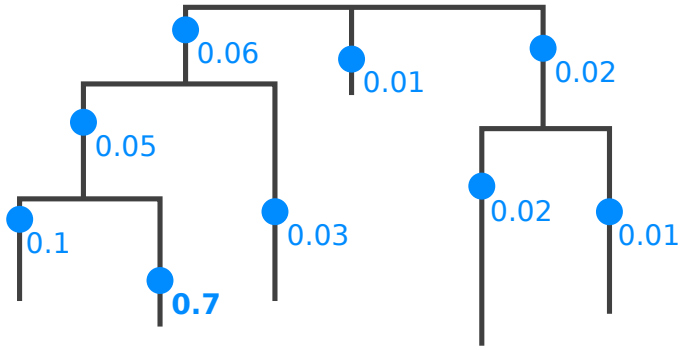

Figure 2.9: Phylogenetic Placement of a Query Sequence. Each branch of the reference tree is tested as a potential insertion position, called a *placement* (blue dots; pendant lengths are ignored here). Note that placements have a specific position on their branch, due to the branch length optimization process. A probability of how likely it is that the sequence belongs to a specific branch is computed (numbers next to dots), which is called the *likelihood weight ratio* (LWR). The bold number (0.7) denotes the most probable placement of the sequence.

Accelerations

The most expensive part of the placement computation are the branch lengths optimizations. Thus, several techniques have been developed to accelerate the placement process [10].

Firstly, above, all branches of the tree were optimized when evaluation a placement location, which gives the most accurate results. In practice however, it suffices to only optimize the three branches of the placement location without losing too much accuracy.

Secondly, because the reference tree is fixed (except for the temporarily created nodes during the computation), the CLVs of all possible subtrees can be precomputed, which drastically accelerates the likelihood evaluation. Using Figure 2.8 as an example, with two CLVs of the nodes D and P, and one application of the Felsenstein Pruning Algorithm (see Section 2.4.4), the CLV of node C can be computed.

Then, the final likelihood can be evaluated as the edge likelihood of the pendant edge, using this CLV as well as the pseudo-CLV of node `Q`.

Thirdly, further acceleration can be achieved with a *pre-placement* heuristic: A first approximate evaluation of a placement location can be conducted without branch length optimization by using distal and proximal lengths based on the respective original branch length of the tree, and a fixed default pendant length. Then, only the most likely fraction of locations is thoroughly evaluated, that is, including branch length optimizations. This way, millions or even billions of sequences can be placed within acceptable time [10]. This is however a heuristic that might lead to suboptimal results by ignoring placement locations whose LWR is significantly improved by the branch length optimization process. If the RT is however well suited for the QSs, this situation is generally not expected to occur.

Placement Result

The placement process is repeated independently for every QS. That is, for each QS, the algorithm starts calculating placements from scratch on the original RT. The result thus classifies each QS in the phylogenetic context of the RT, without resolving the evolutionary relationships between the QSs.

The data is usually stored in so-called `jplace` files [159], which is a standard based on the `JSON` format [25, 40]. It stores the RT in `Newick` format, including tip names and branch lengths, and extended by a post-order numbering of the edges to be referenced by the placements. The main part of the file is a list of lists: For each QS, its list of placements is stored. A placement is described by its edge number (referencing the RT), the LWR, the pendant length, and the distal length. The proximal length is usually omitted, as it can be inferred from the branch length of the tree. Furthermore, the format can summarize multiple identical QSs by allowing several names for each list of placements, where each name can also have a weight (called its *multiplicity*). Lastly, usually not all placement locations are stored in the file, as the ones with low LWR do not contribute much to post-analysis methods anyway.

2.5.2 Use Cases and Applications

Phylogenetic placement is a flexible tool that yields useful biological information *per se*, but that also can be utilized for a variety of downstream analyses.

Comparison to Existing Methods

A typical task in metagenomic studies is to identify and classify the environmental sequences with respect to known reference sequences, either in a taxonomic or phylogenetic context, as mentioned in Section 2.2.2. Conventional methods for this task, such as BLAST [3], are based on sequence similarity. Such methods are fast, but only attain satisfying accuracy levels if the query sequences are sufficiently similar to the reference sequences. Furthermore, the best BLAST hit does often *not* represent the most closely related species [118]. This is particularly true for environments

where available reference databases do not exhibit sufficient taxon coverage [149]. Moreover, as insufficient taxon coverage cannot be detected by methods that are based on sequence similarity, they can potentially bias downstream analyses.

More recent methods can alleviate some of these issues, for instance by using machine learning techniques to obtain a taxonomic classification of metagenomic sequences [244], or by utilizing a phylogeny inferred from metagenomic sequence clusters to classify microbial communities [235]. However, the common shortcoming of these methods is that they lack a way of incorporating phylogenetic information of known sequences.

A phylogenetic placement analysis *does* incorporate known phylogenetic relationships, and hence provides a more accurate means for read identification and classification. For example, the classification of query sequences can be summarized by means of sequence abundances [99, 185], or to obtain taxonomic assignments [119].

Furthermore, phylogenetic placement also allows for more elaborate downstream analyses. Firstly, the reference tree usually offers a higher resolution than simple per-taxon abundance counts, and the amount of mapped QSs per branch can be directly visualized on the RT [149], as shown in Section 4.1. Secondly, established methods such as Edge PCA and Squash Clustering [155], which we introduce in Section 2.5.5, allow for identifying subtle differences between distinct samples, thus enabling comparative studies directly based on phylogenetic placement. Lastly, we proposed novel methods for visualizing and clustering phylogenetic placement data [43], which we describe in Chapter 4 and Chapter 5. Another typical task in metagenomic studies is to discover novel diversity within the samples, which can be conducted using phylogenetic placement, as outlined in Chapter 6.

Variants and Derived Tools

The placement algorithm presented above uses the standard ML framework for evaluating placement locations on the tree. There also exist variants of phylogenetic placement that use maximum parsimony [16] and minimum evolution [73] instead of maximum likelihood, and variants that calculate Bayesian posterior probabilities [158]. Moreover, the boosting method SEPP has been proposed to improve the accuracy of the placements [173]. The recently proposed tool RAPPAS [131] is an alignment-free approach that is not based on ML, but yields comparable results to standard phylogenetic placement implementations.

Phylogenetic placement has further been used for a variety of applications and derived pipelines, such as species delimitation as in PTP [272] and mPTP [109], genome and metagenome analysis as in PhyloSift [44], taxonomic identification and phylogenetic profiling as in TIPP [180], and identification and correction of taxonomically mislabeled sequences as in Sativa [119].

2.5.3 Placement Processing

When placing multiple environmental samples, for instance, from different geographical locations, typically, the same RT is used, in order to allow for comparisons of

the phylogenetic composition of these samples. In this context, it is important to consider how to properly normalize the samples. Normalization is required as the sample size (often also called library size), that is, the number of sequences per sample, can vary by several orders of magnitude. This is due to technical aspects in the sequencing process, such as efficiency variations, or biases introduced by the amplification process, as explained in Section 2.2.2. In consequence, metagenomic sequence data are inherently compositional [82], which needs to be considered in all steps of data analysis.

Selecting an appropriate normalization strategy hence constitutes a common problem in many metagenomic studies. The appropriateness depends on data characteristics [256], but also on the biological question asked. For example, estimating indices such as the species richness are often implemented via so-called *rarefaction* and rarefaction curves [84] by randomly re-drawing sequences from the set of sequences in a sample to obtain comparable sample sizes. This however ignores a potentially large amount of the available valid data [164]. Furthermore, the specific type of input sequence data has to be taken into account for normalization: Biases induced by the amplification process can potentially be avoided if, instead of amplicons, data based on shotgun sequencing are used, such as $_{\text{mi}}$tags [138]. Moreover, similar sequences can be clustered prior to phylogenetic placement analysis, for instance, by constructing so-called *operational taxonomic units* (OTUs) [21, 222], using programs such as UCLUST [65], VSEARCH [204], or SWARM [147, 148]. Analyses using OTUs focus on species diversity instead of simple abundances. OTU clustering substantially reduces the number of sequences, and hence greatly decreases the computational cost for placement analyses. Lastly, one may completely ignore the abundances (which correspond to the *multiplicities* of placements) of the placed sequences, reads, or OTUs, and only be interested in their presence/absence when comparing samples.

Which of the above analysis strategies is deployed, depends on the specific design of the study and the research question at hand. The common challenge is that the number of sequences per sample differs, which affects most post-analysis methods.

In the following, we therefore explain how the necessary normalizations of sample sizes can be performed. We also introduce established terminology, and describe general techniques for interpreting and working with phylogenetic placement data. These are not methods of their own, but they are tools and building blocks that are necessary for the analysis methods explained and introduced later.

Edge Masses

Methods that compare samples directly based on their sequences, such as the UniFrac distance [140, 142] (see Section 2.5.4), can benefit from rarefaction [256]. However, in the context of phylogenetic placement, rarefaction is not necessary. Thus, more valid data can be kept. To this end, it is convenient to think of the reference tree as a graph. Then, the per-branch LWRs for a single QS can be interpreted as mass points distributed over the edges of the RT, including their respective placement positions on the branches, as shown in Figure 2.9. We call this the *mass interpretation* of

the placed QSs, and henceforth use mass and LWR interchangeably. For simplicity, we here ignore that typically not all placements are stored in `jplace` files, meaning that the mass per QS can also be < 1. Hence, each QS is assumed to have a total accumulated mass of 1.0 on the RT. The *mass of an edge* or *edge mass* refers to the sum of the LWRs on that edge for all QSs of a sample, as shown in Figure 2.10(a). The *total mass* of a sample is then the sum over all edge masses, which is identical to the number of QSs in the sample,

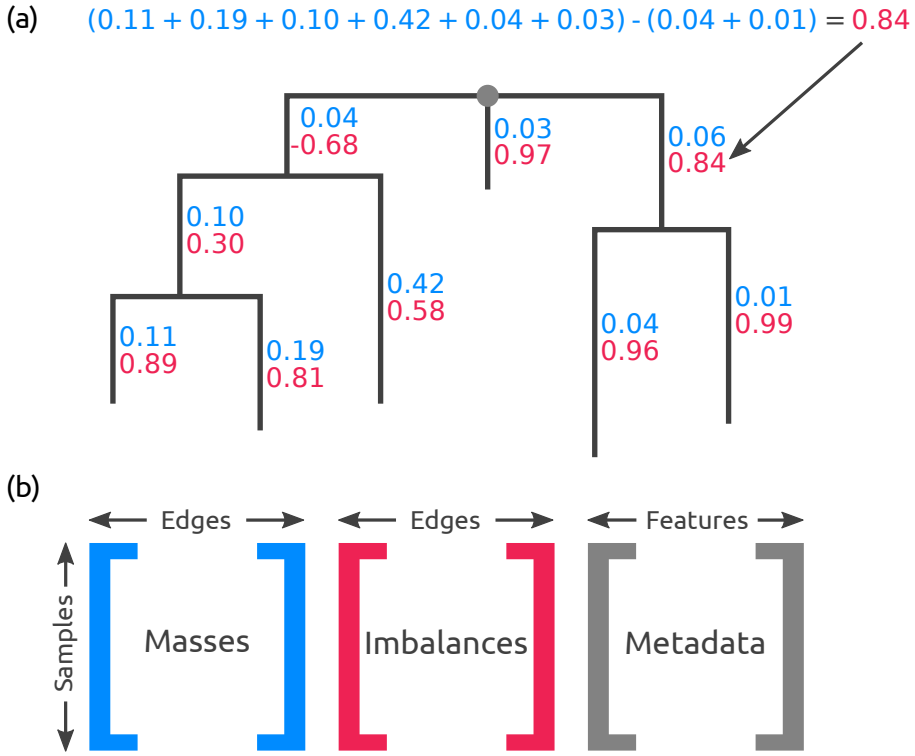

Figure 2.10: Edge Masses and Imbalances. (a) Reference tree where each edge is annotated with the normalized mass (first value, blue) and imbalance (second value, red) of the placements in a sample. The imbalance is the sum of masses on the root side of the edge minus the sum of the masses on the non-root side. The depicted tree is unrooted, hence, its top-level trifurcation (gray dot) is used as "root" node. An exemplary calculation of the imbalance is given at the top. Because terminal edges only have a root side, their imbalance is not informative. (b) The masses and imbalances for the edges of a sample constitute the rows of the first two matrices. The third matrix contains the available meta-data features for each sample. These matrices are used to calculate, for instance, the edge principal components or correlation coefficients.

The key idea is to use the distribution of placement mass points over the edges of the RT to characterize a sample. This allows for normalizing samples of different size by scaling the total sample mass to unit mass 1.0. In other words, absolute abundances—which are inappropriate for analyses of metagenomic sequences due to the compositional nature of the data [82]—are converted into relative abundances. This way, rare species, which might have been removed by rarefaction, can be kept,

as they only contribute a negligible mass to the branches into which they have been placed. This approach is analogous to using proportional values for methods based on OTU count tables. For each OTU, such tables store a count of how often it appeared in each sample, which can be made proportional by scaling each sample/column of the table by its sum of OTU counts [256]. Most of the methods presented here use normalized samples, that is, they use relative abundances. As relative abundances are compositional data, certain caveats occur [2, 81, 139], which we discuss where appropriate.

When working with large numbers of QSs, the mass interpretation allows to further simplify and reduce the data: The masses on each edge of the tree can be quantized into b discrete bins, that is, each edge is divided into b intervals (or bins) of the corresponding branch length. All mass points on that edge are then accumulated into their respective nearest bin. For example, by accumulating mass points at their nearest interval midpoint, masses are only minimally moved. The parameter b controls the resolution and accuracy of this approximation. In the extreme case of $b := 1$, all masses on an edge are grouped into one single bin. This *branch binning* process drastically reduces the number of mass points that need to be stored and analyzed in several methods we present, while only inducing a negligible decrease in accuracy. As shown in Table 5.1. branch binning can yield a speedup of up to 75% for post-analysis run-times.

Furthermore, using masses allows to summarize a set of samples by annotating the RT with their (weighted) average per-edge mass distribution. This procedure, also called *squashing* [155], sums over all sample masses per edge, and then normalizes them once more to obtain unit mass for this resulting average tree. This normalized tree thereby summarizes the (sub-)set of samples it represents.

Edge Imbalances

So far, we have only considered the per-edge masses. Often, however, it is also of interest to "summarize" the mass of an entire clade, that is, to consider per-clade masses. For example, sequences of the RT that represent species or strains might not provide sufficient phylogenetic signal for properly resolving the phylogenetic placement of short sequences [60]. In these cases, the placement mass of a sequence can be spread across different edges representing the same genus or species, thus blurring analyses based on per-edge masses.

Instead, a clade-based summary can yield clearer analysis results. It can be computed by using the tree structure to appropriately transform the edge masses. Each edge splits the tree into two parts (bipartitions, see Section 2.3.2), of which only one contains the root (or top-level trifurcation) of the tree. For a given edge, its mass difference is then calculated by summing all masses in the root part of the tree and subtracting all masses in the other part, while ignoring the mass of the edge itself [155]. This difference is called the *imbalance* of the edge. It is usually normalized to represent unit total mass, as the absolute (not normalized) imbalance otherwise propagates the effects of differing sample sizes all across the tree. It is irrelevant

where the root of the tree is, as any re-rooting changes the sign of edge imbalance values consistently across different samples.

An example of the imbalance calculation is shown in Figure 2.10(a). The edge imbalance relates the masses on the two sides of an edge to each other. This implicitly captures the RT topology and reveals information about its clades. Furthermore, this transformation can reveal differences in the placement mass distribution of nearby branches of the tree. This is in contrast to the KR distance (see Section 2.5.4 below), which yields low values for masses that are close to each other on the tree. Note that for normalized samples with unit total mass, the imbalance of a leaf edge is simply the total mass of the tree minus the mass of the edge. It thus contains mostly irrelevant information and can often be left out.

Placement Data Matrices

The edge masses and edge imbalances per sample can be summarized by two matrices, which we use for all further downstream edge- and clade-related analyses, respectively. In these matrices, each row corresponds to a sample, and each column to an edge of the RT. Note that these matrices can either store absolute or relative abundances, depending on whether the placement mass was normalized.

Furthermore, many studies provide meta-data for their samples, for instance, the pH value or temperature of the samples' environment. Such meta-data features can also be summarized in a per-sample matrix, where each column corresponds to one feature. The three matrices are shown in Figure 2.10(b). Quantitative meta-data features are the most suitable for computational purposes, as they can be used for calculations such as detecting correlations with the placement mass distributions of samples, see for instance Section 4.2.2.

2.5.4 Distances between Samples

Given a set of metagenomic samples, one key question is how much they differ from each other. Pairwise distances are valuable for downstream tasks, for instance, clustering algorithms such as UPGMA [125, 169, 221] and ordination methods such as *principal component analysis* (PCA) [105, 188], or to discover gradients in microbial communities with respect to meta-data features.

General Metagenomic Distance Measures

In many comparative ecology studies, commonly used methods for assessing sample (dis-)similarity are based on sequence abundances, or OTU count tables (as introduced in Section 2.5.3). For example, the Jaccard index [103] and the related Sørensen-Dice coefficient [55, 223] use the presence/absence of species in two samples to measure their similarity and diversity. The Bray-Curtis dissimilarity [24, 125] furthermore uses abundances, that is, counts of species, in order to measure compositional dissimilarity between samples. Note that working with such similarity indices entails certain pitfalls [22].

These indices do however not take the evolutionary history of the sequences into account. One method that uses the relatedness of the species under study is the UniFrac distance [140, 142]. To compare two metagenomic samples, a phylogenetic tree is employed, either by inference from all sequences of the two samples, or by assigning the sequences to the tips of an existing tree. Then, the branches of the tree are marked as either shared or unique, depending on whether they lead to taxa that appear in both or only one of the samples. The distance is computed as the fraction of total branch lengths that is unique, which satisfies the requirements of a distance metric. The UniFrac distance can be calculated quantitatively (weighted UniFrac), or qualitatively (unweighted UniFrac), depending on whether sequence abundances are considered, or only their presence and absence is used.

The Phylogenetic Kantorovich-Rubinstein Distance

The idea of using phylogenetic distances on a tree to assess sample similarity has been extended and generalized to the context of phylogenetic placement in form of the *phylogenetic Kantorovich-Rubinstein* (KR) distance [68, 155]. In other contexts, the KR distance is also called Wasserstein distance, Mallows distance, or Earth Mover's distance [129, 150, 197, 245]. The KR distance between two metagenomic samples is a metric that describes by at least how much the normalized mass distribution of one sample has to be moved across the RT 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. The distance is symmetrical, and increases the more mass needs to be moved, and the larger the respective displacement (moving distance) is.

The linear case of moving the mass of one distribution to transform it into another distribution is shown in Figure 2.11. This linear case corresponds to the path between two locations on a tree, and is thus a measure of evolutionary distance between these locations. It can be extended to a tree via post-order traversal, by starting at the tips and moving the mass differences towards the (arbitrary) root. In order to enable the transformation, the two samples being compared need to have equal masses. Hence, the KR distance operates on normalized samples; that is, it compares relative abundances.

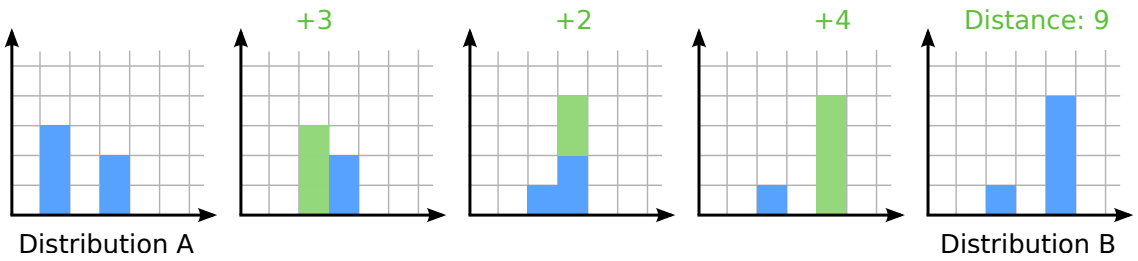

Figure 2.11: Linear KR Distance. Distribution A is transformed into distribution B by moving mass along the axis, while keeping track of the moved distances. For simplicity, masses are discretized here; the continuous case works accordingly.

As the tree needs to be traversed once per pairwise distance calculation, the computation of the phylogenetic KR distance is linear in the tree size, and in the number of placements. It is hence suitable for the large datasets of typical metagenomic studies. Note however that the computation of a pairwise distance matrix between the samples is quadratic in the number of samples. In the special case of assigning mass only to the tips of the tree (for example, by "placing" the QSs via similarity based methods such as BLAST), the KR distance is equivalent to the weighted UniFrac distance [68].

The mathematics of the phylogenetic KR distance have been thoroughly examined by Evans and Matsen [68]. In summary, the KR distance can be formulated as an integral over distances λ along the branches of the tree T. Let $\tau(x)$ denote the subtree below point x on T for an arbitrary rooting, and let P and Q the probability distributions of the two samples on the branches of T. Then, the KR distance can be expressed in closed form as

$$\mathrm{KR}(P, Q) = \int_T \Big| P(\tau(x)) - Q(\tau(x)) \Big| \; \lambda(dx) \tag{2.12}$$

This notation treats the placements as a continuous distribution over the branches of the tree instead of a collection of point masses, and hence describes a more general form of the KR distance. The distance can further be generalized by introducing an additional parameter p, where $0 < p < \infty$ controls the impact of mass relative to transport [198, 199]:

$$\mathrm{KR}_p(P, Q) = \left[\int_T \Big| P(\tau(x)) - Q(\tau(x)) \Big|^p \; \lambda(dx) \right]^{\min(1/p, 1)} \tag{2.13}$$

Large $p > 1$ emphasize the impact of mass differences, while small $p < 1$ increase the influence of the distance traveled. In typical applications however, the default of $p = 1$ is used, which is equal to Equation 2.12 and corresponds to the physical interpretation of mass movements.

TODO: mention NH distance here if we use it later!

2.5.5 Existing Analysis Methods

The pairwise KR distance matrices between metagenomic samples as introduced in Section 2.5.4 above can be used for general-purpose methods, such as PCA [105, 188] and UPGMA [125, 169, 221]. Although appropriate to apply, such methods do not use the fact that the distances were calculated on a phylogenetic tree. Taking this information into account however allows for greater interpretability and visualizability. To this end, the ordination method *Edge PCA*, as well as the clustering method *Squash Clustering* have been developed [155]. As we later compare our novel methods to these existing ones, we introduce them here.

Edge PCA

The *edge principal components analysis* (Edge PCA) [155] is a method that utilizes the imbalance matrix to detect and visualize edges with a high heterogeneity of mass difference between samples. In particular, it computes the principal components of the imbalance matrix, using standard PCA. The result can be interpreted as a weighted sum of variables that maximizes variance between samples.

Similar to standard PCA on other types of input matrices, such as count tables or pairwise distances, the principal components can then be visualized in form of a scatter plot of samples. Samples are separated from each depending on their placement mass distribution, where each principal axis in the plot explains additional variance between the samples. These plots can further be annotated with meta-data features, for instance, by coloring, thus establishing a connection between differences in samples and differences in their meta-data [225]. Examples of this are shown later in Figure 3.8 and Figure 5.3.

In contrast to standard PCA, using the imbalance matrix allows for further visualizations. As the columns of the imbalance matrix correspond to edges of the tree (see Section 2.5.3), the resulting eigenvectors (principal components) can be projected back onto to tree. Hence, while the scatter plots show *how* samples separate from each other, these visualizations on the tree explains *why* they separate. Each principal component results in a tree visualization, where edges are highlighted that are responsible for the observed differences in the corresponding principal axis of the plot. An example of this is shown later in Figure 4.3.

Squash Clustering

A fundamental task for a set of metagenomic samples consists in finding clusters of samples that are similar to each other according to some distance measure, such as the KR distance. Standard linkage-based clustering methods like UPGMA are solely based on the distances between samples, that is, they calculate the distances of clusters as a function of pairwise sample distances.

In contrast, *Squash Clustering* [155] is a method that also takes into account the intrinsic structure of phylogenetic placement data. It uses the KR distance (see Section 2.5.4) to perform agglomerative hierarchical clustering of samples. Instead of using pairwise sample distances, however, it merges (*squashes*, see Section 2.5.3) clusters of samples by calculating their weighted average per-edge placement mass.

Thus, in each step, Squash Clustering operates on the same type of data, namely, mass distributions on the RT. This results in a hierarchical clustering tree of samples that has meaningful, and hence interpretable, branch lengths: The distances in the cluster tree correspond to the KR distance between merged samples. Examples of such cluster trees are shown later in Figure 3.6 and Figure 5.1. Furthermore, as the inner nodes of the cluster tree are again mass distributions, they can be visualized, and thus allow to interpret the features of each set of merged samples.

3. Ancillary Methods for Phylogenetic Placement

This chapter is based on the peer-reviewed publication:

- **Lucas Czech**, Pierre Barbera, and Alexandros Stamatakis. "Methods for Automatic Reference Trees and Multilevel Phylogenetic Placement." *Bioinformatics*, 2018.

Contributions: Lucas Czech... Pierre Barbera... Alexandros Stamatakis... and...

3.1 Background and Motivation

Molecular sequencing costs are decreasing exponentially, leading to unprecedented amounts of genetic sequence data, as explained in Section 2.2.2. In most metagenomic studies, an initial analysis step consists in assessing the evolutionary provenance of the sequences. Phylogenetic placement, as introduced in Section 2.5, can be employed to determine the evolutionary position of sequences with respect to a given reference phylogeny.

This is particularly helpful for studying new, unexplored environments, for which no closely related sequences exist in reference databases yet [149]. However, the selection process of suitable reference sequences for inferring a reference tree is typically conducted manually. This constitutes a major challenge and hindrance for studying such environments with placement methods.

This limitation also concerns the use of phylogenetic placement for taxonomic assignment. In studies that specifically look for certain kinds of organisms, e.g, protists

[149], it usually suffices to use a taxonomy covering the organisms of interest, potentially including outgroups from more distantly related species. As metagenomic analyses get cheaper, it is however to be expected that researchers want to target more than one group of organisms within one study. Particularly in cases where the environment contains a yet unknown diversity of organisms, this hence necessitates to use a broad reference that covers many taxonomic clades. At the same time however, the number of taxa in the reference phylogeny should be small enough to allow for visually inspecting and interpreting the results.

Lastly, phylogenetic placement methods have generally already reached their scalability limits: They require a higher computational effort with respect to the placement algorithms *per se*, but also the pre- and post-processing, than, for instance, similarity-based methods such as BLAST.

3.2 Methods and Implementation

Here, we introduce methods to overcome the aforementioned limitations, that is, to (1) automatically obtain a high-quality reference tree for conducting phylogenetic placement, (2) split up the placement process into two steps using smaller phylogenies, and (3) accelerate the computation of placements via appropriate data pre-processing approaches. All methods are implemented as part of our GAPPA tool; see Appendix C for implementation details.

3.2.1 Phylogenetic Automatic (Reference) Trees

Motivation

Molecular environmental sequencing studies, particularly those that aim to conduct phylogenetic placement of query sequences (QSs), often rely on a set of manually selected and aligned reference sequences to infer a reference tree (RT) [51, 149, 237, 240]. Creating and maintaining databases of such reference sequences constitutes a labor-intensive and potentially error-prone process. Moreover, this approach is impractical for highly diverse samples that comprise sequences from many taxonomic clades, or samples obtained from unexplored environments, where it is yet unknown which reference sequences are necessary. Lastly, even if a large RT is available, the visualization of placements on such an RT might be confusing and thus hard to interpret.

The RT used for phylogenetic placement should ideally (a) cover all major taxonomic groups that occur in the QSs, (b) use high-quality error-free reference sequences, and (c) not be too large to allow for unambiguous visualization and interpretation. These criteria can be met for small datasets by manually selecting curated sequences from databases, potentially informed by literature describing these sequences. In order to increase coverage, often additional sequences are selected based on their similarity to the already selected ones. For large and taxonomically diverse samples one key challenge is that sequence databases such as GREENGENES [54], UNITE [1], PR2 [88], EZTAXON [113], SILVA [196], and RDP [35] maintain reference collections of

thousands to millions of taxonomically annotated sequences. Therefore, one needs to appropriately sub-sample sequences such that the RT can be inferred in reasonable time *and* sufficiently covers the diversity of the sample.

Previous approaches mainly relied on phylogenetic diversity [70, 172, 186] and related methods [160]. The major drawback is that they require a comprehensive phylogeny as input. Inferring such large comprehensive phylogenies with hundreds of thousands of taxa, to subsequently reduce the taxon set again, is computationally inefficient and in certain cases infeasible.

To this end, we present a computationally efficient approach for obtaining sequences from large databases to infer an RT. This RT is then used for conducting phylogenetic placement analyses. The input of our method is a database of aligned sequences of known species, including their taxonomic labels. Our approach then identifies sets of sequences that are similar to each other based on their entropy. It subsequently reduces the sequences in these sets to a predefined number of consensus sequences. This set of sequences is the output of our method. It represents the taxonomic clades and is then used to infer the RT.

Sequence Entropy

Conventional methods for sequence similarity are often based on edit distance and other pairwise comparison methods [3, 177, 220]. This however necessitates to transform the pairwise distances to some form of ensemble measure that describes the similarity of all sequences to each other, for which there is no obvious approach [274]. There also exist methods that describe genetic variation and nucleotide diversity of sets of sequences [20, 178] which could be used for this purpose.

We here use entropy [216] to define a measure for quantifying the ensemble similarity of a set s of sequences. Variants of sequence entropy have been used before in numerous biological and phylogenetic contexts, for example, to asses the information content of sequences [36, 39, 130, 213, 246–248], or to measure substitution saturation [261]. Here, we use entropy for alignment sites, that is, we define the entropy (uncertainty) H at alignment site i as

$$H_i \;=\; -\sum_c f_{c,i} \times \log f_{c,i} \tag{3.1}$$

where $c \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}, \mathtt{-}\}$ is the set of nucleotide states including gaps, and $f_{c,i}$ is the frequency of character c at site i of the alignment. Including gaps (`-`) in the summation reduces the contribution of sites that contain a large fraction of gaps. Their contribution is weighed down as all standard phylogenetic inference tools model gaps as undetermined states, that is, they do not contribute anything to the likelihood score. The entropy is 0 for sites that only contain a single character. It increases the more different characters an alignment site contains, *and* the more similar their frequencies are. Its maximum occurs if all characters appear with the same frequency (each of them 20%). Note that we also treat ambiguous characters

as gaps (see Section 2.2.4). As only 0.008% of the non-gap characters in our test database (SILVA) are ambiguous, their influence is negligible. Ambiguous characters could however be incorporated by using fractional character counts.

Finally, the total entropy of a set s of aligned sequences is simply the sum over all per-site entropies: $H(s) = \sum_i H_i$. It is also possible to normalize this value by dividing it by the length of the alignment to get comparable values across alignments. Here, this however does not make a difference, as we are always comparing sequences with the same alignment length. We use this entropy to quantify the ensemble similarity of a set of sequences. This can be regarded as an information content estimate of the sequences.

Sequence Grouping

The goal of this step is to group the sequences of a database into a given target number of groups/sets, such that the groups reflect the diversity of the sequences in the database. At the same time, the number of sequences needs to be small enough such that a maximum likelihood RT can be inferred in reasonable time.

A possible approach would be to use agglomerative clustering: In each step, the sequences that have the lowest entropy are clustered until the desired number of reference sequences is reached. A supposed advantage of this approach is that it does not rely on any taxonomic information (in contrast to our approach presented below). This procedure is however computationally expensive, having a complexity in $\mathcal{O}(n^2 \log n)$, and is thus not applicable to large databases with millions of sequences. Furthermore, the resulting sequence clusters can generally not be assigned unambiguous taxonomic labels, that is, they lack an interpretable naming scheme. This severely limits the types of useful post-analyses that can be executed; we did therefore not explore this approach.

Instead, we use the taxonomic information (see Section 2.3.1) of the sequence database to identify potential candidate groups of sequences that could be represented by a consensus sequence (see Section 2.2.4). We interpret a taxonomy as a sequence labeling, where similar sequences have related labels. Thus, a taxonomy represents a pre-classification of similar sequences that can be exploited to group them.

For a clade t of the taxonomic tree, we denote by $H(t)$ the entropy of all sequences that belong to that clade, including all sequences in its sub-clades, that is, its lower taxonomic ranks. Clades with low entropy imply that they contain highly similar sequences that can in turn be represented by a consensus sequence without sacrificing too much diversity. Inversely, clades with high entropy contain diverse sequences, implying that a consensus sequence is not likely to sufficiently capture the inherent sequence diversity. It is thus better to expand these clades and construct separate consensus sequences for their respective sub-clades. An example is shown in Figure 3.1. As the clade structure of a taxonomy forms a tree, this criterion can then be applied recursively, as shown in Algorithm 3.1.

TODO: the formatting is currently off here...

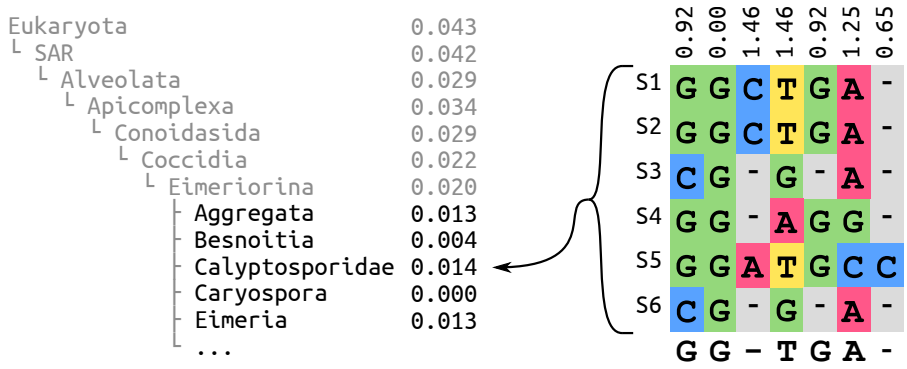

Figure 3.1: Entropy and consensus sequence of a taxonomic clade. The left hand side shows the exemplary clade *Eimeriorina* in its taxonomic context, listing its super- and sub-clades with the normalized entropy of their respective sequences. The right hand side is an excerpt from the alignment of six sequences that belong to the *Calyptosporidae* sub-clade. At its top, the per-site entropies for the alignment columns are shown. At the bottom, the majority rule consensus sequence is shown, which is used to represent the sub-clade.

Algorithm 3.1 Taxonomy Expansion

1: *Candidates* ← list of highest ranking clades
2: *TaxaCount* ← size of *Candidates*
3: **while** *TaxaCount* < *TargetCount* **do**
4: *MostDiverse* ← $\arg\max_{t \in Candidates} H(t)$
5: remove *MostDiverse* from *Candidates*
6: add sub-clades of *MostDiverse* to *Candidates*
7: *TaxaCount* ← *TaxaCount* − 1 + size of *MostDiverse*
8: **return** *Candidates*

The algorithm works as follows: We initialize a list of candidate clades with the highest ranking clades that we want to consider. In the most general case, these can be "Archaea", "Bacteria", and "Eukaryota". We then select the most diverse candidate clade, that is, the clade t whose sequences exhibit the highest entropy $H(t)$. This clade is then expanded, and we do not consider it as a potential candidate for building a consensus sequence. The high entropy clade is then removed from our list and its immediate sub-clades are added as new candidates to the list. Finally, the current count of how many candidates we have already selected is updated accordingly. By expanding clades with high entropy, we descend into the lower ranks of the taxonomy. On average, this decreases the entropy, because low ranking clades generally tend to contain more similar sequences. This process is repeated until our list contains approximately as many candidate clades as the desired target count of reference sequences, which is provided as input. As the sizes of expanded clades can vary substantially, the target count cannot always be met exactly. In our tests, the average deviation was 0.2%, as shown later in Table 3.1.

Given this list of clades from different taxonomic ranks, we can now compute the consensus sequences. For each clade, all sequences in that clade and its sub-clades are used to construct a consensus sequence, which represents the clade diversity, and serves as the reference sequence for that clade. This has several advantages: If only a few sequences diverge from the majority of that clade, the entropy might underestimate the molecular diversity of a clade. The consensus sequence for such a clade however compensates for this. Using consensus sequences furthermore levels out spurious and erroneous sequences in the database. A simple per-site majority rule consensus [49, 161] works well, but we also assessed alternative methods; see Figure 3.4 and Figure 3.5 for details.

The algorithm can start at any rank of the taxonomy in order to only group sequences from specific clades. It is computationally cheap compared to pairwise sequence comparison, while still yielding reasonable representative sequences for large taxonomic clades. Note that it would also be possible to directly use the relative character frequencies at each site to obtain more accurate representations. Maximum likelihood-based phylogenetic inference tools do, in principle, not require discrete input sequences, as explained in Section 2.4.4. The likelihood model allows to account for uncertainty in the input data [72], although this is generally not implemented in the mainstream software packages. The above process yields a set of consensus reference sequences which capture the diversity of distinct taxonomic clades.

Inferring a Reference Tree

Once we have identified the consensus sequences, which are already aligned to each other, we can use them to infer a maximum likelihood tree, which we call a *Phylogenetic Automatic (Reference) Tree* (PhAT). As each consensus sequence is associated with a taxonomic clade, the corresponding taxonomic path can be used to label the tips of the tree. Note that since clades with low entropy might not be expanded, the tip labels do not necessarily correspond to species or genus level. Also, the

PhAT will not necessarily be congruent to the taxonomy, unless the tree search is specifically constrained in that way (see Section 2.4.3).

A PhAT satisfies all criteria we listed above: (a) All taxonomic groups occurring in the QSs can be covered by using a suitable taxonomy as input. (b) By using consensus sequences, potential sequencing errors can be alleviated. (c) The size of the tree can be specified by the user. However, the resolution of the trees is limited by the underlying taxonomy, see Section 3.3.3 and Section 3.3.4 for details. Thus, one needs to verify that the resulting tree is appropriate for the dataset to be placed on it. This however also holds for manually selected reference sequences, and is hence not a specific disadvantage of our method. Furthermore, using consensus sequences may obscure the degree of sequence diversity in sub-clades, which in turn can affect the accuracy of subsequent phylogenetic placements on that tree. Our algorithm as described here can not fully compensate for this. We present a method to address both issues (tree resolution and obscured diversity) in the next Section.

3.2.2 Multilevel Placement

When conducting phylogenetic placement, the computationally limiting factors are (i) the number of QSs to be placed (addressed in the following Section 3.2.3) and (ii) the size of the RT (number of taxa) and corresponding alignment length (addressed here). Using RTs with more taxa increases the phylogenetic resolution of the placements, at the cost of increased computational effort for inferring the RT, aligning the QSs, and placing the QSs. Furthermore, longer reference alignments (if appropriate data is available) are required to accurately infer large trees under the maximum likelihood criterion [262], thus further increasing the computational costs. Lastly, placement on large trees that comprise reference sequences with high evolutionary distances can reduce placement accuracy [173]. Thus, using a large number of reference sequences is not always desirable in practice.

One solution is to divide the tree and its alignment into more conquerable subsets, for example as implemented in SATÉ [134, 135]. This approach has also been extended to phylogenetic placement in SEPP [173] and TIPP [180], which divide the tree into disjoint subsets of taxa and conduct placement on each of them. While yielding more accurate placements and taxonomic classifications in less computing time, this method might still result in large reference trees, which are hard to inspect and visualize.

To address this issue, we present an approach called *Multilevel* or *Russian Doll* Placement, which is summarized in Figure 3.2. Instead of working with one large RT comprising *all* taxa of interest, we use a smaller, but taxonomically broad backbone tree (BT) for pre-classifying the QSs (first level), and a set of refined clade trees (CTs) for the final, more accurate placements (second level). These CTs comprise the reference sequences that are of interest for a particular study. For example, if a study is concerned with *Apicomplexa* and *Cercozoa*, a broad *Eukaryotes* BT can be used for the first level, and two respective CTs for the second level, in analogy to [149]. Each CT is associated with the set of branches of a specific BT clade.

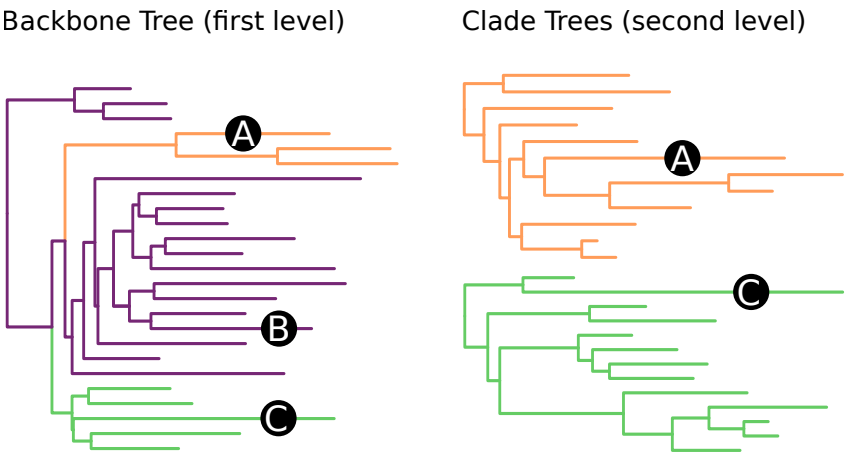

Figure 3.2: Multilevel Placement. The left shows a backbone tree (BT); the right shows two clade trees (CTs) in orange and green. Branches in the BT that are associated with a CT are marked in its color. The trees "overlap" each other, meaning that each CT is represented by multiple branches in the BT. Three sequences A, B, and C are placed on the BT, which is the first level. A and C are placed on branches associated with a CT. Hence, their second level placement is conducted on the respective CT. B is placed on a branch that is not associated with any CT, and thus not used in the second level.

The method then works in three steps:

1. Align and place the QSs using the BT (first level).
2. For each CT, collect the QSs that are placed on the BT branches associated with the CT.
3. Align and place these QSs again, using their specific CTs (second level).

While this approach requires some additional bookkeeping, the total computational cost is reduced, because the QSs do not have to be placed on all branches of all CTs. The speed gain depends on the sizes of the BT *and* the CTs with respect to the size of the substantially larger (often one order of magnitude or more) comprehensive tree. For example, by splitting a tree with 10 000 taxa into a BT and 10 CTs with 1000 taxa each, the computational cost decreases by a factor of 5 (two placement levels with 10% of the cost each). Furthermore, at each level, the amount of required computer memory is reduced by a factor of 10 compared to the large tree. Lastly, this method allows for fine-grained control over the clades of interest at both placement levels:

Firstly, the BT provides a means for phylogenetically informed sequence filtering—that is, to identify and remove "spurious" QSs. Sequences with low similarity to known references are often removed in environmental sequencing studies [230]. However, using sequence similarity as a filter criterion can remove too many QSs, particularly when studying new, unexplored environments [149]. By using phylogenetic

placement as a filter instead, substantially more sequences can be retained for downstream analyses. Only the QSs that are placed onto the inner branches of the BT, that is, branches with no associated CT, are omitted at the second placement level. Such placements may indicate that suitable reference sequences are missing from the RT, or that the respective QSs represent novel species. Either way, as these QSs are not well represented by the RT, they are not informative for most downstream analyses and can thus be removed. This thus represents a phylogenetically informed sequence filtering method as an alternative to sequence similarity.

Secondly, using specific clade trees for lower level taxonomic clades offers the phylogenetic resolution that is necessary for downstream analyses and for biological reasoning. It is, for example, possible to use manually curated "expert" trees for each clade of interest.

In this setup, the BT is only used for pre-classification, and can, for example, use our PhAT method as presented in Section 3.2.1. The aforementioned issue of obscured diversity in sub-clades can be circumvented by "overlapping" the CTs with the BT. That is, a CT can be associated with several branches of the BT, so that placements on each of these BT branches are collected and placed onto the same CT. See Figure 3.2 and Figure 3.10 for examples. We recommend to ensure that the branches of the BT that are associated with one CT are monophyletic, meaning that there is one split that separates these branches from the rest of the BT. This can be achieved by inferring the BT with a high-level constraint that maintains the monophyly of the CTs. It ensures phylogenetic consistency between the BT and the CTs, and improves the accuracy of the first placement level, as shown in Section 3.3.5. Lastly, it is also possible to use more than two levels, which might become necessary when working with RTs and datasets even larger than what is currently available.

3.2.3 Data Preprocessing for Phylogenetic Placement

Apart from the RT size, handling the sheer number of QSs also induces computational limitations for conducting phylogenetic placements. Most metagenomic studies publish their data in unprocessed formats, which are sometimes filtered to contain only reads from certain barcoding or marker regions. For instance, they store the raw sequencing output in `fasta` [189] or `fastq` [34] format (see Section 2.2.1). Those data often contain duplicates of exactly identical sequences, both *within* and *across* samples. Identical sequences are however treated the same in phylogenetic placement algorithms and therefore induce unnecessary computational overhead. Furthermore, sample sizes, that is, the number of sequences per sample, can vary by several orders of magnitude. For example, the "HM16STR" dataset of the HMP [101, 166] contains an average of 12 911 sequences per sample, but also an outlier sample with 0 sequences and one with 403 211 sequences. If the placement algorithm is parallelized over samples, this leads to an uneven load balance across compute nodes. A potential solution is to initially cluster the sequences into OTUs (see Section 2.5.3), which however negates the accuracy benefit of using individually placed sequences.

In order to solve these issues, that is, reduce computational cost and achieve good load balancing, one can pre-process the sequences as follows (see Appendix C for

implementation details). First, sequences are de-duplicated across all samples and fused into chunks of equal size. The chunk size should be chosen to allow aligning and placing a chunk within wall time on the intended hardware; we recommend chunk sizes of 50 000 or larger. Our tool assigns an identifier to each unique sequence, and computes a list of abundance counts for each sequence in a sample. This way, each strictly identical sequence is only processed once in the next steps.

Given an RT and its underlying alignment, the QS chunks are then aligned to the reference multiple sequence alignment, and subsequently phylogenetically placed on the RT, as explained in Section 2.5.1. The resulting per-chunk placement result files in combination with the per-sample abundance counts can then be used to generate final per-sample placement files, containing a placement for each sequence in the original sample.

The speedup induced by this preprocessing is proportional to the ratio of total versus unique sequences; the gain in parallel efficiency depends on the ratio of smallest to larges sample (in number of sequences). This approach allows to analyze datasets that are orders of magnitude larger than in previous published studies. For example, in 2012, an analysis of Bacterial Vaginosis (BV) data placed a total of 426 612 sequences, thereof 15 060 unique, on an RT with 796 tips [225]. Using a prototype of our implementation, we were able to analyze a neotropical soils dataset with 50 118 536 total sequences, thereof 10 567 804 unique, with an RT comprising 512 taxa [149]. To demonstrate the scalability of our method, we analyzed datasets with up to 116 520 289 total sequences, thereof 63 221 538 unique, from the HMP [101, 166], using RTs with up to 2059 tips. This corresponds to a computational effort that is four orders of magnitude greater than for the BV study. See Appendix B for an overview of the datasets used in our evaluation.

3.3 Evaluation and Results

3.3.1 Reference Tree Setup

To test the Phylogenetic Automatic (Reference) Tree (PhAT) method, we used the "SSU Ref NR 99" sequences of the Silva database [196] version 123.1 and the corresponding taxonomic framework [269]. The database contains 598 470 aligned sequences from all three domains of life, classified into 11 860 distinct taxonomic labels, and mainly contains bacterial sequences. In detail, there are

- 22 913 sequences with 347 taxonomic labels for the *Archaea*,
- 62 436 sequences with 7441 taxonomic labels for the *Eukaryota*, and
- 513 121 sequences with 4072 taxonomic labels for the *Bacteria*.

The overall number of taxonomic labels is counted here, that is, it includes higher level labels. We use the Silva alignment as-is, thus assuming that it is of sufficient quality for our purposes; see Section 3.3.2 for an evaluation of this.

Sequence Selection

We constructed four sets of consensus sequences from the SILVA database: a *General* set ("all of life"), as well as separate sets for the domains *Archaea*, *Bacteria*, and *Eukaryota*. The target sizes of the recursive expansion of taxonomic clades (see Section 3.2.1) where chosen to be large enough to cover the diversity well, while still being computationally feasible and visually interpretable for the subsequent steps. The target size for the *General* tree was 2000 taxa, while the *Bacteria* and *Eukaryota* tree were targeting 1800 domain-specific taxa, which is approximately reached, but not exactly (see Table 3.1). This is because the sizes of sub-clades in the taxonomy vary. Because each tip of the tree is a consensus sequence that represents the respective lowest taxonomic level, the number of available taxa is smaller than the total number of taxonomic labels in the SILVA database. For example, the *Archaea* have a total of 347 taxonomic labels across all ranks, but only 248 labels at *Genus* level. Thus, the *Archaea* tree used here comprises 248 taxa, which represents the *Archaea* taxonomy fully resolved at the *Genus* level. In the three domain-specific trees, we furthermore included consensus sequences at the *Phylum* level of the respective two remaining domains, in order to make sure that the evaluation also works well if such "outgroups" are included. The assembly of these four data sets required in total about 30 min and 10 GB of memory on a standard laptop computer. This includes counting alignment characters, calculating entropies and constructing consensus sequences. The resulting data set and tree sizes, as well as the fraction of sequences from each domain the PhATs contain are shown in Table 3.1.

Table 3.1: Taxonomic composition of the four PhATs. The table lists the four trees used in our evaluation and their sizes (in number of sequences/tips), as well as how many of these tips originate from each of the three domains of life. The underlined values represent the resulting tree sizes, which slightly deviate from the intended target sizes (2000 and 1800 taxa, respectively).

		Thereof number of		
Tree	Size	*Archaea*	*Bacteria*	*Eukaryota*
General	<u>1998</u>	210	508	1280
Archaea	511	248	205	58
Bacteria	1914	59	<u>1797</u>	58
Eukaryota	2059	59	205	<u>1795</u>

TODO: ref to implementation / gappa here? Our implementation of the method contains some further details that are worth mentioning for reproducibility: It is possible to constrain the maximal size of clades in order to not build a consensus sequence for an overly large clade, which might not be a good representative of that clade. For the same reason, it is possible to first expand the highest ranks of the taxonomy into separate candidates. We used conservative values for these two

constraints (a maximal clade size of 2000 and an expansion of only the first two taxonomic ranks), in order to give more weight to the sequence entropy. Lastly, some clades contain only one sub-clade. Those were immediately expanded, as they do not change the length of the candidate list during the algorithm.

Tree Inference

Givens the four sets of consensus sequences, we then inferred unconstrained and constrained maximum likelihood trees, running 50 independent tree searches for each tree and selecting the best-scoring tree. Unconstrained trees were inferred using RAxML 8.2.8 [228]. Constrained trees were inferred with Sativa 0.9-55 [119], which internally again relies on RAxML, and offers a convenient way to transform a taxonomy into a constraint tree. The unconstrained trees adhere to the phylogenetic signal of the sequences and thus usually work better for conducting phylogenetic placement. The constrained trees comply with the Silva taxonomy, which might be necessary in comparative studies. They are used here to assess how taxonomic constraints affect the phylogenetic placement and the subsequent analyses. In total, our setup hence yields eight distinct RTs for evaluation: the *General* tree, the three domain trees, and the respective taxonomically constrained variants. Figure 3.10 shows the unconstrained *Bacteria* tree as an example.

The relative Robinson-Foulds distances [202] (see Section 2.3.2) between the four pairs of trees (unconstrained versus constrained) are between 45.8% and 49.7%. The differences probably occur because our trees span diverse clades, whose ancient branches are hard to resolve. Also, single gene data might not be sufficient to resolve these clades. The differences between the trees however mostly concern inner branches. When conducting phylogenetic placement, QSs generally tend to be placed more towards the terminal branches of the tree. As these branches are more stable across our trees, the differences in the inner branches thus are acceptable for our evaluation purposes. Furthermore, we performed significance tests comparing the unconstrained trees to the constrained ones, as shown in Table 3.2. The tests show that in all cases, the unconstrained trees fit the sequence data significantly better, and are hence preferable in cases where congruence with the taxonomy is not needed.

3.3.2 Accuracy

Measurement Method

Using the eight trees described above, we assess how using our PhAT affects phylogenetic placement accuracy. Each terminal branch of our RTs represents a consensus sequence, which is computed from species level sequences in Silva that share the same taxonomic label. We evaluate an RT by placing these species sequences onto the RT: Each species sequence is expected to be placed onto the branch leading to the consensus sequence that represents this particular species sequence. As the consensus sequences are derived from the taxonomy, all terminal branches of the tree have taxonomic labels. These labels thus identify the expected placement position

Table 3.2: Tree Topology Significance Tests. Here, we report typical significance tests comparing the four pairs of unconstrained (U) and constrained (C) trees used in our evaluation. The tests were performed with IQ-TREE v1.5.6 [179] under the "GTR+G" model (see Section 2.4.2 and Section 2.4.3; the "+G" stands here for the Γ model of rate heterogeneity) and 10 000 resamplings using the RELL method [116]. The table shows that the unconstrained trees fit the sequence data significantly better in all four cases and in all tests.
Columns are as follows. logL and deltaL: log likelihood and difference between constrained and unconstrained tree. bp: bootstrap proportion using RELL method [116]. p-(W)KH: p-value of the one sided and the weighted Kishino-Hasegawa test [115]. p-(W)SH: p-value of the (weighted) Shimodaira-Hasegawa test [218]. c-ELW: Expected Likelihood Weight [231]. p-AU: p-value of approximately unbiased (AU) test [217].

Tree	logL	deltaL	bp	p-KH	p-WKH	p-SH	p-WSH	c-ELW	p-AU
General (U)	-725199.040		1.0	1.0	1.0	1.0	1.0	1.0	0.9987
General (C)	-731949.568	6750.528	0.0	0.0	0.0	0.0	0.0	0.0	0.0012
Archaea (U)	-131862.815		1.0	1.0	1.0	1.0	1.0	1.0	1.0000
Archaea (C)	-133110.463	1247.648	0.0	0.0	0.0	0.0	0.0	0.0	0.0000
Bacteria (U)	-405028.378		1.0	1.0	1.0	1.0	1.0	1.0	1.0000
Bacteria (C)	-412464.820	7436.442	0.0	0.0	0.0	0.0	0.0	0.0	0.0000
Eukaryota (U)	-745442.969		1.0	1.0	1.0	1.0	1.0	1.0	1.0000
Eukaryota (C)	-753944.998	8502.030	0.0	0.0	0.0	0.0	0.0	0.0	0.0000

for each species sequence. For example, sequences `S1-6` in Figure 3.1 are represented by the consensus sequence for the *Calyptosporidae* clade, which is shown below the 6 sequences in the Figure. They are thus expected to be placed onto the *Calyptosporidae* branch in the RT.

In order to conduct the accuracy evaluation, we placed the respective subset of the SILVA database species sequences onto each of the eight RTs. As the sequences in SILVA are already aligned to each other, no alignment step was necessary for this. We further removed sites consisting entirely of gaps from the alignment because they contain no phylogenetic signal, in order to reduce the memory footprint of downstream steps. Phylogenetic placement was conducted using EPA-NG [10].

We quantify placement accuracy for a sequence by the distance to its expected placement branch. More precisely, we measured (a) the (discrete) number of branches between the actual placement and the expected branch, and (b) the (continuous) distance in branch lengths units. The former is more important in the context of phylogenetic placement, as the placement branch of a sequence is more significant in most analysis methods than its exact location on that branch. As a sequence can have multiple placement locations, the distances are in fact weighted averages incorporating the placement probabilities (LWRs, see Section 2.5.1). For sequences with a clear phylogenetic signal, that is, one placement with a high LWR, the averaging procedure only slightly affects the measurements. However, sequences with less clear placement locations are measured more comprehensively this way.

Results for the Unconstrained and Constrained Trees

The results for the four unconstrained and the four constrained trees are shown in Figure 3.3. Further details are provided in Table 3.3.

Considering the size of the trees, most sequences are placed in close vicinity to their expected branches. This is corroborated by the short average distances reported in Table 3.3. Furthermore, the average expected distance between placement locations (EDPL) [158] is low, indicating that the placements of a specific sequence mostly cluster in a small neighborhood of the tree. We observed that errors occur mostly in parts of the tree with short branches, which might be explained by the inability of 16S SSU sequences to properly resolve certain clades [104]. Also, the placement likelihood differences are small between neighboring, short branches, such that the placement signal is fuzzy.

With 77% of the sequences placed exactly on their expected branch, the accuracy is generally lowest for the *Bacteria* tree. This might be because the *Bacteria* have the most sequences in SILVA, and exhibit a high diversity. In the other three trees, more than 90% of the sequences are placed at most one branch away from their respective expected branch. The constrained trees exhibit similar placement accuracy, indicating that the differences in the inner branches of the trees indeed do not substantially affect the placement accuracy. Finally, we note that the results are reported without any manual corrections, and use overly broad RTs. Thus, in real

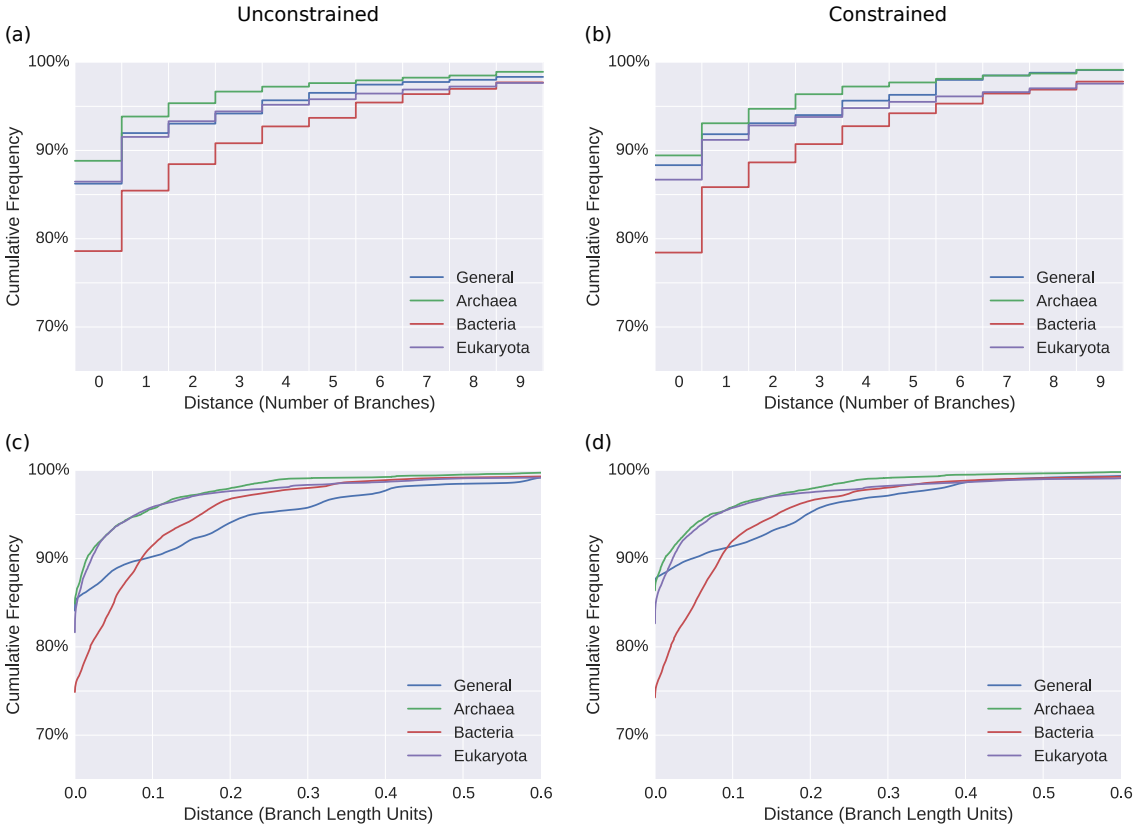

Figure 3.3: Accuracy of the unconstrained and constrained Phylogenetic Automatic (Reference) Trees (PhATs). We evaluated the accuracy of our PhATs by placing sequences and measuring the weighted distances to their respective expected placement branches. The figure shows the cumulative frequencies of number of sequences versus distances, measured in number of branches (top row, Subfigures (a) and (b)) and in branch length units (bottom row, Subfigures (c) and (d)). In other words, it shows how many sequences are placed within a certain radius from their expected branches. For example, in (a), more than 85% of the sequences of the *Bacteria* (red) are placed within a radius of at most one branch from their expected branch, and in (c), more than 95% of the Eukaryota (purple) are within a radius of 0.1 branch length units from their expected branches.
The figure compares the accuracy of using the unconstrained trees (left, Subfigures (a) and (c)) to using the SILVA taxonomy as constraint for the tree inference (right, Subfigures (b) and (d)). As explained in Section 3.3.1, the differences between the unconstrained and constrained trees mostly concern their inner branches, and thus are not expected to affect the accuracy much. This is confirmed by the fact that, overall, the results are similar between the pairs of trees. A slight improvement can be observed for the constrained General tree (blue), which performs better according to both distance measures. In most other cases, no significant differences can be observed.

Table 3.3: Overview of the Phylogenetic Automatic (Reference) Trees (PhATs) and their evaluation statistics. Details of four unconstrained (U) and four constrained (C) trees are shown. "Size" is the number of leaves of the tree, that is, the number of consensus sequences that the tree was inferred from, see Table 3.2. "% Seqs." the percentage of sequences from SILVA placed on it. The *General* tree does not cover all sequences, because there are some sequence labels in the database that could not be mapped to the taxonomy. "∅ Br. Len." is the average branch length in the tree. The evaluation results are reported in the remaining columns: Average distances of the sequences to their respective expected branch are listed in numbers of branches (Discrete) and in branch length units (Continuous), as explained in the text. Furthermore, "Exp. Br. Hits" shows how often the most probable placement was placed exactly on the expected branch. Lastly, the average expected distance between placement locations (EDPL) is shown. The EDPL is the sum of the distances between the placements of a sequences weighted by their probability [158].

				Average Distance			
Reference Tree	Size	% Seqs.	∅ Br. Len.	Discrete	Continuous	Exp. Br. Hits	∅ EDPL
General (U)	1998	98.7%	0.084	0.63	0.034	85.9%	0.00058
General (C)	1998	98.7%	0.086	0.57	0.027	88.2%	0.00046
Archaea (U)	511	3.4%	0.070	0.46	0.013	86.4%	0.00038
Archaea (C)	511	3.4%	0.071	0.45	0.013	88.2%	0.00041
Bacteria (U)	1914	84.6%	0.067	1.13	0.031	77.0%	0.00095
Bacteria (C)	1914	84.6%	0.071	1.11	0.031	76.6%	0.00091
Eukaryota (U)	2059	10.0%	0.080	0.79	0.022	84.9%	0.00032
Eukaryota (C)	2059	10.0%	0.083	0.81	0.024	85.7%	0.00031

world studies, where trees are often more specific for a clade of interest, better results are to be expected. Particularly when using Multilevel Placement with overlapping RTs, placement differences of a few branches on the first level tree are acceptable, as they do not change the second level tree on which the sequence is placed; see Section 3.3.5 for details.

Different Consensus Methods

As outlined in the method description (see Section 3.2.1), we represent clade diversity via majority rule consensus sequences. To assess the impact of the consensus method on the placement accuracy, we repeated the above evaluation using alternative consensus methods. In particular, we used Cavener's method [31, 32], and threshold consensus sequences [48, 49]. As shown in Figure 3.4, we found little difference between the methods.

By using alternative consensus methods, the consensus sequences and thus the sites in the alignments change. Furthermore, we used random initial starting trees for the tree inference. Hence, the obtained reference trees (not shown) differ substantially from each other. Across the corresponding trees of the tested consensus methods, we observed an average relative Robinson-Foulds distance [202] (see Section 2.3.2) of 49.5%. This is similar to our findings depicted above, e.g., in Figure 3.3. For the different consensus methods, again, the accuracy of their respective constrained variants of these trees (data not shown) does not change much compared to the accuracy obtained for the unconstrained trees shown in Figure 3.4. Thus, the differences in accuracy seen in the Figure are most likely due to the interplay of alignment and placed sequences (which is what we are interested in), and not due to differences in the trees (which are not of interest here).

The first three plots in Figure 3.4(a)–(c) exhibit similar accuracies. On average, majority rule, Cavener's, and low threshold ($\leq 70\%$) consensus methods place 82–83% of the sequences on the expected branch. As a general trend, the *Archaea*, being the smallest tree, tend to have the highest accuracy. On the other hand, the *Bacteria*, having the most sequences in SILVA, score worst. This changes for high consensus thresholds. At high thresholds, many sites contain ambiguity characters, thus blurring the phylogenetic signal. The *General* tree, representing the highest diversity, is most affected by this, as can be seen in the last two plots Figure 3.4(c) and Figure 3.4(d).

Actual Sequences

For the above evaluations of the PhAT method, we used some form of consensus sequence representation for the clades of the taxonomy, see e.g., Figure 3.3 and Figure 3.4. However, we also tested how the method behaves when using actual sequences from the database instead to represent the taxonomic clades, thus avoiding to unnecessarily blur the phylogenetic signal, and other potential drawback of consensus sequences.

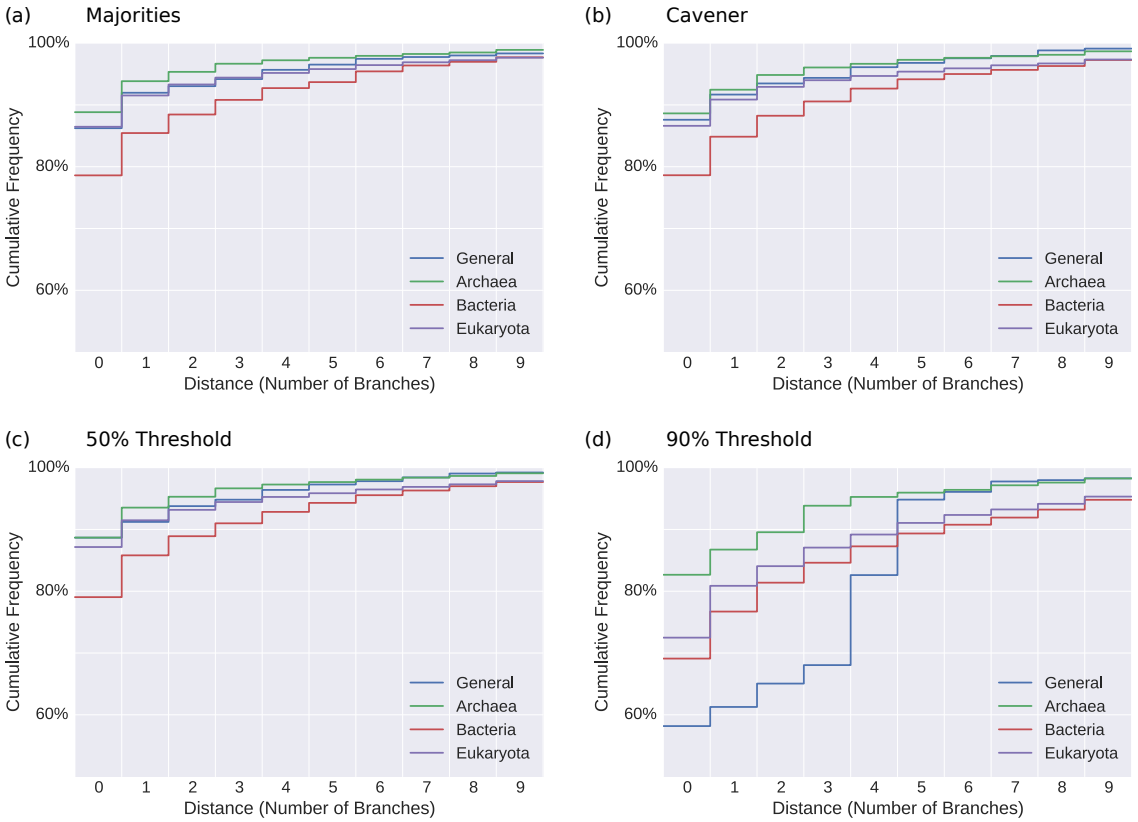

Figure 3.4: Effect of different consensus sequence methods on accuracy. In the main evaluation of our PhAT method, we use reference trees and alignments based on majority rule consensus sequences [49, 161] of the SILVA database sequences. Here, we evaluate the effect of using other consensus sequence methods on phylogenetic placement accuracy. In addition to (a) majority rule consensus, we tested (b) Cavener's method [31, 32], as well as threshold consensus sequences [48, 49] using thresholds of 50%, 60%, 70%, 80%, and 90%, of which two are shown in (c) and (d). The three remaining threshold methods exhibit accuracies almost exactly in between the shown plots, that is, accuracy decreases with increasing thresholds. For comparison, we also included Figure 3.3(a) again, here as Subfigure (a), using the same y-axis scaling as the other plots. All trees used in this part of the evaluation are unconstrained. We only show distances measured in number of branches here, because this is more relevant in the context of our methods.

As manually selecting representative sequences from the database was not practical, we used the following automated approach. First, we took the 90% threshold consensus sequences of the PhAT method that were already evaluated in Figure 3.4(d). By using a high threshold, most of the diversity of each clade is included. Then, for each such consensus sequence, we calculated a score for all sequences from the database that were used to construct this consensus sequence. This score is the number of different nucleotides between the consensus sequence and the database sequence. The sequence with the lowest score (that is, with most matching nucleotides) was then used as representative of the clade. Thereby, the taxonomic clades are represented by actual sequences from the database. However, as these sequences are close to the respective consensus sequence, they are still good representatives of the diversity of the clade. This resulted in a set of sequences of the same size as the original set of consensus sequences. Using these sequences, we then again inferred a tree and conducted the evaluation procedure by placing all sequences of the database on that tree, as described before. The results for the four unconstrained trees is shown in Figure 3.5.

We found that this approach yields trees that are less accurate for phylogenetic placement. The resulting accuracy is worse in all cases. That is, on average, the sequences were placed further from their respective expected branch. We suspect that this is (i) because single sequences do not capture the diversity of their clade as well as consensus sequences, and (ii) because they do not incorporate as much biological information (e.g., in form of ambiguity characters). We hence conclude that using consensus sequences to represent clades in our PhAT approach is superior.

Further Aspects and Observations

In the evaluations above, we generally found that using a constraint when inferring the tree only slightly changes the accuracy of our evaluation. However, when considering only the distance of the most likely placement (highest LWR) to its correct edge instead of using average distances weighted by the LWR per QS, the constrained trees consistently yield better results (data not shown). In other words, the most likely placement is more often on the correct branch of the constrained trees. For example, the most significant change is observed for the Eukaryota tree, with 84% correct placements for the unconstrained tree, but 89% for the constrained one. We suspect that this is an artifact of our evaluation process, as we consider a sequence to be correctly placed if the placement branch belongs to the consensus sequence to which the sequence contributed. As the selection of sequences for each consensus sequence is guided by the taxonomy, using the same taxonomy as constraint for the tree thus might also improve the placement accuracy.

As a final remark, we implicitly assumed the taxonomic label of each sequence to be correct. That is, in the evaluations, we measured the accuracy of the placements using the taxonomic labels of the sequences in Silva as an indicator of the expected branch of each sequence. However, errors are expected due to incongruity between the taxonomy and the phylogeny [175], as well as due to taxonomically mislabeled sequences [119]. For example, Sativa [119], found 9934 mislabeled sequences in

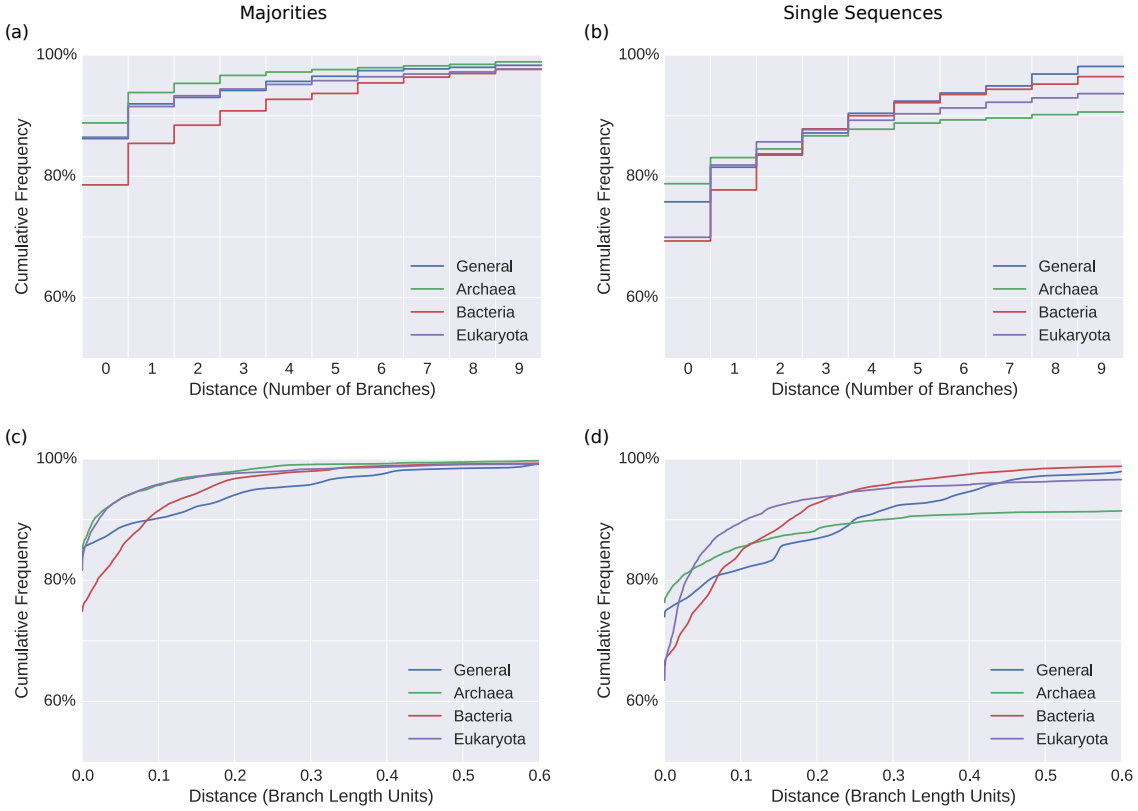

Figure 3.5: Effect of using actual sequences (instead of consensus sequences) on placement accuracy. Subfigures (a) and (c) show the evaluation of the majority rule consensus sequences, and are identical to Figure 3.3(a) and Figure 3.3(c), respectively. They are included here for ease of comparison, however with the y-axis scaled to fit the remaining subfigures. Subfigures (b) and (d) show the evaluation of the approach of using actual sequences from the database (instead of consensus sequences), as explained in the text. The top row (Subfigures (a) and (b)) shows distances in number of branches away from the expected placement branch; the bottom row (Subfigures (c) and (d)) shows distances in branch length units. All trees used for this evaluation are unconstrained.

the Silva database. Furthermore, 17 452 sequences contain one of "incertae", "unclassified" or "unknown" in their name, indicating that those sequences might not be reliable. In total, there are 25 910 (or 4.3%) such dubious sequences in version 123.1 of the Silva database. Not all sequences are hence expected to be placed on their expected branches. We thus evaluated how these dubious sequences affect the accuracy of the trees. To this end, we used the same four trees as used in the main part of the evaluation (that is, they were constructed with all sequences, including the dubious sequences), but for the evaluation step itself excluded the dubious sequences. That is, those sequences were not placed on the trees, and their distance to the expected branch was not used for the evaluation. In most cases, this improved the results slightly, but not by much (data not shown). This shows that our trees are also robust to such uncertain sequences. Therefore, we decided to only report the unfiltered results in the above evaluations.

3.3.3 Empirical Datasets

PhATs are intended for conducting phylogenetic placement of environmental sequences. As the true evolutionary history of such sequences is unknown, we can not repeat the previous accuracy tests on empirical environmental datasets. Instead, we assess if the PhATs yield meaningful quantitative results for typical post-analysis methods. To this end, we placed two empirical metagenomic amplicon barcoding datasets (see Appendix B for their details) on our unconstrained *Bacteria* tree. To asses the placement results obtained from the PhAT, we performed Squash Clustering and Edge PCA [155] post-analyses (see Section 2.5.5) on the placement results.

Bacterial Vaginosis Dataset

We used an empirical sequence dataset of the vaginal microbiome of 220 women with a total of 426 612 sequences [225] for this evaluation. For details on the dataset and its processing, see Appendix B.1. The original study showed associations between the presence of certain bacterial species and the diagnosis of Bacterial Vaginosis (BV), a condition caused by changes in the vaginal microbiome. In the study, the Nugent score [181] was used as a clinical diagnostic criterion for BV, which ranges from 0 (healthy) to 10 (severe illness). We placed the sequences of the dataset on their original tree and on our unconstrained *Bacteria* tree, and reproduce some of the results from the original study to assess differences induced by using distinct references trees. The results reveal that the PhAT reproduces certain aspects of the results of previous studies based on custom RTs with manually selected reference sequences, at least to the extend that is expected from its phylogenetic resolution.

First, we conducted Squash Clustering [155] (see Section 2.5.5) of the samples placed on the two trees. The resulting hierarchical cluster trees of the samples are shown in Figure 3.6.

The general features of the two cluster trees are comparable, indicating that our tree is able to distinguish between healthy and sick patients. However, there is a major difference in the lower half of the trees: While (b) shows some small branch lengths

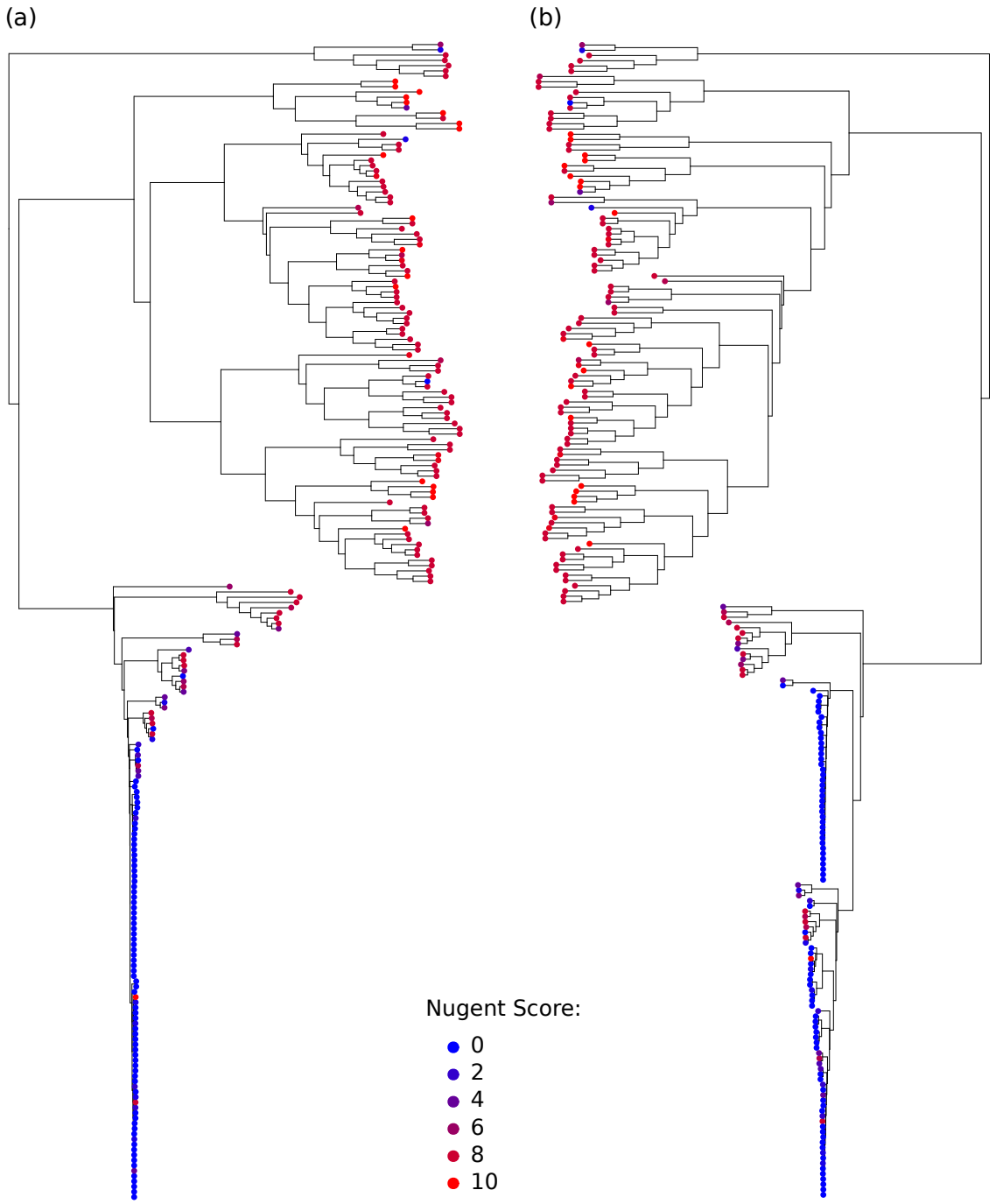

Figure 3.6: Assessment of a PhAT for conducting Squash Clustering. The Figure compares the hierarchical clustering trees resulting from a Squash Clustering analysis (see Section 2.5.5) using (a) our unconstrained *Bacteria* PhAT and (b) the original reference tree of [225]. Subfigure (b) is a recalculation of Figure 1(A) of [225], and horizontally flipped for ease of comparison. The tips of both clustering trees correspond to samples, which are colored by the respective Nugent score of each sample, where higher values indicate women with severe Bacterial Vaginosis.

and even a separated sub-clade of samples with low Nugent score, these branches have a length of virtually zero in (a). As shown in [225], the healthy patients are divided into two classes, based on the presence of two species of *Lactobacillus*. The original reference tree contains sequences of those species, and can thus distinguish between them. Our broad *Bacteria* tree however does not have this degree of species-level resolution and thus treats them the same, yielding a negligible KR distance (Section 2.5.4) between the samples. Although this finding is expected, it serves as an example for the limits of our method.

Second, we conducted an Edge PCA [155] (see Section 2.5.5) of the samples. The resulting scatter plots are shown in Figure 3.7.

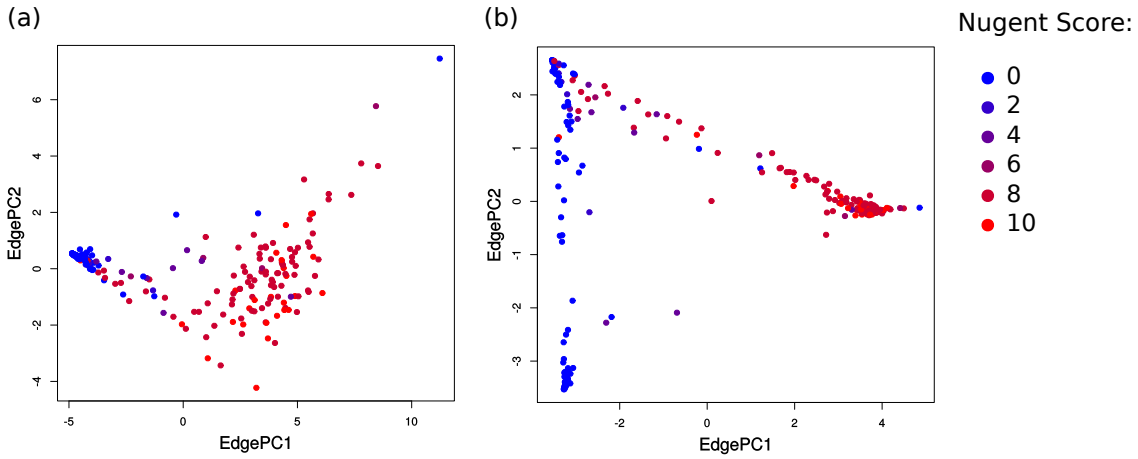

Figure 3.7: Assessment of a PhAT for conducting Edge PCA. The Figure compares the scatter plots resulting from an Edge PCA (see Section 2.5.5) using (a) our unconstrained *Bacteria* PhAT and (b) the original reference tree of [225]. Subfigure (b) is a recalculation of Figure 3(A) of [225]. The items represent samples, which are colored by the respective Nugent score of each sample, where higher values indicate women with severe Bacterial Vaginosis.

The scatter plots show the PhAT is able to separate samples by Nugent score, that is, to classify them into healthy (left, blue items) and sick patients (right, red items). However, as with Squash Clustering, samples that only differ in placements at the species level are not separated from each other in the Edge PCA plot. Hence, the two classes of healthy patients do again not exhibit the separation based on the two *Lactobacillus* species that is apparent when using the original tree. Thus, the samples with low Nugent score form one blob in Figure 3.7(a).

This limitation can be overcome in two ways: On the one hand, one can use a PhAT with finer taxonomic resolution, that is, with more taxa that resolve down to species level. On the other hand, our multilevel placement approach (see Section 3.3.5) can be used with a refined second level tree that for example contains species sequences of the relevant *Lactobacillus* clades.

We however generally note that similar issues of deficient resolution or missing species can potentially also arise when hand-selecting reference sequences, and are thus not an inherent disadvantage of our method. In the end, it is the responsibility of the researcher to make sure that the selected reference sequences are suitable for the dataset to be placed.

Human Microbiome Project Dataset

Next, we tested the unconstrained *Bacteria* tree generated by our PhAT method for placing and analyzing a large sequence dataset. For this, we use the Human Microbiome Project (HMP) [101, 166] data, and selected 9192 samples from different body sites with a total of 117 million sequences. For details on the dataset and its processing, see Appendix B.4. The sequences were placed on the tree, and subsequently analyzed with two different methods, as shown in Figure 3.8.

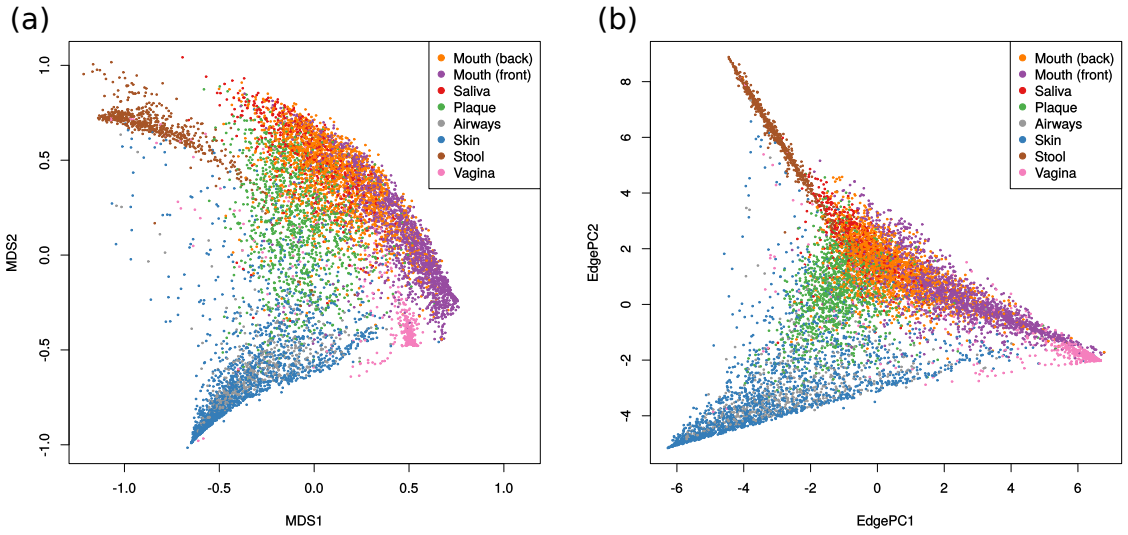

Figure 3.8: Assessment of a PhAT for large dataset analyses. Sequences from the HMP dataset [101, 166] were placed on our unconstrained *Bacteria* tree, and analyzed with two analysis methods. In subfigure (a), we visualized the pairwise KR distances between all samples, using a two-dimensional Multidimensional scaling (MDS). In subfigure (b), we performed Edge PCA (see Section 2.5.5) on the samples. For both plots, we categorized the 19 original body site labels into 8 regions, in order to make the plot more readable. See Table B.2 for the mapping between the original labels and the ones used here.

First, we computed the pairwise KR distance matrix (see (Section 2.5.4)) between all samples. This high-dimensional matrix was then embedded into the plot by performing Multidimensional scaling (MDS). MDS [69, 122, 151] is a dimensionality reduction technique that finds an embedding of a distance matrix into lower dimensions (in this case, 2 dimensions) preserving higher dimensional distances as well as possible. Second, we again performed Edge PCA [155] (see Section 2.5.5) on the samples.

Both subfigures show that the tree, despite only representing higher taxonomic levels, suffices to separate different body site regions from each other. Even different oral regions are mostly separated, although there seems to be quite some overlap. We hence conclude that our PhAT is capable of analyzing such datasets and yields useful results.

3.3.4 Taxonomic Assignment and Profiling

Here, we assess how PhATs perform when used for obtaining a taxonomic profile of a set of samples in conjunction with phylogenetic placement. We emphasize though that taxonomic assignment and profiling are neither the focus of PhATs, nor the intended standard applications of phylogenetic placement.

To perform the evaluation, we conducted parts of the CAMI Challenge [215], which is a community-driven effort to assess taxonomic profiling methods using a common set of benchmark data sets. To assess the feasibility of using trees generated with PhAT to obtain taxonomic profiling of microbiome data, we utilized the *mouse gut* data set of the 2nd CAMI Challenge [26]. See Appendix B.5 for details on our processing of this dataset. In short, we phylogenetically placed the reads of the 16S region of the dataset on our unconstrained and constrained *Bacteria* trees. We then used this placement data to taxonomically assign the reads based on the underlying Silva taxonomy of the trees, in analogy to the method used by Sativa [119].

Unfortunately, the CAMI Challenge requires taxonomic assignments that conform with the NCBI taxonomy [12, 212]. As our reference tree is however based on the Silva taxonomy [269], we thus had to compute a mapping between the two taxonomies. To this end, we developed a dedicated mapping procedure to, in a best effort approach, map our results to NCBI taxonomic names and IDs. The mapping is based on the *loose mapping* procedure by [9]. More specifically, we tried to map taxonomic paths to their name, rank, and ID in the NCBI taxonomy, if we find a name-based match between the two. When this fails, the phylogenetic placement mass assigned to a taxonomic path by our approach is instead added to the last successful mapping further up in the taxonomic hierarchy. By initiating this procedure for each taxonomic path from its root downwards, we ensure that all placement masses are taken into account.

This mapping is a major disadvantage of our approach when using a Silva-based reference, as the Silva and NCBI taxonomies are far from being congruent [9]. Also note that our reference tree is limited to 16S rDNA region. This substantially reduces the volume of data we can evaluate; in this particular test, only $\approx 0.08\%$ of the total mouse gut data was identified as belonging to the 16S region (see also Appendix B.5). This means that our taxonomic profiling only uses a small fraction of the available data.

The resulting per-read assignment was then used to generate a taxonomic profile of the data. We used the CAMI evaluation tool for taxonomic profilers OPAL [215] to compare our approach to competing software and the "gold standard" result for the data set.

Table 3.4: CAMI Scores and Ranks. The table shows the scores and ranks of different tools evaluated with data from, and following the protocol of, the 2nd CAMI challenge [26, 215]. Here, we compare the taxonomic assignment and profiling based on our PhATs to the tools that took part in the 2nd CAMI challenge. For this, we used the unconstrained and constrained *Bacteria* tree, which are abbreviated in the table as "PhAT (U)" and "PhAT (C)", respectively.
Four metrics are used in CAMI for evaluating tools: Recall (completeness), precision (purity), L1 norm error (abbreviated here as L1 NE), and Weighted Unifrac Error (abbreviated here as WUE). For each metric, the comparative scores of the tools are shown, as well as their rankings, relative to each other. Also, the total sum of scores and the total rank are shown, which add up the values of the four metrics. The procedure of the scoring and ranking is explained in detail in the Online Supplement of [215]. Despite the caveats and limitations that are explained in the text, using our PhATs trees to obtain a taxonomic profile yields rankings in the middle of the field for all metrics.

	Total		Recall		Precision		L1 NE		WUE	
Tool	Sum	Rank	Score	Rank	Score	Rank	Score	Rank	Score	Rank
Metaphlan	1814	1	958	3	75	1	731	2	50	1
MetaPhyler	2995	2	314	1	2119	7	322	1	240	5
CommonKmers	3333	3	1448	6	409	2	1318	7	158	3
mOTU	3751	4	1703	8	488	3	1260	5	300	6
PhAT (C)	3784	5	1202	4	1116	4	1280	6	186	4
PhAT (U)	3933	6	1436	5	1153	5	1208	4	136	2
TIPP	4263	7	892	2	2126	8	930	3	315	7
FOCUS	6153	8	2079	9	1636	6	2018	8	420	8
Quikr	6838	9	1488	7	2398	9	2453	9	499	9

We here show the most important OPAL results: Table 3.4 shows the scores and ranks of our approach compared to the other CAMI participants; Figure 3.9 compares the tools based on different error metrics.

TODO: the order of the table and figures is messed up, and the figures of the following section interfere... :-(

The resolution of the assignment is limited by the taxonomy used when running the PhAT method, that is, we could not assign reads at *Species* level. Furthermore, we were only able to use a small fraction of the reads (16S) and had to use incongruent taxonomies. Despite this, we find that the performance of our approach is in the mid-range of the tools evaluated by CAMI. Note that this is a comparison to tools that are dedicated to taxonomic profiling, which also typically can assign more of the available reads. Therefore, our method yields reasonable accuracy for taxonomic assignment and profiling.

3.3.5 Subclades and Multilevel Placement

We selected five bacterial clades to evaluate PhAT accuracy on smaller clades, as well as to assess some properties of the Multilevel Placement approach. The same clades were already scrutinized in SATIVA [119]. Figure 3.10 shows the unconstrained *Bacteria* tree from the previous evaluations, with the branches of these five test clades highlighted.

Subclade Accuracy

First, using the sequences and sub-taxonomies in SILVA of these five clades, we built unconstrained and constrained PhATs. We then conducted the same accuracy analysis as explained before on these ten trees. That is, we placed the SILVA sequences of the five clades onto their respective PhATs and evaluated distances to expected branches. Thereby, we evaluated the accuracy of these PhATs when used as second level clade trees. The results are shown in Figure 3.11.

TODO: maybe add a table similar to Table 3.3, listing sizes and accuracy etc.

The placement accuracy is slightly worse for the sub-clade trees than for the eight comprehensive PhATs evaluated before (see Section 3.3.2), which can be seen by comparison to Figure 3.3. On average, 73.4% of the sequences were placed exactly on their expected branch, dominated by *Proteobacteria* and *Firmicutes*, which combined make up 75% of the sequences in the five clades, and have an accuracy of 71%. The *Actinobacteria* have the highest accuracy, with 82% of their sequences placed on the expected branch.

There are however also differences between the clades. The two smallest clades, *Actinobacteria* and *Cyanobacteria*, exhibit the shortest distances in branch length units. In fact, the longest distance of any sequence from its expected branch in the *Cyanobacteria* clade is around 0.4, which is indicated by the end of the red line in the lower two plots of Figure 3.11. On the other hand, the *Firmicutes* generally

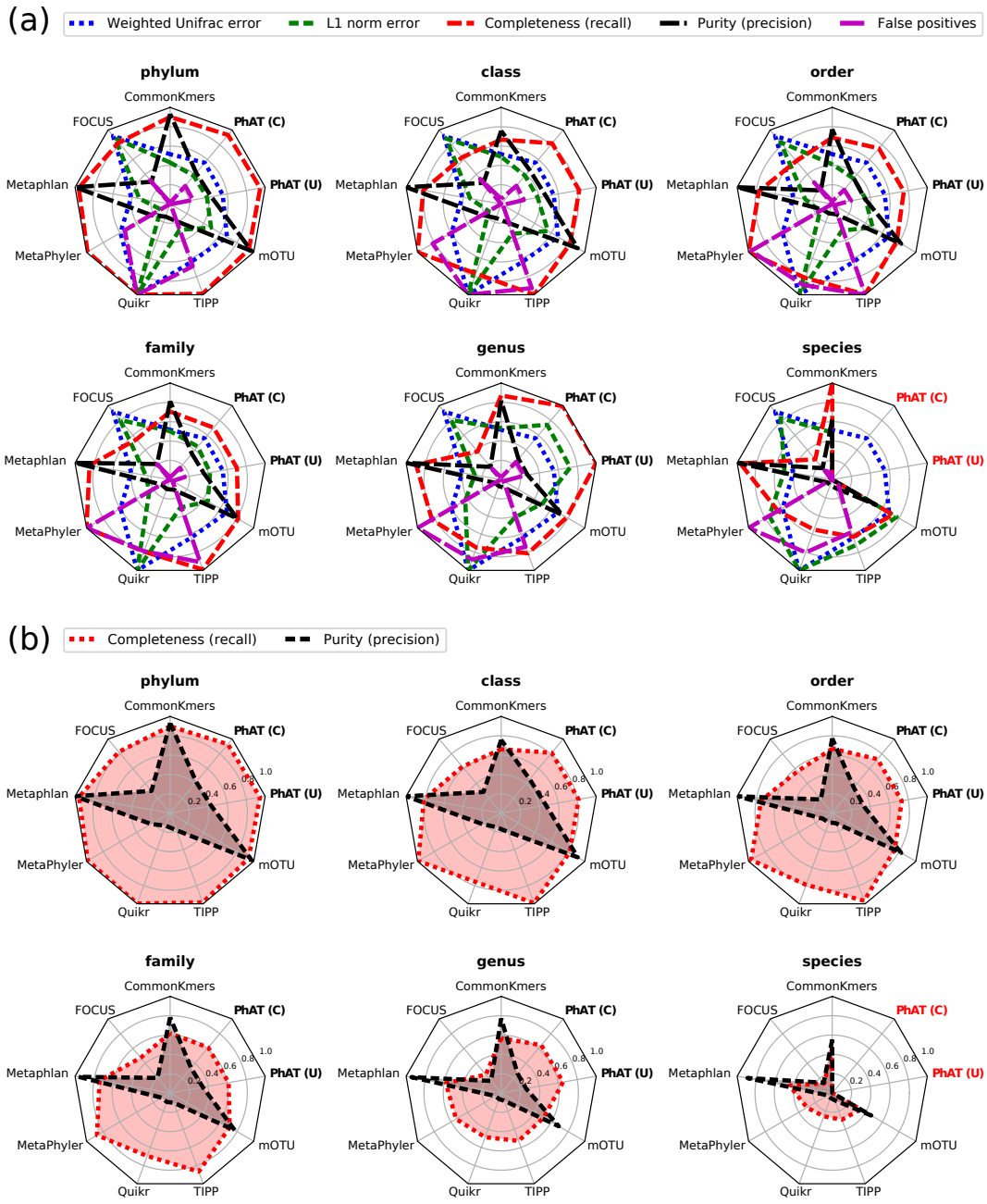

Figure 3.9: CAMI Profiling Results. The figure compares the taxonomic profiling conducted with our *Bacteria* trees to the tools of the 2nd CAMI challenge [26, 215]. The unconstrained and constrained tree are abbreviated here as "PhAT (U)" and "PhAT (C)", respectively. Subfigure (a) shows the *relative* performance of the tools across taxonomic ranks using the error metrics of CAMI: Weighted Unifrac error, L1 norm error, recall (completeness), precision (purity) and false positives. Subfigure (b) shows the *absolute* recall (completeness) and precision (purity) for each tool across the taxonomic ranks. In both subfigures, the red text for our PhAT evaluations indicates that no predictions at the corresponding taxonomic rank were returned. This is because our SILVA-based tree does not have *species* resolution and does hence not allow for taxonomic profiling at this level.

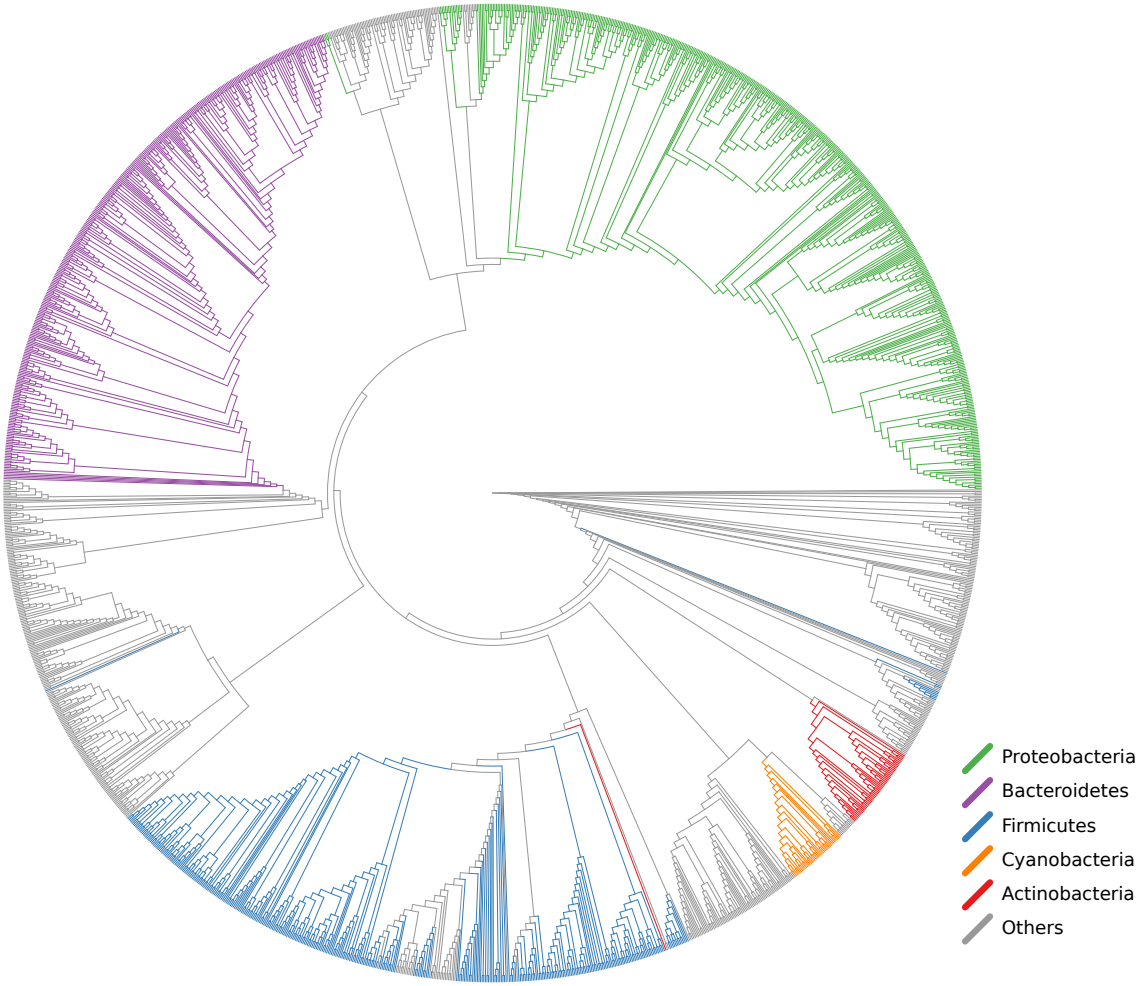

Figure 3.10: Unconstrained *Bacteria* tree with five bacterial sub-clades. This tree is the result of our PhAT method applied to the *Bacteria* sequences in Silva. The tree contains a total of 1914 taxa. Colorized are the five *Phylum* level sub-clades that we used for testing multilevel placement: *Proteobacteria* (505 taxa), *Bacteroidetes* (362 taxa), *Firmicutes* (360 taxa), *Cyanobacteria* (39 taxa) and *Actinobacteria* (53 taxa). The incongruence between taxonomy and phylogeny is visible here as non-monophyletic colored branches. We thus here define a clade to consist of all branches that are part of a monophyletic split of the tree with respect to the taxa in the clade. In other words, all branches on one side of a split are considered to belong to a clade, if that side of the split only contains taxa from that clade. These branches then receive the same color here. Then, for multilevel placement, a sequence is considered to be part of a clade if its most probable placement falls into that clade. For example, a sequence that is placed onto one of the orange branches on this tree is subsequently placed in the *Cyanobacteria* tree for the second level placement. Each of the five sub-clades is represented by multiple branches here, which we call the "overlap" with the *Bacteria* tree.

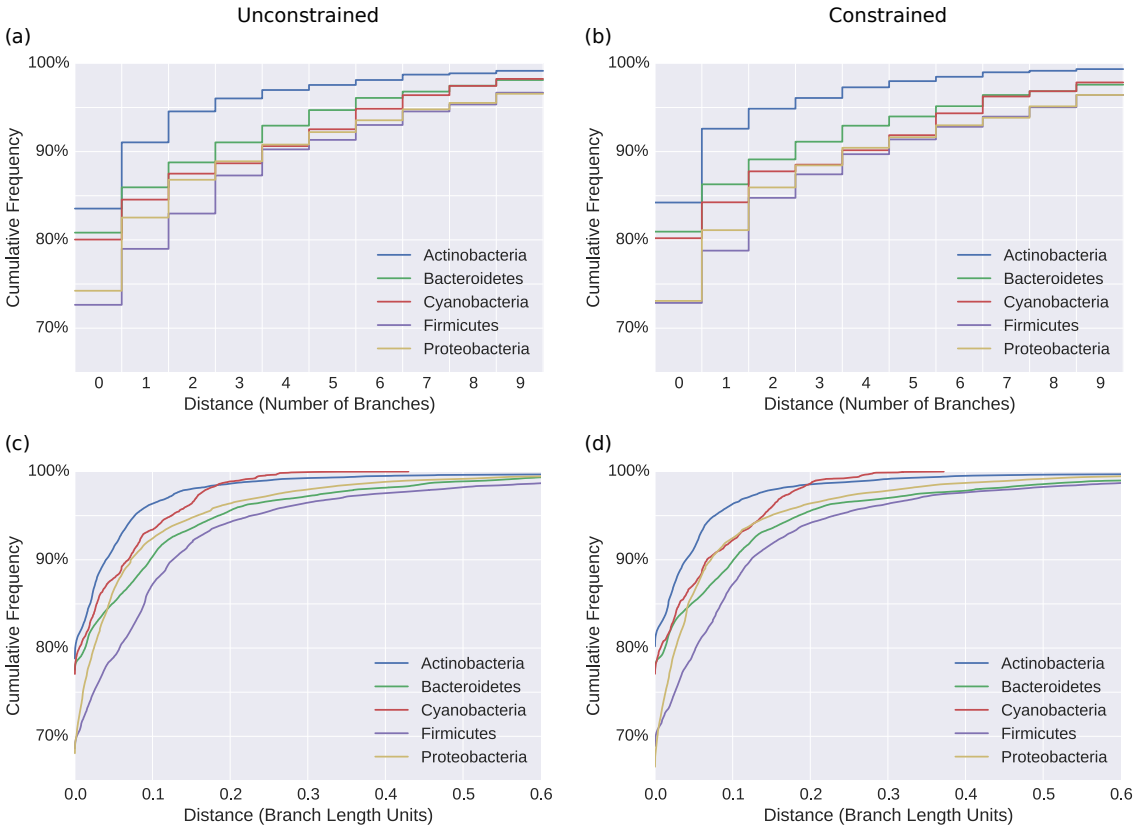

Figure 3.11: Accuracy of the PhATs of five bacterial sub-clades. We used five sub-clades of the *Bacteria* in SILVA, which were already scrutinized in [119], to test how our PhAT method works for less diverse sets of sequences. These five clades are also highlighted in Figure 3.10; see there for a description of the clades. The evaluation was conducted as explained in the text, using the accuracy measurement as before (see Section 3.3.2). In short, we placed the SILVA sequences of the clades on their respective tree, and measured how far each of them is away from the branch of the consensus sequence it is represented by.
The top row (Subfigures (a) and (b)) shows discrete distances in number of branches; the bottom row (Subfigures (c) and (d)) show continuous distances in branch length units. The left side shows the accuracy of the unconstrained trees, the right side shows the accuracy for trees constrained by the SILVA taxonomy.

have the lowest accuracy. In Figure 3.10, which shows the unconstrained *Bacteria* tree, the *Firmicutes* clade exhibits many paraphyletic branches, which is a known issue [187]. This indicates that there is a high incongruence between the *Firmicutes* taxonomy and phylogeny in SILVA, which might explain why the *Firmicutes* score worst in Figure 3.11.

These results are likely due to the inability of 16S SSU sequences to properly resolve lower taxonomic levels [104, 170, 192]. For example, Table 2 of [104] lists 10 bacterial genera that are known to be hard to identify using 16S sequences. These genera account for 7.9% of the 2846 taxa that are represented by the five bacterial trees tested here. Furthermore, 95 553 of the 450 313 sequences that were placed on those trees (21.2%) belong to one of these genera. This might explain the worse scores of these clade trees. Lastly, the consensus sequences at the tips of the trees represent the *Genus* level. Thus, these have short branches, which increases the probability of misplacements.

First Level Accuracy

Next, using the five clades, we evaluated the accuracy of the first placement level when conducting Multilevel Placement (as introduced in Section 3.2.2). So far, our evaluation focused on the distance from a sequence placement to its expected placement branch. For the first placement level on a backbone tree (BT), it is however more important that a sequence is placed into the correct clade than the exact placement branch. A sequence that is placed in the correct clade of the first level tree can subsequently be placed on the correct second level clade tree (CT) tree. Thus, we used the unconstrained *Bacteria* BT again, and assessed how many sequences were placed in the clades shown in Figure 3.10. Of the 450 313 sequences in SILVA in these clades, 98.0% were placed (most likely placement) into a branch of their corresponding clade. Thus, for multilevel placement, they will be assigned to the correct second level clade tree (CT). More specifically, the *Firmicutes* perform worst, as only 94.7% of the *Firmicute* sequences are placed into the corresponding clade. This can be explained by the high amount of paraphyletic branches of this clade, as mentioned above, which is a known issue [187]. The sequences of the other four clades we tested achieve a clade identification accuracy exceeding 99%. This shows that having a high overlap of the clades with with the BT yields high accuracy. In other words, second level clade trees should be represented by multiple branches on the backbone tree.

As mentioned in Section 3.2.2 before, a high-level taxonomic constraint can improve the accuracy of placing a sequence into the correct BT clade. To show this, we inferred the *Bacteria* RT again, but used a *Phylum* level constraint that separates the five clades from each other and from the rest of the tree. All branches within the clades were resolved using maximum likelihood. The tree (not shown) is similar to the tree in Figure 3.10, but all five clades are now monophyletic. Using this tree, 99.3% of the sequences were placed into the correct clade. Particularly the accuracy for *Firmicutes* improved, yielding an accuracy of 99.5%.

Overall, our experiments show that the first level placement is highly accurate, even if an extremely diverse "all bacteria" backbone tree is used. The accuracy on the second level is slightly worse when using PhATs as CTs.

3.4 Conclusion and Outlook

We presented methods and algorithms to facilitate and accelerate phylogenetic placement of large environmental sequencing studies. TODO: ref to implementation section in the supplement?!

The Phylogenetic Automatic (Reference) Tree (PhAT) method (Section 3.2.1) provides a means for automatically obtaining suitable reference trees by using the taxonomy of large sequence databases. Using the SILVA database as a test case, we showed that it can be applied for accurately (pre-)placing environmental sequences into taxonomic clades. In combination with our multilevel placement approach (Section 3.3.5), even very broad PhATs achieve high accuracy, particularly when using high-level clade constraints. The method can also be used for rapid data exploration in environmental sequencing studies: A PhAT might be useful to obtain an overview of the taxa that are necessary to capture the diversity of a sequence dataset, without the substantial human effort and potential bias of manually selecting reference sequences. As we showed, PhATs can also be used to obtain taxonomic assignments and profiles for a set of samples, in conjunction with phylogenetic placement (Section 3.3.4). To capture clade diversity with finer resolution, for example for a second placement level, clade-specific PhATs can be inferred. If species-level resolution is required, we recommend that the sequences are inspected by an expert, in order to confirm that the tree is appropriate for the dataset to be placed on it. Furthermore, as our automated approach inevitably suffers from errors in the database it is based on, we recommend using SATIVA [119] to identify potentially mislabeled sequences in the database. One should also keep in mind that phylogenetic placement does not necessarily provide resolution at the *species* level [60].

As we show, our multilevel placement method (Section 3.2.2) as well as the preprocessing pipeline (Section 3.2.3) accelerate the placement process without sacrificing accuracy. By first placing the query sequences on a broad backbone tree (BT), as described in the method, novel environments with sequences of unknown evolutionary origin can be classified without having to process a large tree comprising all taxa of interest. The method hence offers the benefits of high resolution reference trees, without suffering the mentioned downsides that usually come with large alignments. A second placement level on a set of clade trees (CTs) provides sufficient taxonomic resolution for biological interpretation. Placement accuracy can be further improved by inferring the BT with a high-level constraint that separates the clades of the CTs from each other and thus ensures monophyly of these clades. Furthermore, for the practical applicability and relevance of this approach, we refer to [149].

Apart from exploring read data from unknown environments, we see online services as a potential application of our methods. A web service that offers phylogenetic

placement of user-submitted sequences is confronted with two issues: Firstly, the potentially large number of query sequences, and secondly, their unknown provenance. Both can be solved by using a broad "all-of-life" backbone tree for pre-classification, and subsequently distributing the second-level placement to different compute nodes.

TODO: implementation note and ref to supplement?

4. Visualization

This chapter is based on parts of the peer-reviewed publication:

- **Lucas Czech** and Alexandros Stamatakis. "Scalable Methods for Post-Processing, Visualizing, and Analyzing Phylogenetic Placements." *PLOS One*, 2018.

Contributions: Lucas Czech... Alexandros Stamatakis...

4.1 Background and Motivation

When analyzing a set of metagenomic sequence samples (Section 2.2.2) using phylogenetic placement (Section 2.5), a first step is often to visualize the data. For small samples, it is possible to mark individual placement locations on the reference tree (RT), as offered for example by iTOL [128], or even to create a tree where the most probable placement per query sequence (QS) is attached as a new branch, as implemented in the GUPPY tool from the PPLACER suite [158], RAxML-EPA [16, 228], and our tool GAPPA. TODO: link to supplement or paper, or something For larger samples, one can alternatively display the per-edge placement mass, either by adjusting the line widths of the edges according to their mass, or by using a color scale, as offered in GGTREE [271], GUPPY, and GAPPA. Using per-edge colors corresponds to binning all placement of an edge into one bin (see Section 2.5.3). For large datasets, the per-edge masses can vary by several orders of magnitude. In these cases, it is often preferable to use a logarithmic scaling, as shown in [149]. In addition to visualizing each sample separately, the average mass distribution (after *squashing* the masses, see Section 2.5.3) gives an overview of a set of samples.

The visualizations provide an overview of the species abundances over the tree. They can be regarded as a more detailed version of abundance charts [101], which

are typically shown in the form of pie or bar plots [126, 149]. For instance, Figure 4.1 shows a comparison of a simple pie chart of abundances compared to a visualization of per-branch abundances on a reference tree for the same dataset. Although there exist more advanced variants such as hierarchical pie charts offered by the Krona tool [183], it is apparent that the tree visualization provides more in-depth information. In Figure 4.1(b), we actually combined the information of the pie chart with the tree visualization by adding the abundances next to each clade.

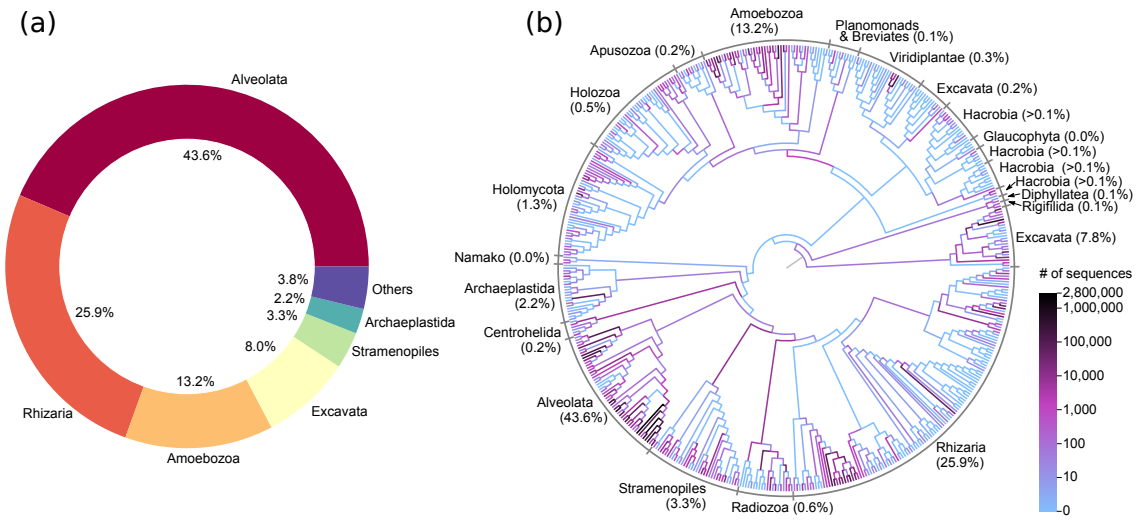

Figure 4.1: Visualizations of sequence abundances. The figure shows an example of (a) a typical pie chart of taxonomic abundances and (b) the much more informative per-branch mass visualization using phylogenetic placement on a reference tree. The data is from [149], see Appendix B.2 for details. The branches of the tree are colored by abundances on a logarithmic scale, and clades of the tree are annotated with the per-clade abundances, effectively combining the information of the pie chart with the tree visualization.

Such visualizations directly depict the placement masses on the tree. When visualizing the accumulated masses of multiple samples at once, it is important to chose the appropriate normalization strategy for the task at hand, as explained in Section 2.5.3. For example, if samples represent different locations, one might prefer to use normalized masses, as comparing relative abundances is common for this type of data. On the other hand, if samples from the same location are combined (e.g., from different points in time, or different size fractions), it might be preferable to use absolute abundances instead, so that the total number of sequences per sample can be visualized.

When placing OTUs (see Section 2.5.3), or ignoring sequence abundances, the resulting visualizations can be interpreted as a depiction of species diversity. Moreover, these visualizations can be used to assess the quality of the RT. For example, placements into inner branches of the RT may indicate that appropriate reference sequences (i) have not been included or (ii) are simply not yet available. This com-

plements the sequence filtering that relies on so-called backbone trees as previously described in Section 3.2.2.

These visualizations are useful tools for initially exploring a dataset and its features in terms of species abundances. However, when working with a set of multiple samples, they do not immediately reveal comparative differences between samples that might hint at underlying biological or ecological properties of the samples or their environment.

4.2 Methods and Implementation

Here, we introduce visualization methods for phylogenetic placement of a set of metagenomic sequence samples that highlight (i) regions of the tree with a high variance in their placement distribution (called *Edge Dispersion*), and (ii) regions with a high correlation to meta-data features (called *Edge Correlation*).

Both methods take as input a set of samples, each consisting of a set of query sequence (QS) placed on a fixed reference tree (RT). They then use the edge masses matrix and the edge imbalances matrix as introduced in Section 2.5.3 to calculate per-branch quantities, which are subsequently visualized on the RT.

4.2.1 Edge Dispersion

The Edge Dispersion is derived from the edge masses or edge imbalances matrix by calculating a measure of dispersion for each of the matrix columns, for example the standard deviation σ. Because each column corresponds to an edge, this information can be mapped back to the tree, and visualized, for instance, via color coding. This allows to examine which edges exhibit a high heterogeneity of placement masses across samples, and hence indicates which edges discriminate samples. As edge mass values can span many orders of magnitude, it might be necessary to scale the variance logarithmically.

Often, one is more interested in the branches with high placement mass, as they indicate the most abundant species in the samples. In these cases, using the standard deviation or variance is appropriate, as they also indicate the mean mass per edge. On the other hand, by calculating the per-edge Index of Dispersion [69], that is, the variance-mean-ratio σ^2/μ, differences on edges with little mass also become visible. Note that this is a valid operation, as edge masses are a zero-based dimension. The Index of Dispersion is useful to explore heterogeneity on edges with low species abundances.

As Edge Dispersion relates placement masses from different samples to each other, the choice of the normalization strategy *is* important (see also Section 2.5.3). When using normalized masses, the magnitude of dispersion values needs to be cautiously interpreted [139]. The Edge Dispersion can also be calculated for edge imbalances in form of the standard deviation. As edge imbalances are usually normalized to $[-1.0, 1.0]$, their dispersion can be visualized directly without any further normalization steps. However, because imbalances can be negative, the Index of Dispersion

is not applicable to them. An example for an Edge Dispersion visualization is shown in Figure 4.2(a), and discussed in Section 4.3.

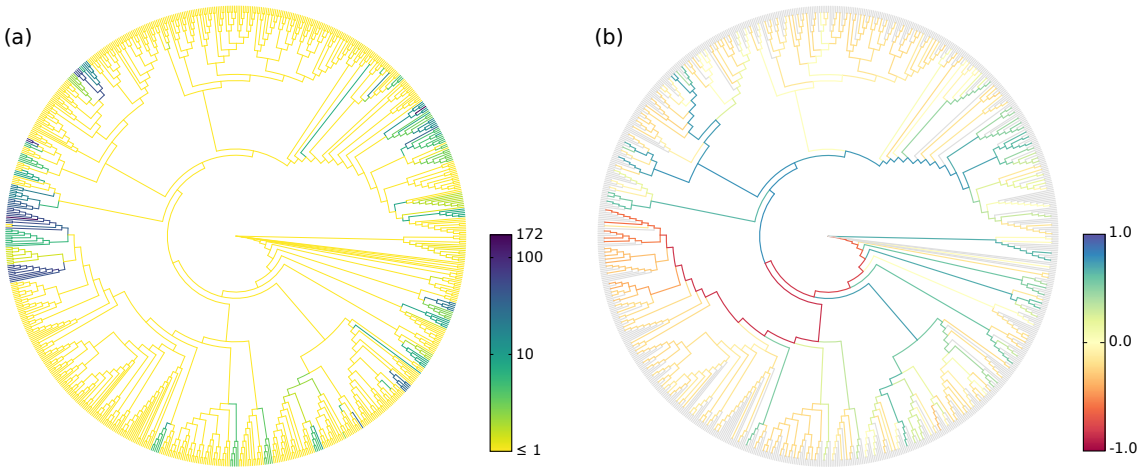

Figure 4.2: Examples of Edge Dispersion and Edge Correlation. We applied our novel visualization methods to the Bacterial Vaginosis (BV) dataset (see Appendix B.1 for details) to compare them to the existing examinations of the data. (a) Edge Dispersion, measured as the standard deviation of the edge masses across samples, logarithmically scaled. (b) Edge Correlation, in form of Spearman's Rank Correlation Coefficient between the edge imbalances and the Nugent score. Tip edges are gray, because they do not have a meaningful imbalance. This example also shows the characteristics of edge masses and edge imbalances: The former highlights individual edges, the latter paths to clades.

4.2.2 Edge Correlation

In addition to the per-edge masses, the Edge Correlation further takes a specific meta-data feature into account, that is, a column of the meta-data matrix. The Edge Correlation is calculated as the correlation between each edge column and the feature column, for example by using the Pearson Correlation Coefficient or Spearman's Rank Correlation Coefficient [69]. This yields a per-edge correlation of the placement masses or imbalances with the meta-data feature, and can again be visualized via color coding of the edges.

It is inexpensive to calculate and hence scales well to large datasets. As typical correlation coefficients are within $[-1.0, 1.0]$, there is again no need for further normalization. This yields a tree where edges or clades with either a high linear or monotonic correlation with the selected meta-data feature are highlighted. Figure 4.2(b) shows an exemplary visualization of this method.

In contrast to Edge PCA [155] that can use meta-data features to annotate samples in its scatter plots (see Section 2.5.5), our Edge Correlation method directly represents the influence of a feature on the branches or clades of the tree. It can thus, for example, help to identify and visualize dependencies between species abundances

and environmental factors such as temperature or nutrient levels. Again, the choice of normalization strategy is important to draw meaningful conclusions. However, the correlation is *not* calculated between samples or sequence abundances. Hence, even when using normalized samples, the pitfalls regarding correlations of compositional data [139] do not apply here.

4.3 Evaluation and Results

4.3.1 BV Dataset

We re-analyzed the Bacterial Vaginosis (BV) dataset (see Appendix B.1 for details) by inferring a tree from the original reference sequence set and conducting phylogenetic placement of the 220 samples. The characteristics of this dataset were already explored in [225] and [155]. We use it here to give exemplary interpretations of our Edge Dispersion and Edge Correlation methods, and to evaluate them in comparison to existing methods.

Figure 4.2 shows our novel visualizations of the BV dataset. Edge Dispersion is shown in Figure 4.2(a), while Figure 4.2(b) shows Edge Correlation with the so-called Nugent score. The Nugent score [181] is a clinical standard for the diagnosis of Bacterial Vaginosis, ranging from 0 (healthy) to 10 (severe illness). The connection between the Nugent score and the abundance of placements on particular edges was already explored in [155], but only visualized indirectly (i.e., not on the RT itself). For example, Figure 6 of the original study plots the first two Edge PCA components colorized by the Nugent score. We recalculated this figure for comparison, and show it later in Figure 5.3(i).

In contrast, our Edge Correlation measure directly reveals the connection between Nugent score and placements on the reference tree: The clade on the left hand side of the tree, to which the red and orange branches lead to, are *Lactobacillus iners* and *Lactobacillus crispatus*, respectively, which were identified in [225] to be associated with a healthy vaginal microbiome. Thus, their presence in a sample is anti-correlated with the Nugent score, which is lower for healthy subjects. The branches leading to this clade are hence colored in red. On the other hand, there are several other clades that exhibit a positive correlation with the Nugent score, that is, were green and blue paths lead to in the figure, again a finding already reported in [225].

Both trees in Figure 4.2 highlight the same parts of the tree: The dark branches with high deviation in Figure 4.2(a) represent clades attached to either highly correlated (blue) or anti-correlated (red) paths Figure 4.2(b). This indicates that edges that have a high dispersion also vary between samples of different Nugent score. This indicates that both methods reveal the clades that are relevant for discriminating samples of this dataset.

We further compared our methods to the visualization of Edge PCA components on the reference tree. To this end, we recalculated Figures 4 and 5 of [155], and visualized them with our color scheme in Figure 4.3 for ease of comparison. They show

the first two components of Edge PCA, mapped back to the RT. The first component, Figure 4.3(a), reveals that the *Lactobacillus* clade represents the axis with the highest heterogeneity across samples, while the second component, Figure 4.3(b), further distinguishes between the two aforementioned clades within *Lactobacillus*. Edge Correlation also highlights the *Lactobacillus* clade as shown in Figure 4.2(b), but does not distinguish further between its sub-clades. This is because a high Nugent score is associated with a high abundance of placements in either of the two relevant *Lactobacillus* clades.

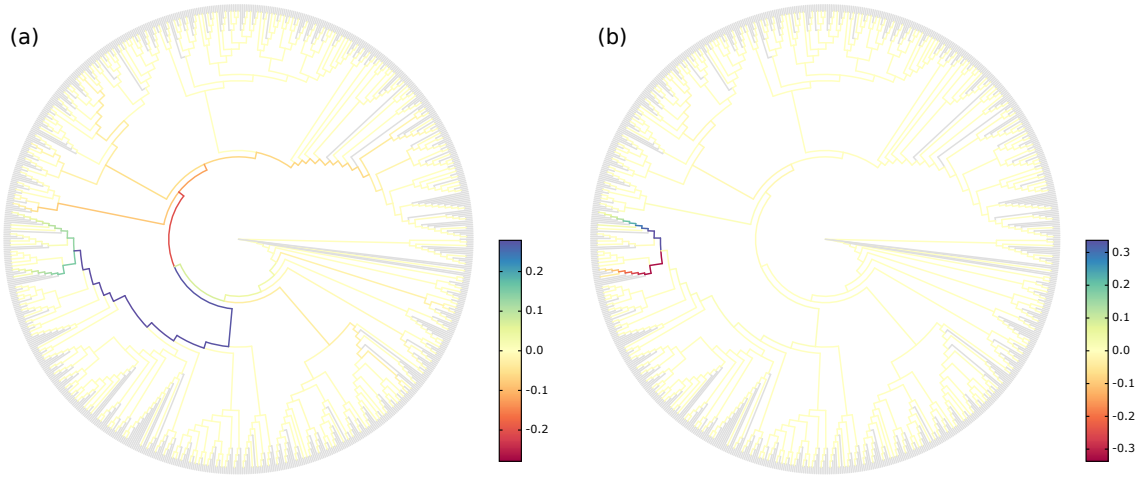

Figure 4.3: Recalculation of the Edge PCA tree visualization. Subfigures (a) and (b) are recalculations of Figures 4 and 5 of [155], respectively. However, we show them here in our coloring scheme in order to facilitate comparison with other figures. The original publication instead uses two colors for a positive and a negative sign of the principal components, and branch width to show their magnitude. Note that the actual sign is arbitrary, as it is derived from principal components.
The figure shows the first two Edge PCA components, visualized on the reference tree. This form of visualization is useful to interpret results such as the Edge PCA projection plot as shown later in Figure 5.1(e). It reveals which edges are mainly responsible for separating the samples into the PCA dimensions. Here, the first principal component in (a) indicates that the main PCA axis separates samples based on the presence of placements in the *Lactobacillus* clade, which is what the blue and green path leads to. The second component in (b) then further distinguishes between two species in this clade, namely *Lactobacillus iners* and *Lactobacillus crispatus*.

Further examples of variants of Edge Dispersion on the BV dataset are shown in Figure 4.4. In Figure 4.4(a), which is linearily scaled, it is striking that one outlier, marked with an arrow, is dominating, thus hiding the values on less variable edges. This outlier occurs at the species *Prevotella bivia* in one of the 220 samples, where 2781 out of 2782 sequences in the sample have placement mass on that branch. Upon close examination, this outlier can also be seen in Figure 1D of [225], but is less apparent there. Thus, our novel visualization can help to detect such outlier

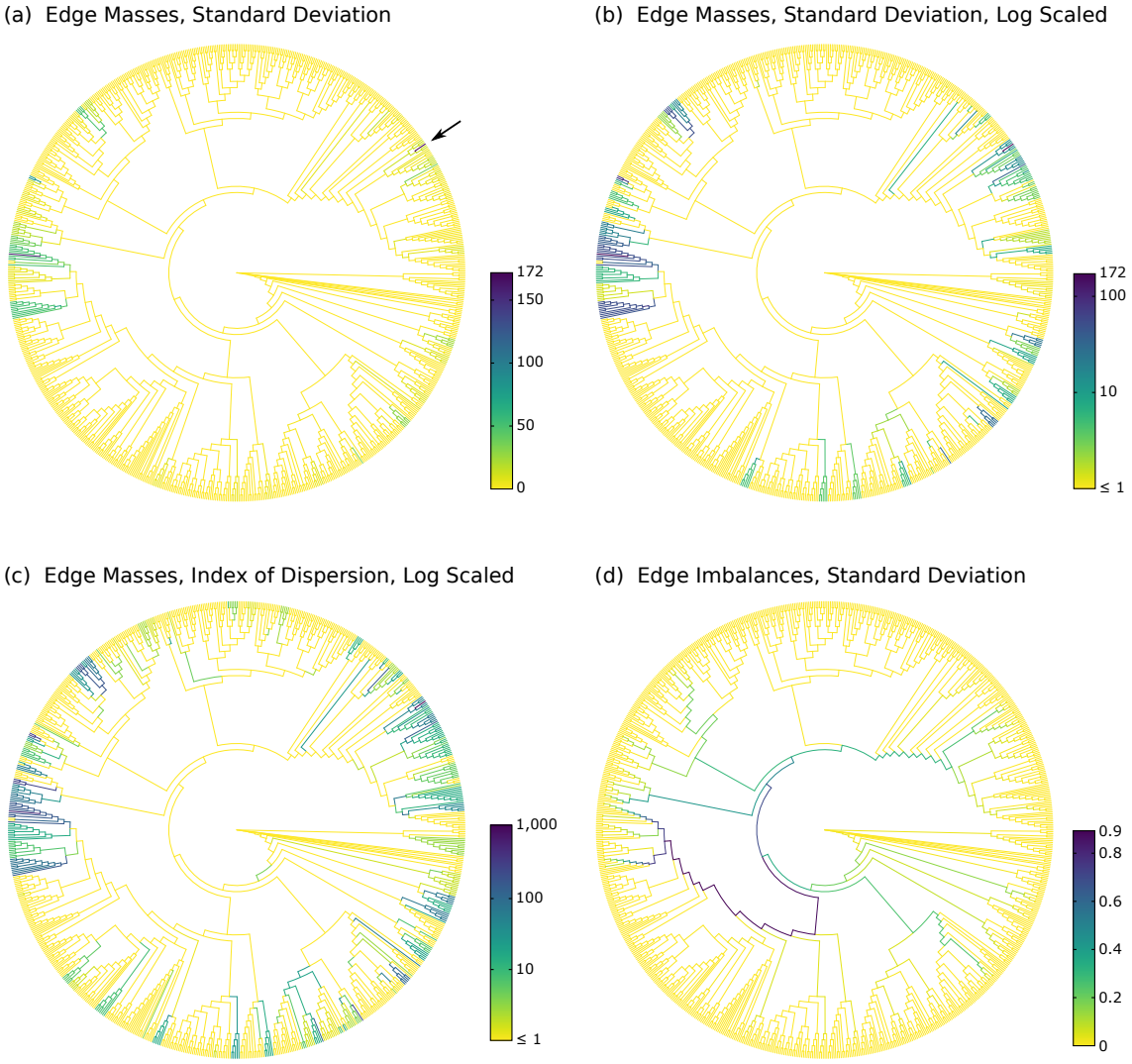

Figure 4.4: Examples of variants of Edge Dispersion. The Figure shows further visualizations of Edge Dispersion on the BV dataset. All subfigures highlight the same branches and clades as found by other methods such as Edge PCA. Subfigure (a) shows the standard deviation of the absolute edge masses, without any further processing. Subfigure (b) is identical to Figure 4.2(a), for comparison, and shows the standard deviation again, but this time using logarithmic scaling, thus revealing more details on the edges with lower placement mass variance. Subfigure (c) shows the Index of Dispersion of the edge masses, that is, the variance normalized by the mean. Hence, edges with a higher number of placements are also allowed to have a higher variance. The figure reveals more details on the edges with lower variance, highlighted in medium green colors. Subfigure (d) shows the standard deviation of edge imbalances. Because we used imbalances of unit mass samples, the values are already normalized. Note that imbalances can be negative; thus, the Index of Dispersion is not applicable to them.

samples. In Figure 4.4(b) and Figure 4.4(c), we used logarithmic scaling instead, in order to reveal more details on the edges with lower placement mass variance. When comparing these two Figures to Figure 4.5, we see that the same clades that exhibit a high correlation or anti-correlation with meta-data there are also highlighted here. There are only few medium values, which indicates that there are two classes of edges: Those which have a high heterogeneity in placement mass (and thus can help to distinguish patients), and those who have almost no placement on them at all. Lastly, Figure 4.4(d) shows Edge Dispersion of the edge imbalances. The path to the *Lactobacillus* clade is again clearly visible, indicating that the placement mass in this clade has a high variance across samples.

In Figure 4.5, we show further examples of variants of Edge Correlation on the BV dataset. All subfigures show red edges or red paths at the *Lactobacillus* clade. This indicates that presence of placements in this clade is anti-correlated with the Nugent score, which is consistent with the findings of [225] and [155]. In other words, presence of *Lactobacillus* correlates with a healthy vaginal microbiome. On the other hand, blue and green edges, which represent positive correlation, are indicative of edges that correlate to Bacterial Vaginosis. The extent of correlation is larger for Spearman's Coefficient, indicating that the correlation is monotonic, but not strictly linear.

Lastly, we conducted Edge Correlation using additional meta-data features that are available for the BV dataset, in order to further confirm the consistency of our methods with existing results. In particular, we visualizes the correlation with Amsel's criteria [4] and the vaginal pH value in Figure 4.6, both of which were already used in [225] as additional indicators of Bacterial Vaginosis. We again found similar correlations compared to the Nugent score.

4.3.2 Tara Oceans Dataset

We analyzed the Tara Oceans (TO) dataset (see Appendix B.3 for details) to provide further exemplary use cases for our visualization methods. To this end, we used the unconstrained *Eukaryota* RT with 2059 taxa as described in Section 3.3.1. The meta-data features of the TO dataset that best lend themselves to our methods are the sensor values for chlorophyll, nitrate, and oxygen concentration, as well as the salinity and temperature of the water samples. Other available meta-data features such as longitude and latitude are available, but would require more involved methods than the ones presented here. This is because geographical coordinates yield pairwise distances between samples, whose integration into our correlation analysis methods is challenging. The Edge Correlation of the 370 samples with the nitrate concentration, the salinity, the chlorophyll concentration, and the water temperature are shown in Figure 4.7.

We selected the *Diatoms* and the *Animals* as two exemplary clades for closer examination of the results. Diatoms are mainly photosynthetic, and thus depend on nitrates as key nutrients [141, 194], which is clearly visible by the high correlation of the clade with the nitrate concentration in Figure 4.7(a). Furthermore, the diatoms

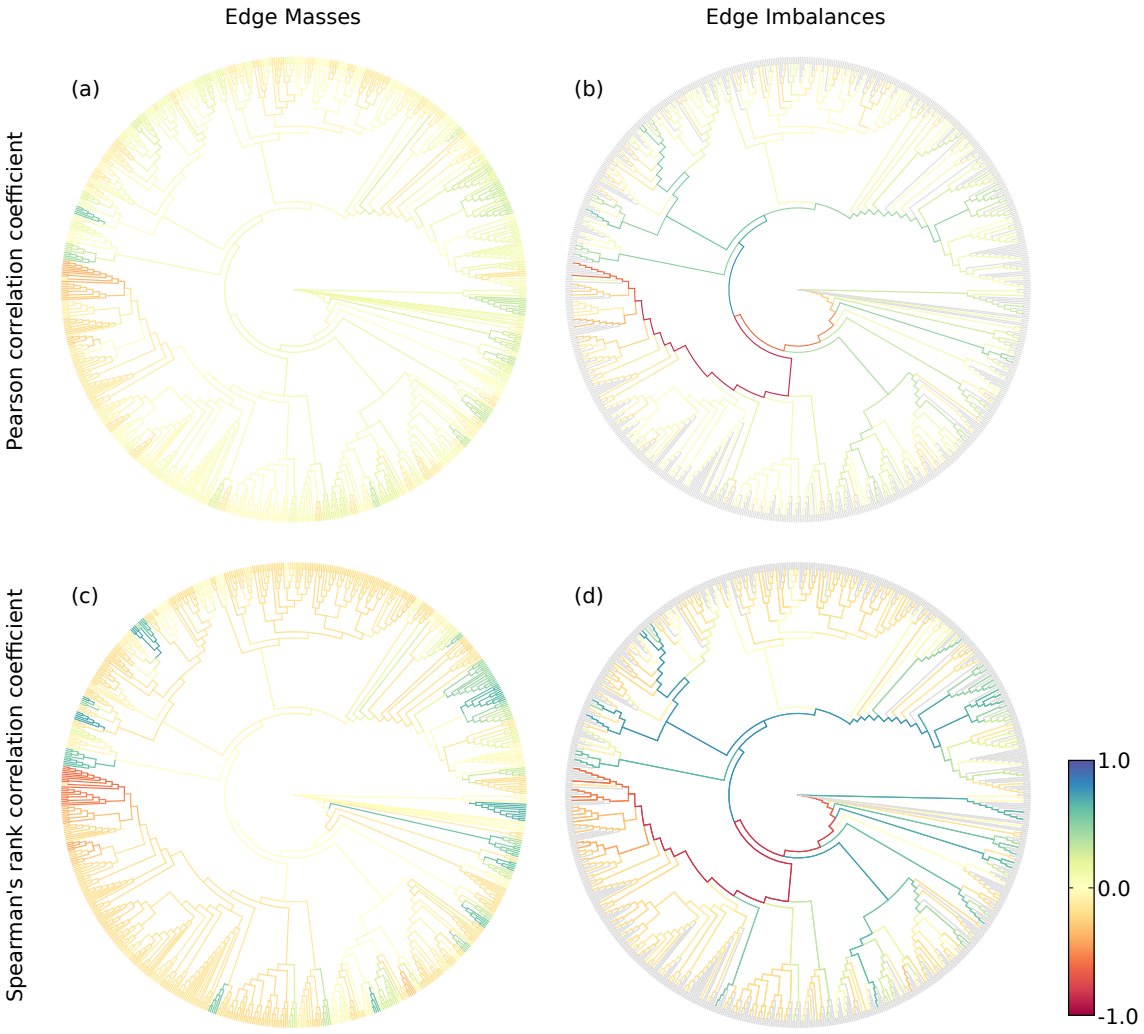

Figure 4.5: Examples of variants of Edge Correlation. The Figure shows the correlation of edge masses and imbalances with the Nugent score on the BV dataset. The Nugent score measures the severeness of Bacterial Vaginosis, and ranges from 0 for healthy subjects to 10 for heavily affected patients. Subfigures (a) and (b) use the Pearson Correlation Coefficient, that is, they show the linear correlation with the meta-data feature, while subfigures (c) and (d) use Spearman's Rank Correlation Coefficient, and thus show monotonic correlations. Subfigure (d) is identical to Figure 4.2(b), for comparison.

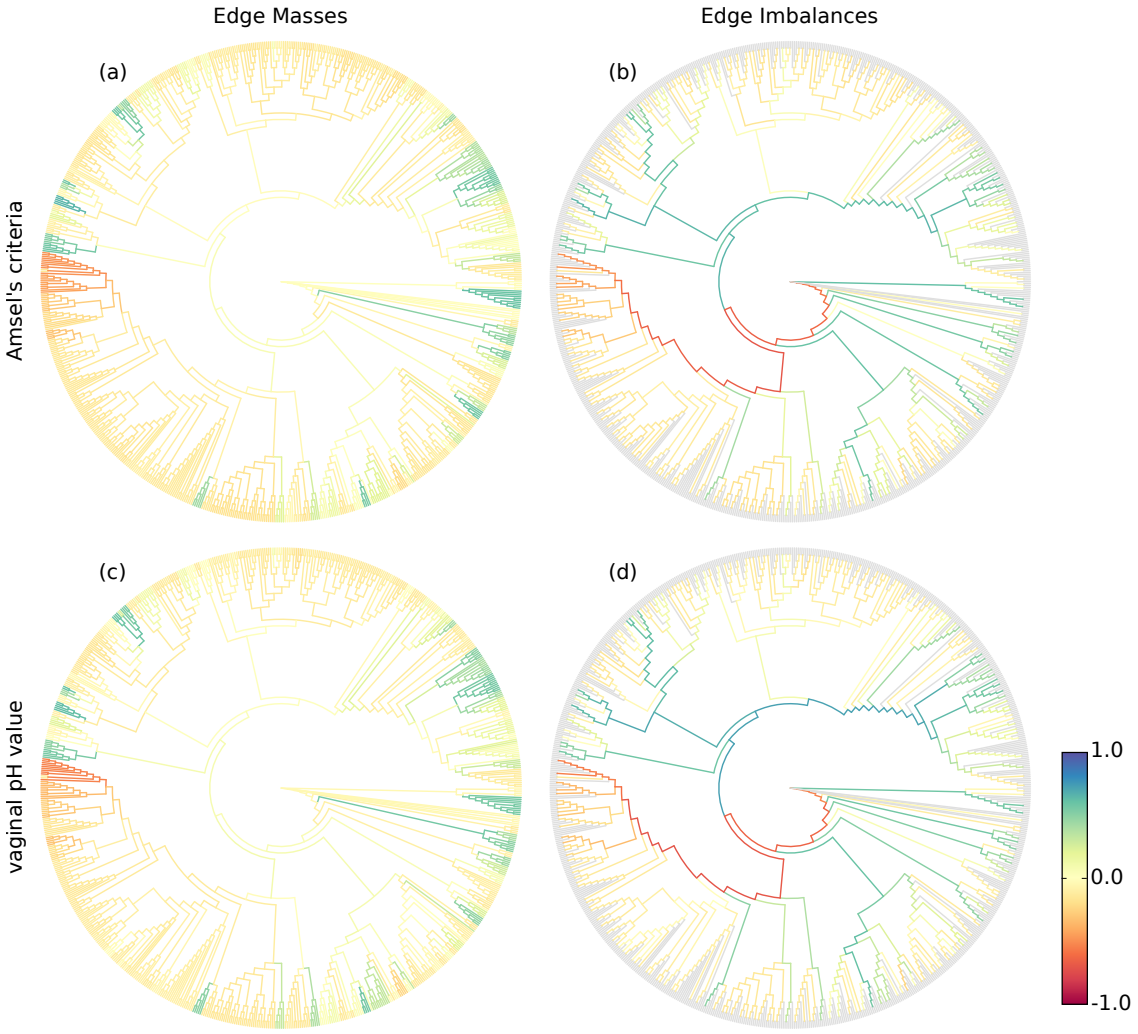

Figure 4.6: Edge Correlation with more meta-data features. Here, we use additional meta-data features of the BV dataset to show that Edge Correlation yields consistent results with existing methods. In particular, we calculated Spearman's Coefficient with Amsel's criteria [4] in subfigures (a) and (b), as well as with the vaginal pH value in subfigures (c) and (d). Both features were also used in [225] as additional indicators of Bacterial Vaginosis. The figures are almost identical to the ones shown in Figure 4.5; that is, they yield results that are consistent with the previously used Nugent score, as well as consistent with existing methods.

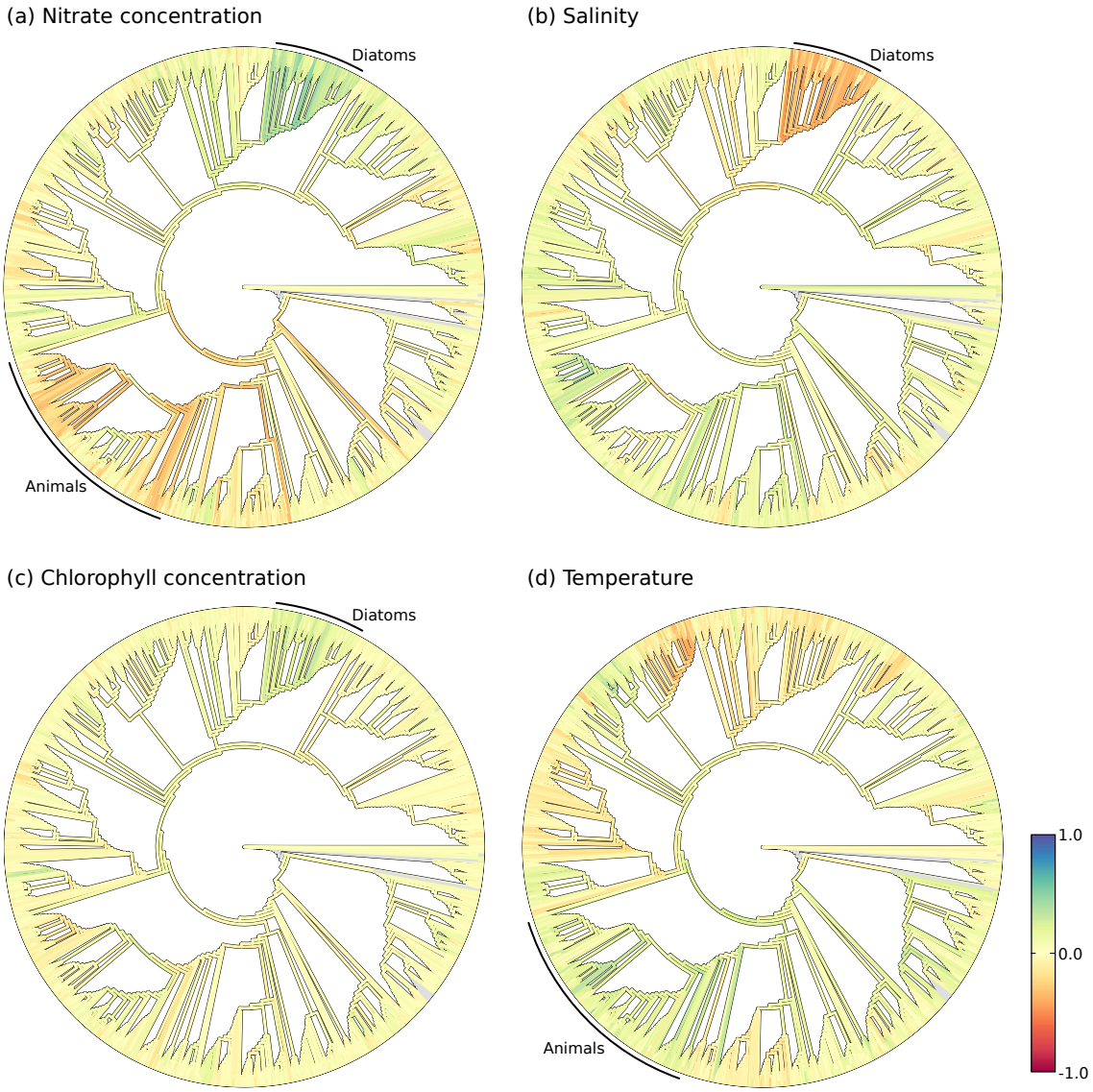

Figure 4.7: Examples of Edge Correlation using Tara Oceans samples. The figure shows the correlation of Tara Oceans sequence placements with (a) the nitrate, (b) the salinity, (c) the chlorophyll, and (d) the temperature sensor data of each sample. The sensor values range from -2.2 to $33.1\,\mu\text{mol/l}$ (nitrate), from 33.2 to 40.2 psu (salt), from -0.02 to $1.55\,\text{mg/m}^3$ (chlorophyll), and from -0.8 to $30.5\,^\circ\text{C}$ (temperature), respectively. The negative nitrate and chlorophyll concentrations are values below the detection limit of the measurement method (pers. comm. with L. Guidi), and hence simply denote low concentrations. We used Spearman's Rank Correlation Coefficient in all subfigures, and examine two exemplary clades, namely the *Animals* and the *Diatoms*, which are marked by arcs around the tree here.

exhibit positive correlation with the chlorophyll concentration Figure 4.7(c), which again is indicative of their photosynthetic behavior. On the other hand, they prefer environments with low salt concentrations, and thus show a high anti-correlation with the salt content Figure 4.7(b). Salinity is a strong environmental factor which heavily affects community structures and species abundances [141], particularly diatoms [194].

The correlations of the animal clade are less pronounced. They exhibit a negative correlation with nitrate Figure 4.7(a), as well as an increase in absolute abundance with higher temperatures Figure 4.7(d). While these findings are not surprising, they show that the method is able to find meaningful relationships in the data.

These findings indicate that the Edge Correlation method is able to identify known relationships. It will therefore also be useful to investigate or discover insights of novel relationships between sequence abundances and environmental parameters.

4.3.3 Performance

Both methods (Edge Dispersion and Edge Correlation) are computationally inexpensive, as they only need a few operations per entry of the input matrices. They are thus applicable to large datasets. The calculation of the above visualizations took about 30 s each, which were mainly required for reading in the data. The required main memory for these computations is also relatively low, and mostly determined by the size of the input matrices, which contain $s \cdot b$ floating point numbers for a dataset of s samples placed on a tree with b branches.

Furthermore, in order to scale to large datasets, we reimplemented Edge PCA (Section 2.5.5), which was originally implemented as a command in the GUPPY program [158]. For the BV dataset with 220 samples (Appendix B.1), GUPPY required 9 min and used 2.2 GB of memory, while our implementation only required 33 s on a single core, using less than 600 MB of main memory. Furthermore, we tested our reimplementation of Edge PCA on the large Human Microbiome Project (HMP) dataset (see Appendix B.4 for details). For this dataset, GUPPY took 11 days and 75.1 GB memory, as it is only single-threaded and seems to have a slow parser for the `jplace` input format (Section 2.5.1), while our implementation needed 7.5 min on 16 cores and used 43.5 GB memory.

4.4 Conclusion and Outlook

The chapter presented two novel methods to explore phylogenetic placement data in order to derive biological and ecological knowledge and unravel new patterns in the data. The methods complement existing analysis tools such as Edge PCA, and yield consistent results on known datasets.

Edge Dispersion is a first exploratory tool that highlights branches of the phylogenetic tree which exhibit variations in the number of placements across samples. It thus allows to identify "interesting" regions of the tree with a high placement heterogeneity. In contrast to Edge Correlation, it can however not explain the reasons of heterogeneity.

Edge Correlation additionally takes meta-data features into account, and identifies branches of the tree that correlate with quantitative features, such as the temperature or the pH value of the environmental samples.

As mentioned above, the methods are currently limited to correlations with simple, singular value meta-data features. In their current form however, they do not allow for more challenging analyses, such as finding pattern and correlations depending on multiple features at once, or taking more complex data into account, such as the geographical distribution of the samples. For example, in biogeographic and ecological studies, one might be interested in questions such as (i) how the regional diversity per area in a rain forest depends on distances between these regions [126], or (ii) how oceanic currents influence species diversity and distribution in the global oceans [234]. While it is unlikely that such questions can be answered in a single visualization, it might still be interesting and helpful to explore more involved methods using phylogenetic placement data to help answering them.

5. Clustering

This chapter is based on the peer-reviewed publication:

- **Lucas Czech** and Alexandros Stamatakis. "Scalable Methods for Post-Processing, Visualizing, and Analyzing Phylogenetic Placements." *PLOS One*, 2018.

Contributions: Lucas Czech... Alexandros Stamatakis...

TODO: distance measures, nhd, simulations, mantel test

5.1 Motivation

Given a set of metagenomic samples, one key question is how much they differ from each other. TODO: see Section 2.5.4

5.2 Phylogenetic k-means

The number of tips in the resulting clustering tree obtained through Squash Clustering is equal to the number n of samples that are being clustered. Thus, for datasets with more than a few hundred samples, the clustering result becomes hard to inspect and interpret visually. We propose a variant of k-means clustering [143] to address this problem, which we call *Phylogenetic k-means*. It uses a similar approach as Squash Clustering, but yields a predefined number of k clusters. It is hence able to work with arbitrarily large datasets. Note that we are clustering samples here, instead of sequences [112]. We discuss choosing a reasonable value for k later.

The underlying idea is to assign each of the n samples to one of k cluster centroids, where each centroid represents the average mass distribution of all samples assigned

to it. Note that all samples and centroids are of the same data type, namely, they are mass distributions on a fixed RT. It is thus possible to calculate distances between samples and centroids, and to calculate their average mass distributions, as described earlier. Our implementation follows Lloyd's algorithm [136], as shown in Algorithm 5.1.

Algorithm 5.1 Phylogenetic k-means

```
initialize k Centroids
while not converged do
    assign each Sample to nearest Centroid
    update Centroids as mass averages of their Samples
return Assignments and Centroids
```

By default, we use the k-means++ initialization algorithm [8] to obtain a set of k initial centroids. It works by subsequent random selection of samples to be used as initial centroids, until k centroids have been selected. In each step, the probability of selecting a sample is proportional to its squared distance to the nearest already selected sample. An alternative initialization is to select samples as initial clusters entirely at random. This is however more likely to yield sub-optimal clusterings [108].

Then, each sample is assigned to its nearest centroid, using the KR distance. Lastly, the centroids are updated to represent the average mass distribution of all samples that are currently assigned to them. This iterative process alternates between improving the assignments and the centroids. Thus, the main difference to normal k-means is the use of phylogenetic information: Instead of euclidean distances on vectors, we use the KR distance, and instead of averaging vectors to obtain centroids, we use the average mass distribution.

The process is repeated until it converges, that is, the cluster assignments do not change any more, or until a maximum number of iterations have been executed. The second stopping criterion is added to avoid the super-polynomial worst case running time of k-means, which however almost never occurs in practice [7, 23].

The result of the algorithm is an assignment of each sample to one of the k clusters. As the algorithm relies on the KR distance, it clusters samples with similar relative abundances. The cluster centroids can be visualized as trees with a mass distribution, analogous to how Squash Clustering visualizes inner nodes of the clustering tree. That is, each centroid can be represented as the average mass distribution of the samples that were assigned to it. This allows to inspect the centroids and thus to interpret how the samples were clustered. Examples of this are shown in Figure 5.4.

The key question is how to select an appropriate k that reflects the number of "natural" clusters in the data. There exist various suggestions in the literature [19, 92, 190, 206, 241, 242]; we assessed the Elbow method [241] as explained in Figure 5.6, which is a straight forward method that yielded reasonable results for

our test datasets. Additionally, for a quantitative evaluation of the clusterings, we used the k that arose from the number of distinct labels based on the available meta-data for the data. For example, the HMP samples are labeled with 18 distinct body sites, describing where each sample was taken from, c.f. Figure 5.2.

5.2.1 Algorithmic Improvements

In each assignment step of the algorithm, distances from all samples to all centroids are calculated, which has a time complexity of $\mathcal{O}(n \cdot k)$. In order to accelerate this step, we can apply branch binning as introduced in Section 2.5.3. For the BV dataset, we found that even using just 2 bins per edge does not alter the cluster assignments. Branch binning reduces the number of mass points that have to be accessed in memory during KR distance calculations; however, the costs for tree traversals remain. Thus, we observed a maximal speedup of 75% when using one bin per branch, see Table 5.1 for details.

Furthermore, during the execution of the algorithm, empty clusters can occur, for example, if k is greater than the number of natural clusters in the data. Although this problem did not occur in our tests, we implemented the following solution: First, find the cluster with the highest variance. Then, choose the sample of that cluster that is furthest from its centroid, and assign it to the empty cluster instead. This process is repeated if multiple empty clusters occur at once.

5.3 Imbalance k-means

We further propose *Imbalance k-means*, which is a variant of k-means that makes use of the edge imbalance transformation, and thus also takes the clades of the tree into account. In order to quantify the difference in imbalances between two samples, we use the euclidean distance between their imbalance vectors (that is, rows of the imbalance matrix). This is a suitable distance measure, as the imbalances implicitly capture the tree topology as well as the placement mass distributions. As a consequence, the expensive tree traversals required for Phylogenetic k-means are not necessary here. The algorithm takes the edge imbalance matrix of normalized samples as input, as shown in Figure 2.10(b), and performs a standard euclidean k-means clustering following Lloyd's algorithm.

This variant of k-means tends to find clusters that are consistent with the results of Edge PCA, as both use the same input data as well as the same distance measure. Furthermore, as the method does not need to calculate KR distances, and thus does not involve tree traversals, it is several orders of magnitude faster than the Phylogenetic k-means. For example, on the HMP dataset, it runs in mere seconds, instead of several hours needed for Phylogenetic k-means; see Section Performance for details.

5.4 Results

We now evaluate our Phylogenetic k-means clustering (which uses edge masses and KR distances) and Imbalance k-means clustering (which uses edge imbalances and

euclidean distances) methods in terms of their clustering accuracy. We used the BV as an example of a small dataset to which methods such as Squash Clustering [155] are still applicable, and the HMP dataset to showcase that our methods scale to datasets that are too large for existing methods.

5.4.1 BV Dataset

We again use the re-analyzed BV dataset to test whether our methods work as expected, by comparing them to the existing analysis of the data in [225] and [155]. To this end, we ran both Phylogenetic k-means and Imbalance k-means on the BV dataset. We chose $k := 3$, inspired by the findings of [225]. They distinguish between subjects affected by Bacterial Vaginosis and healthy subjects, and further separate the healthy ones into two categories depending on the dominating clade in the vaginal microbiome, which is either *Lactobacillus iners* or *Lactobacillus crispatus*. Any choice of $k > 3$ would simply result in smaller, more fine-grained clusters, but not change the general findings of these experiments. An evaluation of the number of clusters using the Elbow method is shown in Figure 5.6. We furthermore conducted Squash Clustering and Edge PCA on the dataset, thereby reproducing previous results, in order to allow for a direct comparison between the methods, see Figure 5.1. The figure shows the results of Squash Clustering, Edge PCA, and two alternative dimensionality reduction methods, colorized by the cluster assignments *PKM* of Phylogenetic k-means (in red, green, and blue) and *IKM* of the Imbalance k-means (in purple, orange, and gray), respectively. We use two different color sets for the two methods, in order to make them distinguishable at first glance. Note that the mapping of colors to clusters is arbitrary and depends on the random initialization of the algorithm.

As can be seen in Figure 5.1(a), Squash Clustering as well as Phylogenetic k-means can distinguish healthy subjects from those affected by Bacterial Vaginosis. Healthy subjects constitute the lower part of the cluster tree. They have shorter branches between each other, indicating the smaller KR distance between them, which is a result of the dominance of *Lactobacillus* in healthy subjects. The same clusters are found by Phylogenetic k-means: As it uses the KR distance, it assigns all healthy subjects with short cluster tree branches to one cluster (shown in red). The green and blue clusters are mostly the subjects affected by the disease.

The distinguishing features between the green and the blue cluster are not apparent in the Squash cluster tree. This can however be seen in Figure 5.1(c), which shows a Multidimensional scaling (MDS) plot of the pairwise KR distances between the samples. MDS [69, 122, 151] is a dimensionality reduction method that can be used for visualizing levels of similarity between data points. Given a pairwise distance matrix, it finds an embedding into lower dimensions (in this case, 2 dimensions) that preserves higher dimensional distances as well as possible. Here, the red cluster forms a dense region, which is in agreement with its short branch lengths in the cluster tree. At the same time, the green and blue cluster are separated in the MDS plot, but form a coherent region of low density, indicating that $k := 3$ might be too

large with Phylogenetic k-means on this dataset. That is, the actual clustering just distinguishes healthy from sick patients (Figure 5.6).

A similar visualization of the pairwise KR distances is shown in Figure 5.1(d). It is a recalculation of Figure 4 in the preprint [156], which did not appear in the final published version [155]. It is a recalculation of Figure 4 of [156]. but can also be found at [157]. The figure shows a standard Principal Components Analysis (PCA) [69, 122] applied to the distance matrix by interpreting it as a data matrix, and was previously used to motivate Edge PCA. However, although it is mathematically sound, the direct application of PCA to a distance matrix lacks a simple interpretation. Again, the red cluster is clearly separated from the rest, but this time, the distinction between the green and the blue cluster is not as apparent.

In Figure 5.1(b), we compare Squash Clustering to Imbalance k-means. Here, the distinction between the two *Lactobacillus* clades can be seen by the purple and orange cluster assignments. The cluster tree also separates those clusters into clades. The separate small group of orange samples above the purple clade is an artifact of the tree ladderization. The diseased subjects are all assigned to the gray cluster, represented by the upper half of the cluster tree. It is apparent that both methods separate the same samples from each other.

Lastly, Figure 5.1(e) compares Imbalance k-means to Edge PCA. The plot is a recalculation of Figure 3 of [156], which also appeared in Figure 6 in [155] and Figure 3 in [225], but colored using our cluster assignments. Because both methods work on edge imbalances, they group the data in the same way, that is, they clearly separate the two healthy groups and the diseased one from each other. Edge PCA forms a plot with three corners, which are colored by the three Imbalance k-means cluster assignments.

In Figure 5.3, we report more details of the comparison of our k-means variants to the dimensionality reduction methods used here. Furthermore, examples visualizations of the cluster centroids are shown in Figure 5.4, which further supports that our methods yield results that are in agreement with existing methods.

5.4.2 HMP Dataset

The HMP dataset is used here as an example to show that our method scales to large datasets. To this end, we used the unconstrained *Bacteria* RT with 1914 taxa as provided by our Automatic Reference Tree method [42]. The tree represents a broad taxonomic range of *Bacteria*, that is, the sequences were not explicitly selected for the HMP dataset, in order to test the robustness of our clustering methods. We then placed the 9192 samples of the HMP dataset with a total of 118 701 818 sequences on that tree, and calculated Phylogenetic and Imbalance k-means on the samples. The freely available meta-data for the HMP dataset labels each sample by the body site were it was taken from. As there are 18 different body site labels, we used $k := 18$. The result is shown in Figure 5.2. Furthermore, in Figure 5.5, we show a clustering of this dataset into $k := 8$ broader body site regions to exemplify the effects of using

different values of k. This is further explored by using the Elbow method as shown in Figure 5.6.

Ideally, all samples from one body site would be assigned to the same cluster, hence forming a diagonal on the plot. However, as there are several nearby body sites, which share a large fraction of their microbiome [101], we do not expect a perfect clustering. Furthermore, we used a broad reference tree that might not be able to resolve details in some clades. Nonetheless, the clustering is reasonable, which indicates a robustness against the exact choice of reference taxa, and can thus by used for distinguishing among samples. For example, stool and vaginal samples are clearly clustered. Furthermore, the sites that are on the surface of the body (ear, nose, and arm) also mostly form two blocks of cluster columns.

5.4.3 Performance

The complexity of Phylogenetic k-means is in $\mathcal{O}(k \cdot i \cdot n \cdot e)$, with k clusters, i iterations, and n samples, and e being the number of tree edges, which corresponds to the number of dimensions in standard euclidean k-means. As the centroids are randomly initialized, the number of iterations can vary; in our tests, it was mostly below 100. For the BV dataset with 220 samples and a reference tree with 1590 edges, using $k := 3$, our implementation ran 9 iterations, needing 35 s and 730 MB of main memory on a single core. For the HMP dataset with 9192 samples and 3824 edges, we used $k := 18$, which took 46 iterations and ran in 2.7 h on 16 cores, using 48 GB memory.

In contrast to this, Imbalance k-means does not need to conduct any expensive tree traversals, and instead operates on compact vectors, using euclidean distances. It is hence several orders of magnitude faster than Phylogenetic k-means. For example, using again $k := 18$ for the HMP dataset, the algorithm executed 75 iterations in 2 s. It is thus also applicable to extremely large datasets.

Furthermore, as the KR distance is used in Phylogenetic k-means as well as other methods such as Squash Clustering, our implementation is highly optimized and outperforms the existing implementation in GUPPY [158] by orders of magnitude (see below for details). The KR distance between two samples has a linear computational complexity in both the number of QSs and the tree size. As a test case, we computed the pairwise distance matrix between sets of samples. Calculating this matrix is quadratic in the number of samples, and is thus expensive for large datasets. For example, in order to calculate the matrix for the BV dataset with 220 samples, GUPPY can only use a single core and required 86 min. Our KR distance implementation in GENESIS is faster and also supports multiple cores. It only needed 90 s on a single core; almost half of this time is used for reading input files. When using 32 cores, the matrix calculation itself only took 8 s. This allows to process larger datasets: The distance matrix of the HMP dataset with 9192 samples placed on a tree with 3824 branches for instance took less than 10 h to calculate using 16 cores in GENESIS. In contrast, GUPPY needed 43 days for this dataset. Lastly, branch binning can be used to achieve additional speedups, as shown in Table 5.1.

5.5 Conclusion and Outlook

Furthermore, we presented adapted variants of the k-means method, which exploit the structure of phylogenetic placement data to identify clusters of environmental samples. The method builds upon ideas such as Squash Clustering and can be applied to substantially larger datasets, as it uses a pre-defined number of clusters. For future exploration, other forms of cluster analyses could be extended to work on phylogenetic placement data, for example, soft k-means clustering [18, 58] or density-based methods [121]. The main challenge when adopting such methods consists in making them phylogeny-aware, that is, to use mass distributions on trees instead of the typical $\mathbb{R}^n$ vectors.

Table 5.1: Effect of Branch Binning on the KR Distance of the HMP Dataset. Here we show the effect of per-branch placement binning on the run-time and on the resulting relative error when calculating the pairwise KR distance matrix between samples, by example of the Human Microbiome Project (HMP) [101, 166] dataset. Because of the size of the dataset (9192 samples) and reference tree (1914 taxa), we executed this evaluation in parallel on 16 cores. The first row shows the baseline performance, that is, without binning. When using fewer bins per branch, the run-time decreases, at the cost of slightly increasing the average relative error. Still, even when compressing the placement masses into only one bin per branch (that is, just using per-branch masses), the average relative error of the KR distances is around 1%, which is acceptable for most applications. However, considering that the run-time savings are not substantially better for a low number of bins, we recommend using a relatively large number of bins, e.g., 32 or more. This is because run-times of KR distance calculations also depend on other effects such as the necessary repeated tree traversals. We also conducted these tests on the BV dataset, were the relative error is even smaller.

Bins	Time (h:mm)	Speedup	Relative Δ
-	9:46	1.00	0.000000
256	6:58	1.40	0.000008
128	6:39	1.47	0.000015
64	6:30	1.50	0.000035
32	6:25	1.52	0.000124
16	6:13	1.57	0.000272
8	6:08	1.59	0.000669
4	6:07	1.60	0.002747
2	6:04	1.61	0.004284
1	5:35	1.75	0.011585

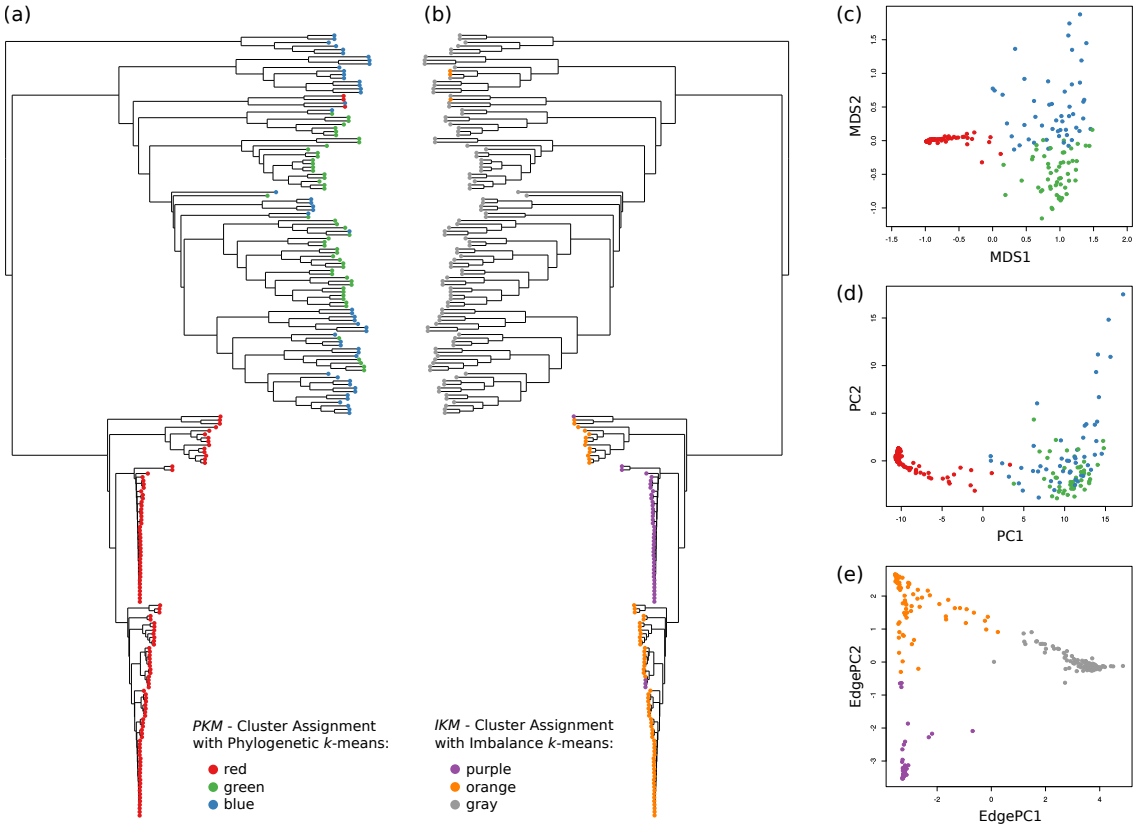

Figure 5.1: Comparison of k-means clustering to Squash Clustering and Edge PCA. We applied our variants of the k-means clustering method to the BV dataset in order to compare them to existing methods. See [225] for details of the dataset and its interpretation. We chose $k := 3$, as this best fits the features of the dataset. For each sample, we obtained two cluster assignments: First, by using Phylogenetic k-means, we obtained the cluster assignment *PKM*. Second, by using Imbalance k-means, we obtained assignment *IKM*. In each subfigure, the 220 samples are represented by colored circles: red, green, and blue show the cluster assignments *PKM*, while purple, orange, and gray show the cluster assignments *IKM*. (a) Hierarchical cluster tree of the samples, using Squash Clustering. The tree is a recalculation of Figure 1(A) of [225]. Each leaf represents a sample; branch lengths are KR distances. We added color coding for the samples, using *PKM*. The lower half of red samples are mostly healthy subjects, while the green and blue upper half are patients affected by Bacterial Vaginosis. (b) The same tree, but annotated by *IKM*. The tree is flipped horizontally for ease of comparison. The healthy subjects are split into two sub-classes, discriminated by the dominating species in their vaginal microbiome: orange and purple represent samples were *Lactobacillus iners* and *Lactobacillus crispatus* dominate the microbiome, respectively. The patients mostly affected by BV are clustered in gray. (c) Multidimensional scaling using the pairwise KR distance matrix of the samples, and colored by *PKM*. (d) Principal component analysis applied to the distance matrix by interpreting it as a data matrix. This is a recalculation of Figure 4 of [156], but colored by *PKM*. (e) Edge PCA applied to the samples, which is a recalculation of Figure 3 of [156], but colored by *IKM*.

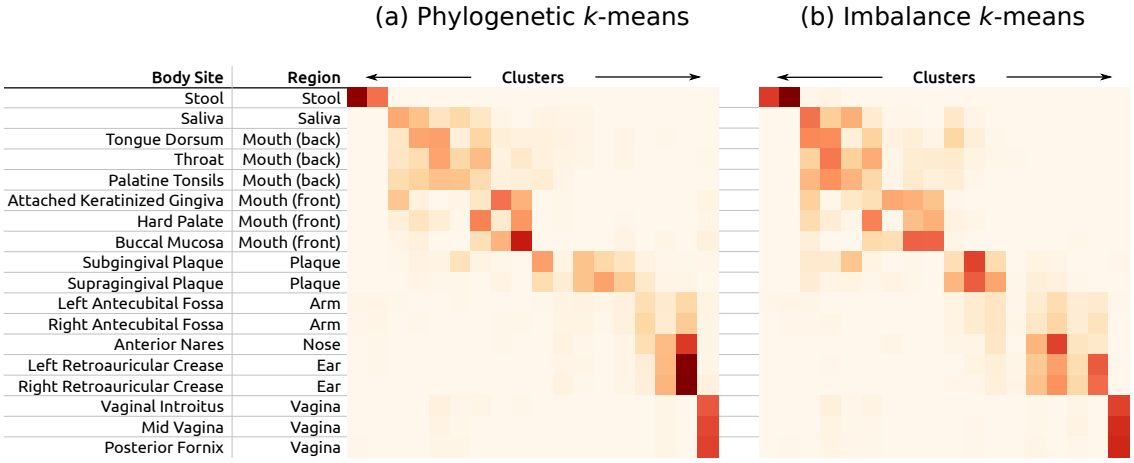

Figure 5.2: k-means cluster assignments of the HMP dataset with $k := 18$. Here, we show the cluster assignments as yielded by Phylogenetic k-means (a) and Imbalance k-means (b) of the HMP dataset. We used $k := 18$, which is the number of body site labels in the dataset, in order to compare the clusterings to this "ground truth". Each row represents a body site; each column one of the 18 clusters found by the algorithm. The color values indicate how many samples of a body site were assigned to each cluster. Similar body sites are clearly grouped together in coherent blocks, indicated by darker colors. For example, the stool samples were split into two clusters (topmost row), while the three vaginal sites were all put into one cluster (rightmost column). However, the algorithm cannot always distinguish between nearby sites, as can be seen from the fuzziness of the clusters of oral samples. This might be caused by our broad reference tree, and could potentially be resolved by using a tree more specialized for the data/region (not tested). Lastly, the figure also lists how the body site labels were aggregated into regions as used in Figure 5.5.
Although the plots of the two k-means variants generally exhibit similar characteristics, there are some differences. For example, the samples from the body surface (ear, nose, arm) form two relatively dense clusters (columns) in (a), whereas those sites are spread across four of five clusters in (b). On the other hand, the mouth samples are more densely clustered in (b).

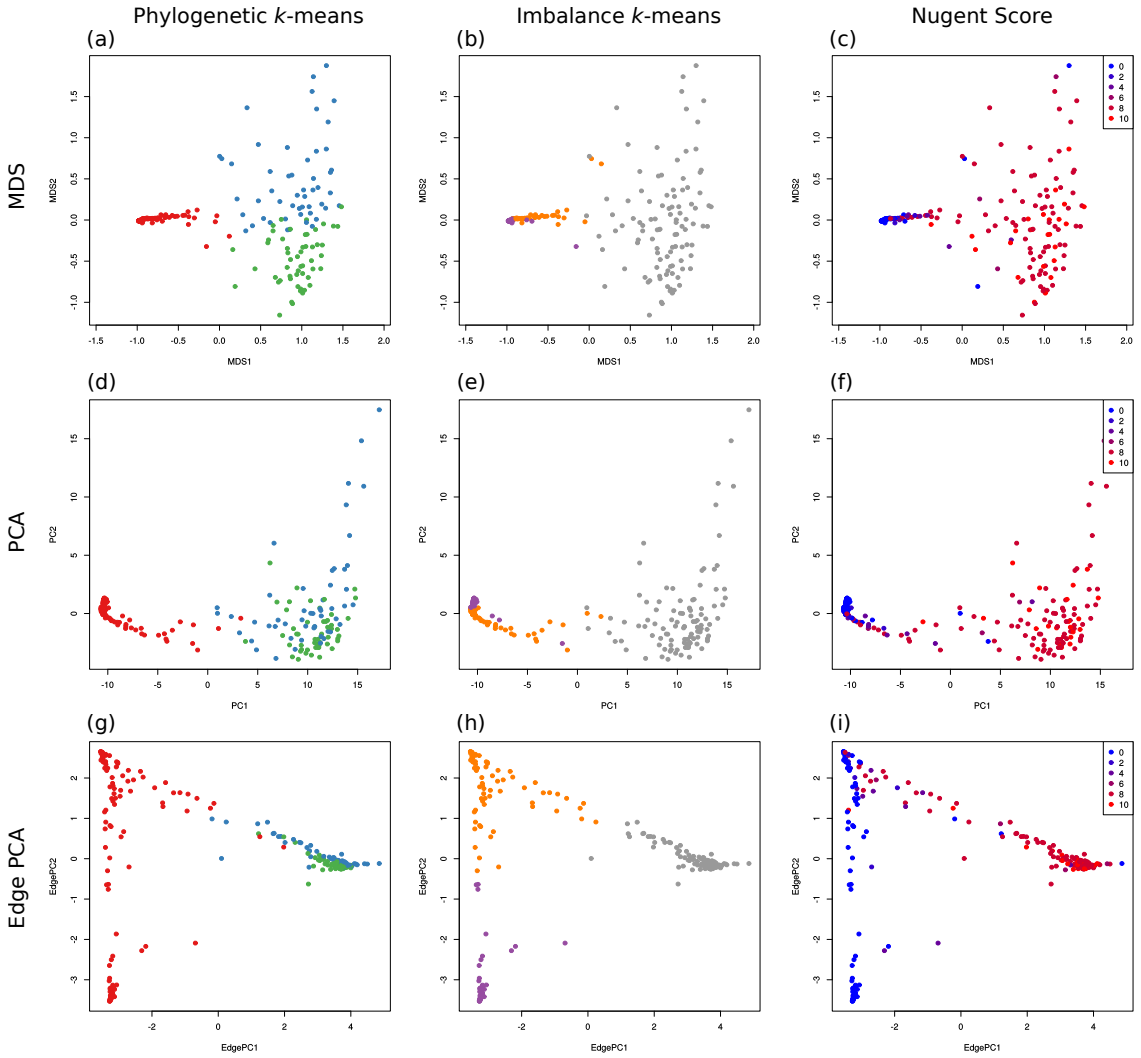

Figure 5.3: Comparison of k-means clustering to MDS, PCA, and Edge PCA. Here, we show and compare the dimensionality reduction methods MDS, PCA, and Edge PCA (one per row). MDS and PCA were calculated on the pairwise KR distance matrix of the BV dataset, Edge PCA was calculated using the placements on the re-inferred RT of the original publication [225]. The plots are colored by the cluster assignments as found by our k-means variants (first two columns), and by the Nugent score of the samples (last column). The Nugent score is included to allow comparison of the health status of patients with the clustering results. (a), (d) and (h) are identical to Figure 5.1(c), (d) and (e) of the main text, respectively. (f) and (i) are recalculations of Figures 4 and 3 of [156], respectively. This figure reveals additional details about how the k-means method works, that is, which samples are assigned to the same cluster. For example, the purple cluster found by Imbalance k-means forms a dense cluster of close-by samples on the left in (b) and (e), which is in accordance with the short branch lengths of this cluster as shown in Figure 5.1(b) of the main text.

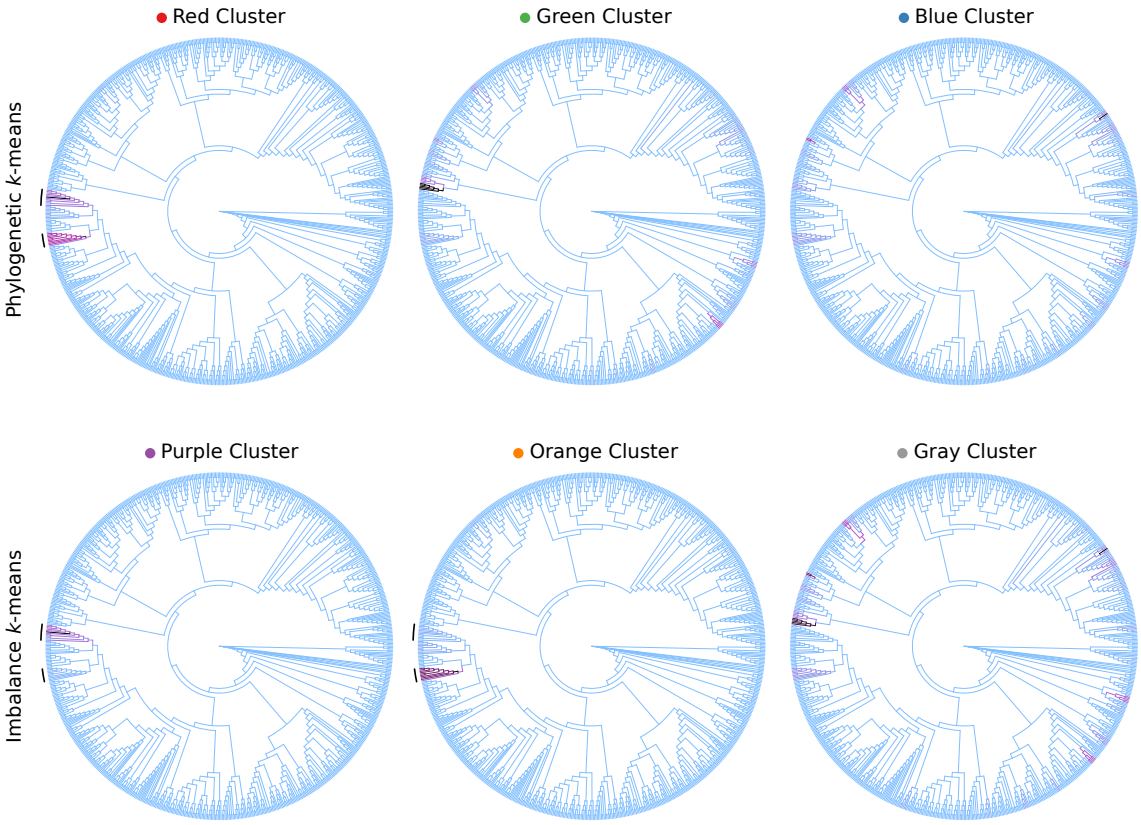

Figure 5.4: Example of k-means cluster centroids visualization. Here we show the cluster centroids as found by our k-means variants using the BV dataset, visualized on the reference tree via color coding. The cluster assignments are the same as in Figure 5.1 of the main text; the first row show the three clusters found by Phylogenetic k-means, the second row the clusters found by Imbalance k-means. Each tree represents one centroid around which the samples were clustered, that is, it shows the combined masses of the samples that were assigned to that cluster. The edges are colored relative to each other, using a linear scaling of light blue (no mass), purple (half of the maximal mass) and black (maximal mass).
As explained in the main text, the samples can be split into three groups: The diseased subjects, which have placement mass in various parts of the tree, as well as two groups of healthy subjects, with placement mass in one of two *Lactobacillus* clades (marked with black arcs on the left of the trees). This grouping is also clearly visible in these trees. The red cluster for example represents all healthy subjects, and thus most of its mass is located in the two *Lactobacillus* clades. The purple and orange clusters on the other hand show a difference in placement mass between those clades. Furthermore, the placement mass of the gray cluster is mostly a combination of the masses of the green and blue cluster, all of which represent diseased subjects. These observations are in accordance with previous findings as explained in the main text.

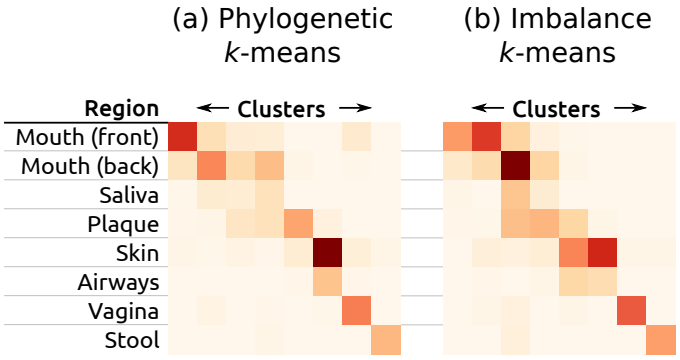

Figure 5.5: Clustering using Phylogenetic k-means on the HMP dataset. k is set to 8, instead of $k := 18$ as in the main text, based on a coarse aggregation of the original body site labels. See Figure 5.1 for the cluster assignment where k is set to the original number of labels; there, we also list how the labels were aggregated. Each row represents a body site; each column one of the 8 clusters. The color values indicate how many samples of a body site were assigned to each cluster. Some of the body sites can be clearly separated, while particularly the samples from the oral region are distributed over different clusters. This might be due to homogeneity of the oral samples.

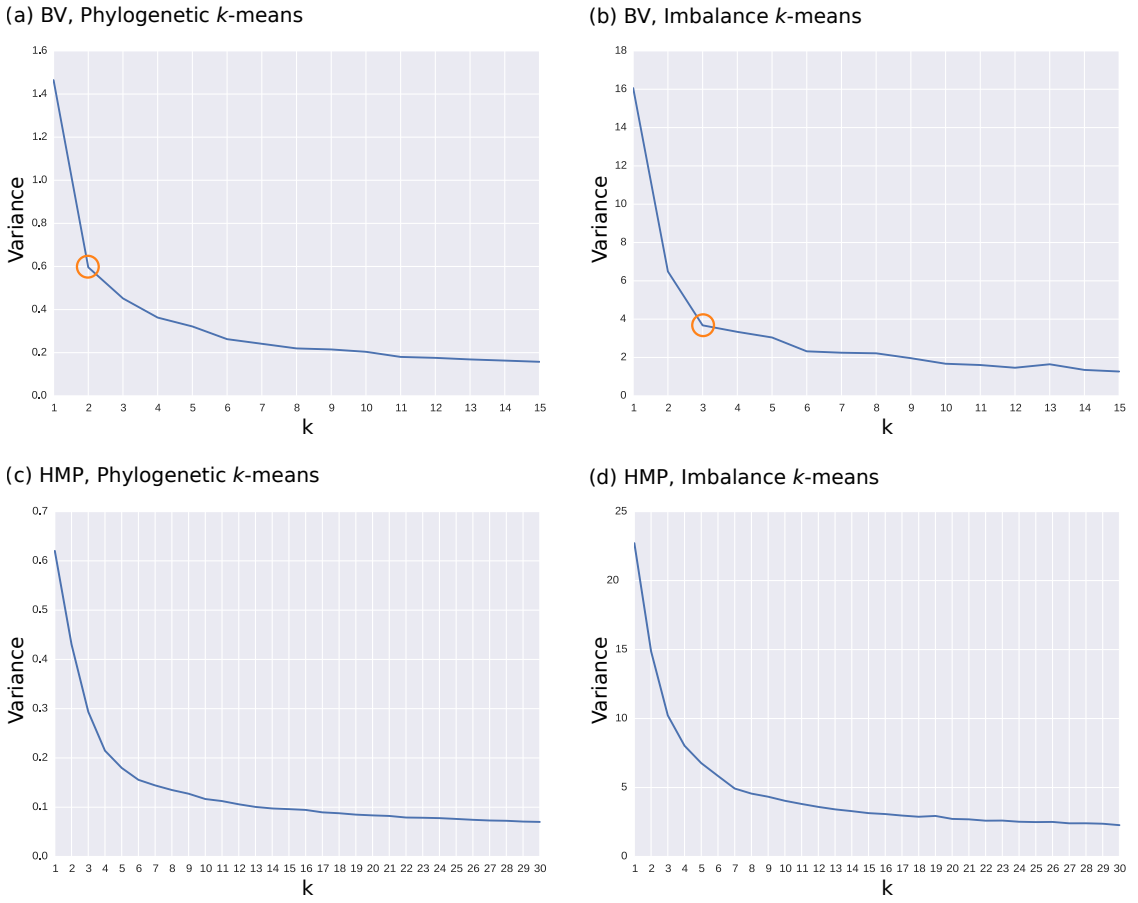

Figure 5.6: Variances of k-means clusters in our test datasets. The figures show the cluster variance, that is, the average squared distance of the samples to their assigned cluster centroids, for different values of k. The first row are clusterings of the BV dataset, the second row of the HMP dataset. They were clustered using Phylogenetic k-means (first column), and Imbalance k-means (second column), respectively. Accordingly, (a) and (c) use the KR distance, while (b) and (d) use the euclidean distance to measure the variance. These plots can be used for the Elbow method in order to find the appropriate number of clusters in a dataset [241]. Low values of k induce a high variance, because many samples exhibit a large distance from their assigned centroid. On the other hand, at a given point, higher values of k only yield a marginal gain by further splitting clusters. Thus, if the data has a natural number of clusters, the corresponding k produces an angle in the plot, called the "elbow".
For example, (a) and (b) exhibit the elbow at $k := 2$ and 3, respectively, which are marked with orange circles. These values are consistent with previous findings, for instance, Figure 5.1: There, Phylogenetic k-means splits the samples into a distinct red cluster and the nearby green and blue clusters, while Imbalance k-means yields three separate clusters in purple, orange, and gray.
For the HMP dataset, the elbow is less pronounced. We suspect that this is due to the broad reference tree not being able to adequately resolve fine-grained differences between samples, see Section B for details. Likely candidates for k are $4 - 6$ for (c) and around 7 for (d). These values are consistent with the number of coherent "blocks" of clusters, which can be observed in Figure 5.2. Clearer results for this dataset might be obtained with other methods for finding "good" values for k, although we did not test them here.

6. Conclusion and Outlook

we showed...

in the future... future work

scrapp: species count estimation, used to discover novelty! (the novelty aspect is mentioned in the foundations chapter!)

A. Supporting Information

Table A.1: IUPAC notation of nucleobases and ambiguity characters. The table lists the character representations of nucleobases and their combinations (which are used to denote ambiguity) as suggested by the IUPAC Commission [102]. The names and symbols for ambiguity characters are chosen based on bio-chemical properties of the nucleobases. See Section 2.2.4 for more information.

Symbol	Description	Represented Bases					Complement
A	**A**denine	A				1	T
C	**C**ytosine		C			1	G
G	**G**uanine			G		1	C
T	**T**hymine				T	1	A
U	**U**racil				U	1	A
W	**W**eak	A			T	2	W
S	**S**trong		C	G		2	S
M	a**M**ino	A	C			2	K
K	**K**eto			G	T	2	M
R	pu**R**ine	A		G		2	Y
Y	p**Y**rimidine		C		T	2	R
B	not A (**B** comes after A)		C	G	T	3	V
D	not C (**D** comes after C)	A		G	T	3	H
H	not G (**H** comes after G)	A	C		T	3	D
V	not T (**V** comes after T and U)	A	C	G		3	B
N	any **N**ucleotide (not a gap)	A	C	G	T	4	N
Z	**Z**ero					0	Z

TODO: maybe add the entropy examples from /home/lucas/Dropbox/HITS/manuscripts/automatic-reference-trees/svg here

B. Empirical Datasets

This chapter is partially based on the peer-reviewed publications:

- **Lucas Czech**, Pierre Barbera, and Alexandros Stamatakis. "Methods for Automatic Reference Trees and Multilevel Phylogenetic Placement." *Bioinformatics*, 2018.
- **Lucas Czech** and Alexandros Stamatakis. "Scalable Methods for Post-Processing, Visualizing, and Analyzing Phylogenetic Placements." *PLOS One*, 2018.

Contributions: Lucas Czech... Alexandros Stamatakis...

We used three real world datasets to evaluate our methods:

- Bacterial Vaginosis (BV) [225]. This small dataset was already analyzed with phylogenetic placement in the original publication. We used it as an example of an established study to compare our results to. It has 220 samples with a total of 15 060 unique sequences.
- Neotropical Soils (NTS) [149]. TODO: check We already analyzed this medium-sized dataset in its publication and found that the sequences are highly diverse and not well covered in existing reference databases. This was a particular challenge for classical approaches for metagenomic studies. It is thus used as an example of a difficult dataset here. It contains 154 samples with 10 567 804 unique sequences in total.
- Tara Oceans (TO) [87, 110, 234]. This world-wide sequencing effort of the open oceans provides a rich set of meta-data, such as geographic location,

temperature, and salinity. Unfortunately, the sample analysis for creating the official data repository is still ongoing. We thus were only able to use 370 samples with 27 697 007 unique sequences.

- Human Microbiome Project (HMP) [101, 166]. This large data repository intends to characterize the human microbiota. It contains 9192 samples from different body sites with a total of 63 221 538 unique sequences. There is additional meta-data such as age and medical history, which is available upon special request. We only used the publicly available meta-data.

- TODO: mouse gut

Details of the datasets (download links, data statistics, data preprocessing, etc.) are provided in TODO: S1 Text. At the time of writing, about one year after we initially downloaded the data, the TO repository has grown to 1170 samples, while the HMP even published a second phase and now comprises 23 666 samples of the 16S region. This further emphasizes the need for scalable methods to analyze such data.

These datasets represent a wide range of environments, number of samples, and sequence lengths. We use them to evaluate our methods and to exemplify which method is applicable to what kind of data. To this end, the sequences of the datasets were placed on appropriate phylogenetic RTs as explained in TODO: S1 Text, in order to obtain phylogenetic placements that our methods can be applied to. In the following, we present the respective results, and also compare our methods to other methods where applicable. As the amount and type of available meta-data differs for each dataset, we could not apply all methods to all datasets. Lastly, we also report the run-time performance of our methods on these data.

The analyses and figures presented here were conducted on distinct reference alignments and trees. Firstly, for the BV dataset, we used the original set of reference sequences, and re-inferred a tree on them. Secondly, for the TO and HMP datasets, we used our Phylogenetic Automatic (Reference) Tree PhAT method [42] to construct sets of suitable reference sequences from the Silva database [196, 269]. We used the 90% threshold consensus sequences; see [42] for details.

For all analyses, we used the following software setup: Unconstrained maximum likelihood trees were inferred using RAxML v8.2.8 [228]. For aligning reads against reference alignments and reference trees, we used a custom MPI wrapper for PaPaRa 2.0 [14, 15], which is available at [13]. We then applied the `chunkify` procedure as explained in [42] to split the sequences into chunks of unique sequences prior to conducting the phylogenetic placement, in order to minimize processing time. Phylogenetic placement was conducted using EPA-ng [10], which is a faster and more scalable phylogenetic placement implementation than RAxML-EPA [16] and pplacer [158]. Lastly, given the per-chunk placement files produced by EPA-ng, we executed the `unchunkify` procedure of [42] to obtain per-sample placement files. These subsequently served as the input data for the methods presented here.

Table B.1: Overview of the dataset dimensions. The "Samples" columns show how many metagenomic samples there were in the originally downloaded data and how many of those we actually used for our experiments after filtering out spurious ones. The columns "Filtered Sample Sizes" show how many sequences each of the filtered samples has. The "Sequence Count" columns show the total number of sequences in the filtered samples, and how many of them are unique. The columns "Sequence Length" show statistics of the length of the sequences. Lastly, the "Chunks" column shows into how many chunks of size 50 000 the samples were distributed.

	Samples		Filtered Sample Sizes			Filtered Sequence Count		Filtere
Dataset	Source	Filtered	Min	Max	Avg	Total	Unique	Min
Bacterial Vaginosis		220				426,612	15,060	
Neotropical Soils		154				50,118,536	10,567,804	
Tara Oceans		370				49,023,231	27,697,007	
Human Microbiome	9,815	9,194				118,701,818	63,221,538	

B.1 Bacterial Vaginosis

We used the Bacterial Vaginosis dataset [225] in order to compare our novel methods to existing ones such as Edge PCA and Squash Clustering [68, 155]. The dataset contains metabarcoding sequences of the vaginal microbiome of 220 women, and was kindly provided by Sujatha Srinivasan. This small dataset with a total of 426 612 query sequences, thereof 15 060 unique, was already analyzed with phylogenetic placement methods in the original publication. We re-inferred the reference tree of the original publication using the original alignment, which contains 797 reference sequences specifically selected to represent the vaginal microbiome. As the query sequences were already prepared, no further preprocessing was applied prior to phylogenetic placement. The available per-sample quantitative meta-data for this dataset comprises the Nugent score [181], the value of Amsel's criteria [4], and the vaginal pH value. We used all three meta-data types in our analyses.

TODO: from art:

For testing the accuracy of our unconstrained *Bacteria* tree on real data, we used a vaginal microbiome dataset of 220 women [225], which was provided by Sujatha Srinivasan. See Figure 3.6 and Figure 3.7 for the respective results. This small dataset with a total of 426 612 query sequences, thereof 15 060 unique, was already analyzed with phylogenetic placement methods in the original publication. We used it as an example of a well-designed study to asses our results using an PhAT as reference tree. In addition to the *Bacteria* PhAT, we re-inferred the reference tree of the original publication using their alignment, again using RAxML 8.2.8 [228]. The query sequences of the dataset were then aligned to our reference tree and alignment, as well as to the reference alignment of the original publication and our

re-inferred tree. For aligning, we used a custom MPI wrapper of PaPaRa 2.0 [14, 15], which is available at [13]. Finally, the query sequences were placed on these trees using EPA-ng [10], and the analyses were subsequently performed as explained in Figure 3.6 and Figure 3.7. TODO: the above is not up to date!

B.2 Neotropical Soils

B.3 Tara Oceans

The Tara Oceans (TO) dataset [87, 110, 234] that we used here contains amplicon sequences of protists, and is available at https://www.ebi.ac.uk/ena/data/view/PRJEB6610. At the time of download, there were 370 samples available with a total of 49 023 231 sequences. As the available data are raw `fastq` files, we followed [146] to generate cleaned per-sample `fasta` files. For this, we used our tool PEAR [273] to merge the paired-end reads; cutadapt [153] for trimming tags as well as forward and reverse primers; and vsearch [204] for filtering erroneous sequences and generating per-sample `fasta` files. We filtered out sequences below 95 bps and above 150 bps, to remove potentially erroneous sequences. No further preprocessing (such as chimera detection) was applied. This resulted in a total of 48 036 019 sequences, thereof 27 697 007 unique. The sequences were then used for phylogenetic placement as explained above. We placed the sequences on the unconstrained *Eukaryota* reference tree obtained via our PhAT method [42], which comprises 2059 taxa, thereof 1795 eukaryotic sequences. The remaining 264 taxa are *Archaea* and *Bacteria*, and were included as a broad outgroup. The TO dataset has a rich variety of per-sample meta-data features; in the context of this paper, we mainly focus on quantitative features such as temperature, salinity, as well as oxygen, nitrate and chlorophyll content of the water. Furthermore, each sample has meta-data features indicating the date, longitude and latitude, depth, etc. of the sampling location. This data might be interesting for further correlation analyses based on geographical information. We did not use them here, as for example longitude and latitude would require a more involved method that also accounts for, e.g., ocean currents. Furthermore, geographical coordinates yield pairwise distances between samples, which are not readily usable with our correlation analysis. Lastly, in order to use features such as the date, that is, in order to analyze samples over time, the same sampling locations would need to be visited at different times during the year, which is not the case for the Tara Oceans expedition.

B.4 Human Microbiome Project

We used the Human Microbiome Project (HMP) dataset [101, 166] for testing the scalability of our methods. In particular, we used the "HM16STR" data of the initial phase "HMP1", which are available from http://www.hmpdacc.org/hmp/. The dataset consists of trimmed 16S rRNA sequences of the `V1V3`, `V3V5`, and `V6V9` regions. The data are further divided into a "by_sample" set and a "healthy" set, which we merged in order to obtain one large dataset, with a total of 9811 samples. We

then removed 10 samples that were larger than 70 MB as well as 605 samples that had fewer than 1500 sequences, because we considered them as defective or unreliable outliers. Finally, we also removed 2 samples that did not have a valid body site label assigned to them. This resulted in a set of 9192 samples containing a total of 118 702 967 sequences with an average length of 413 bps. From these samples, sequences with a length of less than 150 bps as well as sequences longer than 540 bps were removed, as we considered them potentially erroneous. No further preprocessing (such as chimera detection) was applied. This resulted in a total of 116 520 289 sequences, of which 63 221 538 were unique. We then used the unconstrained *Bacteria* tree of our PhAT method [42] for phylogenetic placement. The tree comprises 1914 taxa, thereof 1797 bacterial sequences. The remaining 117 taxa are *Archaea* and *Eukaryota*, and were included as a broad outgroup. Each sample is labeled with one of 18 human body site locations where it was sampled. This is the only publicly available meta-data feature.

TODO: from art:

We used the Human Microbiome Project (HMP) dataset [101, 166] for further testing our methods (see Figure 3.8). In particular, we used the "HM16STR" data of their initial phase "HMP1", which are available from http://www.hmpdacc.org/hmp/. The dataset consists of trimmed 16S rRNA sequences of the `V1V3`, `V3V5`, and `V6V9` regions. Each sample of the dataset is labeled with the human body site where it was obtained. See Table B.2 for an overview of those labels. The data are further divided into a "by_sample" set and a "healthy" set, which we merged in order to obtain one large dataset, with a total of 9811 samples. We then removed 10 samples that were larger than 70 MB as well as 605 samples that had fewer than 1500 sequences, because we considered them as outliers, and finally 2 samples that did not have a valid body site label assigned to them. This resulted in a set of 9192 samples containing a total of 118 702 967 sequences with an average length of 413 bps. From these samples, sequences with a length less than 150 bps as well as sequences longer than 540 bps were removed. No further preprocessing (e.g., chimera detection) was applied. This resulted in a total of 116 520 289 sequences, of which 63 221 538 were unique. These were split into 1265 chunks of size 50 000 each, which were subsequently aligned to and placed on the unconstrained *Bacteria* tree with 2059 tips using the steps explained above. The chunk placements were then transformed again into per-sample placement files, before finally running the steps explained in Figure 3.8.

B.5 Mouse Gut

This dataset was used for the eval of tax assign in Section 3.3.4

CAMI Challenge [215]. The CAMI Challenge is a community-driven effort to assess taxonomic profiling methods using a common set of benchmark data sets.

we utilized the *mouse gut* data set of the 2nd CAMI Challenge [26].

More specifically, we used the unpaired HiSeq reads of the mouse gut data set from CAMI, which comprises 64 samples of simulated reads. The preprocessing involved

Table B.2: HMP Dataset Overview. The table lists the 19 body site labels used by the Human Microbiome Project (HMP) [101, 166]. We used this dataset to evaluate the applicability of typical analysis methods for phylogenetic placement using our PhATs, see Section B and Figure 3.8 for details. In order to simplify the visualization in Figure 3.8, we summarized some of the labels into eight location regions, as shown in the second column. The last column lists how many samples from each body site were used in our evaluation.

Body Site	Region	Samples
Tongue Dorsum	Mouth (back)	610
Palatine Tonsils	Mouth (back)	599
Throat	Mouth (back)	638
Attached Keratinized Gingiva	Mouth (front)	600
Hard Palate	Mouth (front)	566
Buccal Mucosa	Mouth (front)	597
Saliva	Saliva	529
Supragingival Plaque	Plaque	608
Subgingival Plaque	Plaque	595
Anterior Nares	Airways	541
Left Retroauricular Crease	Skin	596
Right Retroauricular Crease	Skin	604
Left Antecubital Fossa	Skin	290
Right Antecubital Fossa	Skin	328
Stool	Stool	600
Vaginal Introitus	Vagina	292
Mid Vagina	Vagina	298
Posterior Fornix	Vagina	301
Sum		9192

read de-interleaving following [254], paired-end read merging using PEAR [273], as well as quality filtering and conversion to `fasta` using VSEARCH2 [204]. This yielded a total of 800 341 409 reads. As our trees are based on small ribosomal subunit sequences, we also performed read filtering to obtain reads from the 16S rDNA region (see Section 2.2.2). This filtering was performed using the protocol of [138], which relies on HMMER [62, 63], and respective profiles for the 16S rDNA locus. We performed a global identity based de-replication step on the resulting reads that yielded 616 405 query sequences. We aligned these query sequences to our *Bacteria* reference alignment using PaPaRa 2.0 [14, 15]. We then performed phylogenetic placement of the aligned query sequences onto the unconstrained and constrained reference trees, respectively, using EPA-ng [10]. We performed de-de-replication to obtain per-sample data again, resulting in 64 `jplace` files (one per

original sample) with placements of the 16S rDNA sequences, for each of the two trees.

Finally, we performed taxonomic assignment and taxonomic profiling of the per-sample results using the `assign` command implemented in GAPPA, which works analogously to the method used in SATIVA [119]. Its basic steps are described in Appendix C.

C. Pipeline and Implementation

This chapter is based on the peer-reviewed publications:

- **Lucas Czech**, Pierre Barbera, and Alexandros Stamatakis. "Methods for Automatic Reference Trees and Multilevel Phylogenetic Placement." *Bioinformatics*, 2018.
- **Lucas Czech** and Alexandros Stamatakis. "Scalable Methods for Post-Processing, Visualizing, and Analyzing Phylogenetic Placements." *PLOS One*, 2018.

Contributions: Lucas Czech... Alexandros Stamatakis...

TODO: add genesis and gappa logos. because we can.

The methods described here are implemented in our tool GAPPA, which is freely available under GPLv3 at http://github.com/lczech/gappa. GAPPA internally uses our C++ library GENESIS, which offers functionality for working with phylogenies and phylogenetic placement data, and also contains methods to work with taxonomies, sequences and many other data types. GENESIS is also freely available under GPLv3 at http://github.com/lczech/genesis.

GAPPA offers a command line interface for conducting typical tasks when working with phylogenetic placements. The methods that we described here are implemented via the following sub-commands:

- `dispersion`: The command takes a set of jplace files (called samples), and calculates and visualizes the Edge Dispersion per edge of the reference tree.

- `correlation`: The command takes a set of jplace samples, as well as a table containing metadata features for each sample. It then calculates and visualizes the Edge Correlation with the metadata features per edge of the reference tree.
- `phylogenetic-kmeans` and `imbalance-kmeans`: Performs k-means clustering of a set of jplace files according to our methods.
- `squash` and `edgepca`: Reimplementations of the two existing methods [68, 155].

These are the GAPPA commands that are relevant for this paper. The tool also offers additional commands that are useful for phylogenetic placement data, such as visualization or filtering. At the time of writing this manuscript, GAPPA is under active development, with more functions planned in the near future. Lastly, we provide prototype implementations, scripts, data, and other tools used for the tests and figures in this paper at http://github.com/lczech/placement-methods-paper.

TODO: ART:

An implementation of the methods described here is freely available in our tool GAPPA, which is published under GPLv3 at http://github.com/lczech/gappa. GAPPA is based on our C++ library GENESIS, which offers functionality concerning phylogenies and phylogenetic placement data, but also has functions to work with sequences, taxonomies and many other data types. GENESIS is also published under GPLv3 and is available at http://github.com/lczech/genesis.

GAPPA offers a command line interface for typical tasks of working with phylogenetic placements. The methods described in this paper are implemented via four subcommands:

- `phat`: Phylogenetic Automatic (Reference) Tree method. The command expects a taxonomy file and a sequence file of a sequence database, e.g., SILVA [196, 269], as well as the target number of consensus sequences to be generated for the intended phylogeny. The result is a `fasta` file with consensus sequences representing taxonomic clades. The command can be further customized, e.g., by changing the consensus sequence method, using only a specified subclade of the taxonomy for running the algorithm, as well as several detail settings for the method. It can also output additional info files that allow to inspect details of the calculations, like the number of sequences and their entropy per clade.
- `extract`: Extract/collect placements in specific sub-clades of the tree. The command performs the main step of the multilevel placement approach. Its input is a set of `jplace` files containing placements on the backbone tree, as well as a file listing the clade name that each taxon of the backbone tree belongs to. For each clade, it then writes a new `jplace` file, containing all queries that were placed in that clade with more than a customizable threshold of their

placement mass.
Furthermore, if provided with the sequence files that were used to make the input `jplace` files, the corresponding sequence of each query are also written to `fasta` files per clade. Thus, a per-clade collection of sequences is created, where each result file contains the sequences that were placed in this clade of the backbone tree. These can then be used for the second level placement on separate clade-specific trees.

- `chunkify`: Split a set of `fasta` files into chunks of equal size, and write abundance maps. The command re-names the sequences using a configurable hash function (MD5, SHA1 or SHA256), and de-duplicates across all input sequences. Its output are chunk files of sequences, as well as an abundance map file for each input sequences file. The sequence chunk files can then be used to perform phylogenetic placement to obtain per-chunk `jplace` files.

- `unchunkify`: Take the per-chunk `jplace` files as well as the abundance map files, and generate a `jplace` for each original sequence file, including the correct abundances. This command is the second step of the `chunkify` command, and reverts its effect, so that the resulting `jplace` files are as if they were created using the original sequence files.

- `assign`: Perform taxonomic assignment using phylogenetic placements. While this is not the main focus of this work, we briefly introduce this method here. The command uses a taxonomic labeling of the tips of the reference tree to annotate all inner branches of the tree with the longest common taxonomic label for the induced subtree of the inner branch, in analogy to Sativa [119]. Then, each query sequence in the provided `jplace` files is taxonomically assigned according to the labels of the branches where it does have placement mass. This can subsequently either be used for taxonomic assignment of the query sequences themselves, or to obtain a taxonomic profile of one or more samples.

These are the commands of gappa relevant for this paper, but it also offers more commands that are useful when working with phylogenetic placements. For details on the commands, and additional potentially useful commands, see the gappa documentation at https://github.com/lczech/gappa/wiki. At the time of writing, it is under active development, and more functions are planned for the near future. Furthermore, we provide prototype implementations, scripts, data and other tools used for the tests and figures in this paper at http://github.com/lczech/placement-methods-paper.

D. List of Publications

unieuk: [17] art: [42] pppp: [43]

TODO: workshops and conferences

1. **L. Czech**, S. Berger, D. Krompaß, J. Zhang, P. Kapli, P. Pavlidis and A. Stamatakis. Evolutionary Placement of Short Reads - Methods, Applications, and Visualization. Poster at EMBO/EMBL Symposium: A New Age of Discovery for Aquatic Microeukaryotes, Heidelberg, Germany, January 2016, and at Hellenic Bioinformatics Conference (HBio) 2016, Thessaloniki, Greece, November 2016.

2. F. Mahé, C. de Vargas, D. Bass, **L. Czech**, A. Stamatakis, E. Lara, D. Singer, J. Mayor, J. Bunge, S. Sernaker, T. Siemensmeyer, I. Trautmann, S. Romac, C. Berney, A. Kozlov, E. A. D. Mitchell, C. V. W. Seppey, E. Egge, G. Lentendu, R. Wirth, G. Trueba, and M. Dunthorn. Parasites dominate hyperdiverse soil protist communities in Neotropical rainforests. *Nature Ecology & Evolution*, vol. 1, no. 4, p. 0091, 2017.

3. **L. Czech**, J. Huerta-Cepas, and A. Stamatakis. A Critical Review on the Use of Support Values in Tree Viewers and Bioinformatics Toolkits. *Molecular Biology and Evolution*, vol. 17, no. 4, pp. 383–384, 2017.

4. T. Flouri, J. Zhang, **L. Czech**, K. Kobert, A. Stamatakis. An efficient approach to merging paired-end reads and the incorporation of uncertainties. Chapter in Algorithms for Next-Generation Sequencing Data: Techniques, Approaches and Applications. 1st ed., M. Elloumi, Ed. Springer International Publishing AG, 2017, pp. 299–326.

5. P. Barbera, A. Kozlov, T. Flouri, D. Darriba, **L. Czech** and A. Stamatakis. Massively Parallel Evolutionary Placement of Genetic Sequences. Poster at ISC 2017 PhD Symposium, Frankfurt am Main, Germany, June 2017.

6. C. Berney, A. Ciuprina, S. Bender, J. Brodie, V. Edgcomb, E. Kim, J. Rajan, L. W. Parfrey, S. Adl, S. Audic, D. Bass, D. A. Caron, G. Cochrane, **L. Czech**, M. Dunthorn, S. Geisen, F. O. Glöckner, F. Mahé, C. Quast, J. Z. Kaye, A. G. B. Simpson, A. Stamatakis, J. del Campo, P. Yilmaz, and C. de Vargas. UniEuk : Time to Speak a Common Language in Protistology! *Journal of Eukaryotic Microbiology*, vol. 38, no. 1, pp. 42–49, 2017.

7. X. Zhou, S. Lutteropp, **L. Czech**, A. Stamatakis, M. von Looz, A. Rokas. Quartet-based computations of internode certainty provide accurate and robust measures of phylogenetic incongruence. *bioRxiv*, 168526, 2017.

8. P. Barbera, A. Kozlov, **L. Czech**, B. Morel, A. Stamatakis. EPA-ng: Massively Parallel Evolutionary Placement of Genetic Sequences. *bioRxiv*, 291658, 2018.

9. D. Bass, **L.Czech**, B. Williams, C. Berney, M. Dunthorn, F. Mahe, G. Torruella, G. Stentiford and T. Williams. Clarifying the Relationships between Microsporidia and Cryptomycota. *Journal of Eukaryotic Microbiology*, 2018.

10. **L. Czech**, A. Stamatakis. Methods for Automatic Reference Trees and Multilevel Phylogenetic Placement. *bioRxiv*, 299792, 2018.

11. **L. Czech**, A. Stamatakis. Scalable Methods for Post-Processing, Visualizing, and Analyzing Phylogenetic Placements. *bioRxiv*, 346353, 2018.

12. TODO: 1KITE

13. TODO: long reads

14. TODO: swarm 3?

15. TODO: genesis and gappa

Bibliography

[1] K. Abarenkov, R. Henrik Nilsson, K.-H. Larsson, I. J. Alexander, U. Eberhardt, S. Erland, K. Høiland, R. Kjøller, E. Larsson, T. Pennanen, and Others. The UNITE database for molecular identification of fungi–recent updates and future perspectives. *New Phytologist*, 186(2):281–285, 2010.

[2] J. Aitchison. *The statistical analysis of compositional data*. Chapman and Hall London, 1986.

[3] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic Local Alignment Search Tool. *Journal of Molecular Biology*, 215(3):403–410, 1990.

[4] R. Amsel, P. A. Totten, C. A. Spiegel, K. C. S. Chen, D. Eschenbach, and K. K. Holmes. Nonspecific vaginitis: Diagnostic Criteria and Microbial and Epidemiologic Associations. *The American Journal of Medicine*, 74(1):14–22, 1983.

[5] S. Anderson. Shotgun DNA sequencing using cloned DNase I-generated fragments. *Nucleic Acids Research*, 9(13):3015–3027, 1981.

[6] J. Archie, W. H. Day, W. Maddison, C. Meacham, F. J. Rohlf, D. Swofford, and J. Felsenstein. The Newick tree format, 1986. Online: http://evolution.genetics.washington.edu/phylip/newicktree.html. Accessed: 2015-07-26.

[7] D. Arthur and S. Vassilvitskii. How Slow is the K-means Method? In *Proceedings of the Twenty-second Annual Symposium on Computational Geometry*, SCG '06, pages 144–153, New York, NY, USA, 2006. ACM.

[8] D. Arthur and S. Vassilvitskii. k-means++ : The Advantages of Careful Seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.*, pages 1027–1035. Society for Industrial and Applied Mathematics Philadelphia, PA, USA, 2007.

[9] M. Balvočiūtė and D. H. Huson. SILVA, RDP, Greengenes, NCBI and OTT — how do these taxonomies compare? *BMC Genomics*, 18(2):114, 2017.

[10] P. Barbera, A. M. Kozlov, L. Czech, B. Morel, and A. Stamatakis. EPA-ng: Massively Parallel Evolutionary Placement of Genetic Sequences. *bioRxiv*, 2018.

[11] J. M. S. Bartlett and D. Stirling. A Short History of the Polymerase Chain Reaction. pages 3–6, 2003.

[12] D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and E. W. Sayers. GenBank. *Nucleic Acids Research*, 37(Database):D26–D31, 2009.

[13] S. Berger and L. Czech. PaPaRa 2.0 with MPI, 2016. Online: https://github.com/lczech/papara_nt. Accessed: 2017-11-04.

[14] S. Berger and A. Stamatakis. Aligning short reads to reference alignments and trees. *Bioinformatics*, 27(15):2068–2075, 2011.

[15] S. Berger and A. Stamatakis. PaPaRa 2.0: A Vectorized Algorithm for Probabilistic Phylogeny-Aware Alignment Extension. Technical report, Heidelberg Institute for Theoretical Studies, Heidelberg, 2012.

[16] S. Berger, D. Krompass, and A. Stamatakis. Performance, accuracy, and web server for evolutionary placement of short sequence reads under maximum likelihood. *Systematic Biology*, 60(3):291–302, 2011.

[17] C. Berney, A. Ciuprina, S. Bender, J. Brodie, V. Edgcomb, E. Kim, J. Rajan, L. W. Parfrey, S. Adl, S. Audic, D. Bass, D. A. Caron, G. Cochrane, L. Czech, M. Dunthorn, S. Geisen, F. O. Glöckner, F. Mahé, C. Quast, J. Z. Kaye, A. G. B. Simpson, A. Stamatakis, J. del Campo, P. Yilmaz, and C. de Vargas. UniEuk : Time to Speak a Common Language in Protistology! *Journal of Eukaryotic Microbiology*, 38(1):42–49, 2017.

[18] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Advanced applications in pattern recognition. Plenum Press, 1981.

[19] H. Bischof, A. Leonardis, and A. Selb. MDL Principle for Robust Vector Quantisation. *Pattern Analysis & Applications*, 2(1):59–72, 1999.

[20] B. E. Blaisdell. A measure of the similarity of sets of sequences not requiring sequence alignment. *Proceedings of the National Academy of Sciences*, 83(14): 5155–5159, 1986.

[21] M. Blaxter, J. Mann, T. Chapman, F. Thomas, C. Whitton, R. Floyd, and E. Abebe. Defining operational taxonomic units using DNA barcode data. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 360(1462):1935–43, 2005.

[22] S. A. Bloom. Similarity Indices in Community Studies: Potential Pitfalls. *Marine Ecology Progress Series*, 5(2):125–128, 1981.

[23] L. Bottou and Y. Bengio. Convergence properties of the k-means algorithms. In *Advances in neural information processing systems*, pages 585–592, 1995.

[24] J. R. Bray and J. T. Curtis. An ordination of the upland forest communities of southern Wisconsin. *Ecological Monographs*, 27(4):325–349, 1957.

[25] T. Bray. The JavaScript Object Notation (JSON) Data Interchange Format, RFC, 2014. Online: https://tools.ietf.org/html/rfc7159. Accessed: 2018-08-14.

[26] A. Bremges and A. C. McHardy. Critical Assessment of Metagenome Interpretation Enters the Second Round. *mSystems*, 3(4), 2018.

[27] R. P. Brent. An algorithm with guaranteed convergence for finding a zero of a function. *The Computer Journal*, 14(4):422–425, 1971.

[28] S. M. Brown, Y. Hao, H. Chen, B. P. Laungani, T. A. Ali, C. Dong, C. Lijeron, B. Kim, K. Krampis, and Z. Pei. Fast functional annotation of metagenomic shotgun data by DNA alignment to a microbial gene catalog. *bioRxiv*, page 120402, 2017.

[29] M. Brudno, S. Malde, A. Poliakov, C. B. Do, O. Couronne, I. Dubchak, and S. Batzoglou. Glocal alignment: finding rearrangements during alignment. *Bioinformatics*, 19(Suppl 1):i54–i62, 2003.

[30] T. Cavalier-Smith. The neomuran origin of archaebacteria, the negibacterial root of the universal tree and bacterial megaclassification. *International Journal of Systematic and Evolutionary Microbiology*, 52(1):7–76, 2002.

[31] D. R. Cavener. Comparison of the consensus sequence flanking translational start sites in Drosophila and vertebrates. *Nucleic Acids Research*, 15(4), 1987.

[32] D. R. Cavener and S. C. Ray. Eukaryotic start and stop translation sites. *Nucleic Acids Research*, 19(12):3185–3192, 1991.

[33] B. Chor and T. Tuller. Maximum Likelihood of Evolutionary Trees Is Hard. pages 296–310. Springer, Berlin, Heidelberg, 2005.

[34] P. J. A. Cock, C. J. Fields, N. Goto, M. L. Heuer, and P. M. Rice. The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Research*, 38(6):1767–1771, 2009.

[35] J. R. Cole, Q. Wang, J. A. Fish, B. Chai, D. M. McGarrell, Y. Sun, C. T. Brown, A. Porras-Alfaro, C. R. Kuske, and J. M. Tiedje. Ribosomal database project: data and tools for high throughput rRNA analysis. *Nucleic Acids Res*, 42, 2014.

[36] M. Comin and D. Verzotto. Alignment-free phylogeny of whole genomes using underlying subwords. *Algorithms for molecular biology : AMB*, 7(1):34, 2012.

[37] F. Crick. Central Dogma of Molecular Biology. *Nature*, 227(5258):561–563, 1970.

[38] F. H. C. Crick. On Protein Synthesis. In *Symposia of the Society for Experimental Biology*, volume 12, page 8, 1958.

[39] A. Criscuolo and S. Gribaldo. BMGE (Block Mapping and Gathering with Entropy): a new software for selection of phylogenetic informative regions from multiple sequence alignments. *BMC Evolutionary Biology*, 10(1):210, 2010.

[40] D. Crockford. The application/json Media Type for JavaScript Object Notation (JSON), RFC, 2006. Online: https://tools.ietf.org/html/rfc4627. Accessed: 2018-08-14.

[41] L. Czech. DNA double helix and nucleobases, 2018. The image is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license, see https://creativecommons.org/licenses/by-sa/3.0/deed.en. It is based on [267] and [268], and parts of [224], which itself is based on [165]. We changed some colors, added background colors, and connected the original images with boxes and lines.

[42] L. Czech and A. Stamatakis. Methods for Automatic Reference Trees and Multilevel Phylogenetic Placement. *bioRxiv*, page 299792, 2018.

[43] L. Czech and A. Stamatakis. Scalable Methods for Post-Processing, Visualizing, and Analyzing Phylogenetic Placements. *bioRxiv*, 2018.

[44] A. E. Darling, G. Jospin, E. Lowe, F. A. Matsen, H. M. Bik, and J. a. Eisen. PhyloSift: phylogenetic analysis of genomes and metagenomes. *PeerJ*, 2:e243, 2014.

[45] C. Darwin. First diagram of an evolutionary tree, from the first notebook on Transmutation of Species, 1837. Online: https://en.wikipedia.org/wiki/File:Darwin_tree.png. Accessed: 2018-08-01. The image is public domain due to its age.

[46] C. Darwin. *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*. John Murray, 1859.

[47] R. Dawkins. *The Selfish Gene*. Oxford University Press, Oxford, 1989.

[48] W. H. E. Day and F. R. McMorris. Threshold consensus methods for molecular sequences. *Journal of Theoretical Biology*, 159(4):481–489, 1992.

[49] W. H. E. Day and F. R. McMorris. Critical comparison of consensus methods for molecular sequences. *Nucleic acids research*, 20(5):1093–1099, 1992.

[50] M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt. A model of evolutionary change in proteins. *Atlas of Protein Sequence and Structure*, 5:345–352, 1978.

[51] C. de Vargas, S. Audic, N. Henry, J. Decelle, F. Mahe, R. Logares, E. Lara, C. Berney, N. Le Bescot, I. Probert, M. Carmichael, J. Poulain, S. Romac, S. Colin, J.-M. Aury, L. Bittner, S. Chaffron, M. Dunthorn, S. Engelen, O. Flegontova, L. Guidi, A. Horak, O. Jaillon, G. Lima-Mendez, J. Luke, S. Malviya, R. Morard, M. Mulot, E. Scalco, R. Siano, F. Vincent, A. Zingone, C. Dimier,

M. Picheral, S. Searson, S. Kandels-Lewis, S. G. Acinas, P. Bork, C. Bowler, G. Gorsky, N. Grimsley, P. Hingamp, D. Iudicone, F. Not, H. Ogata, S. Pesant, J. Raes, M. E. Sieracki, S. Speich, L. Stemmann, S. Sunagawa, J. Weissenbach, P. Wincker, E. Karsenti, E. Boss, M. Follows, L. Karp-Boss, U. Krzic, E. G. Reynaud, C. Sardet, M. B. Sullivan, and D. Velayoudon. Eukaryotic plankton diversity in the sunlit ocean. *Science*, 348(6237):1261605–1261605, 2015.

[52] K. Deiner, H. M. Bik, E. Mächler, M. Seymour, A. Lacoursière-Roussel, F. Altermatt, S. Creer, I. Bista, D. M. Lodge, N. de Vere, M. E. Pfrender, and L. Bernatchez. Environmental DNA metabarcoding: Transforming how we survey animal and plant communities. *Molecular Ecology*, (August):5872–5895, 2017.

[53] N. Desai, D. Antonopoulos, J. A. Gilbert, E. M. Glass, and F. Meyer. From genomics to metagenomics. *Current Opinion in Biotechnology*, 23(1):72–76, 2012.

[54] T. Z. DeSantis, P. Hugenholtz, N. Larsen, M. Rojas, E. L. Brodie, K. Keller, T. Huber, D. Dalevi, P. Hu, and G. L. Andersen. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Applied and environmental microbiology*, 72(7):5069–5072, 2006.

[55] L. R. Dice. Measures of the Amount of Ecologic Association Between Species. *Ecology*, 26(3):297–302, 1945.

[56] M. S. Dodd, D. Papineau, T. Grenne, J. F. Slack, M. Rittner, F. Pirajno, J. O'neil, and C. T. S. Little. Evidence for Early Life in Earth's Oldest Hydrothermal Vent Precipitates. *Nature*, 543(7643):60, 2017.

[57] M. A. Donk. Typification and Later Starting-Points. *Taxon*, 6(9):245, 1957.

[58] J. C. Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3(3):32–57, 1973.

[59] J. C. Dunning Hotopp. Horizontal gene transfer between bacteria and animals. *Trends in genetics : TIG*, 27(4):157–63, 2011.

[60] M. Dunthorn, J. Otto, S. A. Berger, A. Stamatakis, F. Mahé, S. Romac, C. De Vargas, S. Audic, A. Stock, F. Kauff, T. Stoeck, B. Edvardsen, R. Massana, F. Not, N. Simon, and A. Zingone. Placing environmental next-generation sequencing amplicons from microbial eukaryotes into a phylogenetic context. *Molecular Biology and Evolution*, 31(4):993–1009, 2014.

[61] A. Ö. C. Dupont, R. I. Griffiths, T. Bell, and D. Bass. Differences in soil micro-eukaryotic communities over soil pH gradients are strongly driven by parasites and saprotrophs. *Environmental Microbiology*, 18(6):2010–2024, 2016.

[62] S. R. Eddy. Profile hidden Markov models. *Bioinformatics*, 14(9):755—-763, 1998.

[63] S. R. Eddy. A new generation of homology search tools based on probabilistic inference. In *Genome Informatics*, volume 23, pages 205–211. World Scientific, 2009.

[64] R. C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32(5):1792–1797, 2004.

[65] R. C. Edgar. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, 26(19):2460–2461, 2010.

[66] D. J. Edwards and K. E. Holt. Beginner's guide to comparative bacterial genome analysis using next-generation sequence data. *Microbial informatics and experimentation*, 3(1):2, 2013.

[67] A. Escobar-Zepeda, A. Vera-Ponce De León, and A. Sanchez-Flores. The road to metagenomics: From microbiology to DNA sequencing technologies and bioinformatics. *Frontiers in Genetics*, 6(348):1–15, 2015.

[68] S. N. Evans and F. A. Matsen. The phylogenetic Kantorovich-Rubinstein metric for environmental sequence samples. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 74:569–592, 2012.

[69] B. S. Everitt and A. Skrondal. *The Cambridge Dictionary of Statistics*. Cambridge University Press, 4th edition, 2010.

[70] D. P. Faith. Conservation evaluation and phylogenetic diversity. *Biological Conservation*, 61:1–10, 1992.

[71] J. Felsenstein. Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, 1981.

[72] J. Felsenstein. *Inferring Phylogenies*. Sinauer Associates Sunderland, MA, 2 edition, 2004.

[73] A. Filipski, K. Tamura, P. Billing-Ross, O. Murillo, and S. Kumar. Phylogenetic placement of metagenomic reads using the minimum evolution principle. *BMC genomics*, 16(1):6947, 2015.

[74] R. Fletcher. *Practical Methods of Optimization*. Wiley, 1987.

[75] E. Gaba. A phylogenetic tree of living things, based on RNA data and proposed by Carl Woese, showing the separation of bacteria, archaea, and eukaryotes, 2006. Online: https://en.wikipedia.org/wiki/File:Phylogenetic_tree.svg. Accessed: 2018-08-01. The image is released into the public domain.

[76] P. A. Gagniuc. *Markov Chains: From Theory to Implementation and Experimentation*. Wiley, 1st edition, 2017.

[77] N. M. Gericke and M. Hagberg. Definition of historical models of gene function and their relation to students' understanding of genetics. *Science & Education*, 16(7-8):849–881, 2007.

[78] C. R. Giner, I. Forn, S. Romac, R. Logares, and C. D. Vargas. Environmental Sequencing Provides Reasonable Estimates of the Relative Abundance of Specific Picoeukaryotes. *Applied and Environmental Microbiology*, 82(15): 4757–4766, 2016.

[79] P. D. Gingerich. Evolution and the fossil record: patterns, rates, and processes. *Canadian Journal of Zoology*, 65(5):1053–1060, 1987.

[80] E. M. Glass, J. Wilkening, A. Wilke, D. Antonopoulos, and F. Meyer. Using the Metagenomics RAST Server (MG-RAST) for Analyzing Shotgun Metagenomes. *Cold Spring Harbor Protocols*, 2010(1):pdb.prot5368–pdb.prot5368, 2010.

[81] G. B. Gloor, J. R. Wu, V. Pawlowsky-Glahn, and J. J. Egozcue. It's all relative: analyzing microbiome data as compositions. *Annals of epidemiology*, 26(5):322–9, 2016.

[82] G. B. Gloor, J. M. Macklaim, V. Pawlowsky-Glahn, and J. J. Egozcue. Microbiome Datasets Are Compositional: And This Is Not Optional. *Frontiers in Microbiology*, 8:2224, 2017.

[83] S. Goodwin, J. D. McPherson, and W. R. McCombie. Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, 17(6):333–351, 2016.

[84] N. J. Gotelli and R. K. Colwell. Quantifying biodiversity: procedures and pitfalls in the measurement and comparison of species richness. *Ecology Letters*, 4(4):379–391, 2001.

[85] S. Gran-Stadniczeñko, L. Šupraha, E. D. Egge, and B. Edvardsen. Haptophyte Diversity and Vertical Distribution Explored by 18S and 28S Ribosomal RNA Gene Metabarcoding and Scanning Electron Microscopy. *Journal of Eukaryotic Microbiology*, pages 1–19, 2017.

[86] A. J. F. Griffiths, J. H. Miller, D. T. Suzuki, and R. C. Lewontin. *An Introduction to Genetic Analysis*. W.H. Freeman, 2000.

[87] L. Guidi, S. Chaffron, L. Bittner, D. Eveillard, A. Larhlimi, S. Roux, Y. Darzi, S. Audic, L. Berline, J. Brum, L. P. Coelho, J. C. I. Espinoza, S. Malviya, S. Sunagawa, C. Dimier, S. Kandels-Lewis, M. Picheral, J. Poulain, S. Searson, Tara Oceans coordinators, L. Stemmann, F. Not, P. Hingamp, S. Speich, M. Follows, L. Karp-Boss, E. Boss, H. Ogata, S. Pesant, J. Weissenbach, P. Wincker, S. G. Acinas, P. Bork, C. de Vargas, D. Iudicone, M. B. Sullivan, J. Raes, E. Karsenti, C. Bowler, and G. Gorsky. Plankton networks driving carbon export in the oligotrophic ocean. *Nature*, 532(7600):465–470, 2016.

[88] L. Guillou, D. Bachar, S. Audic, D. Bass, C. Berney, L. Bittner, C. Boutte, G. Burgaud, C. De Vargas, J. Decelle, and Others. The Protist Ribosomal Reference database (PR2): a catalog of unicellular eukaryote small sub-unit rRNA sequences with curated taxonomy. *Nucleic acids research*, 41(D1):D597—-D604, 2012.

[89] R. S. Gupta. Life's Third Domain (Archaea): An Established Fact or an Endangered Paradigm?: A New Proposal for Classification of Organisms Based on Protein Sequences and Cell Structure. *Theoretical Population Biology*, 54 (2):91–104, 1998.

[90] P. Halasz. Biological Classification, 2007. Online: https://en.wikipedia.org/wiki/File:Biological_classification_L_Pengo_vflip.svg. Accessed: 2018-08-03. The image is released into the public domain.

[91] B. Hall, B. Hallgrímsson, and M. W. Strickberger. *Strickberger's Evolution*. Jones & Bartlett Learning, 2008.

[92] G. Hamerly and C. Elkan. Learning the k in k-means. In S. Thrun, L. K. Saul, and P. B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 281–288. MIT Press, 2004.

[93] M. V. Han and C. M. Zmasek. phyloXML: XML for evolutionary biology and comparative genomics. *BMC Bioinformatics*, 10:356, 2009.

[94] M. Hasegawa, H. Kishino, and T.-a. Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *Journal of Molecular Evolution*, 22(2):160–174, 1985.

[95] P. D. N. Hebert, A. Cywinska, S. L. Ball, and J. R. DeWaard. Biological Identifications Through DNA Barcodes. *Proceedings in Biological Sciences*, 270(1512):313–21, 2003.

[96] J. Hein. A heuristic method to reconstruct the history of sequences subject to recombination. *Journal of Molecular Evolution*, 36(4):396–405, 1993.

[97] D. G. Higgins and P. M. Sharp. CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. *Gene*, 73(1):237–44, 1988.

[98] D. Hillis and J. Wiens. Molecules versus morphology in systematics: conflicts, artifacts, and misconceptions. *Phylogenetic Analysis of Morphological Data*, pages 1–19, 2000.

[99] P. Hugenholtz, B. M. Goebel, and N. R. Pace. Impact of Culture-Independent Studies on the Emerging Phylogenetic View of Bacterial Diversity. *Journal of Bacteriology*, 180(18):4765–4774, 1998.

[100] D. H. Huson and C. Scornavacca. A Survey of Combinatorial Methods for Phylogenetic Networks. *Genome Biology and Evolution*, 3:23–35, 2011.

[101] C. Huttenhower, D. Gevers, R. Knight, S. Abubucker, J. H. Badger, A. T. Chinwalla, H. H. Creasy, A. M. Earl, M. G. FitzGerald, R. S. Fulton, M. G. Giglio, K. Hallsworth-Pepin, E. A. Lobos, R. Madupu, V. Magrini, J. C. Martin, M. Mitreva, D. M. Muzny, E. J. Sodergren, J. Versalovic, A. M. Wollam, K. C. Worley, J. R. Wortman, S. K. Young, Q. Zeng, K. M. Aagaard, O. O. Abolude, E. Allen-Vercoe, E. J. Alm, L. Alvarado, G. L. Andersen, S. Anderson, E. Appelbaum, H. M. Arachchi, G. Armitage, C. A. Arze, T. Ayvaz, C. C. Baker, L. Begg, T. Belachew, V. Bhonagiri, M. Bihan, M. J. Blaser, T. Bloom, V. Bonazzi, J. Paul Brooks, G. A. Buck, C. J. Buhay, D. A. Busam, J. L. Campbell, S. R. Canon, B. L. Cantarel, P. S. G. Chain, I.-M. A. Chen, L. Chen, S. Chhibba, K. Chu, D. M. Ciulla, J. C. Clemente, S. W. Clifton, S. Conlan, J. Crabtree, M. A. Cutting, N. J. Davidovics, C. C. Davis, T. Z. DeSantis, C. Deal, K. D. Delehaunty, F. E. Dewhirst, E. Deych, Y. Ding, D. J. Dooling, S. P. Dugan, W. Michael Dunne, A. Scott Durkin, R. C. Edgar, R. L. Erlich, C. N. Farmer, R. M. Farrell, K. Faust, M. Feldgarden, V. M. Felix, S. Fisher, A. A. Fodor, L. J. Forney, L. Foster, V. Di Francesco, J. Friedman, D. C. Friedrich, C. C. Fronick, L. L. Fulton, H. Gao, N. Garcia, G. Giannoukos, C. Giblin, M. Y. Giovanni, J. M. Goldberg, J. Goll, A. Gonzalez, A. Griggs, S. Gujja, S. Kinder Haake, B. J. Haas, H. A. Hamilton, E. L. Harris, T. A. Hepburn, B. Herter, D. E. Hoffmann, M. E. Holder, C. Howarth, K. H. Huang, S. M. Huse, J. Izard, J. K. Jansson, H. Jiang, C. Jordan, V. Joshi, J. A. Katancik, W. A. Keitel, S. T. Kelley, C. Kells, N. B. King, D. Knights, H. H. Kong, O. Koren, S. Koren, K. C. Kota, C. L. Kovar, N. C. Kyrpides, P. S. La Rosa, S. L. Lee, K. P. Lemon, N. Lennon, C. M. Lewis, L. Lewis, R. E. Ley, K. Li, K. Liolios, B. Liu, Y. Liu, C.-C. Lo, C. A. Lozupone, R. Dwayne Lunsford, T. Madden, A. A. Mahurkar, P. J. Mannon, E. R. Mardis, V. M. Markowitz, K. Mavromatis, J. M. McCorrison, D. McDonald, J. McEwen, A. L. McGuire, P. McInnes, T. Mehta, K. A. Mihindukulasuriya, J. R. Miller, P. J. Minx, I. Newsham, C. Nusbaum, M. O'Laughlin, J. Orvis, I. Pagani, K. Palaniappan, S. M. Patel, M. Pearson, J. Peterson, M. Podar, C. Pohl, K. S. Pollard, M. Pop, M. E. Priest, L. M. Proctor, X. Qin, J. Raes, J. Ravel, J. G. Reid, M. Rho, R. Rhodes, K. P. Riehle, M. C. Rivera, B. Rodriguez-Mueller, Y.-H. Rogers, M. C. Ross, C. Russ, R. K. Sanka, P. Sankar, J. Fah Sathirapongsasuti, J. A. Schloss, P. D. Schloss, T. M. Schmidt, M. Scholz, L. Schriml, A. M. Schubert, N. Segata, J. A. Segre, W. D. Shannon, R. R. Sharp, T. J. Sharpton, N. Shenoy, N. U. Sheth, G. A. Simone, I. Singh, C. S. Smillie, J. D. Sobel, D. D. Sommer, P. Spicer, G. G. Sutton, S. M. Sykes, D. G. Tabbaa, M. Thiagarajan, C. M. Tomlinson, M. Torralba, T. J. Treangen, R. M. Truty, T. A. Vishnivetskaya, J. Walker, L. Wang, Z. Wang, D. V. Ward, W. Warren, M. A. Watson, C. Wellington, K. A. Wetterstrand, J. R. White, K. Wilczek-Boney, Y. Wu, K. M. Wylie, T. Wylie, C. Yandava, L. Ye, Y. Ye, S. Yooseph, B. P. Youmans, L. Zhang, Y. Zhou, Y. Zhu, L. Zoloth, J. D. Zucker, B. W. Birren, R. A. Gibbs, S. K. Highlander, B. A. Methé, K. E. Nelson, J. F. Petrosino, G. M. Weinstock, R. K. Wilson, and O. White. Structure, function and diversity of the healthy human microbiome. *Nature*,

486(7402):207–214, 2012.

[102] IUPAC-IUB Commission on Biochemical Nomenclature (CBN). Abbreviations and symbols for nucleic acids, polynucleotides, and their constituents. *Biochemistry*, 9(20):4022–4027, 1970.

[103] P. Jaccard. Etude de la distribution florale dans une portion des Alpes et du Jura. *Bulletin de la Societe Vaudoise des Sciences Naturelles*, 37:547–579, 1901.

[104] J. M. Janda and S. L. Abbott. 16S rRNA Gene Sequencing for Bacterial Identification in the Diagnostic Laboratory: Pluses, Perils, and Pitfalls. *Journal of Clinical Microbiology*, 45(9):2761–2764, 2007.

[105] I. Jolliffe. *Principal Component Analysis*. Springer, New York, 2nd ed. edition, 2002.

[106] T. H. Jukes, C. R. Cantor, and Others. Evolution of Protein Molecules. *Mammalian Protein Metabolism*, 3(21):132, 1969.

[107] W. Just. Computational Complexity of Multiple Sequence Alignment with SP-Score. *Journal of Computational Biology*, 8(6):615–623, 2001.

[108] T. Kanungo, D. M. Mount, N. S. Netanyahu, A. Y. Wu, C. D. Piatko, R. Silverman, and A. Y. Wu. A Local Search Approximation Algorithm for k-Means Clustering. *Computational Geometry*, 28(2-3):89–112, 2003.

[109] P. Kapli, S. Lutteropp, J. Zhang, K. Kobert, P. Pavlidis, A. Stamatakis, and T. Flouri. Multi-rate Poisson tree processes for single-locus species delimitation under maximum likelihood and Markov chain Monte Carlo. *Bioinformatics*, 33(11):1630–1638, 2017.

[110] E. Karsenti, S. G. Acinas, P. Bork, C. Bowler, C. de Vargas, J. Raes, M. Sullivan, D. Arendt, F. Benzoni, J. M. Claverie, M. Follows, G. Gorsky, P. Hingamp, D. Iudicone, O. Jaillon, S. Kandels-Lewis, U. Krzic, F. Not, H. Ogata, S. Pesant, E. G. Reynaud, C. Sardet, M. E. Sieracki, S. Speich, D. Velayoudon, J. Weissenbach, and P. Wincker. A holistic approach to marine Eco-systems biology. *PLoS Biology*, 9(10):7–11, 2011.

[111] K. Katoh, K. Misawa, K. Kuma, and T. Miyata. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Research*, 30(14):3059–3066, 2002.

[112] D. R. Kelley and S. L. Salzberg. Clustering metagenomic sequences with interpolated Markov models. *BMC Bioinformatics*, 11(1):544, 2010.

[113] O.-S. Kim, Y.-J. Cho, K. Lee, S.-H. Yoon, M. Kim, H. Na, S.-C. Park, Y. S. Jeon, J.-H. Lee, H. Yi, and Others. Introducing EzTaxon-e: a prokaryotic 16S rRNA gene sequence database with phylotypes that represent uncultured

species. *International journal of systematic and evolutionary microbiology*, 62 (3):716–721, 2012.

[114] M. Kimura. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution*, 16(2):111–120, 1980.

[115] H. Kishino and M. Hasegawa. Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the branching order in hominoidea. *Journal of Molecular Evolution*, 29(2):170–179, 1989.

[116] H. Kishino, T. Miyata, and M. Hasegawa. Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. *Journal of Molecular Evolution*, 31(2):151–160, 1990.

[117] E. V. Koonin. Orthologs, Paralogs, and Evolutionary Genomics. *Annual Review of Genetics*, 39(1):309–338, 2005.

[118] L. B. Koski and G. B. Golding. The Closest BLAST Hit is Often not the Nearest Neighbor. *Journal of Molecular Evolution*, 52(6):540–2, 2001.

[119] A. M. Kozlov, J. Zhang, P. Yilmaz, F. O. Glöckner, and A. Stamatakis. Phylogeny-aware identification and correction of taxonomically mislabeled sequences. *Nucleic Acids Research*, 44(11):5022–5033, 2016.

[120] W. J. Kress and D. L. Erickson. DNA Barcodes: Genes, Genomics, and Bioinformatics. *Proceedings of the National Academy of Sciences of the United States of America*, 105(8):2761–2, 2008.

[121] H.-P. Kriegel, P. Kröger, J. Sander, and A. Zimek. Density-based clustering. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(3): 231–240, 2011.

[122] W. J. Krzanowski and F. Marriott. *Multivariate Analysis*. Wiley, 1994.

[123] M. K. Kuhner and J. Felsenstein. A Simulation Comparison of Phylogeny Algorithms under Equal and Unequal Evolutionary Rates. *Molecular Biology and Evolution*, 11(3):459–468, 1994.

[124] S. Q. Le and O. Gascuel. An Improved General Amino Acid Replacement Matrix. *Molecular Biology and Evolution*, 25(7):1307–1320, 2008.

[125] P. Legendre and L. F. J. Legendre. *Numerical Ecology*. Developments in Environmental Modelling. Elsevier Science, 1998.

[126] G. Lentendu, F. Mahé, D. Bass, S. Rueckert, T. Stoeck, and M. Dunthorn. Consistent patterns of high alpha and low beta diversity in tropical parasitic and free-living protists. *Molecular Ecology*, 27(13):2846–2857, 2018.

[127] A. M. Leroi. *The Lagoon: How Aristotle Invented Science*. Bloomsbury Circus, 1st edition, 2014.

[128] I. Letunic and P. Bork. Interactive tree of life (iTOL) v3: an online tool for the display and annotation of phylogenetic and other trees. *Nucleic Acids Research*, 44(W1):W242–5, 2016.

[129] E. Levina and P. Bickel. The earth mover's distance is the Mallows distance: some insights from statistics. *Eighth IEEE International Conference on Computer Vision*, pages 251–256, 2001.

[130] C. Li and J. Wang. Relative entropy of DNA and its application. *Physica A: Statistical Mechanics and its Applications*, 347:465–471, 2005.

[131] B. Linard, K. Swenson, and F. Pardi. Rapid alignment-free phylogenetic identification of metagenomic sequences. *bioRxiv*, page 328740, 2018.

[132] C. Linnaeus. *Systema Naturae*. Haak, Leiden, 1735.

[133] C. Linnaeus. *Species Plantarum*. Laurentius Salvius, Stockholm, 1753.

[134] K. Liu, S. Raghavan, S. Nelesen, C. R. Linder, and T. Warnow. Rapid and Accurate Large-Scale Coestimation of Sequence Alignments and Phylogenetic Trees. *Science*, 324(5934):1561—-1564, 2009.

[135] K. Liu, T. J. Warnow, M. T. Holder, S. M. Nelesen, J. Yu, A. P. Stamatakis, and C. R. Linder. SATé-II: Very Fast and Accurate Simultaneous Estimation of Multiple Sequence Alignments and Phylogenetic Trees. *Systematic Biology*, 61(1):90, 2012.

[136] S. P. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[137] R. Logares, T. H. Haverkamp, S. Kumar, A. Lanzén, A. J. Nederbragt, C. Quince, and H. Kauserud. 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, 2012.

[138] R. Logares, S. Sunagawa, G. Salazar, F. M. Cornejo-Castillo, I. Ferrera, H. Sarmento, P. Hingamp, H. Ogata, C. de Vargas, G. Lima-Mendez, J. Raes, J. Poulain, O. Jaillon, P. Wincker, S. Kandels-Lewis, E. Karsenti, P. Bork, and S. G. Acinas. Metagenomic 16S rDNA Illumina tags are a powerful alternative to amplicon sequencing to explore diversity and structure of microbial communities. *Environmental Microbiology*, 16(9):2659–2671, 2014.

[139] D. Lovell, V. Pawlowsky-Glahn, J. J. Egozcue, S. Marguerat, and J. Bähler. Proportionality: A Valid Alternative to Correlation for Relative Data. *PLOS Computational Biology*, 11(3):e1004075, 2015.

[140] C. Lozupone and R. Knight. UniFrac: a New Phylogenetic Method for Comparing Microbial Communities. *Applied and Environmental Microbiology*, 71 (12):8228–8235, 2005.

[141] C. A. Lozupone and R. Knight. Global patterns in bacterial diversity. *Proceedings of the National Academy of Sciences*, 104(27):11436–11440, 2007.

[142] C. A. Lozupone, M. Hamady, S. T. Kelley, and R. Knight. Quantitative and Qualitative β Diversity Measures Lead to Different Insights into Factors That Structure Microbial Communities. *Applied and Environmental Microbiology*, 73(5):1576–1585, 2007.

[143] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1(233):281–297, 1967.

[144] D. R. Maddison, D. L. Swofford, and W. P. Maddison. NEXUS: an extensible file format for systematic information. *Systematic biology*, 46(4):590–621, 1997.

[145] W. P. Maddison and J. J. Wiens. Gene Trees in Species Trees. *Systematic Biology*, 46(3):523–536, 1997.

[146] F. Mahé. Fred's metabarcoding pipeline, 2016. Online: https://github.com/frederic-mahe/swarm/wiki/Fred's-metabarcoding-pipeline. Accessed: 2018-01-15.

[147] F. Mahé, T. Rognes, C. Quince, C. de Vargas, and M. Dunthorn. Swarm: Robust and fast clustering method for amplicon-based studies. *PeerJ*, 2:1–12, 2014.

[148] F. Mahé, T. Rognes, C. Quince, C. De Vargas, and M. Dunthorn. Swarm v2: Highly-scalable and high-resolution amplicon clustering. *PeerJ*, 2015.

[149] F. Mahé, C. de Vargas, D. Bass, L. Czech, A. Stamatakis, E. Lara, D. Singer, J. Mayor, J. Bunge, S. Sernaker, T. Siemensmeyer, I. Trautmann, S. Romac, C. Berney, A. Kozlov, E. A. D. Mitchell, C. V. W. Seppey, E. Egge, G. Lentendu, R. Wirth, G. Trueba, and M. Dunthorn. Parasites dominate hyperdiverse soil protist communities in Neotropical rainforests. *Nature Ecology & Evolution*, 1(4):0091, 2017.

[150] C. L. Mallows. A Note on Asymptotic Joint Normality. *Annals of Mathematical Statistics*, 43(2):508–515, 1972.

[151] K. V. Mardia. Some Properties of Classical Multi-Dimesional Scaling. *Communications in Statistics-Theory and Methods*, 7(13):1233–1241, 1978.

[152] E. R. Mardis. Next-Generation Sequencing Platforms. *Annual Review of Analytical Chemistry*, 6(1):287–303, 2013.

[153] M. Martin. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal*, 17(1):10, 2011.

[154] F. A. Matsen. Phylogenetics and the Human Microbiome. *Systematic Biology*, 64(1):e26–e41, 2015.

[155] F. A. Matsen and S. N. Evans. Edge principal components and squash clustering: using the special structure of phylogenetic placement data for sample comparison. *PLOS ONE*, 8(3):1–17, 2011.

[156] F. A. Matsen and S. N. Evans. Edge principal components and squash clustering: using the special structure of phylogenetic placement data for sample comparison. *arXiv*, 2011.

[157] F. A. Matsen and S. N. Evans. Edge principal components analysis example, 2011. Online: http://matsen.fredhutch.org/pplacer/demo/pca.html. Accessed: 2018-01-15.

[158] F. A. Matsen, R. B. Kodner, and E. V. Armbrust. pplacer: linear time maximum-likelihood and Bayesian phylogenetic placement of sequences onto a fixed reference tree. *BMC Bioinformatics*, 11(1):538, 2010.

[159] F. A. Matsen, N. G. Hoffman, A. Gallagher, and A. Stamatakis. A format for phylogenetic placements. *PLoS ONE*, 7(2):1–4, 2012.

[160] F. A. Matsen, A. Gallagher, and C. O. McCoy. Minimizing the average distance to a closest leaf in a phylogenetic tree. *Systematic Biology*, 62(6):824–836, 2013.

[161] K. O. May. A set of independent necessary and sufficient conditions for simple majority decision. *Econometrica: Journal of the Econometric Society*, pages 680–684, 1952.

[162] E. Mayr. Two empires or three? *Proceedings of the National Academy of Sciences of the United States of America*, 95(17):9720–3, 1998.

[163] E. Mayr and W. J. Bock. Classifications and other ordering systems. *Journal of Zoological Systematics and Evolutionary Research*, 40(4):169–194, 2002.

[164] P. J. McMurdie and S. Holmes. Waste Not, Want Not: Why Rarefying Microbiome Data Is Inadmissible. *PLoS Computational Biology*, 10(4):e1003531, 2014.

[165] MesserWoland. DNA structure and bases, 2006. Online: https://commons.wikimedia.org/wiki/File:DNA_structure_and_bases.svg. Accessed: 2018-08-04. The image is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license, see https://creativecommons.org/licenses/by-sa/3.0/deed.en.

[166] B. A. Methé, K. E. Nelson, M. Pop, H. H. Creasy, M. G. Giglio, C. Huttenhower, D. Gevers, J. F. Petrosino, S. Abubucker, H. Jonathan, A. T. Chinwalla, A. M. Earl, M. G. Fitzgerald, R. S. Fulton, K. Hallsworth-Pepin, E. A. Lobos, R. Madupu, V. Magrini, J. C. Martin, M. Mitreva, D. M. Muzny, E. J. Sodergren, A. M. Wollam, K. C. Worley, J. R. Wortman, S. K. Young, Q. Zeng, K. M. Aagaard, O. O. Abolude, E. Allen-vercoe, J. Eric, L. Alvarado, G. L. Andersen, S. Anderson, E. Appelbaum, H. M. Arachchi, G. Armitage, C. A. Arze, T. Ayvaz, C. C. Baker, L. Begg, T. Belachew, V. Bhonagiri, M. Bihan, M. J. Blaser, T. Bloom, J. V. Bonazzi, P. Brooks, G. A. Buck, J. Christian, D. A. Busam, J. L. Campbell, S. R. Canon, B. L. Cantarel, P. S. Chain, I.-M. A. Chen, L. Chen, S. Chhibba, K. Chu, M. Dawn, J. C. Clemente, S. W. Clifton, S. Conlan, J. Crabtree, A. Cutting, N. J. Davidovics, C. C. Davis, T. Z. Desantis, K. D. Delehaunty, F. E. Dewhirst, E. Deych, Y. Ding, J. H. Badger, A. T. Chinwalla, A. M. Earl, M. G. Fitzgerald, R. S. Fulton, K. Hallsworth-Pepin, E. A. Lobos, R. Madupu, V. Magrini, J. C. Martin, M. Mitreva, D. M. Muzny, E. J. Sodergren, J. Versalovic, A. M. Wollam, K. C. Worley, J. R. Wortman, S. K. Young, Q. Zeng, K. M. Aagaard, O. O. Abolude, E. Allen-vercoe, E. J. Alm, L. Alvarado, G. L. Andersen, S. Anderson, E. Appelbaum, H. M. Arachchi, G. Armitage, C. A. Arze, T. Ayvaz, C. C. Baker, L. Begg, T. Belachew, V. Bhonagiri, M. Bihan, M. J. Blaser, T. Bloom, V. R. Bonazzi, P. Brooks, G. A. Buck, C. J. Buhay, D. A. Busam, J. L. Campbell, S. R. Canon, B. L. Cantarel, P. S. Chain, I.-M. A. Chen, L. Chen, S. Chhibba, K. Chu, D. M. Ciulla, J. C. Clemente, S. W. Clifton, S. Conlan, J. Crabtree, M. A. Cutting, N. J. Davidovics, C. C. Davis, T. Z. Desantis, C. Deal, K. D. Delehaunty, F. E. Dewhirst, E. Deych, Y. Ding, D. J. Dooling, S. P. Dugan, W. Michael Dunne, A. Scott Durkin, R. C. Edgar, R. L. Erlich, C. N. Farmer, R. M. Farrell, K. Faust, M. Feldgarden, V. M. Felix, S. Fisher, A. A. Fodor, L. Forney, L. Foster, V. Di Francesco, J. Friedman, D. C. Friedrich, C. C. Fronick, L. L. Fulton, H. Gao, N. Garcia, G. Giannoukos, C. Giblin, M. Y. Giovanni, J. M. Goldberg, J. Goll, A. Gonzalez, A. Griggs, S. Gujja, B. J. Haas, H. A. Hamilton, E. L. Harris, T. A. Hepburn, B. Herter, D. E. Hoffmann, M. E. Holder, C. Howarth, K. H. Huang, S. M. Huse, J. Izard, J. K. Jansson, H. Jiang, C. Jordan, V. Joshi, J. A. Katancik, W. A. Keitel, S. T. Kelley, C. Kells, S. Kinder-Haake, N. B. King, R. Knight, D. Knights, H. H. Kong, O. Koren, S. Koren, K. C. Kota, C. L. Kovar, N. C. Kyrpides, P. S. La Rosa, S. L. Lee, K. P. Lemon, N. Lennon, C. M. Lewis, L. Lewis, R. E. Ley, K. Li, K. Liolios, B. Liu, Y. Liu, C.-C. Lo, C. A. Lozupone, R. Dwayne Lunsford, T. Madden, A. A. Mahurkar, P. J. Mannon, E. R. Mardis, V. M. Markowitz, K. Mavrommatis, J. M. McCorrison, D. McDonald, J. McEwen, A. L. McGuire, P. McInnes, T. Mehta, K. A. Mihindukulasuriya, J. R. Miller, P. J. Minx, I. Newsham, C. Nusbaum, M. O'Laughlin, J. Orvis, I. Pagani, K. Palaniappan, S. M. Patel, M. Pearson, J. Peterson, M. Podar, C. Pohl, K. S. Pollard, M. E. Priest, L. M. Proctor, X. Qin, J. Raes, J. Ravel, J. G. Reid, M. Rho, R. Rhodes, K. P. Riehle, M. C. Rivera, B. Rodriguez-Mueller, Y.-H. Rogers, M. C. Ross, C. Russ, R. K. Sanka, P. Sankar, J. Fah Sathirapongsasuti,

J. A. Schloss, P. D. Schloss, T. M. Schmidt, M. Scholz, L. Schriml, A. M. Schubert, N. Segata, J. A. Segre, W. D. Shannon, R. R. Sharp, T. J. Sharpton, N. Shenoy, N. U. Sheth, G. A. Simone, I. Singh, C. S. Smillie, J. D. Sobel, D. D. Sommer, P. Spicer, G. G. Sutton, S. M. Sykes, D. G. Tabbaa, M. Thiagarajan, C. M. Tomlinson, M. Torralba, T. J. Treangen, R. M. Truty, T. A. Vishnivetskaya, J. Walker, L. Wang, Z. Wang, D. V. Ward, W. Warren, M. A. Watson, C. Wellington, K. A. Wetterstrand, J. R. White, K. Wilczek-Boney, Y. Qing Wu, K. M. Wylie, T. Wylie, C. Yandava, L. Ye, Y. Ye, S. Yooseph, B. P. Youmans, L. Zhang, Y. Zhou, Y. Zhu, L. Zoloth, J. D. Zucker, B. W. Birren, R. A. Gibbs, S. K. Highlander, G. M. Weinstock, R. K. Wilson, and O. White. A framework for human microbiome research. *Nature*, 486(7402): 215–221, 2012.

[167] M. L. Metzker. Sequencing Technologies—The Next Generation. *Nature Reviews Genetics*, 11(1):31–46, 2010.

[168] A. Meyer, C. Todt, N. T. Mikkelsen, and B. Lieb. Fast evolving 18S rRNA sequences from Solenogastres (Mollusca) resist standard PCR amplification and give new insights into mollusk substitution rate heterogeneity. *BMC Evolutionary Biology 2010 10:1*, 10(1):70, 2010.

[169] C. D. Michener and R. R. Sokal. A quantitative approach to a problem in classification. *Evolution*, 11(2):130–162, 1957.

[170] S. Mignard and J. P. Flandrois. 16S rRNA sequencing in routine bacterial identification: a 30-month experiment. *Journal of microbiological methods*, 67 (3):574–581, 2006.

[171] D. P. Mindell. The Tree of Life: Metaphor, Model, and Heuristic Device. *Systematic Biology*, 62(3):479–489, 2013.

[172] B. Minh, S. Klaere, and A. Haeseler. Phylogenetic Diversity within Seconds. *Systematic Biology*, 55(5):769–773, 2006.

[173] S. Mirarab, N. Nguyen, and T. Warnow. SEPP: SATé-Enabled Phylogenetic Placement. *Biocomputing*, pages 247–258, 2012.

[174] A. Monier, J.-M. Claverie, and H. Ogata. Taxonomic distribution of large DNA viruses in the sea. *Genome biology*, 9(7):R106, 2008.

[175] D. Moreira and H. Philippe. Molecular phylogeny: pitfalls and progress. *International Microbiology*, 3(1):9–16, 2000.

[176] J. L. Morgan, A. E. Darling, and J. A. Eisen. Metagenomic sequencing of an in vitro-simulated microbial community. *PLoS ONE*, 5(4):1–10, 2010.

[177] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, 1970.

[178] M. Nei and W. H. Li. Mathematical model for studying genetic variation in terms of restriction endonucleases. *Proceedings of the National Academy of Sciences of the United States of America*, 76(10):5269–5273, 1979.

[179] L.-T. T. Nguyen, H. A. Schmidt, A. Von Haeseler, and B. Q. Minh. IQ-TREE: A fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. *Molecular Biology and Evolution*, 32(1):268–74, 2015.

[180] N. P. Nguyen, S. Mirarab, B. Liu, M. Pop, and T. Warnow. TIPP: Taxonomic identification and phylogenetic profiling. *Bioinformatics*, 30(24):3548–3555, 2014.

[181] R. P. Nugent, M. A. Krohn, and S. L. Hillier. Reliability of diagnosing bacterial vaginosis is improved by a standardized method of gram stain interpretation. *Journal of Clinical Microbiology*, 29(2):297–301, 1991.

[182] H. Ochman, J. G. Lawrence, and E. A. Groisman. Lateral gene transfer and the nature of bacterial innovation. *Nature*, 405(6784):299–304, 2000.

[183] B. D. Ondov, N. H. Bergman, and A. M. Phillippy. Interactive metagenomic visualization in a Web browser. *BMC Bioinformatics*, 12(1):385, 2011.

[184] A. Oulas, C. Pavloudi, P. Polymenakou, G. A. Pavlopoulos, N. Papanikolaou, G. Kotoulas, C. Arvanitidis, and I. Iliopoulos. Metagenomics: Tools and Insights for Analyzing Next-Generation Sequencing Data Derived from Biodiversity Studies. *Bioinformatics and Biology Insights*, 9:75–88, 2015.

[185] N. R. Pace. A molecular view of microbial diversity and the biosphere. *Science*, 276(5313):734–740, 1997.

[186] F. Pardi and N. Goldman. Species Choice for Comparative Genomics: Being Greedy Works. *PLOS Genetics*, 1(6):1, 2005.

[187] D. H. Parks, M. Chuvochina, D. W. Waite, C. Rinke, A. Skarshewski, P.-A. Chaumeil, and P. Hugenholtz. A proposal for a standardized bacterial taxonomy based on genome phylogeny. *bioRxiv*, 2018.

[188] K. Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[189] W. R. Pearson and D. J. Lipman. Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences*, 85(8):2444–2448, 1988.

[190] D. Pelleg, A. W. Moore, and Others. X-means: Extending K-means with Efficient Estimation of the Number of Clusters. In *ICML*, volume 1, pages 727–734, 2000.

[191] E. Pettersson, J. Lundeberg, and A. Ahmadian. Generations of Sequencing Technologies. *Genomics*, 93(2):105–111, 2009.

[192] C. A. Petti. Detection and identification of microorganisms by gene amplification and sequencing. *Clinical Infectious Diseases*, 44(8):1108–1114, 2007.

[193] V. O. Polyanovsky, M. A. Roytberg, and V. G. Tumanyan. Comparative analysis of the quality of a global algorithm and a local algorithm for alignment of two sequences. *Algorithms for Molecular Biology*, 6(1):25, 2011.

[194] M. Potapova. Patterns of Diatom Distribution In Relation to Salinity. In J. Kociolek and J. Seckbach, editors, *The Diatom World*, pages 313–332. Springer, 2011.

[195] M. N. Price, P. S. Dehal, and A. P. Arkin. FastTree 2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS ONE*, 5(3):e9490, 2010.

[196] C. Quast, E. Pruesse, P. Yilmaz, J. Gerken, T. Schweer, P. Yarza, J. Peplies, and F. O. Glockner. The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucleic Acids Research*, 41(D1): D590–D596, 2013.

[197] S. T. Rachev. The Monge-Kantorovich Mass Transference Problem and its Stochastic Applications. *Theory of Probability and its Applications*, 29(4): 647–676, 1985.

[198] S. T. Rachev. *Probability Metrics and the Stability of Stochastic Models.* Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. John Wiley & Sons Ltd, Chichester, 1991.

[199] S. T. Rachev and L. Rüschendorf. *Mass Transportation Problems. normalfont Volume 1: Theory.* Springer-Verlag, New York, 1 edition, 1998.

[200] R. Ren, Y. Sun, Y. Zhao, D. Geiser, H. Ma, and X. Zhou. Phylogenetic Resolution of Deep Eukaryotic and Fungal Relationships Using Highly Conserved Low-Copy Nuclear Genes. *Genome Biology and Evolution*, 8(9):2683–701, 2016.

[201] J. A. Reuter, D. V. Spacek, and M. P. Snyder. High-Throughput Sequencing Technologies. *Molecular Cell*, 58(4):586–97, 2015.

[202] D. F. Robinson and L. R. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1):131–147, 1981.

[203] K. M. Robinson, K. B. Sieber, and J. C. Dunning Hotopp. A Review of Bacteria-Animal Lateral Gene Transfer May Inform Our Understanding of Diseases like Cancer. *PLoS Genetics*, 9(10):e1003877, 2013.

[204] T. Rognes, T. Flouri, B. Nichols, C. Quince, and F. Mahé. VSEARCH: a versatile open source tool for metagenomics. *PeerJ*, 4:e2584, 2016.

[205] F. Ronquist and J. P. Huelsenbeck. MrBayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19(12):1572–1574, 2003.

[206] P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[207] N. Saitou and M. Nei. The Neighbor-Joining Method: A new Method for Reconstructing Phylogenetic Trees. *Molecular Biology and Evolution*, 4(4): 406–425, 1987.

[208] F. Sanger and A. Coulson. A Rapid Method for Determining Sequences in DNA by Primed Synthesis with DNA Polymerase. *Journal of Molecular Biology*, 94(3):441–448, 1975.

[209] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12):5463–7, 1977.

[210] D. Sankoff. Minimal Mutation Trees of Sequences. *SIAM Journal on Applied Mathematics*, 28(1):35–42, 1975.

[211] V. Savolainen, R. S. Cowan, A. P. Vogler, G. K. Roderick, and R. Lane. Towards Writing the Encyclopedia of Life: An Introduction to DNA Barcoding. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 360(1462):1805–11, 2005.

[212] E. W. Sayers, T. Barrett, D. A. Benson, S. H. Bryant, K. Canese, V. Chetvernin, D. M. Church, M. DiCuccio, R. Edgar, S. Federhen, M. Feolo, L. Y. Geer, W. Helmberg, Y. Kapustin, D. Landsman, D. J. Lipman, T. L. Madden, D. R. Maglott, V. Miller, I. Mizrachi, J. Ostell, K. D. Pruitt, G. D. Schuler, E. Sequeira, S. T. Sherry, M. Shumway, K. Sirotkin, A. Souvorov, G. Starchenko, T. A. Tatusova, L. Wagner, E. Yaschenko, and J. Ye. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research*, 37(Database):D5–D15, 2009.

[213] A. O. Schmitt and H. Herzel. Estimating the Entropy of DNA Sequences. *Journal of Theoretical Biology*, 188(3):369–377, 1997.

[214] M. B. Scholz, C. C. Lo, and P. S. G. Chain. Next generation sequencing and bioinformatic bottlenecks: The current state of metagenomic data analysis. *Current Opinion in Biotechnology*, 23(1):9–15, 2012.

[215] A. Sczyrba, P. Hofmann, P. Belmann, D. Koslicki, S. Janssen, J. Dröge, I. Gregor, S. Majda, J. Fiedler, E. Dahms, A. Bremges, A. Fritz, R. Garrido-Oter, T. S. Jørgensen, N. Shapiro, P. D. Blood, A. Gurevich, Y. Bai, D. Turaev,

M. Z. DeMaere, R. Chikhi, N. Nagarajan, C. Quince, F. Meyer, M. Balvočiūtė, L. H. Hansen, S. J. Sørensen, B. K. H. Chia, B. Denis, J. L. Froula, Z. Wang, R. Egan, D. Don Kang, J. J. Cook, C. Deltel, M. Beckstette, C. Lemaitre, P. Peterlongo, G. Rizk, D. Lavenier, Y.-W. Wu, S. W. Singer, C. Jain, M. Strous, H. Klingenberg, P. Meinicke, M. D. Barton, T. Lingner, H.-H. Lin, Y.-C. Liao, G. G. Z. Silva, D. A. Cuevas, R. A. Edwards, S. Saha, V. C. Piro, B. Y. Renard, M. Pop, H.-P. Klenk, M. Göker, N. C. Kyrpides, T. Woyke, J. A. Vorholt, P. Schulze-Lefert, E. M. Rubin, A. E. Darling, T. Rattei, and A. C. McHardy. Critical Assessment of Metagenome Interpretation—a benchmark of metagenomics software. *Nature Methods*, 14(11):1063–1071, 2017.

[216] C. E. Shannon and W. Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, 1951.

[217] H. Shimodaira. An Approximately Unbiased Test of Phylogenetic Tree Selection. *Systematic Biology*, 51(3):492–508, 2002.

[218] H. Shimodaira and M. Hasegawa. Multiple Comparisons of Log-Likelihoods with Applications to Phylogenetic Inference. *Molecular Biology and Evolution*, 16(8):1114, 1999.

[219] J.-J. Shu. A new integrated symmetrical table for genetic codes. *Biosystems*, 151:21–26, 2017.

[220] T. F. Smith and M. S. Waterman. Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, 147(1):195–197, 1981.

[221] R. R. Sokal and C. Michener. A Statistical Method for Evaluating Systematic Relationships. *University of Kansas Science Bulletin*, 38:1409–1438, 1958.

[222] R. R. Sokal and P. H. A. Sneath. *Principles of Numerical Taxonomy*. W.H. Freeman, San Francisco, 1963.

[223] T. Sørensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. *Biol. Skr.*, 5:1–34, 1948.

[224] Sponk. Comparison of a single-stranded RNA and a double-stranded DNA with their corresponding nucleobases, 2010. Online: https://commons.wikimedia.org/wiki/File:Difference_DNA_RNA-DE.svg. Accessed: 2018-08-04. The image is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license, see https://creativecommons.org/licenses/by-sa/3.0/deed.en. It is based on [165], which is published under the same license.

[225] S. Srinivasan, N. G. Hoffman, M. T. Morgan, F. A. Matsen, T. L. Fiedler, R. W. Hall, F. J. Ross, C. O. McCoy, R. Bumgarner, J. M. Marrazzo, and D. N. Fredricks. Bacterial communities in women with bacterial vaginosis: High resolution phylogenetic analyses reveal relationships of microbiota to clinical criteria. *PLOS ONE*, 7(6):e37818, 2012.

[226] R. Staden. A strategy of DNA sequencing employing computer programs. *Nucleic Acids Research*, 6(7):2601–2610, 1979.

[227] A. Stamatakis. Phylogenetic models of rate heterogeneity: A high performance computing perspective. In P. Spirakis and H. J. Siegel, editors, *20th International Parallel and Distributed Processing Symposium, (IPDPS) 2006.*, page 278, Rhodes Island, Greece, 2006. IEEE, IEEE.

[228] A. Stamatakis. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 2014.

[229] L. Stein. Genome Annotation: From Sequence to Biology. *Nature Reviews Genetics*, 2(7):493–503, 2001.

[230] T. Stoeck, D. Bass, M. Nebel, R. Christen, M. D. M. Jones, H.-W. BREINER, and T. A. Richards. Multiple marker parallel tag environmental DNA sequencing reveals a highly complex eukaryotic community in marine anoxic water. *Molecular Ecology*, 19(s1):21–31, 2010.

[231] K. Strimmer and A. Rambaut. Inferring confidence sets of possibly misspecified gene trees. *Proceedings of the Royal Society of London B: Biological Sciences*, 269(1487):137–142, 2002.

[232] M. A. Suchard, P. Lemey, G. Baele, D. L. Ayres, A. J. Drummond, and A. Rambaut. Bayesian phylogenetic and phylodynamic data integration using BEAST 1.10. *Virus Evolution*, 4(1), 2018.

[233] S. Sunagawa, D. R. Mende, G. Zeller, F. Izquierdo-Carrasco, S. a. Berger, J. R. Kultima, L. P. Coelho, M. Arumugam, J. Tap, H. B. Nielsen, S. Rasmussen, S. Brunak, O. Pedersen, F. Guarner, W. M. de Vos, J. Wang, J. Li, J. Doré, S. D. Ehrlich, A. Stamatakis, P. Bork, J. Dore, S. D. Ehrlich, A. Stamatakis, and P. Bork. Metagenomic species profiling using universal phylogenetic marker genes. *Nature Methods*, 10(12):1196, 2013.

[234] S. Sunagawa, L. P. Coelho, S. Chaffron, J. R. Kultima, K. Labadie, G. Salazar, B. Djahanschiri, G. Zeller, D. R. Mende, A. Alberti, F. M. Cornejo-castillo, P. I. Costea, C. Cruaud, F. Ovidio, S. Engelen, I. Ferrera, J. M. Gasol, L. Guidi, F. Hildebrand, F. Kokoszka, C. Lepoivre, F. D\textquoterightOvidio, S. Engelen, I. Ferrera, J. M. Gasol, L. Guidi, F. Hildebrand, F. Kokoszka, C. Lepoivre, G. Lima-Mendez, J. Poulain, B. T. Poulos, M. Royo-Llonch, H. Sarmento, S. Vieira-Silva, C. Dimier, M. Picheral, S. Searson, S. Kandels-Lewis, C. Bowler, C. de Vargas, G. Gorsky, N. Grimsley, P. Hingamp, D. Iudicone, O. Jaillon, F. Not, H. Ogata, S. Pesant, S. Speich, L. Stemmann, M. B. Sullivan, J. Weissenbach, P. Wincker, E. Karsenti, J. Raes, S. G. Acinas, and P. Bork. Structure and function of the global ocean microbiome. *Science*, 348 (6237):1–10, 2015.

[235] O. Tanaseichuk, J. Borneman, and T. Jiang. Phylogeny-based classification of microbial communities. *Bioinformatics*, 30(4):449–456, 2014.

[236] S. Tavaré. Some probabilistic and statistical problems in the analysis of DNA sequences. *American Mathematical Society: Lectures on Mathematics in the Life Sciences*, 17:57–86, 1986.

[237] L. Tedersoo, M. Bahram, S. Põlme, U. Kõljalg, N. S. Yorou, R. Wijesundera, L. V. Ruiz, A. M. Vasco-Palacios, P. Q. Thu, A. Suija, and Others. Global diversity and geography of soil fungi. *Science*, 346(6213):1256688, 2014.

[238] B. Temperton and S. J. Giovannoni. Metagenomics: Microbial diversity through a scratched lens. *Current Opinion in Microbiology*, 15(5):605–612, 2012.

[239] J. D. Thompson, B. Linard, O. Lecompte, and O. Poch. A Comprehensive Benchmark Study of Multiple Sequence Alignment Methods: Current Challenges and Future Perspectives. *PLoS ONE*, 6(3):e18093, 2011.

[240] L. R. Thompson, J. G. Sanders, D. McDonald, A. Amir, J. Ladau, K. J. Locey, R. J. Prill, A. Tripathi, S. M. Gibbons, G. Ackermann, J. A. Navas-Molina, S. Janssen, E. Kopylova, Y. Vázquez-Baeza, A. González, J. T. Morton, S. Mirarab, Z. Zech Xu, L. Jiang, M. F. Haroon, J. Kanbar, Q. Zhu, S. Jin Song, T. Kosciolek, N. A. Bokulich, J. Lefler, C. J. Brislawn, G. Humphrey, S. M. Owens, J. Hampton-Marcell, D. Berg-Lyons, V. McKenzie, N. Fierer, J. A. Fuhrman, A. Clauset, R. L. Stevens, A. Shade, K. S. Pollard, K. D. Goodwin, J. K. Jansson, J. A. Gilbert, R. Knight, and T. E. M. P. Consortium. A communal catalogue reveals Earth's multiscale microbial diversity. *Nature*, 2017.

[241] R. L. Thorndike. Who belongs in the family? *Psychometrika*, 18(4):267–276, 1953.

[242] R. Tibshirani, G. Walther, and T. Hastie. 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, 2001.

[243] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, J. D. Gocayne, P. Amanatides, R. M. Ballew, D. H. Huson, J. R. Wortman, Q. Zhang, C. D. Kodira, X. H. Zheng, L. Chen, M. Skupski, G. Subramanian, P. D. Thomas, J. Zhang, G. L. Gabor Miklos, C. Nelson, S. Broder, A. G. Clark, J. Nadeau, V. A. McKusick, N. Zinder, A. J. Levine, R. J. Roberts, M. Simon, C. Slayman, M. Hunkapiller, R. Bolanos, A. Delcher, I. Dew, D. Fasulo, M. Flanigan, L. Florea, A. Halpern, S. Hannenhalli, S. Kravitz, S. Levy, C. Mobarry, K. Reinert, K. Remington, J. Abu-Threideh, E. Beasley, K. Biddick, V. Bonazzi, R. Brandon, M. Cargill, I. Chandramouliswaran, R. Charlab, K. Chaturvedi, Z. Deng, V. Di Francesco, P. Dunn, K. Eilbeck, C. Evangelista, A. E. Gabrielian, W. Gan, W. Ge, F. Gong, Z. Gu, P. Guan, T. J. Heiman, M. E. Higgins, R. R. Ji, Z. Ke, K. A. Ketchum, Z. Lai, Y. Lei, Z. Li, J. Li, Y. Liang, X. Lin, F. Lu, G. V. Merkulov, N. Milshina, H. M. Moore, A. K. Naik, V. A. Narayan,

B. Neelam, D. Nusskern, D. B. Rusch, S. Salzberg, W. Shao, B. Shue, J. Sun, Z. Wang, A. Wang, X. Wang, J. Wang, M. Wei, R. Wides, C. Xiao, C. Yan, A. Yao, J. Ye, M. Zhan, W. Zhang, H. Zhang, Q. Zhao, L. Zheng, F. Zhong, W. Zhong, S. Zhu, S. Zhao, D. Gilbert, S. Baumhueter, G. Spier, C. Carter, A. Cravchik, T. Woodage, F. Ali, H. An, A. Awe, D. Baldwin, H. Baden, M. Barnstead, I. Barrow, K. Beeson, D. Busam, A. Carver, A. Center, M. L. Cheng, L. Curry, S. Danaher, L. Davenport, R. Desilets, S. Dietz, K. Dodson, L. Doup, S. Ferriera, N. Garg, A. Gluecksmann, B. Hart, J. Haynes, C. Haynes, C. Heiner, S. Hladun, D. Hostin, J. Houck, T. Howland, C. Ibegwam, J. Johnson, F. Kalush, L. Kline, S. Koduru, A. Love, F. Mann, D. May, S. McCawley, T. McIntosh, I. McMullen, M. Moy, L. Moy, B. Murphy, K. Nelson, C. Pfannkoch, E. Pratts, V. Puri, H. Qureshi, M. Reardon, R. Rodriguez, Y. H. Rogers, D. Romblad, B. Ruhfel, R. Scott, C. Sitter, M. Smallwood, E. Stewart, R. Strong, E. Suh, R. Thomas, N. N. Tint, S. Tse, C. Vech, G. Wang, J. Wetter, S. Williams, M. Williams, S. Windsor, E. Winn-Deen, K. Wolfe, J. Zaveri, K. Zaveri, J. F. Abril, R. Guigó, M. J. Campbell, K. V. Sjolander, B. Karlak, A. Kejariwal, H. Mi, B. Lazareva, T. Hatton, A. Narechania, K. Diemer, A. Muruganujan, N. Guo, S. Sato, V. Bafna, S. Istrail, R. Lippert, R. Schwartz, B. Walenz, S. Yooseph, D. Allen, A. Basu, J. Baxendale, L. Blick, M. Caminha, J. Carnes-Stine, P. Caulk, Y. H. Chiang, M. Coyne, C. Dahlke, A. Mays, M. Dombroski, M. Donnelly, D. Ely, S. Esparham, C. Fosler, H. Gire, S. Glanowski, K. Glasser, A. Glodek, M. Gorokhov, K. Graham, B. Gropman, M. Harris, J. Heil, S. Henderson, J. Hoover, D. Jennings, C. Jordan, J. Jordan, J. Kasha, L. Kagan, C. Kraft, A. Levitsky, M. Lewis, X. Liu, J. Lopez, D. Ma, W. Majoros, J. McDaniel, S. Murphy, M. Newman, T. Nguyen, N. Nguyen, M. Nodell, S. Pan, J. Peck, M. Peterson, W. Rowe, R. Sanders, J. Scott, M. Simpson, T. Smith, A. Sprague, T. Stockwell, R. Turner, E. Venter, M. Wang, M. Wen, D. Wu, M. Wu, A. Xia, A. Zandieh, and X. Zhu. The Sequence of the Human Genome. *Science*, 291(5507):1304–51, 2001.

[244] K. Vervier, P. Mahé, M. Tournoud, J.-B. Veyrieras, and J.-P. Vert. Large-scale machine learning for metagenomics sequence classification. *Bioinformatics*, 32 (7):1023–1032, 2015.

[245] C. Villani. *Optimal transport: old and new*. Springer Science & Business Media, 2008.

[246] S. Vinga. Information theory applications for biological sequence analysis. *Briefings in Bioinformatics*, 15(3):376–389, 2014.

[247] S. Vinga and J. Almeida. Alignment-free sequence comparison - A review. *Bioinformatics*, 19(4):513–523, 2003.

[248] S. Vinga and J. S. Almeida. Rényi continuous entropy of DNA sequences. *Journal of Theoretical Biology*, 231(3):377–388, 2004.

[249] K. V. Voelkerding, S. A. Dames, and J. D. Durtschi. Next-Generation Sequencing: From Basic Research to Diagnostics. *Clinical Chemistry*, 55(4): 641–58, 2009.

[250] C. von Mering, P. Hugenholtz, J. Raes, S. G. Tringe, T. Doerks, L. J. Jensen, N. Ward, and P. Bork. Quantitative Phylogenetic Assessment of Microbial Communities in Diverse Environments. *Science*, 315(5815):1126–1130, 2007.

[251] L. Wang and T. Jiang. On the Complexity of Multiple Sequence Alignment. *Journal of Computational Biology*, 1(4):337–348, 1994.

[252] W.-L. Wang, S.-Y. Xu, Z.-G. Ren, L. Tao, J.-W. Jiang, and S.-S. Zheng. Application of metagenomics in the human gut microbiome. 21(3), 2015.

[253] J. D. Watson and F. H. C. Crick. Molecular Structure of Nucleic Acids. *Nature*, 171:737–738, 1953.

[254] N. S. Watson-Haigh. Deinterleave FASTQ files, 2012. Online: https://gist.github.com/nathanhaigh/3521724. Accessed: 2018-07-04.

[255] W. G. Weisburg, S. M. Barns, D. A. Pelletier, and D. J. Lane. 16S Ribosomal DNA Amplification for Phylogenetic Study. *Journal of Bacteriology*, 173(2): 697–703, 1991.

[256] S. Weiss, Z. Z. Xu, S. Peddada, A. Amir, K. Bittinger, A. Gonzalez, C. Lozupone, J. R. Zaneveld, Y. Vázquez-Baeza, A. Birmingham, E. R. Hyde, and R. Knight. Normalization and microbial differential abundance strategies depend upon data characteristics. *Microbiome*, 5(1):27, 2017.

[257] K. A. Wetterstrand. DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP), 2018. Online: https://www.genome.gov/sequencingcostsdata. Accessed: 2018-07-24.

[258] S. Whelan and N. Goldman. A General Empirical Model of Protein Evolution Derived from Multiple Protein Families Using a Maximum-Likelihood Approach. *Molecular Biology and Evolution*, 18(5):691–699, 2001.

[259] C. R. Woese and G. E. Fox. Phylogenetic structure of the prokaryotic domain: the primary kingdoms. *Proceedings of the National Academy of Sciences of the United States of America*, 74(11):5088–90, 1977.

[260] C. R. Woese, O. Kandler, and M. L. Wheelis. Towards a natural system of organisms: proposal for the domains Archaea, Bacteria, and Eucarya. *Proceedings of the National Academy of Sciences of the United States of America*, 87(12):4576–9, 1990.

[261] X. Xia, Z. Xie, M. Salemi, L. Chen, and Y. Wang. An index of substitution saturation and its application. *Molecular Phylogenetics and Evolution*, 26(1): 1–7, 2003.

[262] Z. Yang. Statistical Properties of the Maximum Likelihood Method of Phylogenetic Estimation and Comparison with Distance Matrix Methods. *Systematic Biology*, 43(3):329–342, 1994.

[263] Z. Yang. Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. *Journal of Molecular Evolution*, 39(3):306–314, 1994.

[264] Z. Yang. A Space-Time Process Model for the Evolution of DNA Sequences. *Genetics*, 139(2), 1995.

[265] Z. Yang. *Computational Molecular Evolution*. Oxford University Press, 2006.

[266] Z. Yang. *Molecular Evolution: A Statistical Approach*. Oxford University Press, 2014.

[267] Yikrazuul. Base pair Adenine Tyhmine (AT), 2008. Online: https://en.wikipedia.org/wiki/File:Base_pair_AT.svg. Accessed: 2018-08-04. The image is released into the public domain.

[268] Yikrazuul. Base pair Guanine Cytosine (GT), 2008. Online: https://en.wikipedia.org/wiki/File:Base_pair_GC.svg. Accessed: 2018-08-04. The image is released into the public domain.

[269] P. Yilmaz, L. W. Parfrey, P. Yarza, J. Gerken, E. Pruesse, C. Quast, T. Schweer, J. Peplies, W. Ludwig, and F. O. Gl?ckner. The SILVA and "All-species Living Tree Project (LTP)" taxonomic frameworks. *Nucleic Acids Research*, 42(D1):D643–D648, 2014.

[270] T. J. Ypma. Historical Development of the Newton-Raphson Method. *SIAM Review*, 37(4):531–551, 1995.

[271] G. Yu, D. K. Smith, H. Zhu, Y. Guan, and T. T.-Y. Lam. ggtree: an r package for visualization and annotation of phylogenetic trees with their covariates and other associated data. *Methods in Ecology and Evolution*, 8(1):28–36, 2017.

[272] J. Zhang, P. Kapli, P. Pavlidis, and A. Stamatakis. A general species delimitation method with applications to phylogenetic placements. *Bioinformatics*, 29(22):2869–2876, 2013.

[273] J. Zhang, K. Kobert, T. Flouri, and A. Stamatakis. PEAR: A fast and accurate Illumina Paired-End reAd mergeR. *Bioinformatics*, 30(5):614–620, 2014.

[274] S. K. Zhou and R. Chellappa. From sample similarity to ensemble similarity: Probabilistic distance measures in reproducing kernel hilbert space. *IEEE transactions on pattern analysis and machine intelligence*, 28(6):917–929, 2006.