UNSUPERVISED CLUSTERING OF TIME SERIES FROM MICROBIAL MARKER-GENE DATA

by

Michael W. Hall

Submitted in partial fulfillment of the requirements
for the degree of Master of Science

at

Dalhousie University
Halifax, Nova Scotia
July 2016

This thesis is dedicated to my parents, Russell and Linda Hall. You have always supported me in all facets of my life, even when they led me half-way across the country. I couldn't have accomplished any of this without your love and support.

Table of Contents

List of Tables

List of Figures

Abstract

Microorganisms interact with each other and the world around us, impacting every environment that they inhabit. DNA sequencing technology allows us to monitor entire communities of microorganisms. Using taxonomic marker genes, the abundance of thousands of microbial species can be tracked across time. Marker-gene data sets are often very large, requiring data reduction techniques for effective analysis. The typical approach involves clustering the DNA sequences by sequence identity, grouping similar sequences into operational taxonomic units. The emergence of marker-gene data sets with a temporal component offers opportunities to cluster genes based on temporal correlation rather than sequence identity; such an approach may be more effective in revealing ecologically meaningful associations. In this work, we describe an algorithm and software package for clustering marker-gene data based on time-series profiles. We present an efficient, interactive, and cross-platform solution that takes the user from raw sequence data to informative visualizations of the inferred clusters. We validate our method on simulated data and apply it to several longitudinal marker-gene data sets including faecal communities from the human gut, and communities from a freshwater lake sampled over eleven years. Within the gut, the segregation of the time series around a food poisoning event was immediately clear. In the freshwater lake, an annual summer bloom seasonal dynamics were isolated and highlighted by our method. We show that high sequence similarity between marker genes does not guarantee similar temporal dynamics. As a result, clustering based on sequence identity alone would hide many important patterns in these data sets. Our algorithm and visualization platform bring these patterns back to the surface. Finally, we demonstrate that multiple time series can be clustered simultaneously, providing a unique way to visualize marker-gene data sets with both longitudinal and cross-sectional components.

List of Abbreviations Used

AMI Adjusted mutual information

bp Base pairs

DBSCAN Density-based spatial clustering of applications with noise

DNA Deoxyribonucleic acid

eLSA Extended local similarity analysis

HDF5 Hierarchical data format, version 5

MC-TIMME Microbial counts trajectories infinite mixture model engine

MI Mutual information

MRI Magnetic resonance imaging

OTU Operational taxonomic unit

RNA Ribonucleic acid

rRNA Ribosomal RNA

STS Short time-series

Acknowledgements

A huge thank you to all of the members of the Beiko and Blouin labs for all of your helpful suggestions, input, and support. I would also like to thank Fiona Whelan, Robin Rohwer, and Jackie Zorz for test driving the software and providing lots of input on which features biologists actually want. Lastly, I would like to acknowledge the things that kept me sane the past few years (in no particular order): coffee, GURPS, beer, board games, knitting, the Halifax music scene, our cat Margo, and hours spent sitting and staring into the Atlantic.

Chapter 1

Introduction

1.1 Describing Microbial Community Structure with Marker Genes

Groups of microorganisms live, grow, and interact all around us, invisible to the naked eye. These communities of microorganisms are more important than many of us realize. For example, it has long been known that microbes play a very significant role in greenhouse gas cycling [62], and more recent work has shown that certain microbial communities can clean up anthropogenic environmental contamination [45]. The links between microorganisms and certain diseases, such as malaria and tuberculosis, have been clear for many decades, but new links are beginning to form connecting microbial communities with a wide range of disorders such as depression [24], obesity [42], asthma [49], and diabetes [80]. Much of the recent work has been enabled by the decreased cost and increased reliability and throughput of biological sequencing technologies (including DNA, RNA, and protein sequencing). Using a combination of these techniques, it is now possible to assess what we refer to as the "microbiome" of an environment; that is, the constituent microorganisms in an environment, their genes and genomes, and the products of those genes and the environment [87]. As a result of these technological improvements, the past several years have seen a concerted push towards assessing and analyzing the microbial community composition of nearly every conceivable environment. These range from the more obvious human-associated microbial communities (e.g., the Human Microbiome Project [79]) to more obscure environments such as indoor rock-climbing walls [8], inflatable children's pools [66], and apple tree flowers [72].

The microbiome of an environment can be studied at many different levels. Each level tries to answer a different question with a different approach:

- "who is there?"
  - "Marker-gene" DNA sequencing to obtain a taxonomic profile

- "what can they do?"
  - "Metagenomic" sequencing of all DNA from an environment to capture genes of entire microbial community
- "what are they doing?"
  - "Transcriptomic" sequencing of RNA transcripts of the genes that are actively being transcribed
  - "Metametabolomic" identification of metabolites and other small molecules
- "who is doing what?"
  - "Multi-omics" techniques that relate and connect the previously mentioned components [25]
  - Validation in the lab with cultures and biochemical assays

Marker-gene studies profile the taxonomic composition of the microorganisms from an environment, and are a widely used approach to addressing the question of "who is there?" (Figure 1.1). The first step after sample collection is to extract and purify all DNA from an environmental sample (Figure 1.1B). A single gene or gene fragment is isolated and amplified through a polymerase chain reaction step. The choice of gene is influenced by a number of factors including presence in the taxonomic groups of interest, specificity of DNA primers, gene copy number, and ability to resolve evolutionary relationships [63]. After isolation and amplification, the genes are sequenced (Figure 1.1D). The result is a set of alignable DNA sequences that contain information that can be used to identify the taxonomic classifications of the microorganisms in the environmental sample.

A common choice for microbial community profiling is a fragment of the 16S small subunit ribosomal RNA (rRNA) gene [47]. This gene, which encodes RNA that makes up a portion of the protein-building ribosome, is universally present in all bacteria and archaea, contains both fast and slow-evolving portions (Figure 1.1C), is not often transferred by lateral gene transfer, and is present in a single copy in most microorganisms [63]. The slowly evolving, or conserved, portions provide a consistent genetic "anchor" as a target for gene isolation and amplification that covers broad taxonomic

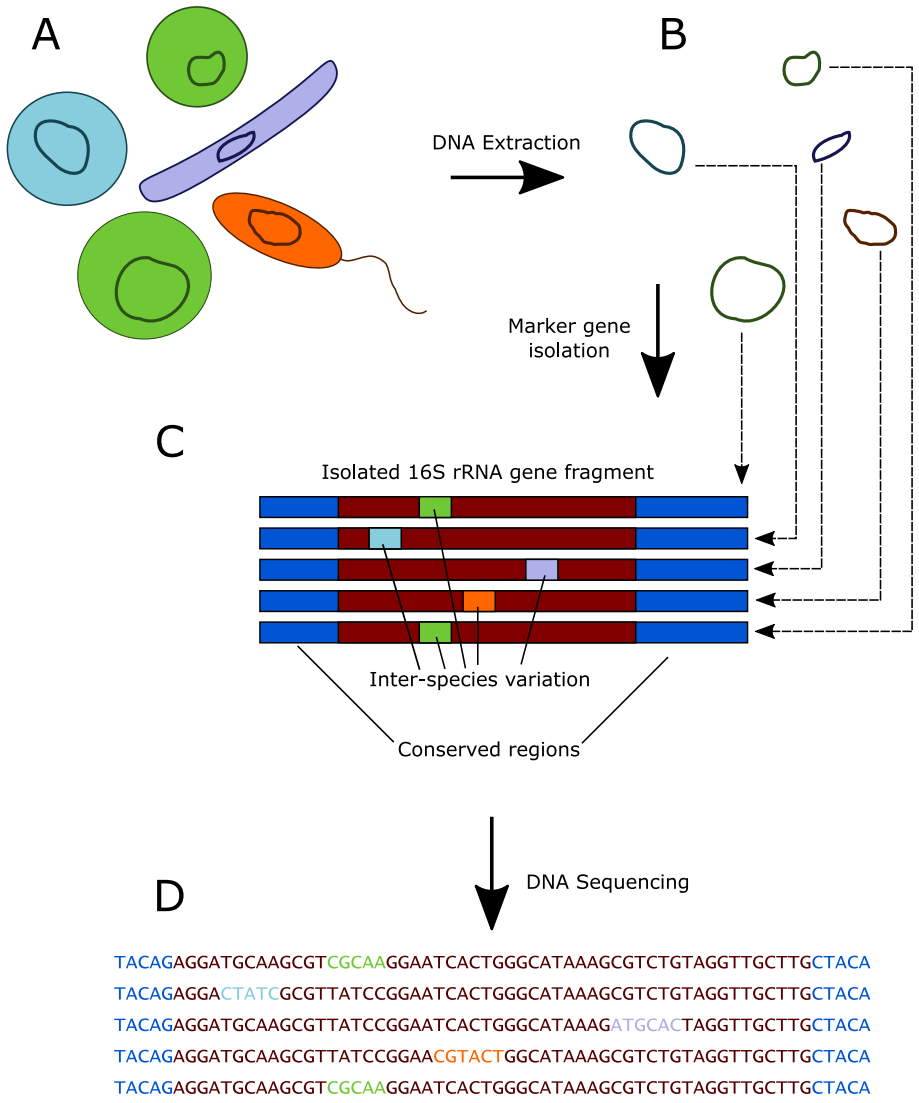

Figure 1.1: Generating the data for microbial community analyses. **A**) A microbial community consisting of four distinct taxa (noted by colour). **B**) The cells are lysed, and the DNA contained within each cell is extracted. **C**) The marker gene (in this case, a fragment of the 16S rRNA gene) is isolated and amplified, and all other DNA is removed. Blue flanking regions represent conserved areas that are consistent across a broad range of taxa. The dark red represents the hypervariable region where inter-species variation is found (denoted by the green, blue, purple, and orange blocks). **D**) The amplified gene region is sequenced, resulting in sequences from each of the constituent taxa. These sequences can be compared against a reference database to identify the taxa.

groups, while the fast-evolving, or "hypervariable", portions provide enough variation between nucleotide sequences to have adequate information to differentiate microbial taxa. Additionally, many reference data sets and alignments exist for the 16S gene (e.g., [17, 60, 14, 39]), easing the task of taxonomic classification and phylogenetic tree building. All data sets used in this thesis are sequences derived from 16S rRNA gene fragments of microbial communities.

1.2 Longitudinal Microbial Community Analysis

Historically, the high cost of sequencing has influenced the number of environmental samples that could be sequenced for a study. In the past decade the throughput of DNA sequencers has rapidly increased, resulting in a significant decrease in the per-kilobase cost of sequencing [73]. To decrease costs even further, protocols have been developed to attach indices (or "barcodes") to sequences that allow dozens or even hundreds of samples to be sequenced in a single run in parallel and sorted *in silico* [4, 29, 11]. Current microbiome studies often contain hundreds of sample points (for example, 761 in [7], and 371 in [59]) which collectively contain millions of sequences. With this increased sequencing capacity, researchers now have the opportunity to look at a wider breadth of environments or to look at a single environment in greater depth.

Microbial communities change over time, sometimes dramatically [70]. This could correspond to a shift from health to disease, as in dysbiosis [56], a catastrophic event like environmental contamination [5], or the forces of selection and drift [57]. As a result, there is value in monitoring a single community across time. By recording these shifts in community composition while they are occurring, we can come closer to understanding their causes and their consequences.

Longitudinal studies make up a small proportion of all microbiome research to date, but the quantity of incoming longitudinal microbial community studies is increasing rapidly [26]. A recent meta-analysis by Shade *et al.* [70] provides an overview and comparison of a selection of longitudinal marker-gene data sets. Notable longitudinal microbiome studies are summarized in Table 1.1. There are diverse ranges of time series lengths, ranging from hours to over a decade, and numbers of sampling points, ranging from under ten to hundreds. In addition to this, sample points

Table 1.1: Summary of notable longitudinal studies.

Study	Environment	Span	Num. of Time Points Per Time Series
Caporaso *et al.*, 2011 [10]	Human stool, skin	1 year	396
David *et al.*, 2014 [16]	Human stool	1 year	191-341
Vergin *et al.*, 2013 [82]	Ocean	9 years	∼108
Gilbert *et al.*, 2013 [12]	Ocean	6 years	72
McMahon *et al.*, 2014 [51]	Lake	11 years	96
Thaiss *et al.* [76]	Mouse stool	48 hours	8

are often unevenly distributed across time, adding an additional challenge for the analysis.

By the nature of longitudinal study design, observations at different time points are not independent. There are often hundreds of thousands of observations which in our case are DNA sequence abundance counts. These abundance values are from thousands of species across hundreds of time points. Techniques from time-series analysis can be applied to this type of data (many of which are summarized in [23]). However, the magnitude of time-series marker-gene data often requires a more efficient implementation of these techniques than what is readily available. A lack of scalability with the implementations of statistical techniques remains a barrier to data analysis. This is particularly acute in analyses such as those involving a time lag. An example of this is identifying shifted time series by quantifying Granger causality [28], a measure of how useful one time series is in predicting another. Run-time increases rapidly as a function of the number of time points, as statistical tests are run for lagged (i.e., shifted) values of a time series.

1.3 Reducing Data Magnitude by Sequence Clustering

The magnitude of a typical marker-gene data set has increased in step with the capacity of modern high-throughput DNA sequencers [65]. The Illumina HiSeq, released in 2011, is able to produce over 50 billion nucleotide base pairs per day, which accounts for approximately 25 million 16S rRNA gene fragments sequences [11]. Newer models

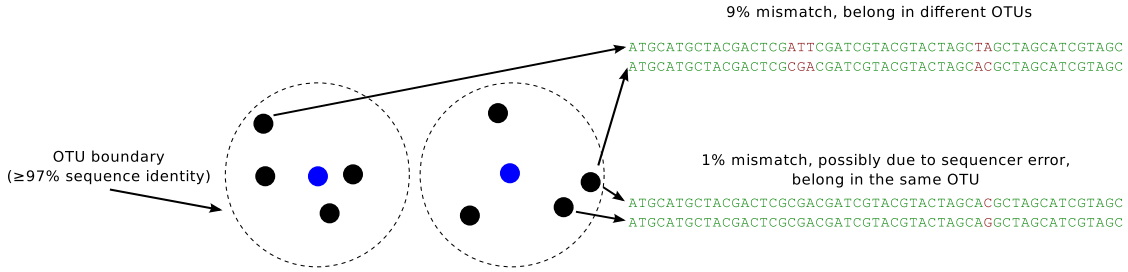

Figure 1.2: Visual representation of OTU clustering. Dots represent DNA sequences. Blue dots are OTU cluster centroids. Dashed lines represent the 97% sequence identity radius around the centroid. All sequences within the dashed lines belong to the same OTU. Example sequences are shown on right, with matching nucleotides shown in green and mismatches in red.

such as the Illumina HiSeq X promise up to 600 billion base pairs per day, or 2 billion gene fragments [36]. As a result, strategies to reduce the magnitude of data are often employed. The most common approach is to group marker genes together based on their similarity. Data set size can be reduced by many orders of magnitude, depending on the grouping method and the diversity of the sampled environment [13].

With 16S rRNA gene data sets, shared sequence identity of aligned sequences is the most frequently used measure of similarity. A 97% identity threshold is the most frequently used cut-off for defining groups of sequences that are similar. Sequences that meet this criterion are grouped together and termed an "operational taxonomic unit" (OTU, Figure 1.2). The OTU is used as a proxy for microbial species, the existence and nature of which is a highly debated topic [2]. A 97% identity cut-off was selected because of strong correlation with results from more traditional DNA-DNA hybridization experiments that are used to determine similarity of overall genetic content between two microorganisms [75]. This method also has the benefit of reducing noise in the data by grouping sequences containing errors (e.g., substitutions are expected in ∼0.1% of DNA bases produced by the Illumina MiSeq platform [46]) together with the corresponding error-free sequence. There are dozens of algorithms for clustering sequences into OTUs (for example, [19, 9, 43, 61, 52, 40, 33]), but they all operate on the same principle of clustering sequences according to their pair-wise sequence identity. As an example, the UPARSE algorithm is an efficient method for creating OTUs from marker-gene data [19]. This greedy algorithm begins by sorting

the sequences by abundance and removing sequences that are below a given abundance threshold. The algorithm starts by aligning all sequences against the most abundant sequence, creating an OTU that contains all sequences which have ≥97% sequence identity to that sequence. The process is then repeated with the set of sequences that have not yet been clustered. As a final step, the low abundance sequences that were initially removed are aligned with the OTU centroids, and are added to the OTU if they meet the ≥97% sequence identity criterion.

1.3.1 Limitations of Clustering by Sequence Identity

Sequence identity-based marker-gene clustering has been the subject of recent criticism. Other studies have analyzed the distribution of the sequences contained within single OTUs and have discovered that there can be dramatic differences in the distribution of sequences that share high sequence identity [78, 20]. Members of a sequence identity cluster are closely related to one another, but evolutionary relatedness does not imply phenotypic or ecological similarity. In particular, lateral gene transfer events can generate two closely related strains with distinct phenotypes [58]. These phenotypic differences can manifest in marker-gene data sets as highly similar sequences with very distinct distributions across the samples. For example, Eren *et al.* (2013) identified strains of *Pelagibacter* that shared 99.57% pairwise sequence identity of a fragment of the 16S rRNA gene, but had anti-correlating relative abundances, with one strain adapted for colder seasons and another for warmer seasons. Similarly, Tikhonov *et al.* (2015) found sequences with >99% sequence similarity that had distinct temporal dynamics, but also sequences with <85% similarity with nearly identical temporal dynamics.

These discrepancies should be a source of concern for biologists who are trying to relate shift in microbial community composition to environmental changes. If the member sequences of an OTU have distinct dynamics, clustering into OTUs risks obscuring them. For example, if one strain is replaced by another closely related strain of equal abundance, this event would go completely unnoticed. Since the OTU abundance is the sum of the abundances of the member sequences, this replacement would appear as an OTU with consistent abundance over time. The extent of this type of within-OTU temporal discordance will vary greatly between data sets, so it is

in the interest of each researcher to identify it in their own data and determine how it may impact their conclusions.

1.4 An Alternative Approach to Sequence Clustering

These observed discrepancies within traditional sequence identity-based clusters, coupled with a desire to reduce the magnitude of marker-gene data sets for the application of statistical analyses, led us to develop an algorithm for the unsupervised clustering of temporal profiles. Our approach measures the similarity between two sequences as a function of their temporal dynamics, rather than the pairwise sequence identities. This results in clusters of sequences that have coordinated increases and decreases in relative abundance over time. These temporal clusters depend only on the distribution of the sequences over time and are therefore agnostic to the underlying phylogeny and taxonomy of the sequences.

This approach is intended to complement traditional sequence identity clustering rather than acting as a replacement. As we will discuss later, closely related sequences most often follow similar temporal patterns. However, contrasting the two methods can reveal interesting cases where the clustering approaches are not in agreement. By reducing a large data set to its distinct temporal profiles, we can facilitate both the application of computationally intensive statistical analyses and the visualization and exploration of microbial temporal dynamics. Algorithms and statistical techniques with poor scaling, such as computing Granger causality, can be run on an otherwise prohibitively large data set after the redundancy has been removed using our method. This approach also allows us to highlight the distinct temporal profiles within an OTU, allowing the researcher to identify and visualize within-OTU temporal discordance.

1.4.1 Overview of Clustering Methods

Clustering algorithms are useful tools for reducing the size of data sets, and identifying structure in data. Hastie *et al.* classify these methods into three general categories: mode-seekers, mixture models, and combinatorial algorithms [34]. A mode-seeking algorithm attempts to discover multiple distinct modes in an estimated probability distribution function, and assigns individual observations (here, the time-series profile for a given sequence) to the closest mode, resulting in a set of clusters. A

mixture-model method attempts to fit the data to a set of pre-defined probability distributions. The clusters are defined by the component probability distributions. Both of these types of clustering methods require an understanding of the underlying probability distributions of the data. The final type, combinatorial algorithms, do not require knowledge of the probability distributions. In most microbiome studies, the complete set of underlying processes that generate the data are not known. This makes combinatorial algorithms ideal for clustering this type of data.

Combinatorial algorithms can be further divided into subcategories: partitioning methods, hierarchical methods, grid-based methods, and density-based methods [30]. Partitioning methods take m data points and attempt to sort them into k clusters, while minimizing the distance between points in a cluster. The data-point-to-cluster relationship can be many-to-one, such as in the popular k-means algorithm [48], or a many-to-many relationship, such as in fuzzy c-means [6]. The k-means algorithm functions by greedy iterative descent, which is efficient, but may give only a locally optimal solution, rather than the global optimum [34].

Conceptually, hierarchical methods generate a tree where the leaves represent the individual data points, and the internal nodes represent the merging (or division) of clusters. These methods do not require the number of clusters, k, to be specified. Cutting the tree at a specified height generates a set of clusters at a given similarity threshold. The tree is created with one of two basic methods: *agglomerative* or *divisive* [34]. In the agglomerative case, the clusters are built from the bottom up, and similar clusters are merged together iteratively. Conversely, the divisive method is a top-down approach where clusters are iteratively split in a manner that maximizes the between-group dissimilarity.

Grid-based clustering methods begin by discretizing the feature space [68]. The density of points in each cell is calculated and sorted, and this is used to identify cluster centers. Points are then added to the clusters by searching the neighbours of their centers. The discretization of the feature space affects both the resolution of the clusters and the runtime of the algorithm. This type of algorithm was designed for large data sets, making it a good candidate for clustering marker-gene data sets.

Similar in concept are the density-based cluster methods. Clusters are defined by areas of high data point density but, unlike grid-based methods, discretization

of the feature space is not required. These methods detect areas of high density that are surrounded by areas of comparably low density to identify clusters. Evenly distributed random noise would most often exist in an area of low density, allowing these methods to properly identify and remove these data points [21]. Density-based methods can make use of any distance measure, making this class of algorithms very flexible.

1.4.2 Unsupervised Clustering of Time Series

Longitudinal data exist in every field. The price of stocks and market trends, natural phenomena from particle collisions to collisions of astral bodies, the electrical signal from a heartbeat that is measured in less than a second, or the evolutionary history of a species that spans hundreds of millenia. Time-series data are ubiquitous, and so the clustering of similar time series has been the focus of other work in the past. Liao (2005) provides an overview of solutions from across different fields. The first distinction between different methods is the data that are being clustered. In some cases, the time series are being directly clustered by way of a pairwise distance that measures the similarity between two time series. Distance measures used by others for this purpose include the Euclidean distance, dynamic time warping [55], Kullback-Leibler divergence, cross-correlation measures, and the short time-series (STS) distance [53]. In other cases, meaningful features are extracted from the time series and clusters are generated based on those features. These features could be the Discrete Wavelet Transform or Discrete Fourier Transform coefficients that correspond to the lower frequencies, helping to eliminate the impact of higher frequency noise [54].

Many different classes of clustering algorithms have been applied to this problem [44]. Fuzzy c-means has been used to cluster time series of functional MRI measurements of brain activity, battle simulations, and microarray gene expression data, while k-means has been used to cluster battle simulations and word recognition data. Other groups have used an agglomerative hierarchical clustering approach to cluster time series of retail sales patterns, earthquake data, wind tunnel flow velocities, and power consumption trends. A hybrid hierarchical and density-based algorithm was developed to cluster longitudinal gene expression profiles in a highly scalable fashion

[37]. In the microbial genomics field, emergent self-organizing maps, a form of artificial neural network, have been harnessed to group metagenomic sequence fragments that have similar distributions over time [18]. Grouping these sequences together allows for more accurate microbial genome assembly. Each type of data has its own distinct properties that helps dictate which algorithm should be used. There is no universal solution, so we must think critically about the structure of marker-gene data when selecting the most appropriate methods.

1.4.3 Longitudinal Marker-Gene Analysis Tools

While the bulk of existing time-series analysis tools have been developed for other applications, a few have been designed specifically for longitudinal marker-gene data. These include the extended local similarity analysis (eLSA) [88] and MC-TIMME [27]. eLSA computes the significance of the similarity between the time series of every pair of OTUs, including time-lagged associations. It can also compute the significance of the similarity between microbial time-series and environmental metadata. The purpose of the tool is to create an association network that can be used to generate hypotheses about interactions between microorganisms. It does not explicitly cluster the data using its results, so it is not considered a clustering algorithm. However, its results could easily be used to generate clusters using the association networks and an appropriate algorithm such as Markov clustering [81]. While useful, it does not adequately solve the problem of time-series clustering. The next tool, MC-TIMME, uses Bayesian statistics to group microbial species together based on how well they fit to *a priori* "prototype" time-series patterns. It can also aid in the design of longitudinal studies by suggesting sampling schemes that focus efforts around time points that require a higher certainty. This method is a model fitting method and would best fit into the "mixture model" category of clustering algorithms. This method is not ideal for the discovery of structure in marker-gene data sets as it presupposes some knowledge of the microbial community dynamics – something that we are unlikely to have.

1.5 Our Contributions

In this chapter, we have discussed the basics of marker-gene sequencing, clustering algorithms, and time-series clustering. Clustering algorithms are used for the reduction in magnitude of marker-gene data sets; a critical step that enables computationally complex analyses of the data to be carried out. While sequence identity is typically used as the similarity criterion for clustering, we could also employ time-series clustering on longitudinal data. In the subsequent chapter, we describe our contribution: Ananke, an algorithm for clustering time-series profiles of microbial marker-gene sequences. We describe the algorithm in detail, discuss how to assess the resulting clusters, and demonstrate how our method facilitates the exploration of large marker-gene data sets. In Chapter 3, we show the results of applying our algorithm to biological data sets, including human-associated and environmental data. We examine the properties of these data sets and discuss interesting patterns that our method is able to highlight. We conclude in the final chapter with a discussion on future work and possible extensions to the algorithm.

Chapter 2

Time-Series Clustering

> Its name was Unaging Time (Chronos). . . . United with it was Ananke . . . incorporeal, her arms extended throughout the universe and touching its extremities.
>
> *The cosmogony according to Damascius* [86]

This work addresses two problems for microbial marker-gene data sets: data reduction and structure discovery. Marker-gene data sets are often too large to permit analyses to be performed directly on the unique sequences. These data sets can contain hundreds of thousands of unique sequences spread across dozens or hundreds of sample points. Investigating each sequence manually is not feasible, and in many cases automated solutions have too high of a run-time or memory requirement to be useful. Removing redundancy by clustering sequences with similar properties addresses our two problems simultaneously. Clustering the data allows us to aggregate the sequences in such a way that we are processing hundreds, instead of hundreds of thousands, of entities. Clustering by sequence identity into OTUs conserves the information contained in the sequences, but we have discussed previously how it can cause the loss of temporal information. Instead, we will cluster sequences by their temporal patterns, preserving the temporal dynamics that are crucial to understanding how microorganisms interact and respond to their environment and one another.

2.1 Problem Formulation

Our input data are m unique sequence strings with abundance data traced over n time points, forming an $m \times n$ data matrix. Our required output is m cluster labels ranging from 0 to $k-1$ such that sequences with similar abundance count distributions are grouped into the same cluster. Neither the number of clusters, k, nor the time-series

similarity cut-off, represented by ϵ, are known *a priori*.

2.2 Ananke: A Time-Series Clustering Algorithm

We have named our algorithm Ananke after the cosmic deity of inevitability and necessity from ancient Greek mythology. She is the consort of Chronos, the Greek personification of time. Ananke is the "goddess who steers all things" [86], humans and microorganisms alike. The Ananke software acts as a companion to time-series data, allowing researchers to explore the temporal patterns that are buried in the chaos of large next-generation sequencing data sets.

An overview of the algorithm is presented in Figure 2.1. The first stage is accepting the input data. The algorithm requires two sets of data: the marker-gene sequences and their sample identifiers in a standard FASTA file format, and a table file that relates the sample identifiers to their respective time points. The input requirements are kept intentionally minimal to ensure the software can be used without complicated preprocessing steps. Sequence data can be filtered beforehand using the user's preferred quality filtering steps. This step is optional, though it is recommended, as the removal of low-quality sequences will decrease run time for later stages.

The second stage is to tabulate the sequence data, thereby generating a time series for each unique sequence. The sequences are represented by a hash, and counts for each hash are maintained for each time point. These counts are stored as an $m \times n$ matrix of integers in an HDF5 formatted file [77]. This is a binary file format that indexes the data with B-trees, increasing the performance substantially over more traditional storage methods such as ASCII-encoded tables.

The next stage is data filtering. Here we remove any time series that does not meet either a presence, abundance, or proportion criterion. Since each data set is unique, the user chooses their preferred filtering method and supplies their own filtering thresholds. Presence filtering removes any time series that have non-zero counts in fewer than a given proportion of time points. Abundance filtering removes any time series with fewer than a given number of counts. Proportion filtering removes any time series that does not represent at least a given proportion of the total count data. There are a few reasons why data filtering is necessary. First, sequences that have low information content would not contribute meaningfully to the clustering step.

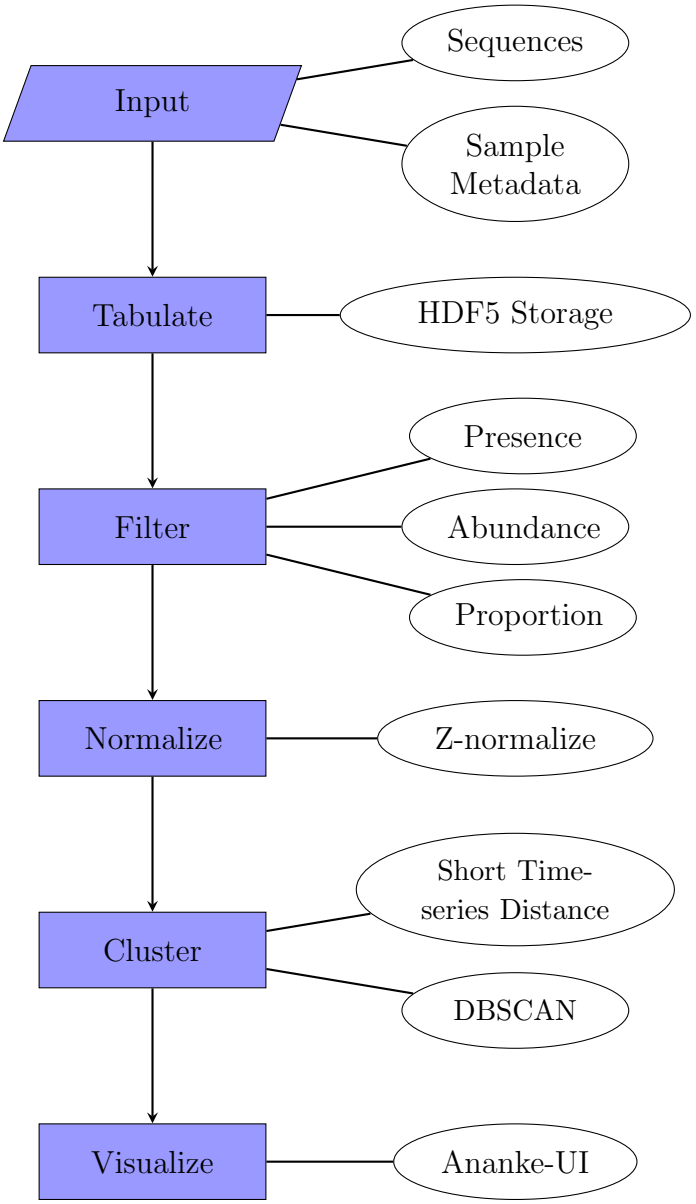

Figure 2.1: Overview of the Ananke algorithm. Steps are shown in blue, with details displayed on the right.

Removing these sequences will decrease run time and memory usage. In the case of clustering by time-series patterns, "low information content" implies sequences that are absent in the majority of sample points. It is also more likely that sequences that are lower in abundance arose from sequencing error [41]. By filtering these sequences out, we remove much of that error from the data. Finally, the clustering stage involves generation of a pairwise distance matrix between sequences and filtering may be necessary to allow this matrix to fit in main memory, especially in instances where the data set is large and limited memory is available. The $m \times n$ data matrix is filtered to become $m^* \times n$ in size, where m^* is the number of time series that meet the user's information content criteria.

The filtered data matrix must then be normalized. This step brings all of the time series onto a common scale for the subsequent pairwise distance calculations. The data are first normalized within-sample to control for discrepancies in sequence depth between time points. This is done by division of the entries in each column by the sum of that column. Next, the rows are Z-normalized by removing the row mean and dividing by the row standard deviation. i.e.,

$$z_i = \frac{x_i - \bar{x_i}}{s_{x_i}}$$

where z_i is the i^{th} row of the data matrix, $\bar{x}_i$ is the mean of the i^{th} row, and s_{x_i} is the standard deviation of the i^{th} row. The Z-score standardization is the recommended procedure for the STS distance, described below [53]. This double-normalization procedure is also used by Dick *et al.* (2009) to allow distance measures to be calculated between time series when time point sampling is uneven [18].

The $m^* \times n$ data matrix is clustered by an algorithm known as DBSCAN [21]. This method is a density-based algorithm that clusters by identifying core points and their neighbours within a given distance threshold, ϵ. Clusters are generated over a range of ϵ values, and the results for each parameter value are stored in the HDF5 data file. In Section 2.2.3 we discuss this clustering algorithm further and justify its use for the problem of time-series clustering.

Once the clusters have been generated, additional metadata can be optionally added to the HDF5 data file prior to visualization. Taxonomic classifications can be generated for each unique sequence using the user's preferred method and reference

database. Similarly, the user can import their pre-existing sequence-based clustering results. This information is utilized and displayed in the visualization component of the software, Ananke-UI. This interactive tool displays data set statistics, plots the time-series clusters as well as sequence-based clusters for contrast, and shows relevant sequence metadata.

2.2.1 Data Storage

The choice of data storage method is critically important for this work. It impacts the run time of the tabulation step and clustering algorithm, as the results are recorded to disk. The data storage format is essential for ease of use. Our algorithm generates different data sets of varying data types (i.e., matrix of integers for the time series, vector of strings for the taxonomic classifications, etc.), and maintaining a single file is critical for ensuring the data are consistent and avoiding problems with file organization and version management. Despite the decreasing cost of disk space, a file format that minimizes the size of the data file would be beneficial. Most importantly, the primary use of our algorithm is for visualization and exploration of large data sets. The data storage format must be capable of handling the retrieval of data from disk efficiently. There must also be compatibility with the format across operating systems and programming languages, since the clustering algorithm was written using Python libraries, while the interactive user interface was built using R libraries. Therefore, our requirements were a storage format that can efficiently retrieve subsets of the data, stores the data in a binary or compressed format, can handle multiple data sets and data types, and is compatible across a wide range of operating systems and programming languages.

Common solutions from within the field did not meet our requirements. Data are often stored as text files in a comma or tab-delimited format. While these can be convenient because they are human readable and easily processed without additional libraries, it is slow to retrieve subsets of the data and is an inefficient use of disk space. There is a file format designed specifically for marker-gene sequence data known as the BIOM format [50]. While utilizing this format would have been ideal for cross-compatibility with other software packages, its current implementation creates unacceptable performance issues. In particular, it maintains two redundant data

structures for each sequence count matrix in parallel: a row-wise sparse matrix, and a column-wise sparse matrix. When inserting new data to these structures, the insertion in one direction is performed in constant time, while the insertion in the other direction is $O(N+m)$, where N is the number of non-zero counts. Our solution is to adapt the HDF5-based BIOM format by removing this redundancy and adding more flexible metadata storage. The HDF5 format meets all of our requirements: subsets can be retrieved efficiently by way of B-tree indices, data are stored in a binary format, multiple data types and data sets can be stored in a single file, libraries exist for Windows, Mac OS, and Linux/UNIX operating systems, and interface libraries exist for both Python and R.

As with the BIOM format, the $m \times n$ sequence count matrix is stored in compressed sparse row format. This data structure stores only non-zero counts, reducing the storage requirements from $O(mn)$ integers to $O(N+m)$ integers. This format stores three vectors: the non-zero counts, their corresponding column indices, and pointers to the beginning of each row. Using this format reduced the storage requirements by between 95% and 97% for the data sets we explore in Chapter 3. This is critical for reducing run time since retrieval of this data from disk can be a slow process on some hardware. The full schema for our file format is shown in Figure 2.2.

2.2.2 Assessing Time-Series Similarity

As discussed in Section 1.4.2, there are many ways to measure the similarity between a pair of time series. These range from simple measures such as the standard Pearson or Spearman correlation coefficients and Euclidean distance, to much more complex measures like local similarity analysis [88] and dynamic time warping [55], which are both able to detect similarity even on a time lag. For our project, we needed to strike a balance between sensitivity and computational scalability. We wanted a distance measure that incorporates the temporal gradient, but is not so computationally complex that it prohibits calculation of pairwise distances for tens of thousands of time series ($\sim 1 \times 10^9$ pairs).

The distance measure that met these requirements was the STS distance [53]. This distance measure was introduced by Möller-Levet *et al.* to compute the similarity of microarray gene expression profiles. It was specifically designed for time series with

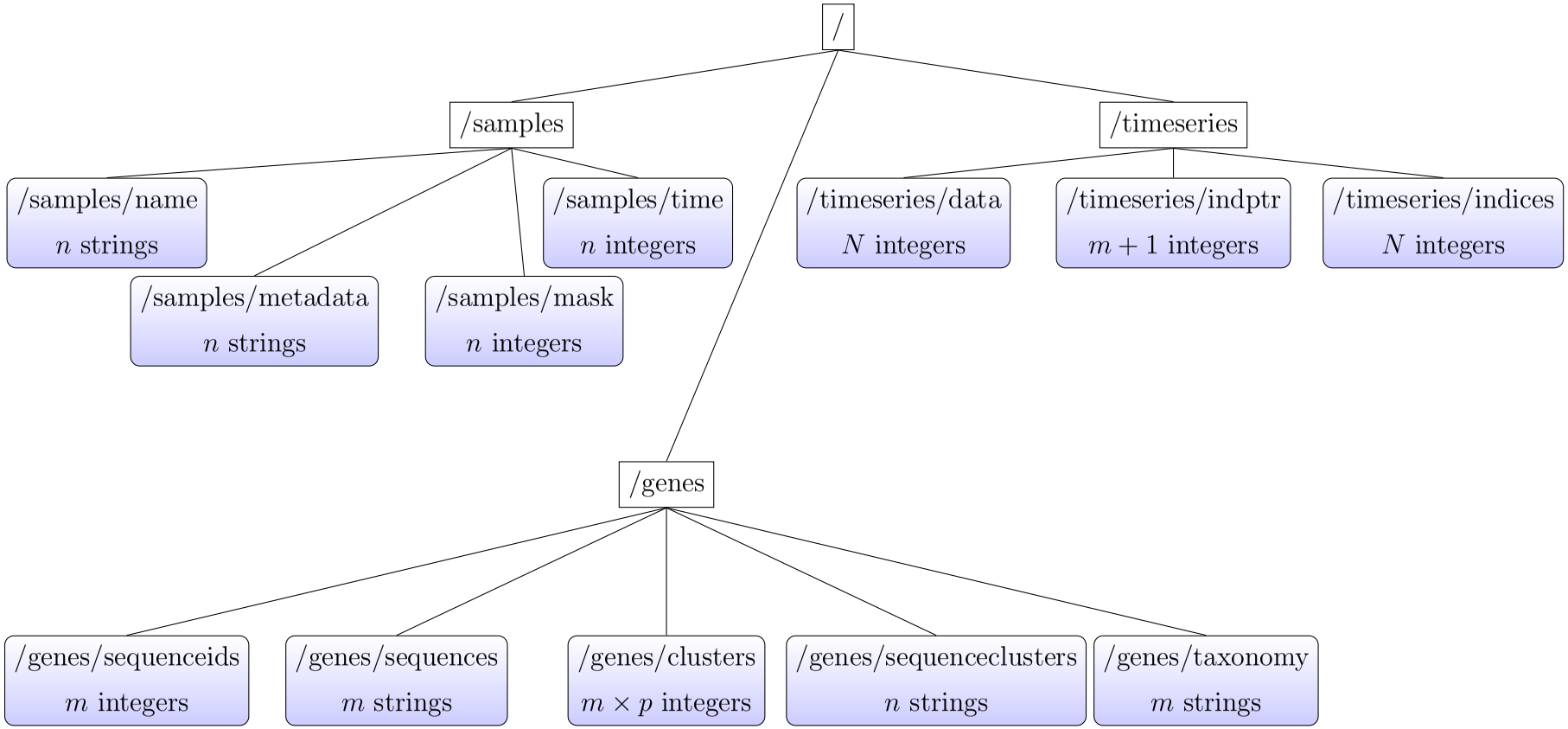

Figure 2.2: HDF5 data storage schema. m is the number of unique sequences, n is the number of time points, N is the number of non-zero sequence counts, p is the number of different values of the ϵ clustering parameter that were computed.

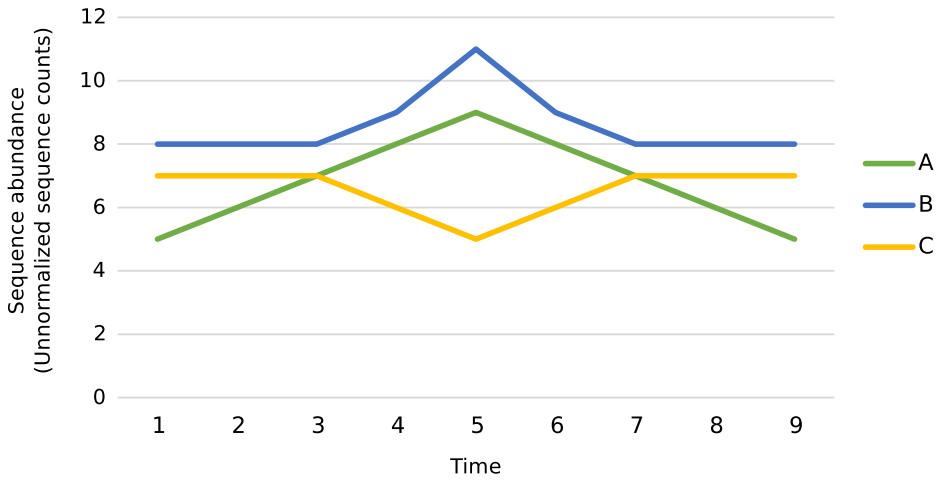

Figure 2.3: A simple example of the differences between Euclidean distance and the STS distance. A, B, and C represent time series of the abundance of individual microbial marker genes. The Euclidean distance between A and B is the same as the Euclidean distance between A and C: $d_{Euclidean}(A, B) = d_{Euclidean}(A, C) = 34$. However, the STS distance between A and B is much smaller than that of A and C: $d_{STS}(A, B) = 6, d_{STS}(A, C) = 20$. In addition to the magnitude of the difference, the direction of change is important for the STS distance.

small numbers of time points. The squared distance between two time series, z_i and z_j, is defined as:

$$d^2_{STS} = \sum_{k=0}^{n-1} \left(\frac{z_{i,k+1} - z_{i,k}}{t_{k+1} - t_k} - \frac{z_{j,k+1} - z_{j,k}}{t_{k+1} - t_k} \right)^2$$

where $z_{i,j}$ is the abundance at the j^{th} time point of the i^{th} time series, and t_k is the time point information (e.g., the number of days elapsed between when sampling began and when the k^{th} sample was taken).

This distance measure computes the slopes between each two adjacent time points and then computes the sum of squared differences between pairs of slopes. By first computing the slopes, this distance measure incorporates the temporal element, as we required, and accounts for uneven sampling by considering the amount of time elapsed between each two adjacent sample points. The STS distance provides a key advantage over simpler measures like the Euclidean distance: it considers the direction of change of the time series being compared. A simple example is given in Figure 2.3. Under the Euclidean distance, only the magnitude of the difference is important

and the direction of the difference is not considered, so time series A and B have the same Euclidean distance as A and C. However, the trajectory of these time series are different: A and B are increasing then decreasing, while the converse is true for C. If these were microbial taxa, we might hypothesize that the populations of taxa A and B have responded positively to a change in the environment that lasts until time point 5, while C has responded negatively. The Euclidean distance cannot account for this important difference in trajectory, and as a result, we recommend using the STS distance for the purpose of clustering time series.

Computation of the STS distance matrix is the most complex step of the Ananke algorithm, both in terms of processing time and memory usage. The filtering step reduces the number of sequences that are clustered from m to m^* and is critical for running the algorithm on systems without large amounts of memory. The run time of each individual STS distance calculation is $O(n)$, and there are $O(m^{*2})$ distances to be calculated. The memory usage is also $O(m^{*2})$, as the STS between each pair of sequences must be stored. The software performs the distance calculations using multiple threads in order to reduce the run time.

2.2.3 Clustering Similar Time Series

We had a number of criteria to guide the selection of a suitable clustering method. First, the method had to scale well for large data sets. Next, since the number of time-series clusters in a given data set is difficult to estimate *a priori*, the method should be parameterized using a distance measure cut-off rather than the number of clusters. This ruled out many methods such as k-means and fuzzy c-means, the method used by the creators of the STS distance [53]. Finally, the ideal method would be able to identify and remove noise in order to avoid creating spurious clusters. These criteria led us to select DBSCAN as our clustering method [21].

DBSCAN is a density-based clustering algorithm that searches for clusters by identifying areas of contiguous high density surrounded by lower-density regions [21]. Points in the feature space are defined as either "core points" or "border points" depending on the definition of two parameters: ϵ, the neighbourhood size parameter, and $MinPts$, the minimum number of points within ϵ distance of a point to consider it a core point. Border points are those within a distance of ϵ of a core point which

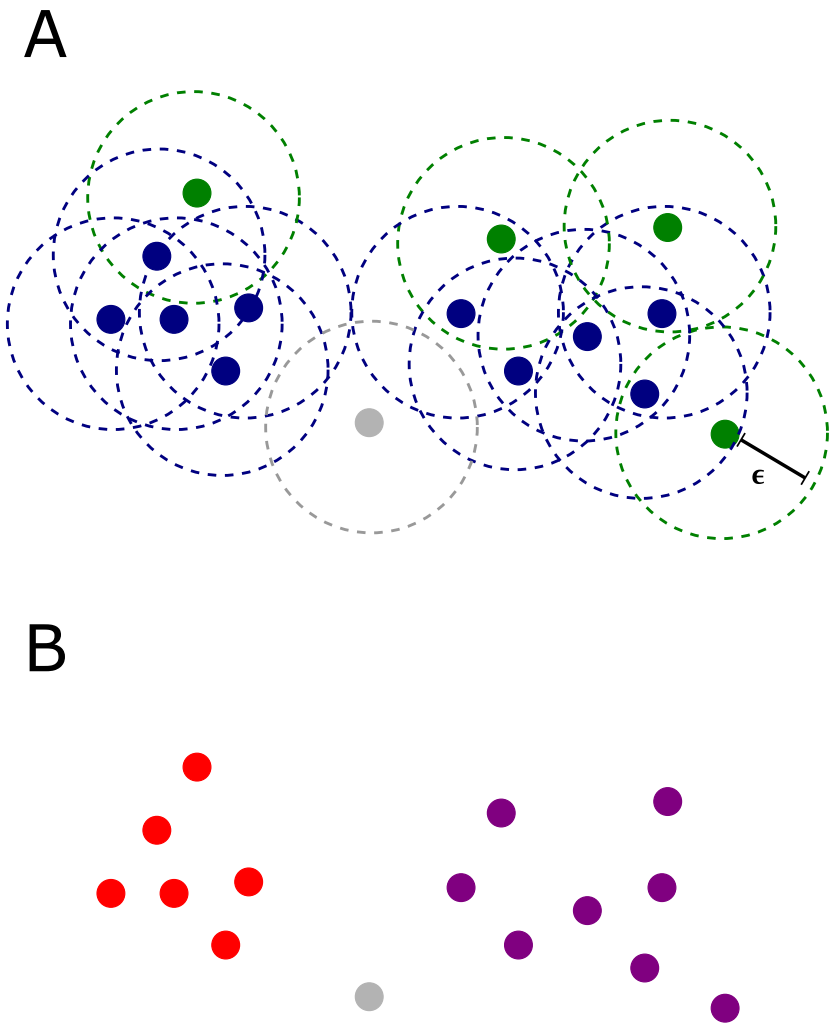

Figure 2.4: An example of DBSCAN clustering with $MinPts = 2$. Points represent sequences being clustered, with the distance between the points representing the level of similarity between the sequences. Points that are closer together are more similar. A) Dashed lines represent the ϵ neighbourhood around a point. Two points are required in the ϵ neighbourhood to be defined as a core point. Blue points are core points. Green points are border points. The grey point is labeled as noise by the DBSCAN algorithm. B) The clusters resulting from the DBSCAN algorithm applied to the points in A. Red and purple are separate clusters and grey is noise.

do not have $MinPts$ neighbours within a distance of ϵ (Figure 2.4A). Clusters are the maximally connected sets of core and border points that are within ϵ of a core point (Figure 2.4B). If a point is not a core point and is not within ϵ of a core point, it is labeled as noise. An interesting property of this algorithm is that it is able to generate non-convex clusters. The run time of the cluster algorithm is $O(mlogm)$, meaning it will scale well with large numbers of unique sequences [21].

The first parameter, $MinPts$, is fixed at 2. This was chosen to maximize the amount of data that is clustered by excluding only singletons as noise. Singletons are sequences with no other time series within a STS distance of ϵ. The algorithm executes quickly on biological data sets, allowing clusters to be computed for large ranges of the ϵ parameter. A default range is provided, but the user can choose any range and step size that they require. The range of STS distances will be different for each data set, since the distances depend on both the number of time points, n, and the number of sequences per time point (also known as sequencing depth). It is not possible to set a range of ϵ parameter values that will work in all cases, but the software can be set to increase the parameter until the clustering results do not change any further.

2.2.4 Multiple Time Series

In some cases, researchers will have several short time series from different areas. For example, several people may be sampled weekly over the course of a month, or a river may be sampled simultaneously at upstream and downstream locations. This study design is known as "cross-sequential" [67] and combines the benefits of cross-sectional and longitudinal study designs. One of the drawbacks to this type of study is that the sampling effort is spread across several distinct time series, rather than being focused on one. It is more difficult to detect and cluster temporal patterns when the time series are very short (see Section 2.3.1). By concatenating the sequence abundance time series from multiple environments, we add more information that can be beneficial to the clustering step.

The implementation of this is straightforward. We can denote the number of individual time series with P. The P time series are normalized within-sample as with the single time series case. For the Z-score normalization of each time series, we

use the within-time-series mean and standard deviation. That is,

$$z_i = \frac{x_i - \bar{x}_{ip}}{s_{x_{ip}}}$$

where p indexes the multiple time series, $\bar{x}_{ip}$ is the mean of time series p, and $s_{x_{ip}}$ is the standard deviation of time series p. Next, the STS distance is taken to be a sum of the STS distances from the P time series:

$$d^2_{STS} = \sum_{p=0}^{P-1} \sum_{k=0}^{n_p-1} \left(\frac{z_{i,k+1} - z_{i,k}}{t_{k+1} - t_k} - \frac{z_{j,k+1} - z_{j,k}}{t_{k+1} - t_k} \right)^2$$

where n_p is the number of time points in time series p. Since this is simply a summation and is therefore commutative, the order in which the time series are considered does not affect the distance measure.

2.3 Assessing Cluster Quality

It is a difficult task to evaluate the clusters generated by our algorithm. There are no biological data sets where the ground-truth for the quantity and variety of temporal dynamics is known. Sequence-based clustering can be evaluated by the use of mock communities, where the proportion of each taxonomic group in the sample is known [69]. As of this writing, a longitudinal mock community data set has not been created. One solution is to use simulated data sets to assess the behaviour of the algorithm, since they are generated with a known ground-truth. Insights into the algorithm's performance on biological data can be obtained by comparing with metadata such as sequence-based clusters and taxonomic classifications.

2.3.1 Simulations

In order to assess the clusters that are output by the algorithm, artificial time series were simulated, clustered by the algorithm, and compared against the ground-truth. We generated artificial patterns of temporal variation that represent ecological events or patterns that users may wish to identify in a large data set (Figure 2.5). Appearance, disappearance, and conditional rarity [71] patterns may indicate a significant change in the environment, so it is important that our method is able to cluster them appropriately. Disappearance and appearance events are defined by a transition to

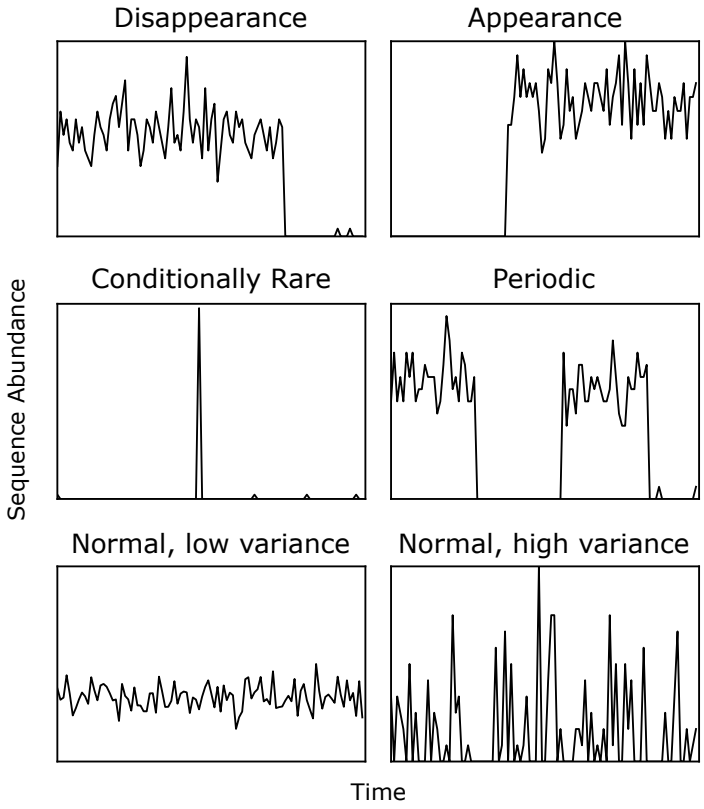

Figure 2.5: Examples of the temporal patterns generated for the data simulations.

or from, respectively, near-zero abundance. Conditionally rare organisms are those that have a bimodal abundance distribution. The first mode is near-zero, with the second mode centered on a positive value [71]. These can be thought of as transient "blooms" where an organism occasionally increases from very low abundance to a relatively high abundance. Periodic patterns are relevant when analyzing seasonal changes in natural environments, so our method must be able to cluster sets of sequences that show multiple coordinated increases and decreases in the time series. Time series that follow a normal distribution with lower variance represent organisms that remain relatively consistent in abundance over time, while those with a high variance represent noisy data. Normally distributed time series are more difficult to cluster as they lack large slopes to influence the STS distance measure.

The simulated time series were generated as follows:

1. Set time series length parameter, n, and number of time series per cluster, r
2. Generate random values for the properties of the time series
    - For disappearance, appearance, or conditionally rare time series, generate a random time point between 1 and n at which the event will occur
    - For periodic events, generate the period of the event and the starting abundance (high or low)
3. Create a binary template of the time series (values are either high or low)
4. Repeat r times:
    (a) For each data point in the template, x_i, add random noise that follows a normal distribution with mean x_i and variance $\frac{x_i}{10} + 0.01$
    (b) Scale the entire time series by a factor generated from a Weibull distribution with $k = 0.3$

The parameters for the normally distributed random noise were chosen to create time series that closely resembled that seen in biological data sets. The Weibull distribution with parameter $k = 0.3$ was chosen for the scaling factor because of its long tail. When applied to a group of time series that originated from the same template, this resulted in many time series with low abundance and few with high

abundance. This was done to mimic the "error cloud" sequence model, where an abundant sequence correlates very well with many low abundance sequences that are the result of errors from the DNA amplification step or errors from sequencing [78].

Data were simulated for time series lengths $n = 5, 10, 25, 100, 250, 500, 1000$ and number of time series per cluster $r = 100$. The clusters were scored by comparing the predicted cluster labels against the ground-truth using the adjusted mutual information (AMI) score. This score is a chance-corrected version of the mutual information (MI) score, which measures the dependence between two variables. In our case, the MI score measures how much information could be obtained about the ground-truth from the predicted clusters. The mutual information between two clustering results, U and V, is defined as [83]:

$$MI(U,V) = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{n_{ij}}{N} log \frac{n_{ij}/N}{a_i b_j / N^2}$$

where R is the number of clusters in U, C is the number of clusters in V, n_{ij} is the number of marker-gene abundance time series that are common to the clusters U_i and V_j, a_i is $\sum_j n_{ij}$, b_j is $\sum_i n_{ij}$, and N is $\sum_{ij} n_{ij}$. The chance correction is calculated as follows:

$$AMI(U,V) = \frac{MI(U,V) - E(MI(U,V))}{max(H(U),H(V)) - E(MI(U,V))}$$

where $E(X)$ is the expectation of X, and $H(U)$ is the entropy of U, defined as $H(U) = -\sum_{i=1}^{R} \frac{a_i}{N} log \frac{a_i}{N}$. The AMI score is in the range $[0, 1]$, where 0 implies no shared information between the two clustering results, and 1 implies that the clusters are identical. The clusters generated by our algorithm matched the ground truth well even for very short time series, but the AMI decreased as the number of time points increased past 100 (Figure 2.6). Our algorithm yielded average AMI scores > 0.8 on time series with as few as ten time points. However, the AMI scores were considerably lower for time series of length 500 (median AMI = 0.67) and 1000 (median AMI = 0.64). The drop in AMI scores for longer time series is a consequence of the STS distance metric. For longer time series, the sum of small differences (which are a result of random noise added to each point) can overwhelm the effect of the true pattern. To reduce the impact of random noise, longer time series could be smoothed by averaging over a sliding window. This would reduce the magnitude of the slopes

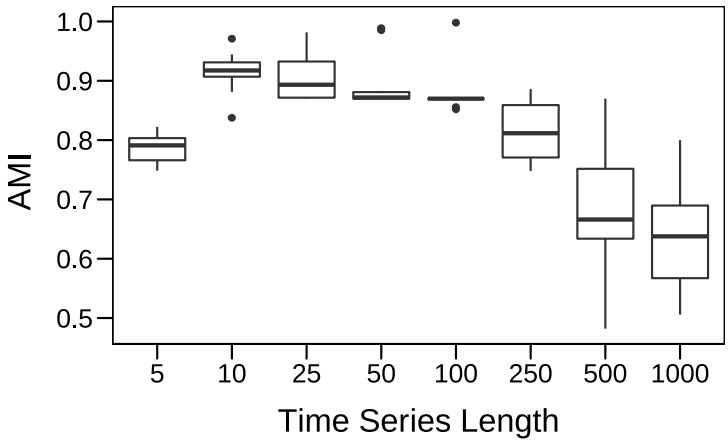

Figure 2.6: Adjusted Mutual Information score for the clustering results of the artificial data. The boxes denote the lower quartile, median, and upper quartile of ten replicates. Individual points are outliers that fall outside of 1.5 times the inter-quartile range.

that are due to random noise, resulting in a smaller cumulative impact on the distance measure.

In the majority of the simulations, the low-variance and high-variance normally distributed time series were flagged by the clustering algorithm as noise. These two types of temporal patterns were often placed into the same cluster, which prevented our method from achieving a perfect AMI score of 1. Since there is no large slope consistently present in these simulated patterns, the STS distance measure is not given enough information to separate the low-variance from the high-variance group.

2.3.2 Assessing Clusters in Biological Data

Simulations are a useful way to assess algorithms in a controlled fashion, but simulated data can only approximate biological data sets. There are many important differences between our simulations and biological data sets, most importantly the number and variety of temporal patterns, and the characteristics of random noise. It is useful to have a way to check the clusters even when no ground-truth is known in order to ensure that the behaviour of the algorithm is as expected. This can be accomplished by analyzing the sequence metadata and contrasting it with the clusters. In particular,

the taxonomic classifications of each sequence are useful for this task.

For this work, we consider each sequence as both a time series and as a series of DNA nucleotides. As discussed in Chapter 1, these DNA nucleotides can contain enough information to identify the taxonomy of the microorganism that contained the sequence in its genome. This is accomplished by comparing each DNA sequence to a reference data set and inferring the closest match. While this process is less accurate for novel microorganisms and those which might challenge our hierarchical Linnaean classification system, it is, in general, an accurate process (>89% accuracy at the genus level with 400bp 16S rRNA gene fragments [85]). For our work, we used the Naïve Bayes classifier from the Ribosomal Database Project (version 2.2) [14] in conjunction with the GreenGenes reference data set (August 2013 revision) [17]. Each unique sequence was classified by comparing the composition of all 8-mers (eight letter nucleotide subsequences) within each query sequence against the 8-mers of the sequences in the reference data set. The sequences were classified at seven taxonomic levels: kingdom, phylum, class, order, family, genus, and species. A $\geq$60% posterior probability was required at each level. The taxonomic classifications are stored in the data file and can be retrieved through Ananke-UI.

A time-series cluster is a group of DNA sequences that show similar abundance patterns over time. Sequences may show comparable distributions over time because they are sequencing or amplification errors associated with a more abundant sequence, or they may have originated in microorganisms that are closely related and therefore have similar phenotypic traits, allowing them react to environmental changes in similar ways. In either of these cases, one would expect that the taxonomic classifications of these sequences would be in agreement, especially at higher taxonomic levels. Therefore, our intuition is that many time-series clusters will have homogeneous taxonomic compositions, with greater homogeneity at higher taxonomic levels such as domain and phylum. Exceptions to this will certainly exist; for example, if two distantly related microorganisms are in a cooperative relationship their temporal dynamics may be in lockstep despite different taxonomic classifications. However, it is anticipated that the majority of time-series clusters will be taxonomically homogeneous, especially at small ϵ values. It is important to note that this intuition does

not extend to the converse scenario. A group of taxonomically consistent microorganisms is not necessarily going to be temporally homogeneous. This is especially true at higher taxonomic levels, but even at lower taxonomic levels, inter-strain diversity can be enough that even closely related taxa demonstrate different phenotypes (and therefore different temporal dynamics).

Taxonomic homogeneity was assessed by calculating the Simpson index of each time-series cluster. The Simpson index is the probability that any two sequences chosen at random will belong to the same taxonomic group [74]. Mathematically, this is defined as:

$$\sum_{i=1}^{N} p_i^2$$

where N is the number of distinct taxa and p_i is the proportion of the sequence abundance that belongs to taxon i. This was calculated at each of the seven taxonomic levels. In general, average Simpson indices were > 0.9, suggesting high levels of taxonomic homogeneity within time-series clusters. Detailed results for each individual data set are shown in Chapter 3.

2.4 Data Exploration and Visualization

The objectives of this algorithm are to reduce the magnitude of marker-gene data sets and to facilitate the discovery of temporal structure within the data. By collapsing the data down to the distinct temporal patterns, our time-series clustering algorithm accomplishes both of these goals. The next task is to ascribe biological meaning to these clusters. That is, we want to determine the biological processes or environmental influences that might explain the temporal abundance patterns. Automated computational methods are certain to play a role in this task, but we must not discount the researcher's intimate knowledge of the environment that they are studying. Allowing the domain expert to interactively explore and visualize the data is critical for discerning the biological significance of the time-series clusters.

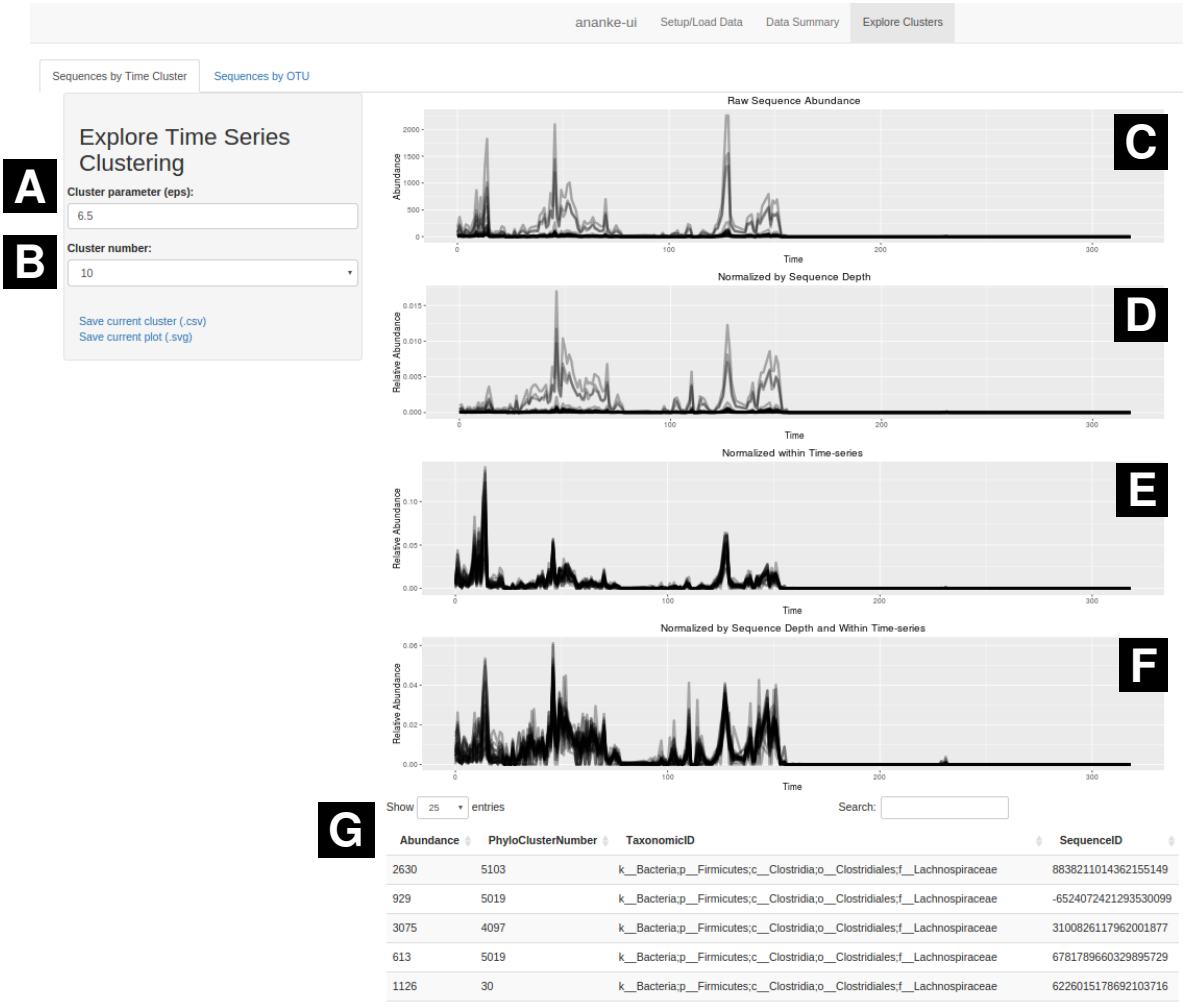

Figure 2.7: An example of the Ananke-UI showing a time-series cluster. A) Control for selecting ϵ clustering parameter. B) Selection box for time-series cluster. C) Time series without normalization. D) Time series normalized within each time point. E) Time series normalized within time series. F) Time series normalized both within time points and within time series. G) Interactive table displaying the metadata for each sequence, including the abundance, sequence-based cluster number, taxonomic classification, and sequence hash.

2.4.1 Ananke-UI

We developed an interactive interface called Ananke-UI using the R language's Shiny library [64]. Our user interface enables exploration of the time-series clusters, presenting them alongside available sequence metadata (Figure 2.7). The web-based, cross-platform utility allows parameters and values to be selected by the user and related visualizations update automatically. The granularity of the clusters can be modified by selecting a new ϵ clustering parameter (Figure 2.7A). This requires no additional computations, as the clusters are pre-computed for a range of ϵ values and are simply read from the data file. Clusters can be easily cycled through with a drop-down menu (Figure 2.7B), and the time-series plots are automatically generated (Figure 2.7C-F). Four time-series plots are displayed to allow the user to compare the effects of different normalization schemes. Some of the plots (Figure 2.7C-D) emphasize the differences in relative abundance of the sequences, while others (Figure 2.7E-F) place all sequences on the same scale to make the temporal dynamics clear. Relevant metadata are presented in a sortable and filterable table (Figure 2.7G). These metadata include the total abundance of each sequence, the sequence-based clustering values (if available), the taxonomic classification (if available), and the unique sequence identifier.

It is also possible to use Ananke-UI to display the sequence-based clusters (i.e., the OTUs). In this case, the sequences are coloured according to the time-series cluster to which they belong. An option exists to exclude from the plot all sequences that were classified as noise by the clustering step, as they may interfere with the visualization. This colouring is meant to draw attention to OTUs that contain multiple distinct temporal patterns. For example, the *Clostridiales* OTU in Figure 2.8 contains two distinct temporal dynamics in spite of their high sequence identity. Around time point 50, three of the sequences respond positively to a change in the environment and are able to increase in abundance relative to a fourth sequence which subsequently decreases in abundance below detection. It is important to detect this type of inconsistency within an OTU as it could impact the conclusions of analyses based on OTUs. If a data set contains a large amount of temporal inconsistency within OTUs, it could signify that the 97% sequence identity threshold is too broad, and a 98%, 99%, or 100% sequence identity threshold may be more suitable.

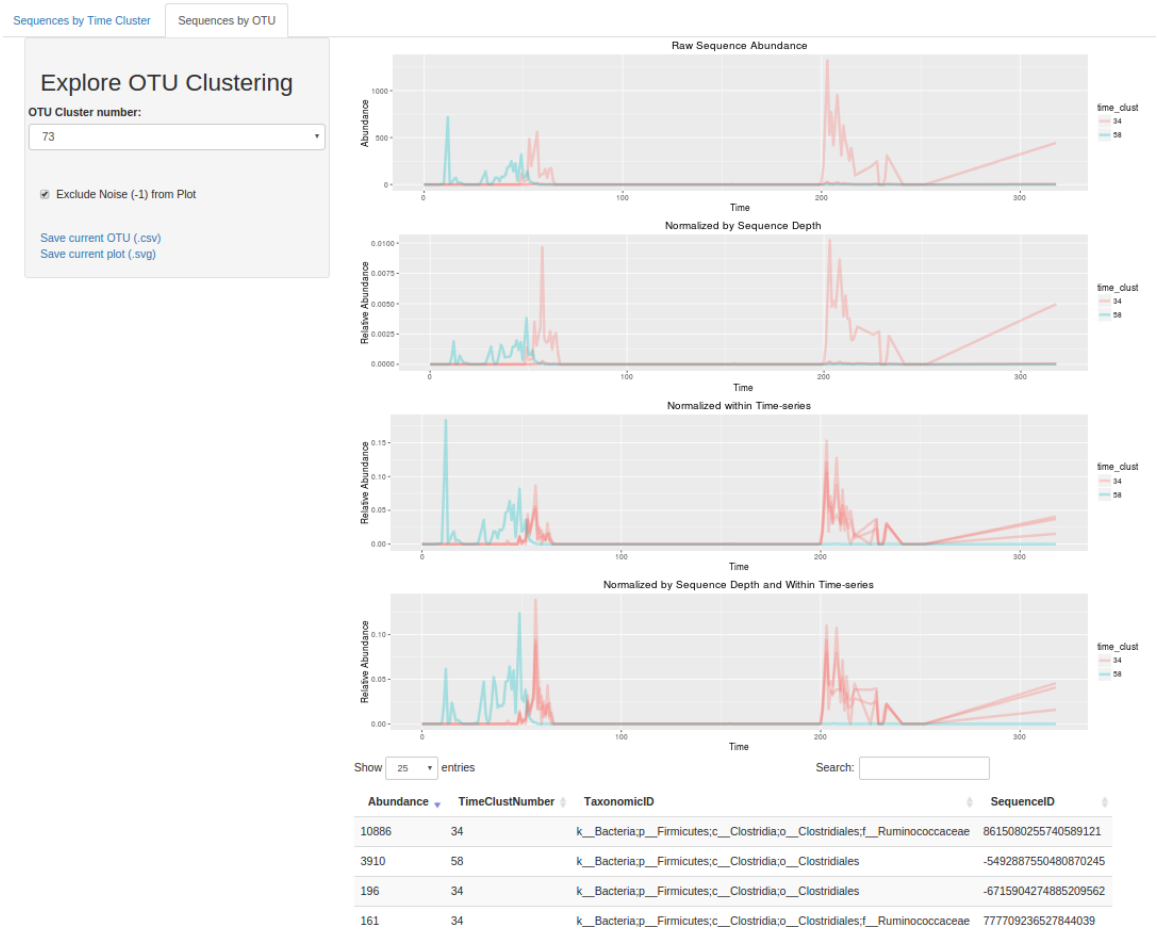

Figure 2.8: An example of the Ananke-UI showing an OTU defined by 97% sequence identity. Individual sequences are coloured by their time-series cluster. This OTU contains sequences that follow two distinct temporal patterns (red and blue).

2.4.2 Time-Series Cluster Granularity

With sequence identity-based clustering methods, a static cut off can be used across different data sets. As discussed previously, 97% sequence identity is the most common criterion used to define sequence-based clusters such that the resulting OTUs can be used as a proxy for microbial species. This particular value is backed up by correlations to the time it takes for 70% DNA-DNA hybridization, an assay used to determine how closely related two microorganisms are [75]. The time-series clusters are based on the ϵ clustering parameter of the DBSCAN algorithm. This parameter defines the radius of the neighbourhood around a point in which we search for additional points to form a cluster [21]. We define similarity between two time series with the STS distance measure, so the ϵ parameter is also expressed as a STS distance. This distance depends on the number of time points as well as the sample sequence depth. Additionally, each user has different research questions that will not all be answerable with a static distance cut-off. As a result, we recommend that users explore various ϵ values and select the values that best suit their research questions.

The ϵ parameter allows the user to control time-series cluster granularity. As this value is increased, the neighbourhood around points grows larger, the clusters more coarse, and the time series contained within a given cluster become more dissimilar. Conversely, as the value is decreased, the time series contained within a given cluster become more similar (Figure 2.9). The level of granularity is controlled through Ananke-UI by selecting the desired ϵ value in a numerical input box. The pre-computed clusters are read from the data file and presented to the user.

As a function of ϵ, the number of time-series clusters follows a regular pattern. At low ϵ values, there are few time-series clusters. As ϵ is increased, the number of clusters increases. After a certain point, a maximum number of clusters is obtained and increasing ϵ further will decrease the number of clusters. This results in a generally concave downwards shape for the number of time-series clusters as a function of ϵ. For low ϵ values, since the neighbourhood around each point is small, most time series have no neighbours within a STS distance of ϵ and are labeled as noise. The increased number of sequences labeled as noise results in a small number of time-series clusters. As ϵ is increased, the radius around each time series grows, often moving them from the noise designation into a time-series cluster. As ϵ increases, the clusters

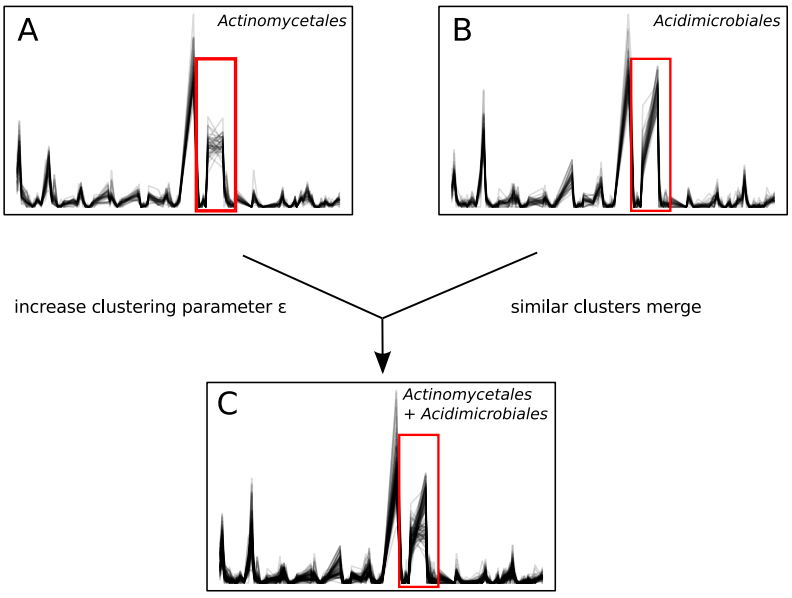

Figure 2.9: Demonstration of the effect of the ϵ cluster parameter. Normalized time-series clusters from two orders of Bacteria, A) *Actinomycetales* and B) *Acidimicrobiales*. The red area highlights an area of the time series that differs significantly. C) As ϵ is increased, the clusters become more coarse. The two time-series clusters merge into one.

begin to merge with one another, resulting in an eventual decreasing trend for the number of clusters. We recommend that the user begins exploring their data with the ϵ value at which the largest number of clusters is obtained. This value will provide a good separation of time series, as well as a balance between the number of sequences labeled as noise and the number of sequences in large, dense clusters that are difficult to analyze.

Chapter 3

Application to Biological Data Sets

In this chapter, we demonstrate the utility of the Ananke algorithm by applying it to biological data sets. Four diverse data sets were selected and processed using the algorithm described in the previous chapter. Two of the data sets contain a single time series each, and the final data sets contains multiple time series. The single time series studies are derived from human stool and a freshwater lake, while the multiple time series data originate from the stool of 43 human subjects. The data sets vary in number of time points, length of each time series, and sequence depth, allowing us to assess the algorithm across various experimental parameters. As discussed in Section 2.3.2, the taxonomic classifications for the sequences were generated by the Ribosomal Database Project's naïve Bayesian classifier (version 2.2) [14] trained against the GreenGenes reference set (August 2013 revision) [17]. Sequence identity-based clusters were computed by UPARSE [19], described briefly in Section 1.3, at a 97% sequence identity threshold with a minimum sequence abundance of two. These methods were chosen because they are frequently used in microbiome studies. This allows for us to contrast our time-series clusters with realistic taxonomic classifications and sequence-identity clusters.

3.1 A Year of Faecal Samples

We first demonstrate our time-series clustering algorithm on the faecal microbial data set from David *et al.* (2014) [16]. The data are 16S rRNA gene fragments taken from one person at 191 time points over 318 days. During the sampling period the subject contracted a food borne illness that resulted in a significant shift in the microbial composition of their stool. As a result, the data contains a variety of temporal patterns that make it an ideal test set for our algorithm.

The data set contains 26,250,106 total and 1,200,847 unique sequences. To reduce the magnitude of the data, they were filtered to include only sequences which appeared

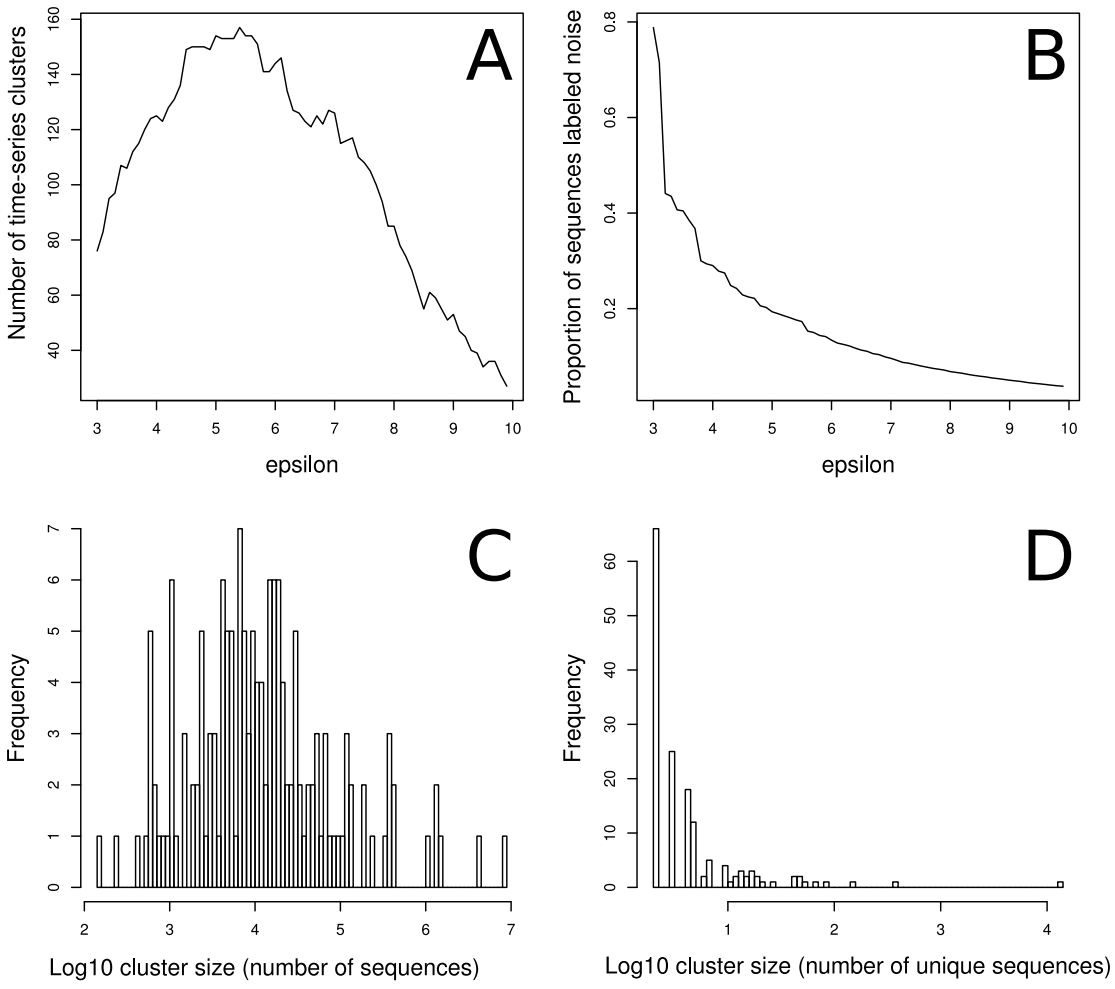

Figure 3.1: Data set statistics for the faecal data. A) Number of time-series clusters vs. ϵ. B) Proportion of data labeled as noise vs. ϵ. C) Histogram of log cluster size by total number of sequences. D) Histogram of log cluster size by number of unique sequences.

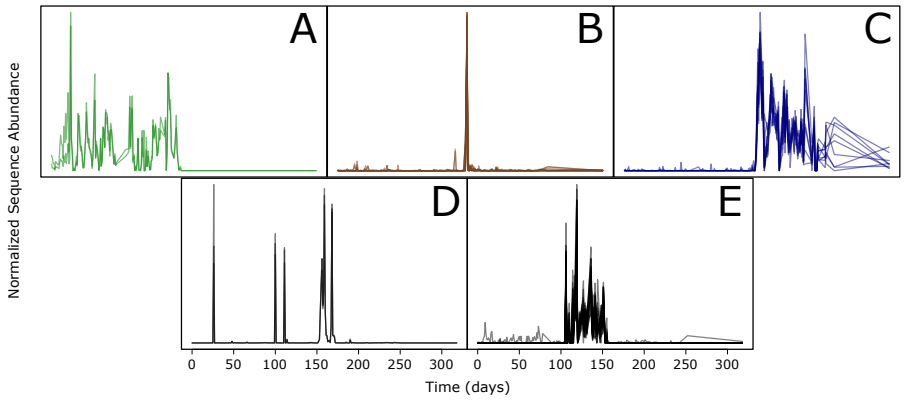

Figure 3.2: Selected time-series clusters of microbial communities from human faecal samples that coincide with the food-borne illness event around day 159. A) Two sequences from the family *Coriobacteriaceae* present only before the event. B) A cluster of seventeen sequences that increase in relative abundance during the food poisoning incident, many belonging to *Enterobacteriaceae*. C) Nine sequences belonging to the family *Lachnospiraceae*, the most abundant classifying to *Clostridium citroniae*. D) Three sequences classifying to the family *Enterobacteriaceae* that are coincident with the food poisoning event and also observed in high relative abundance earlier in the time-series. E) 25 sequences, the majority of which classified to *Ruminococcus bromii*.

in $\geq$15% of time points, reducing the data to 23,533,503 sequences and 14,743 unique sequences. While this drop in unique sequences was significant, only the sequences with low information content were removed and $\sim$90% of the overall data was retained. A maximum of 157 time-series clusters was found at ϵ=5.4, with an average cluster size of 149,894 total sequences and 94 unique sequences. At that ϵ value, 17.6% of the sequence data were labeled as noise.

The subject experienced food poisoning diagnosed to be the result of *Salmonella* sp. just before day 159. The authors of the study showed that the food poisoning event divides the faecal microbial community into three clear segments from days 0-144, 145-162, and 163-240. Once clustered by time series, this segregation is readily apparent (e.g., Figure 3.2A-C). Some sequences seem to disappear completely from the environment after the event, such as a cluster of *Coriobacteriaceae* sequences (Figure 3.2A). Other clusters that were only present in extremely low abundance before the illness, such as a group of sequences classified as *Clostridium citroniae*, appear to

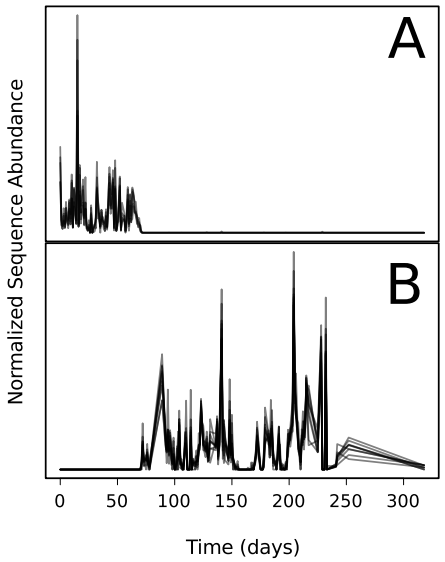

Figure 3.3: Two time-series clusters of sequences that classify to *Akkermansia muciniphila*. One cluster is present only before day 72 (A), while the other is present after day 72 (B).

successfully gain a foothold in the environment (Figure 3.2C). During the food poisoning event, 17 conditionally rare sequences show a large increase in relative abundance (Figure 3.2B). The two most abundant sequences in this spike classify to *Enterobacteriaceae* (the family containing *Salmonella* sp.) and *Haemophilus parainfluenzae*. The remaining sequences belonged to various taxonomic groups including the genera *Leuconostoc*, *Weissella*, *Lactococcus*, and *Turicibacter* from the class *Bacilli*, *Clostridium*, and *Veillonella* from the class *Clostridia*, and two sequences from the genus *Acinetobacter*. A large increase of three abundant *Enterobacteriaceae* sequences was seen at the time of the food poisoning (Figure 3.2D), but these sequences had also been observed earlier in the subject's time points and were assigned to a different time-series cluster.

In addition to the changes induced by the food poisoning event, our method also highlighted several smaller changes in the community that were not noticed by the original authors. Between days 100 and 160, a sudden rapid increase in *Ruminococcus bromii* sequences is observed (Figure 3.2E). This species is known to play an important role in the degradation of resistant starch [90, 1], hinting at a dietary cause for the increase in relative abundance. These *Ruminococcus bromii* sequences

Table 3.1: Average Simpson Index for faecal time-series clusters at ϵ=5.5

Taxonomic Level	Average Simpson Index
Phylum	0.99
Class	0.99
Order	0.99
Family	0.97
Genus	0.92
Species	0.93

are no longer detected in the environment after the food poisoning event. Another event highlighting through time-series clustering occurs at day 72 when several *Akkermansia muciniphila* sequences fall below detectable levels (Figure 3.3A). They are immediately replaced by new sequences that are classified to the same species (Figure 3.3B). *Akkermansia muciniphila* has been linked to obesity [22], diabetes [31], and autism [84]. Further research is required to determine why this replacement occurred and what are the impacts on the host. Our time-series clustering algorithm aids researchers in discovering subtle events like these that could be critical to our understanding of how microbial communities relate to disease.

As discussed in Section 2.3.2, we used the Simpson index to assess the cluster quality. Our assumption is that the majority of time-series clusters will contain taxa with a similar capacity to respond to changes in the environment and as a result they are likely to have high taxonomic homogeneity. We see that our assumption of taxonomic homogeneity of time-series clusters is true for the faecal data set, where the Simpson index was $\geq$0.99 for taxonomic levels above family, and 0.93 for the lowest taxonomic level, species (Table 3.1). Note that the Simpson index calculation includes only sequences which had a classification with $\geq$60% posterior probability at the indicated taxonomic level. This means that some data is excluded from the calculation at species level, explaining why the Simpson index is higher than that of the genus level. The clusters that do not demonstrate taxonomic homogeneity are often the largest clusters, which can contain sequences that share a large coordinated increase or decrease in abundance but are otherwise unrelated. Other taxonomically heterogeneous clusters have sequences with taxonomic classifications that agree at broader taxonomic levels, but the sequence-based classifications are not in agreement

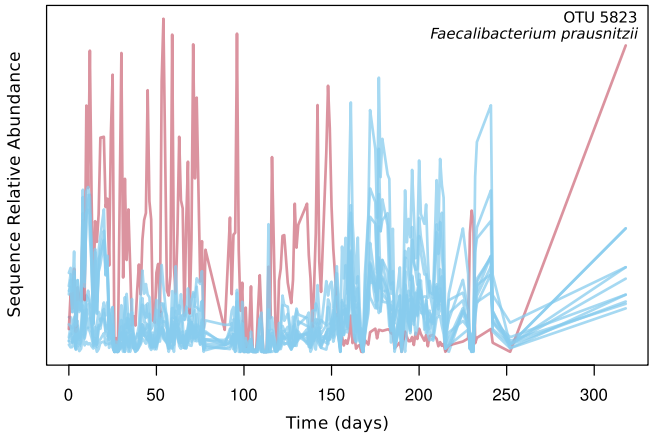

Figure 3.4: A 97% sequence-identity cluster (OTU) of sequences belonging to *Faecalibacterium prausnitzii*. This OTU contains sequences belonging to two time-series clusters that demonstrate different temporal dynamics, coloured red and blue.

at finer taxonomic levels. This may be because these organisms are distinct according to their 16S rRNA genes, but demonstrate similar phenotypic traits under the observed environmental conditions.

Researchers frequently use 97% OTUs as the base ecological unit when making inferences about a microbial community. We have discussed previously that these OTUs can contain temporal inconsistencies (see Section 1.3.1), where the sequences contained in a 97% sequence identity cluster may not have the same distributions across time. This implies that there are multiple phenotypically distinct organisms that are being grouped into the same OTU. Biological data sets show that this is not an uncommon phenomenon. For example, one OTU of the species *Faecalibacterium prausnitzii* showed two clear temporal patterns, situated around the occurrence of the food poisoning event (Figure 3.4). Despite the marker-gene sequences having high identity, the microorganisms demonstrate very distinct behaviours after the event, with one set becoming very low abundance while the other increases in abundance. The relationship between time-series clusters and OTUs is not straightforward. Some OTUs contain multiple time-series clusters, but the converse can be true as well (e.g., Figure 3.5). This can make it difficult to unravel the discrepancies between these two clustering approaches.

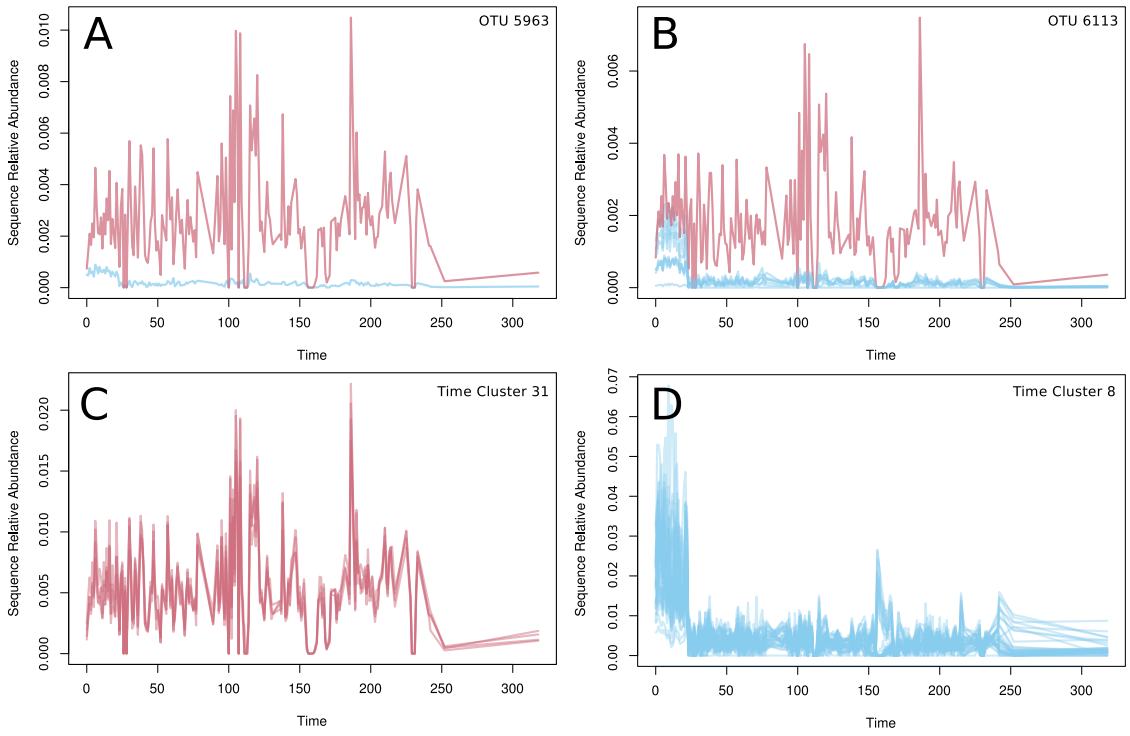

Figure 3.5: A demonstration of the relationship between time-series and sequence-based clusters from the year of faecal samples data set. A) and B) 97% sequence identity clusters (OTUs) of *Bacteroides plebeius* sequences. C) and D) Time-series clusters of *Bacteroides plebeius* sequences. Red sequences belong to time-series cluster 31, while blue sequences belong to time-series cluster 8. Sequences from both time-series clusters are present in both OTUs and vice versa.

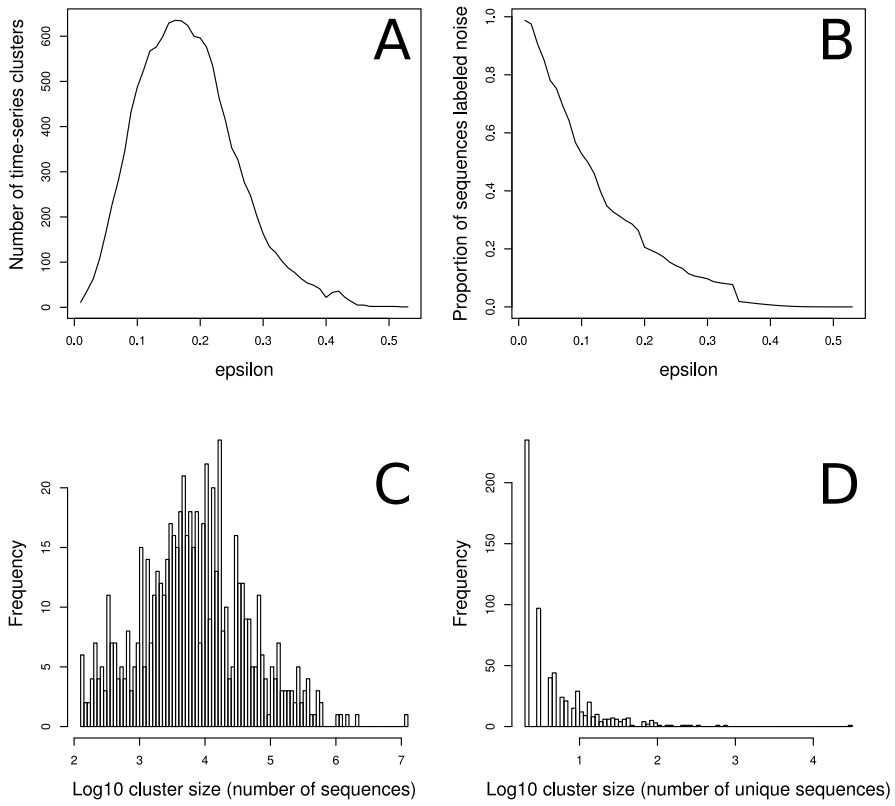

Figure 3.6: Data set statistics for the Lake Mendota data. A) Number of time-series clusters vs. ϵ. B) Proportion of data labeled as noise vs. ϵ. C) Histogram of log cluster size by total number of sequences. D) Histogram of log cluster size by number of unique sequences.

3.2 Lake Mendota, Wisconsin

The second biological time-series data set we analyzed is from Lake Mendota in Wisconsin, USA. This is a eutrophic lake that freezes each winter and thaws each spring [89]. The microbial communities follow strong seasonal trends that are ideal candidates to capture and investigate with our time-series clustering algorithm. The data set spans eleven years (2000-2011), with 96 samples of 16S rRNA gene fragments in total [51]. Samples were taken monthly while the lake was thawed, generally between March and November.

There were 45,094,125 total and 3,058,149 unique sequences. For time-series clustering, the data were filtered to only include sequences which appeared in $\geq$20% of

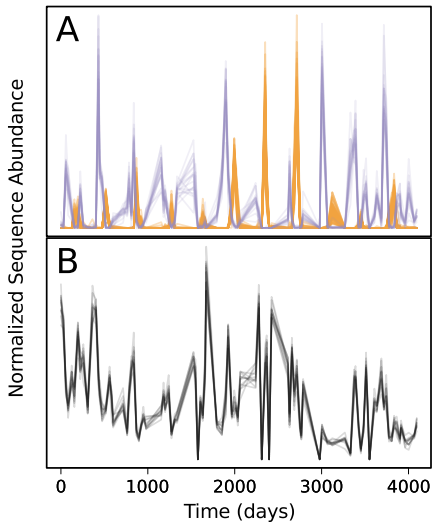

Figure 3.7: Two superimposed time-series clusters of *Flectobacillus* (purple) and *Synechococcus* (orange) sequences displaying clear seasonal dynamics and peaking in different seasons. B) A time-series cluster of *Actinomycetales* sequences displaying persistence through multiple seasons.

Table 3.2: Average Simpson Index for lake time-series clusters at ϵ=5.5

Taxonomic Level	Average Simpson Index
Phylum	0.96
Class	0.95
Order	0.94
Family	0.94
Genus	0.95
Species	0.99

time points, reducing the data to 37,796,894 sequences and 38,204 unique sequences. As this data set was larger than the faecal data from the previous section, a more stringent filter of 20% (vs. 15% for the faecal data) was required for the pairwise distance matrix to fit in memory. A maximum of 626 time-series clusters was found at ϵ=0.16, with an average cluster size of 60,378 total sequences and 61 unique sequences (Figure 3.6). 31% of the data were labeled as noise at this ϵ value. This maximum number of clusters is in contrast to a recent analysis of this data set that grouped 97% OTUs from these sequences into only 14 clusters based on their annual peak [15]. Our clustering was based on the entire time series instead of a single temporal feature, which resulted in many additional clusters.

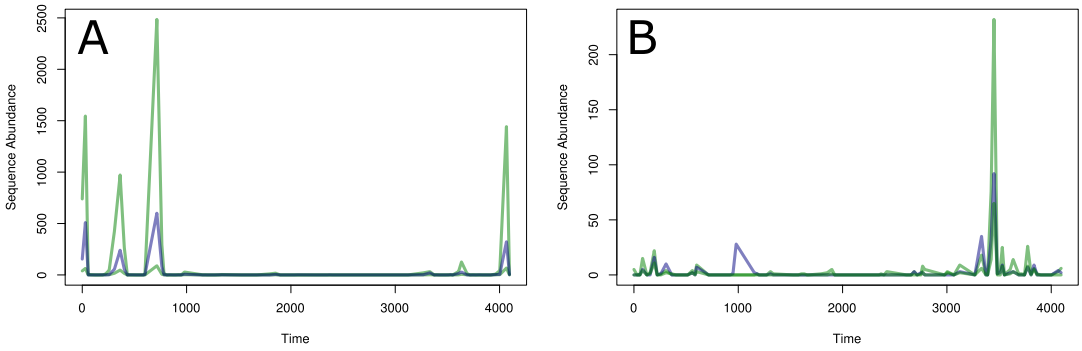

Figure 3.8: Two examples of taxonomically heterogeneous time-series clusters. Green lines are sequences classified as chloroplasts, and blue lines are sequences classified as mitocondria. A) Time-series cluster of sequences from two organelles of the green algal class *Trebouxiophyceae*. B) Time-series cluster of sequences from two organelles of the diatom *Thalassiosira pseudonana*.

When clustered by time series, the seasonal dynamics within this freshwater lake are brought to the surface. For example, sequences classifying to the photosynthetic cyanobacterial genus *Synechococcus* were seen to bloom every year around August, whereas sequences from the genus *Flectobacillus* increased in relative abundance in May of each year (Figure 3.7A), anti-correlating (r = -0.25, p = 0.01) with the *Synechococcus* sequences. Other microorganisms are more persistent through multiple seasons and rarely fall below the detection limit, such as a cluster of *Actinomycetales* sequences (Figure 3.7B).

As before, we used the Simpson index to evaluate the taxonomic homogeneity of recovered clusters. The majority of the time-series clusters were taxonomically homogeneous, especially at higher taxonomic levels, with an average Simpson index of 0.96 at the phylum level and 0.95 at the genus level at ϵ=0.16. Two examples of taxonomically heterogeneous time-series clusters are shown in Figure 3.8. The first contained three sequences that classified to two different groups: chloroplast from the green algal class *Trebouxiophyceae*, and *"Rickettsiales; mitochondria"* (Figure 3.8A). The two chloroplast sequences were four times more abundant than the mitochondrial sequence, which, when compared to a broader database, was found to have the highest similarity to sequences from the class *Trebouxiophyceae*. In Figure 3.8B, we see chloroplast and mitochondrial sequences classified to the diatom *Thalassiosira*

pseudonana have been clustered together by our algorithm. While the presence of these eukaryotic sequences in a data set targeting bacteria and arhaea is not surprising [35, 32], it is a good validation of our method to see that sequences from two organelles of a single taxonomic group follow are properly clustered together.

Like the faecal data set, temporal inconsistencies were a common occurrence with the OTUs of the lake data set. A clear example of temporal discordance from the freshwater lake data is demonstrated by sequences contained within a 97% sequence identity *Sediminibacterium* OTU that exhibits two distinct temporal patterns (Figure 3.9A). Both groups of sequences spike at consistent times each year, the first in late June to early August and the other in late August to early October. This could mean that this OTU includes two distinct types that favour slightly different seasonal conditions. Traditional OTU clustering would have added the abundances of these sequences together and completely obscured the dynamic our clustering method reveals.

While the majority of our OTUs were temporally consistent, discordance was seen more frequently in certain taxonomic groups. These groups included *Actinomycetales* ACK-M1, *Chlorobi* OPB56, *Fluviicola*, and *Sedminibacterium*. This difference could be due to variation in sequence divergence of the hypervariable region, or different processes affecting the functional diversity of these groups. By presenting the relevant sequence metadata alongside the temporal clusters, our user interface helps highlight taxonomic groups that display multiple distinct temporal patterns despite high nucleotide identity. Frequent temporal discordance within the OTUs of a taxonomic group of interest could hint at undetected diversity and may motivate the selection of a different 16S rRNA gene region for future work.

Within both the lake and the faecal data, it was clear that sequence similarity does not guarantee temporal similarity (for example, Figure 3.4 and Figure 3.9A). In the lake data set, we found further evidence that temporal similarity does not imply sequence similarity. As an example, we identified an OTU classified to *Actinomycetales* ACK-M1 that displayed three distinct temporal patterns (Figure 3.9B-E). Two of the time-series clusters (numbers 105 and 214) contained sequences that were all classified to the same group, *Actinomycetales* ACK-M1 (Figure 3.9C-D). Despite the taxonomic similarity, these time-series clusters spanned 27 and 13 (respectively)

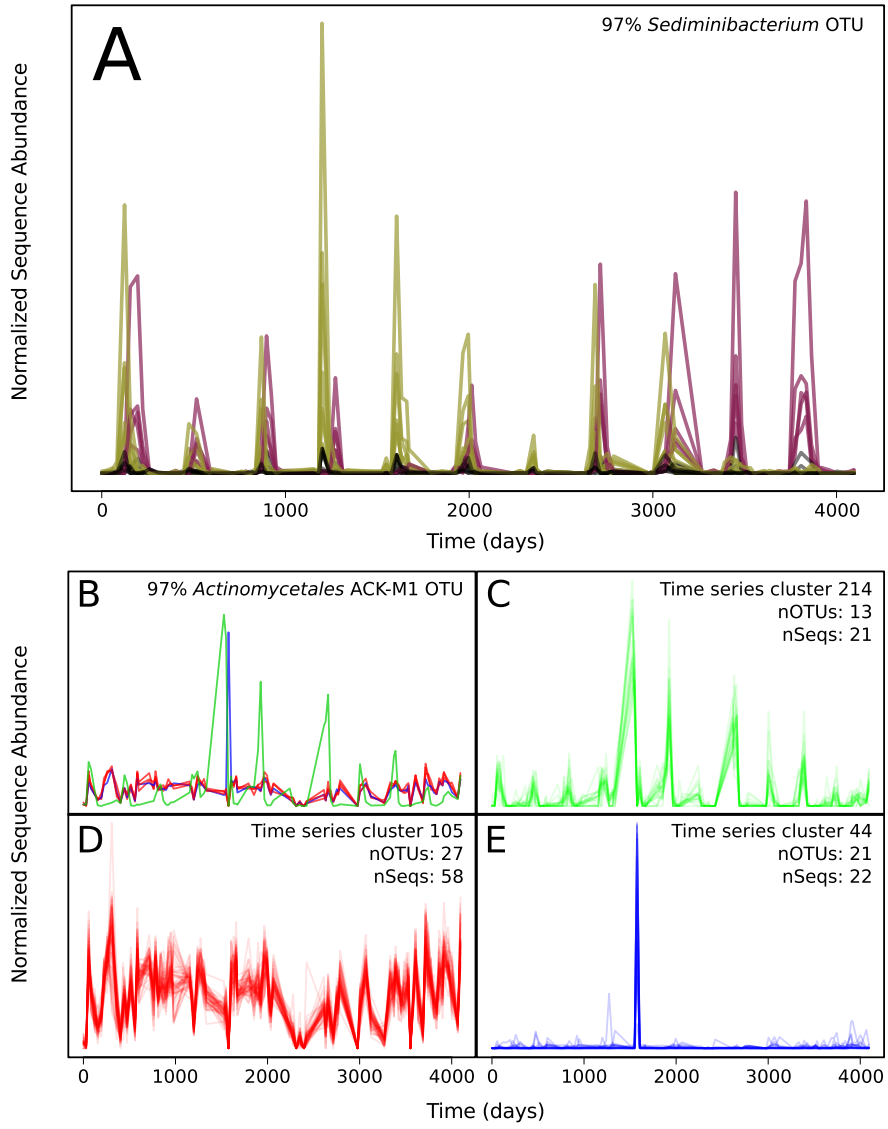

Figure 3.9: Examples of temporal discordance within 97% sequence-identity OTUs in Lake Mendota. A) Sequences from a single 97% sequence identity cluster classified to *Sediminibacterium*, coloured by time-series cluster to highlight temporal discordance. One group of sequences (green) consistently blooms just before a second group (purple). B) A second example of temporal discordance within an OTU. Three distinct temporal patterns were observed in this 97% sequence identity cluster, classified to *Actinomycetales* ACK-M1. The five most abundant sequences are shown, coloured by time-series cluster. C-E) The time-series clusters for each of the three distinct patterns seen in the *Actinomycetales* ACK-M1, along with the number of OTUs and sequences belonging to each cluster.

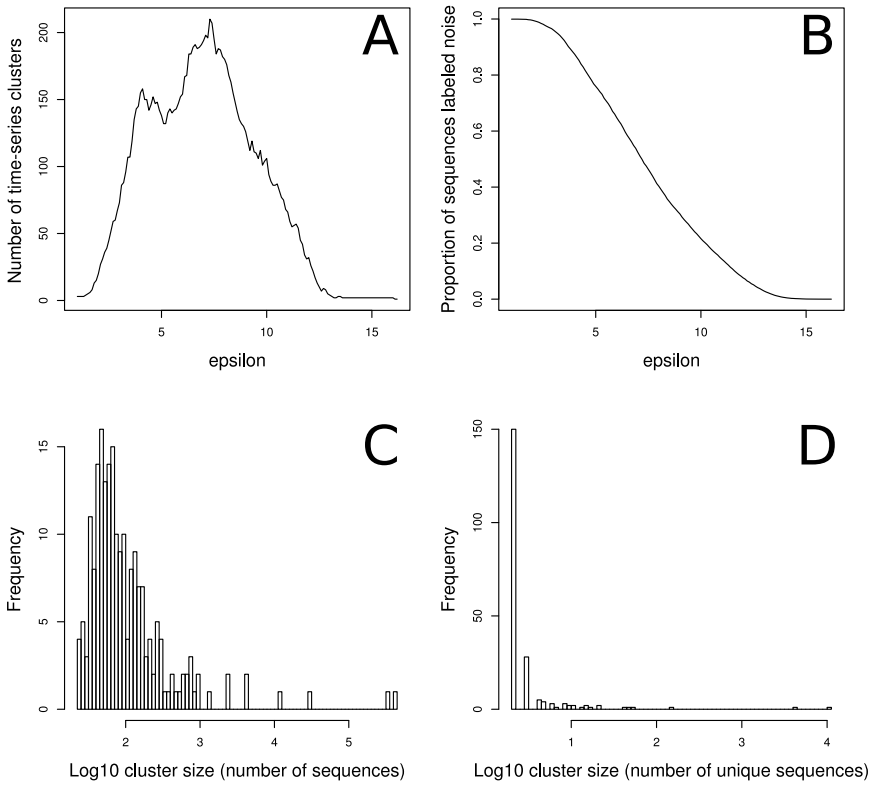

Figure 3.10: Data set statistics for the elder care faecal data. A) Number of time-series clusters vs. ϵ. B) Proportion of data labeled as noise vs. ϵ. C) Histogram of log cluster size by total number of sequences. D) Histogram of log cluster size by number of unique sequences.

distinct 97% sequence-identity OTUs. The third cluster (number 44) contained sequences that classified to six distinct phyla (Figure 3.9E). The spike in abundance around day 1,500 appears to be the primary driver of this cluster. It is unclear whether this spike in abundance is due to some shared trait between very distant strains, contamination, or an artefact from the sequencing or normalization processes.

3.3 Multiple Time Series: Elder Care Facility Faecal Samples

The multiple time series data set is faecal samples from subjects at an elder care facility. Samples were collected from 43 patients over the span of a month. Each subject was sampled between two and five times, for 182 total time points. There were 3,603,610 total sequences and 2,015,820 unique sequences in the data set. After

Table 3.3: Average Simpson Index for elder care faecal time-series clusters at ϵ=6.4

Taxonomic Level	Average Simpson Index
Phylum	0.98
Class	0.97
Order	0.97
Family	0.96
Genus	0.95
Species	0.96

filtering out sequences present in fewer than 5% of sequences, 827,958 sequences and 15,704 unique sequences remained. The maximum number of time-series clusters was 210 at ϵ=6.4. At this ϵ value, the average number of sequences per cluster was 3,943, the average number of unique sequences per cluster was 75, and 48% of the sequences were labeled as noise.

For this data set, the taxonomic homogeneity of the time-series clusters was high (Table 3.3). However, the number of sequences that were labeled as noisy was significantly higher in this data set than the previous two (48% vs. 31% and 17%). That is, there was a lower proportion of sequences that were successfully clustered at the ϵ value that showed the highest separation of the data. This suggests that the data were harder to cluster. In addition, the clusters tended to be very sparse, showing detectable abundance only in a small number of the subjects' time-series (e.g., Figure 3.11). In this time-series cluster, we see that most of the subjects have no detectable occurrence of these sequences. We also see that while the sequences associate clearly across time within some subjects, this is not true for all subjects.

Using our method with multiple time series, we can get a hint of the genetic diversity of an organism and the distribution of strains across sample sites. By contrasting the sequence-identity clusters with the time-series clusters, we can identify those strains which are differentially abundant across sites. For example, a sequence-identity cluster of *Akkermansia muciniphila* was colour-coded by time-series clusters to reveal which subjects shared strains in common (Figure 3.12). Subjects that share the same colours in their time-series plots have the same sequences present in their faecal samples. At least three main groups of strains are revealed to be contained

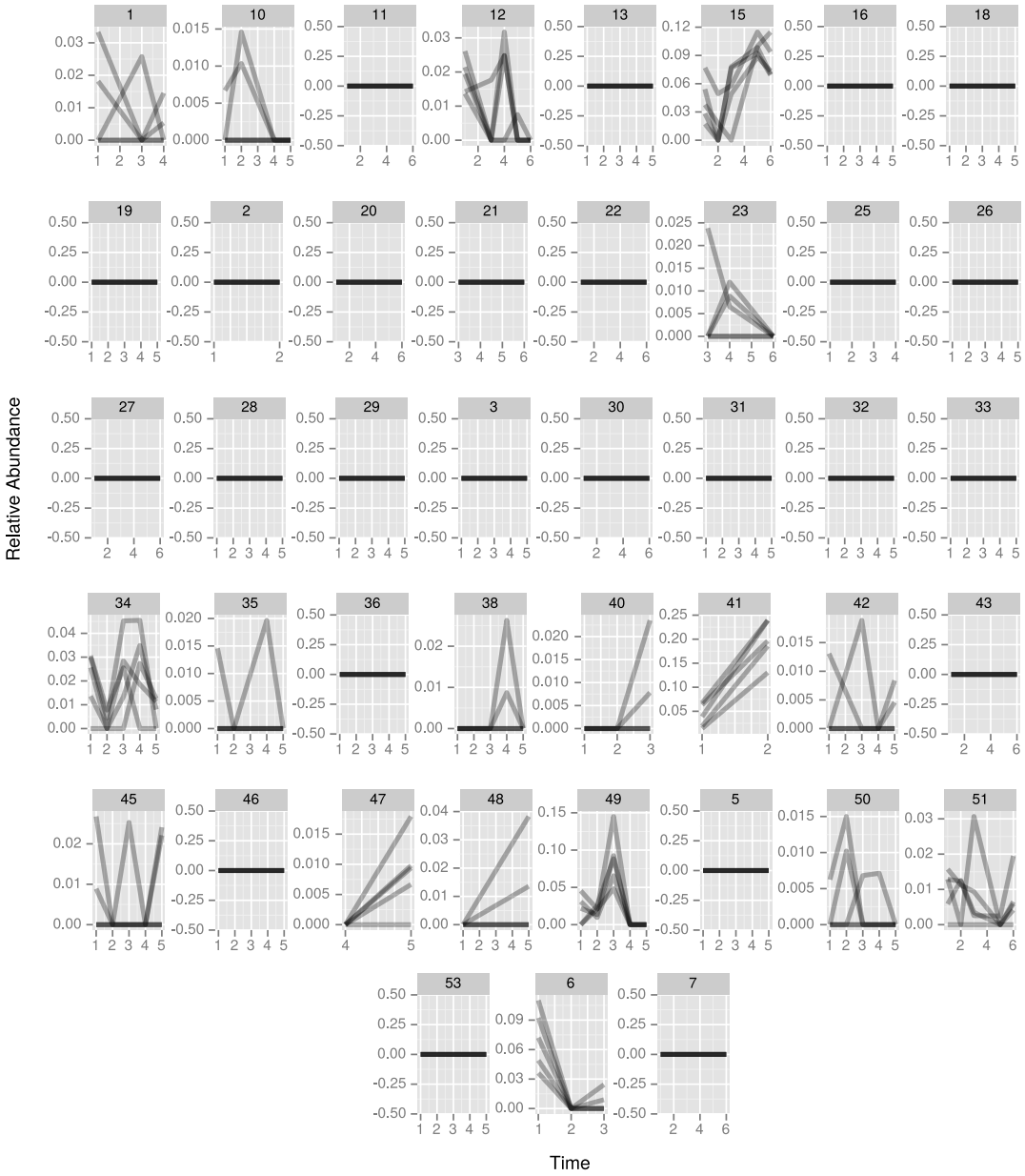

Figure 3.11: A sparse time-series cluster of *Bacteroides caccae* sequences.

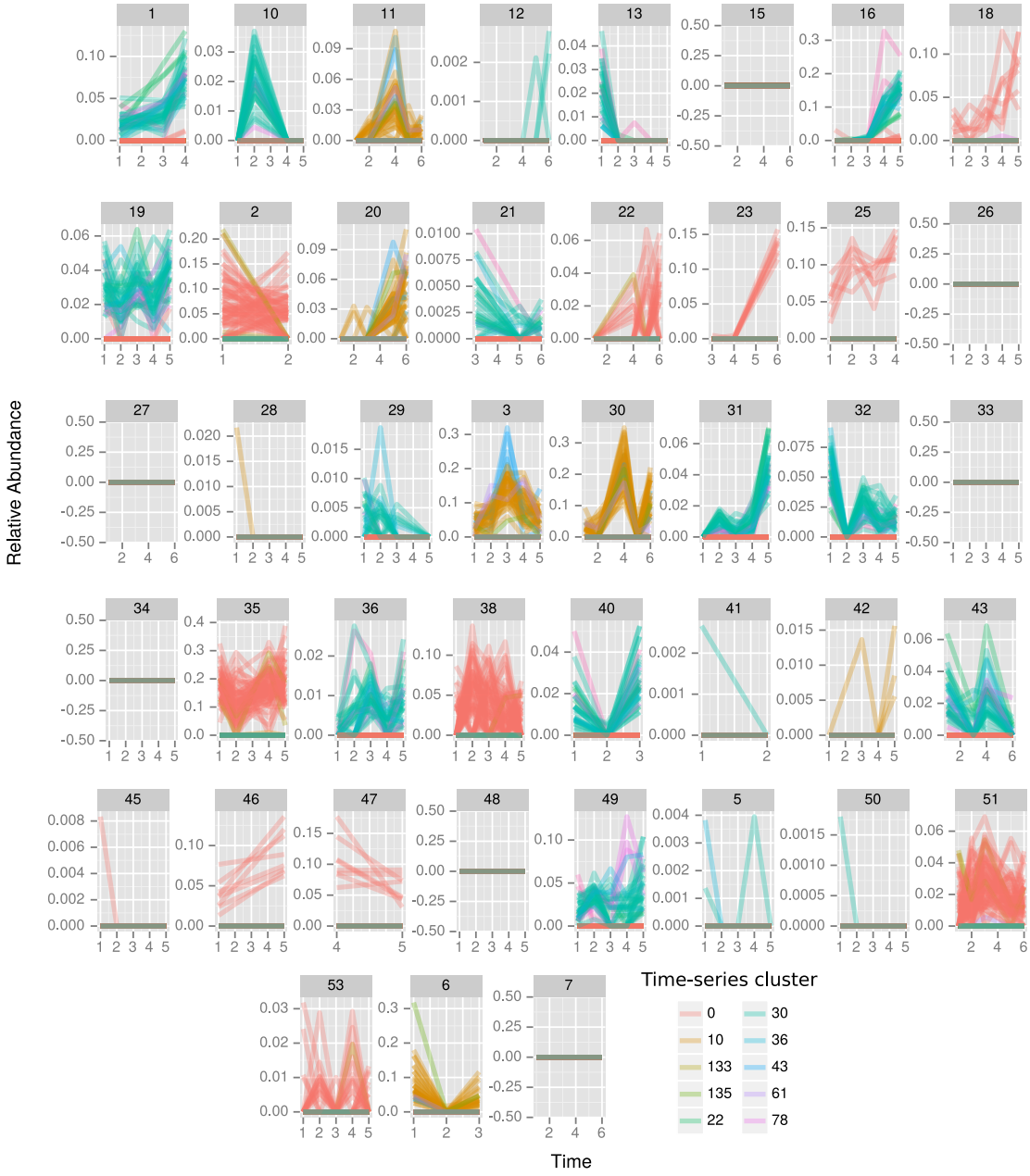

Figure 3.12: Time-series of an OTU of *Akkermansia muciniphila* across 43 subjects. Time-series are coloured by time-series clusters at $\epsilon = 5$.

within this OTU and shared between subjects at the care facility. This type of visualization is useful as it provides a sub-OTU level of detail at a glance. In cases where a 97% sequence-identity OTU is too coarse, this plot uncovers the fine-grained distribution patterns that may be hidden from view.

3.3.1 Complications for Multiple Time Series

Clustering with multiple time series is a more difficult challenge than with one contiguous time series. The proportion of noise is much higher in the multiple time series data set (Figures 3.10B). The distribution of cluster size is much more narrow (Figures 3.10C-D). There are many more small clusters composed of relatively few unique sequences. However, the high taxonomic homogeneity suggests that the clusters that are generated are not spurious.

There are several reasons why clustering with multiple time series results in lower quality clusters. First, each environment has its own distinct microbial strains. This results in very sparse time series where many sequences are present only in one time series and absent in the others. This sparsity results in dense clusters for each time series. It also makes it difficult to cluster across time series, as there must be overlap between the time series for the clusters to be able to form. A possible solution to this problem is a very conservative pre-clustering based on sequence identity. This will add together the different strains from different environments, reducing the sparsity of the data.

When the sequences originate from multiple different environments, there are distinct environmental and microbial contexts that can be detrimental to the clustering process. As an example, suppose we have three microbial species that we will call x, y, and z, represented across two environments (Figure 3.13). In the first environment, x and y follow the same distribution, and z is not present. In the second environment, z is present and harms y, but not x. Although x and y would form a time-series cluster in the first environment, it would not in the second environment. This is only a simple example; the problem compounds with thousands of microbial strains spread across many different time series.

Another shortcoming of using multiple short time series is the difficulty of interpretation of the results. Time-series clusters from a single, long, contiguous time

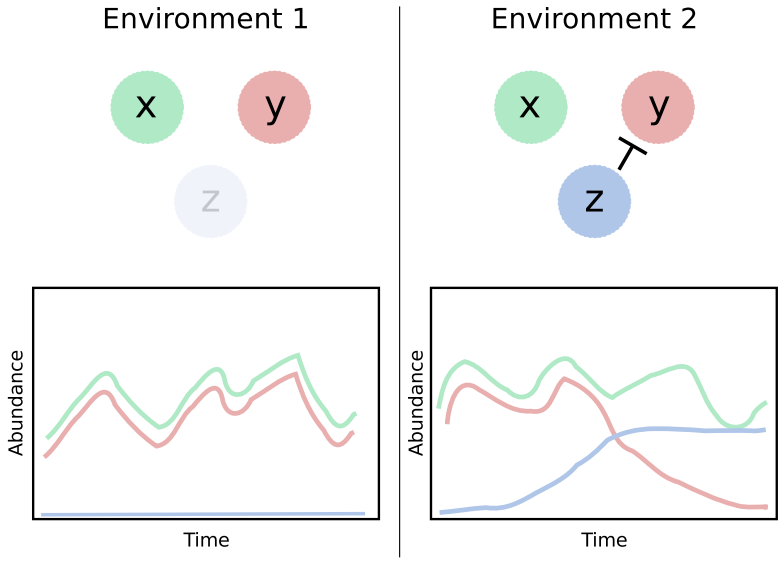

Figure 3.13: An example of how the microbial context of two distinct environments may influence time-series clustering with multiple time-series. The interactions between the taxa are represented by the three coloured circles. In Environment 1, taxon z is not present, so no interactions occur. In Environment 2, taxon z decreases the abundance of taxon y.

series are straightforward to interpret. Major events that affect the microbial community composition are clearly visible in the time-series plots. Changes to baseline abundance levels are evident in the time-series clusters (e.g., Figure 3.3). With many short time series it can be difficult to establish what the baseline is, so detecting changes may not be possible. In the case of the elder care faecal data, there was no intervention and little dynamic metadata. As a result, there were no specific events to monitor or analyze. This made the data difficult to interpret as we could not link a change in abundance of a specific taxon with any event.

In general, multiple time-series clustering produces useful results and allows for the contrasting of multiple sets of longitudinal data. In particular, it is useful to contrast time-series patterns with sequence-identity clustering to identify strains that are present across different environments (e.g., Figure 3.12). However, in light of the weaknesses we have described in this section, we caution that our method may not provide useful results for all longitudinal data sets. Longer time series and more evenly spaced time points contribute to more successful clustering, visualization, and interpretation.

Chapter 4

Conclusions

In this work, we describe a novel algorithm for the clustering of microbial marker-gene sequence data. The goals of this algorithm are to reduce the magnitude of data and to aid in the discovery of temporal structure in the data. With careful consideration of data structures, distance measures, clustering algorithms, we developed a fast and scalable method that is able to meet these goals. Our approach differs from traditional methods by clustering the temporal distribution patterns of each unique sequence rather than clustering the DNA sequences directly. This introduces new challenges, such as the parameterization of the clustering process and processing data sets that contain multiple time series, but provides the benefit of reducing data magnitude significantly while minimizing the loss of temporal dynamics. Our method distills high dimensional time-series data down to its essential temporal patterns, facilitating the exploration of microbial marker-gene data sets.

Ananke and Ananke-UI are effective tools for exploring microbial marker-gene data. We have created an efficient, interactive, and cross-platform solution for the deep exploration of large marker-gene sequence data sets. The clustering step is able to reduce data sets by several orders of magnitude from tens of thousands of unique time series down to hundreds, making manual exploration of marker-gene data feasible. We present the time-series clusters and relevant metadata to the domain expert, which facilitates their efforts to unravel the complicated dynamics of microbial communities. We developed our method with this expert in mind, with a focus on ease of use and speed. This work represents a complete workflow from raw data to visualization, eliminating the barriers present in similar bioinformatics tools.

4.1 Comparisons to Existing Tools

We have briefly discussed other tools that are designed to process longitudinal marker-gene sequence data (see Section 1.4.3). Our method differs from these tools in key

ways. MC-TIMME is a powerful tool that serves a similar purpose: to cluster sequences that share similar temporal distributions [27]. The key difference between our methods is the input requirement. MC-TIMME requires the user to know much more information about the temporal dynamics in their environment *a priori*. This information is used to generate template time series that the queries from the biological data can be matched to. In contrast, our method is completely unsupervised and requires no information from the user about the temporal dynamics. By design, our method requires neither the shapes nor the number of distinct temporal patterns. This allows users to cluster data from environments where all variables cannot be controlled and unknown influences can affect the temporal patterns.

We also discussed eLSA, a tool for computing the similarity of time series. The goal of this tool is to generate association networks using local similarity scores between time series. While it is not designed to explicitly generate time-series clusters, setting a minimum similarity threshold to an association network will create clusters that can be used in a similar fashion. eLSA has several key advantages over our method. First, it is able to detect time-lagged associations between microbial species. This is very useful, as time-lagged associations can indicate that microorganisms are interacting in some fashion. It can also detect local similarity between two time series, in contrast to our method which requires global similarity across all time points. Local similarity detection could identify two time series that are synchronized only until a change in the environment occurs. If the environmental change is known, it could shed light on the adaptation that occurred in the more successful microorganism. Local similarity analysis could also provide a more sensitive way to cluster time series, but it should be noted that there are some who argue that it can generate meaningless clusters if not done with care [38]. The drawback to time-lagged and local similarity analysis is the high computational complexity. eLSA is typically run on small subsets of OTUs, not the much larger set of unique sequences that our algorithm can process. However, our method and eLSA complement one another well. Our method can act as a data reduction step that does not compromise the quality of the temporal patterns. The time-series clusters can be used as input to eLSA, resulting in a very significant decrease in run time.

4.2 Extensions and Future Work

Our work has the potential to be extended to several other applications. Time is not the only gradient that we can sample microbial communities along. Distance, pH, and other variables can be used in an analogous way to time as the independent variable. This affects only the interpretation of the results and will work immediately without any changes to the algorithm. Further, microbial marker-gene data sets are not the only high dimensional time-series data. Our algorithm could be used to cluster any set of time series with only minor changes to the software. As an example from the field of metagenomics, our method could be used to cluster metagenomic fragments to aid in genome assembly, in a similar fashion to the work by Dick *et al.* (2009) [18].

While our comprehensive tool can take the user from raw sequence data to informative visualizations, there are still some improvements to be made. The algorithm is currently limited by the amount of available memory. The pairwise distances between time series are stored in memory which can limit the number of unique sequences to ∼30,000 on a system with 16GB of RAM. While we aim to only filter out time series with low information content, it would be beneficial if we could minimize the amount of data that is lost. A solution to this problem is an out-of-core implementation of DBSCAN that reads distances from our HDF5 data file.

The next issue to address is the degradation of cluster quality for long time series, as determined by the simulated data (Figure 2.6). Smoothing longer time series by averaging over a sliding window could help reduce noise that accumulates in the STS distance computation and overwhelms real signal. We will also investigate additional distance measures, such as those described in Section 1.4.2, to determine if gains can be made in cluster quality for longer time series. Next, we note that some data sets have a high proportion of data labeled as noise by DBSCAN. By using alternative forms of DBSCAN, such as the OPTICS algorithm [3], we can eliminate the need to search for an appropriate ϵ parameter for a given data set. This algorithm selects locally variable ϵ values rather than using a global parameter. This change allows sparse areas to use higher ϵ values to reduce the number of time series that are considered noise.

Our method discovers structure in the data that we aim to exploit with future work. Time-series clusters represent temporally cohesive units that we can use for

downstream analyses of marker-gene data. Using these clusters with techniques from time-series analysis and machine learning, we can model, predict, and draw inferences from these microbial communities in an automated way. These analyses can be incorporated into the existing Ananke framework, making this work a solid foundation for future research.

Bibliography

[1] Abell, G. C., Cooke, C. M., Bennett, C. N., Conlon, M. A., and McOrist, A. L. Phylotypes related to Ruminococcus bromii are abundant in the large bowel of humans and increase in response to a diet high in resistant starch. *FEMS Microbiology Ecology 66*, 3 (2008), 505–515.

[2] Achtman, M., and Wagner, M. Microbial diversity and the genetic nature of microbial species. *Nature Reviews Microbiology 6*, 6 (2008), 431–440.

[3] Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. OPTICS: ordering points to identify the clustering structure. In *ACM Sigmod Record* (1999), vol. 28, ACM, pp. 49–60.

[4] Bartram, A. K., Lynch, M. D., Stearns, J. C., Moreno-Hagelsieb, G., and Neufeld, J. D. Generation of multimillion-sequence 16S rRNA gene libraries from complex microbial communities by assembling paired-end Illumina reads. *Applied and Environmental Microbiology 77*, 11 (2011), 3846–3852.

[5] Beazley, M. J., Martinez, R. J., Rajan, S., Powell, J., Piceno, Y. M., Tom, L. M., Andersen, G. L., Hazen, T. C., Van Nostrand, J. D., Zhou, J., et al. Microbial community analysis of a coastal salt marsh affected by the Deepwater Horizon oil spill. *PLOS ONE 7*, 7 (2012), e41305.

[6] Bezdek, J. C., Ehrlich, R., and Full, W. FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences 10*, 2 (1984), 191–203.

[7] Bosch, A. A., Levin, E., van Houten, M. A., Hasrat, R., Kalkman, G., Biesbroek, G., de Steenhuijsen Piters, W. A., de Groot, P.-K. C., Pernet, P., Keijser, B. J., et al. Development of Upper Respiratory Tract Microbiota in Infancy is Affected by Mode of Delivery. *EBioMedicine* (In press).

[8] Bräuer, S., Vuono, D., Carmichael, M., Pepe-Ranney, C., Strom, A., Rabinowitz, E., Buckley, D., and Zinder, S. Microbial sequencing analyses suggest the presence of a fecal veneer on indoor climbing wall holds. *Current Microbiology 69*, 5 (2014), 681–689.

[9] Cai, Y., and Sun, Y. ESPRIT-Tree: hierarchical clustering analysis of millions of 16S rRNA pyrosequences in quasilinear computational time. *Nucleic Acids Research* (2011), gkr349.

[10] Caporaso, J. G., Lauber, C. L., Costello, E. K., Berg-Lyons, D., Gonzalez, A., Stombaugh, J., Knights, D., Gajer, P., Ravel, J., Fierer, N., et al. Moving pictures of the human microbiome. *Genome Biology 12*, 5 (2011), R50.

[11] Caporaso, J. G., Lauber, C. L., Walters, W. A., Berg-Lyons, D., Huntley, J., Fierer, N., Owens, S. M., Betley, J., Fraser, L., Bauer, M., et al. Ultra-high-throughput microbial community analysis on the Illumina HiSeq and MiSeq platforms. *The ISME Journal 6*, 8 (2012), 1621–1624.

[12] Caporaso, J. G., Paszkiewicz, K., Field, D., Knight, R., and Gilbert, J. A. The Western English Channel contains a persistent microbial seed bank. *The ISME Journal 6*, 6 (2012), 1089–1093.

[13] Chen, W., Zhang, C. K., Cheng, Y., Zhang, S., and Zhao, H. A comparison of methods for clustering 16S rRNA sequences into OTUs. *PLOS ONE 8*, 8 (2013), e70837.

[14] Cole, J. R., Wang, Q., Cardenas, E., Fish, J., Chai, B., Farris, R. J., Kulam-Syed-Mohideen, A., McGarrell, D. M., Marsh, T., Garrity, G. M., et al. The Ribosomal Database Project: improved alignments and new tools for rRNA analysis. *Nucleic Acids Research 37*, suppl 1 (2009), D141–D145.

[15] Dam, P., Fonseca, L. L., Konstantinidis, K. T., and Voit, E. O. Dynamic models of the complex microbial metapopulation of Lake Mendota. *NPJ Systems Biology and Applications 2* (2016), 16007.

[16] David, L. A., Materna, A. C., Friedman, J., Campos-Baptista, M. I., Blackburn, M. C., Perrotta, A., Erdman, S. E., and Alm, E. J. Host lifestyle affects human microbiota on daily timescales. *Genome Biology 15*, 7 (2014), R89.

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

[18] Dick, G. J., Andersson, A. F., Baker, B. J., Simmons, S. L., Thomas, B. C., Yelton, A. P., and Banfield, J. F. Community-wide analysis of microbial genome sequence signatures. *Genome Biology 10*, 8 (2009), 1–16.

[19] Edgar, R. C. UPARSE: highly accurate OTU sequences from microbial amplicon reads. *Nature Methods 10*, 10 (2013), 996–998.

[20] Eren, A. M., Maignien, L., Sul, W. J., Murphy, L. G., Grim, S. L., Morrison, H. G., and Sogin, M. L. Oligotyping: differentiating between closely related microbial taxa using 16S rRNA gene data. *Methods in Ecology and Evolution 4*, 12 (2013), 1111–1119.

[21] Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (1996), vol. 96, pp. 226–231.

[22] Everard, A., Belzer, C., Geurts, L., Ouwerkerk, J. P., Druart, C., Bindels, L. B., Guiot, Y., Derrien, M., Muccioli, G. G., Delzenne, N. M., et al. Cross-talk between Akkermansia muciniphila and intestinal epithelium controls diet-induced obesity. *Proceedings of the National Academy of Sciences 110*, 22 (2013), 9066–9071.

[23] Faust, K., Lahti, L., Gonze, D., de Vos, W. M., and Raes, J. Metagenomics meets time series analysis: unraveling microbial community dynamics. *Current Opinion in Microbiology 25* (2015), 56–66.

[24] Foster, J. A., and Neufeld, K.-A. M. Gut–brain axis: how the microbiome influences anxiety and depression. *Trends in Neurosciences 36*, 5 (2013), 305–312.

[25] Franzosa, E. A., Hsu, T., Sirota-Madi, A., Shafquat, A., Abu-Ali, G., Morgan, X. C., and Huttenhower, C. Sequencing and beyond: integrating molecular'omics' for microbial community profiling. *Nature Reviews Microbiology 13*, 6 (2015), 360–372.

[26] Gerber, G. K. The dynamic microbiome. *FEBS Letters 588*, 22 (2014), 4131–4139.

[27] Gerber, G. K., Onderdonk, A. B., and Bry, L. Inferring dynamic signatures of microbes in complex host ecosystems. *PLOS Computational Biology 8*, 8 (2012), e1002624.

[28] Granger, C. W. Some recent development in a concept of causality. *Journal of Econometrics 39*, 1 (1988), 199–211.

[29] Hamady, M., Walker, J. J., Harris, J. K., Gold, N. J., and Knight, R. Error-correcting barcoded primers allow hundreds of samples to be pyrosequenced in multiplex. *Nature Methods 5*, 3 (2008), 235.

[30] Han, J., Kamber, M., and Pei, J. *Data mining: concepts and techniques*. Elsevier, 2011.

[31] Hansen, C. H. F., Krych, L., Nielsen, D. S., Vogensen, F. K., Hansen, L. H., Sørensen, S. J., Buschard, K., and Hansen, A. Early life treatment with vancomycin propagates *Akkermansia muciniphila* and reduces diabetes incidence in the NOD mouse. *Diabetologia 55*, 8 (2012), 2285–2294.

[32] Hanshew, A. S., Mason, C. J., Raffa, K. F., and Currie, C. R. Minimization of chloroplast contamination in 16S rRNA gene pyrosequencing of insect herbivore bacterial communities. *Journal of Microbiological Methods 95*, 2 (2013), 149–155.

[33] Hao, X., Jiang, R., and Chen, T. Clustering 16S rRNA for OTU prediction: a method of unsupervised Bayesian clustering. *Bioinformatics 27*, 5 (2011), 611–618.

[34] Hastie, T., Tibshirani, R., and Friedman, J. The Elements of Statistical Learning, 2nd edition, 2009. New York: Springer.

[35] Huys, G., Vanhoutte, T., Joossens, M., Mahious, A. S., De Brandt, E., Vermeire, S., and Swings, J. Coamplification of eukaryotic DNA with 16S rRNA gene-based PCR primers: possible consequences for population fingerprinting of complex microbial communities. *Current Microbiology 56*, 6 (2008), 553–557.

[36] Illumina, Inc. *HiSeq X Series of Sequencing Systems Specification Sheet*, 3 2016.

[37] Jiang, D., Pei, J., and Zhang, A. DHC: a density-based hierarchical clustering method for time series gene expression data. In *Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on* (2003), IEEE, pp. 393–400.

[38] Keogh, E., and Lin, J. Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and Information Systems 8*, 2 (2005), 154–177.

[39] Kim, O.-S., Cho, Y.-J., Lee, K., Yoon, S.-H., Kim, M., Na, H., Park, S.-C., Jeon, Y. S., Lee, J.-H., Yi, H., et al. 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 (2012), 716–721.

[40] Kopylova, E., Noé, L., and Touzet, H. SortMeRNA: fast and accurate filtering of ribosomal rnas in metatranscriptomic data. *Bioinformatics 28*, 24 (2012), 3211–3217.

[41] Kunin, V., Engelbrektson, A., Ochman, H., and Hugenholtz, P. Wrinkles in the rare biosphere: pyrosequencing errors can lead to artificial inflation of diversity estimates. *Environmental Microbiology 12*, 1 (2010), 118–123.

[42] Ley, R. E., Turnbaugh, P. J., Klein, S., and Gordon, J. I. Microbial ecology: human gut microbes associated with obesity. *Nature 444*, 7122 (2006), 1022–1023.

[43] Li, W., and Godzik, A. CD-HIT: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics 22*, 13 (2006), 1658–1659.

[44] Liao, T. W. Clustering of time series data - a survey. *Pattern Recognition 38*, 11 (2005), 1857–1874.

[45] Löffler, F. E., and Edwards, E. A. Harnessing microbial activities for environmental cleanup. *Current Opinion in Biotechnology 17*, 3 (2006), 274–284.

[46] Loman, N. J., Misra, R. V., Dallman, T. J., Constantinidou, C., Gharbia, S. E., Wain, J., and Pallen, M. J. Performance comparison of benchtop high-throughput sequencing platforms. *Nature Biotechnology 30*, 5 (2012), 434–439.

[47] Ludwig, W., and Schleifer, K.-H. Phylogeny of bacteria beyond the 16S rRNA standard: Despite expanding data sets and alternative markers for inferring phylogenetic relationships, nothing beats the 16S rRNA gene. *ASM News-American Society for Microbiology 65* (1999), 752–757.

[48] MacQueen, J. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (1967), vol. 1, Oakland, CA, USA., pp. 281–297.

[49] Marri, P. R., Stern, D. A., Wright, A. L., Billheimer, D., and Martinez, F. D. Asthma-associated differences in microbial composition of induced sputum. *Journal of Allergy and Clinical Immunology 131*, 2 (2013), 346–352.

[50] McDonald, D., Clemente, J. C., Kuczynski, J., Rideout, J. R., Stombaugh, J., Wendel, D., Wilke, A., Huse, S., Hufnagle, J., Meyer, F., et al. The Biological Observation Matrix (BIOM) format or: how i learned to stop worrying and love the ome-ome. *GigaScience 1*, 1 (2012), 7.

[51] McMahon, K. North Temperate Lakes Microbial Observatory. `http://lter.limnology.wisc.edu`, 2014.

[52] Mercier, C., Boyer, F., Bonin, A., and Coissac, E. SUMATRA and SUMACLUST: fast and exact comparison and clustering of sequences. In *Programs and Abstracts of the SeqBio 2013 workshop.* (2013), Citeseer, pp. 27–29.

[53] Möller-Levet, C. S., Klawonn, F., Cho, K.-H., and Wolkenhauer, O. Fuzzy clustering of short time-series and unevenly distributed sampling points. In *Advances in Intelligent Data Analysis V.* Springer, 2003, pp. 330–340.

[54] Mörchen, F. Time series feature extraction for data mining using DWT and DFT, 2003. Marburg: Philipps-Marburg University.

[55] Müller, M. Dynamic time warping. *Information Retrieval for Music and Motion* (2007), 69–84.

[56] Nath, S. G., and Raveendran, R. Microbial dysbiosis in periodontitis. *Journal of Indian Society of Periodontology 17*, 4 (2013), 543.

[57] Nemergut, D. R., Schmidt, S. K., Fukami, T., O'Neill, S. P., Bilinski, T. M., Stanish, L. F., Knelman, J. E., Darcy, J. L., Lynch, R. C., Wickey, P., et al. Patterns and processes of microbial community assembly. *Microbiology and Molecular Biology Reviews 77*, 3 (2013), 342–356.

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

[59] Odamaki, T., Kato, K., Sugahara, H., Hashikura, N., Takahashi, S., Xiao, J.-z., Abe, F., and Osawa, R. Age-related changes in gut microbiota composition from newborn to centenarian: a cross-sectional study. *BMC Microbiology 16*, 1 (2016), 1.

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

[61] Rideout, J. R., He, Y., Navas-Molina, J. A., Walters, W. A., Ursell, L. K., Gibbons, S. M., Chase, J., McDonald, D., Gonzalez, A., Robbins-Pianka, A., et al. Subsampled open-reference clustering creates consistent, comprehensive OTU definitions and scales to billions of sequences. *PeerJ 2* (2014), e545.

[62] Rogers, J. E., Whitman, W. B., et al. *Microbial production and consumption of greenhouse gases: methane, nitrogen oxides, and halomethanes.* American Society for Microbiology, 1991.

[63] Roux, S., Enault, F., le Bronner, G., and Debroas, D. Comparison of 16S rRNA and protein-coding genes as molecular markers for assessing microbial diversity (Bacteria and Archaea) in ecosystems. *FEMS microbiology ecology 78*, 3 (2011), 617–628.

[64] RStudio, Inc. *Easy web applications in R.*, 2013. URL: `http://www.rstudio.com/shiny/`.

[65] Sagoff, M. Data deluge and the human microbiome project. *Issues in Science and Technology 28*, 4 (2012), 71.

[66] Sawabe, T., Suda, W., Ohshima, K., Hattori, M., and Sawabe, T. First microbiota assessments of children's paddling pool waters evaluated using 16S rRNA gene-based metagenome analysis. *Journal of Infection and Public Health* (2015).

[67] Schaie, K. W., and Strother, C. R. A cross-sequential study of age changes in cognitive behavior. *Psychological Bulletin 70*, 6p1 (1968), 671.

[68] Schikuta, E. Grid-clustering: An efficient hierarchical clustering method for very large data sets. In *Pattern Recognition, 1996., Proceedings of the 13th International Conference on* (1996), vol. 2, IEEE, pp. 101–105.

[69] Schloss, P. D., Gevers, D., and Westcott, S. L. Reducing the effects of PCR amplification and sequencing artifacts on 16S rRNA-based studies. *PLOS ONE 6*, 12 (2011), e27310.

[70] Shade, A., Caporaso, J. G., Handelsman, J., Knight, R., and Fierer, N. A meta-analysis of changes in bacterial and archaeal communities with time. *The ISME Journal 7*, 8 (2013), 1493–1506.

[71] Shade, A., Jones, S. E., Caporaso, J. G., Handelsman, J., Knight, R., Fierer, N., and Gilbert, J. A. Conditionally rare taxa disproportionately contribute to temporal changes in microbial diversity. *mBio 5*, 4 (2014), e01371–14.

[72] Shade, A., McManus, P. S., and Handelsman, J. Unexpected diversity during community succession in the apple flower microbiome. *mBio 4*, 2 (2013), e00602–12.

[73] Shendure, J., and Ji, H. Next-generation DNA sequencing. *Nature Biotechnology 26*, 10 (2008), 1135–1145.

[74] Simpson, E. H. Measurement of diversity. *Nature* (1949).

[75] Stackebrandt, E., and Goebel, B. Taxonomic note: a place for DNA-DNA reassociation and 16S rRNA sequence analysis in the present species definition in bacteriology. *International Journal of Systematic and Evolutionary Microbiology 44*, 4 (1994), 846–849.

[76] Thaiss, C. A., Zeevi, D., Levy, M., Zilberman-Schapira, G., Suez, J., Tengeler, A. C., Abramson, L., Katz, M. N., Korem, T., Zmora, N., et al. Transkingdom control of microbiota diurnal oscillations promotes metabolic homeostasis. *Cell 159*, 3 (2014), 514–529.

[77] The HDF Group. Hierarchical Data Format, version 5, 1997-NNNN. http://www.hdfgroup.org/HDF5/.

[78] Tikhonov, M., Leach, R. W., and Wingreen, N. S. Interpreting 16S metagenomic data without clustering to achieve sub-OTU resolution. *The ISME Journal 9*, 1 (2015), 68–80.

[79] Turnbaugh, P. J., Ley, R. E., Hamady, M., Fraser-Liggett, C., Knight, R., and Gordon, J. I. The Human Microbiome project: exploring the microbial part of ourselves in a changing world. *Nature 449*, 7164 (2007), 804.

[80] Vaarala, O., Atkinson, M. A., and Neu, J. The "perfect storm" for type 1 diabetes: the complex interplay between intestinal microbiota, gut permeability, and mucosal immunity. *Diabetes 57*, 10 (2008), 2555–2562.

[81] Van Dongen, S. A cluster algorithm for graphs. *Report-Information systems*, 10 (2000), 1–40.

[82] Vergin, K. L., Beszteri, B., Monier, A., Thrash, J. C., Temperton, B., Treusch, A. H., Kilpert, F., Worden, A. Z., and Giovannoni, S. J. High-resolution SAR11 ecotype dynamics at the Bermuda Atlantic Time-series Study site by phylogenetic placement of pyrosequences. *The ISME Journal 7*, 7 (2013), 1322–1332.

[83] Vinh, N. X., Epps, J., and Bailey, J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research 11* (2010), 2837–2854.

[84] Wang, L., Christophersen, C. T., Sorich, M. J., Gerber, J. P., Angley, M. T., and Conlon, M. A. Low relative abundances of the mucolytic bacterium Akkermansia muciniphila and Bifidobacterium spp. in feces of children with autism. *Applied and Environmental Microbiology 77*, 18 (2011), 6718–6721.

[85] Wang, Q., Garrity, G. M., Tiedje, J. M., and Cole, J. R. Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. *Applied and Environmental Microbiology 73*, 16 (2007), 5261–5267.

[86] West, M. L. *The Orphic Poems*. New York: Oxford University Press, 1983.

[87] Whiteside, S. A., Razvi, H., Dave, S., Reid, G., and Burton, J. P. The microbiome of the urinary tract - a role beyond infection. *Nature Reviews Urology 12*, 2 (2015), 81–90.

[88] Xia, L. C., Steele, J. A., Cram, J. A., Cardon, Z. G., Simmons, S. L., Vallino, J. J., Fuhrman, J. A., and Sun, F. Extended local similarity analysis (eLSA) of microbial community and other time series data with replicates. *BMC Systems Biology 5*, 2 (2011), 1.

[89] Yannarell, A., Kent, A., Lauster, G., Kratz, T., and Triplett, E. Temporal patterns in bacterial communities in three temperate lakes of different trophic status. *Microbial Ecology 46*, 4 (2003), 391–405.

[90] Ze, X., Duncan, S. H., Louis, P., and Flint, H. J. *Ruminococcus bromii* is a keystone species for the degradation of resistant starch in the human colon. *The ISME Journal 6*, 8 (2012), 1535–1543.